Synack

black hat USA 2022

Anatomy of AWS Hacking

Jungle Edition

$whoami

- Red Team Member (SRT), Synack Inc.
- Ph.D. Student (Cyber Studies), The University of Tulsa
- Former Information Security Specialist, Augmedix Inc.

ziaurrashid.com https://linkedin.com/in/ziaurrashid https://twitter.com/smziaurrashid

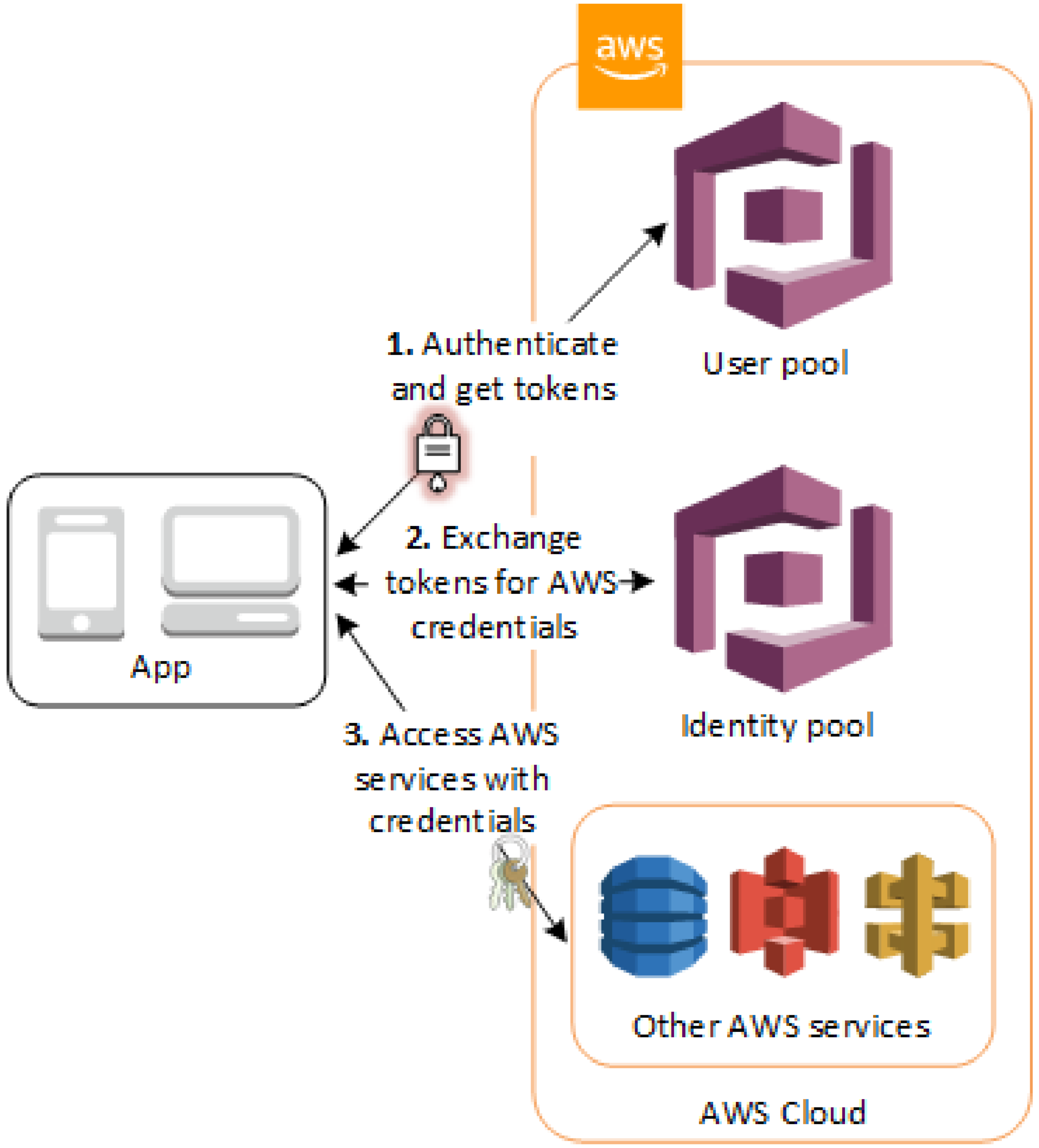
$talkinfo

- Attack scenarios of some common aws services
  - Cognito
  - IAM PrivSec
  - S3 Buckets
  - Data Exfiltration via S3 Replication
- Tools for pentesting, compliance audit and continuous monitoring

AWS Cognito

Attack Vectors

- Leakage Identity Pool ID
  - Hardcoded
  - HTTP Response
- Misconfigured Attributes
- Profile Self-registration allowed



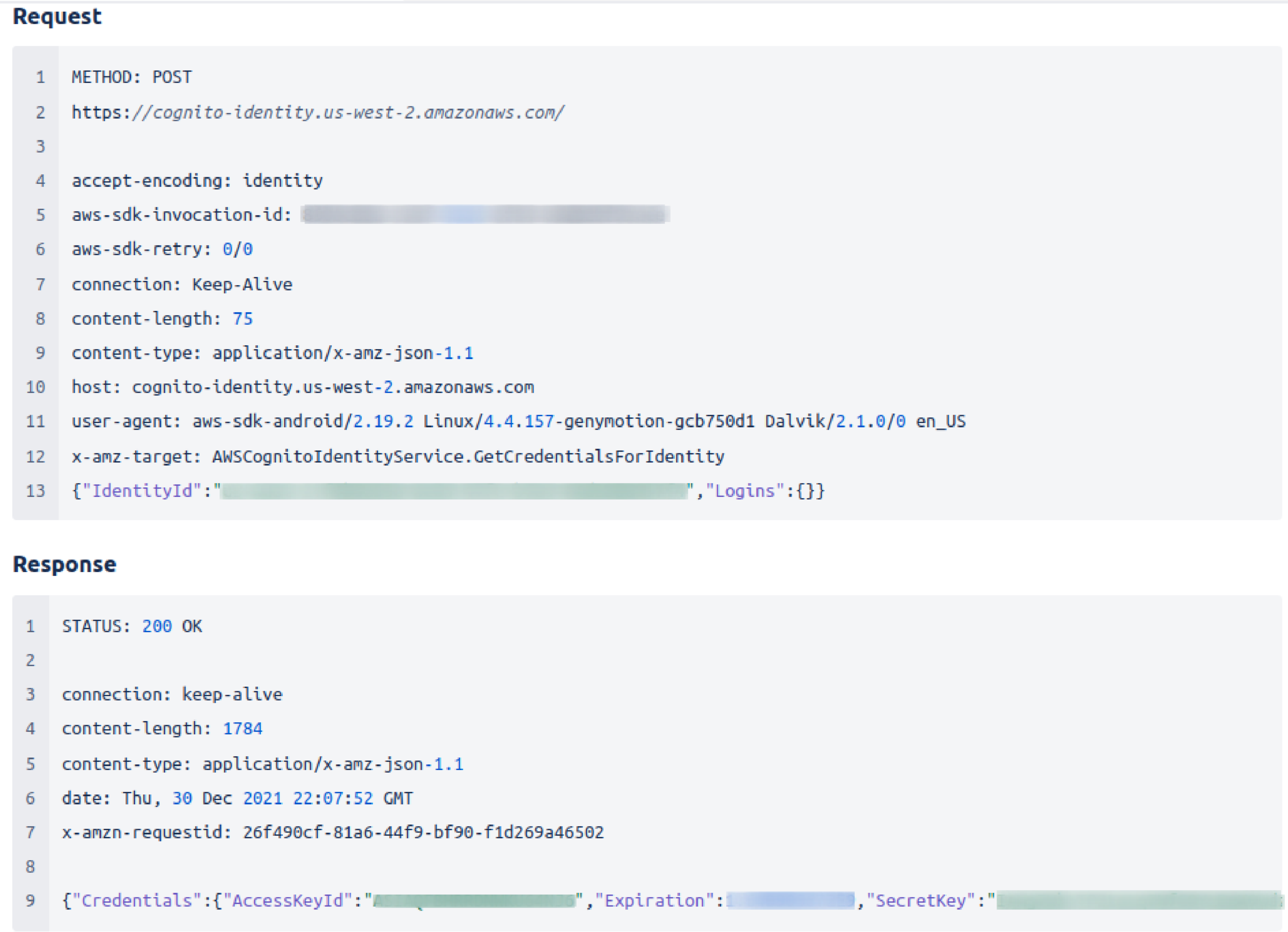
aws
1. Authenticate and get tokens
User pool
App
2. Exchange tokens for AWS credentials
Identity pool
3. Access AWS services with credentials
Other AWS services
AWS Cloud

https://0xn3va.gitbook.io/cheat-sheets/cloud/aws/amazon-cognito
https://blog.appsecco.com/exploiting-weak-configurations-in-amazon-cognito-in-aws-471ce761963
https://notsosecure.com/hacking-aws-cognito-misconfigurations
https://speakerdeck.com/kavisha/amazon-cognito-mis-configurations

EXAMPLE

Identify a hardcoded Cognito Identity Pool by running Nuclei template aws-cognito.yaml on a decompiled Android app.

echo /output_apktool/ | nuclei -t /file/Keys/aws-cognito.yaml



Request

```
METHOD: POST
https://cognito-identity.us-west-2.amazonaws.com/

accept-encoding: identity
aws-sdk-invocation-id: [redacted]
aws-sdk-retry: 0/0
connection: Keep-Alive
content-length: 75
content-type: application/x-amz-json-1.1
host: cognito-identity.us-west-2.amazonaws.com
user-agent: aws-sdk-android/2.19.2 Linux/4.4.157-genymotion-gcb750d1 Dalvik/2.1.0/0 en_US
x-amz-target: AWSCognitoIdentityService.GetCredentialsForIdentity
{"IdentityId":"[redacted]","Logins":{}}
```

Response

```
STATUS: 200 OK

connection: keep-alive
content-length: 1784
content-type: application/x-amz-json-1.1
date: Thu, 30 Dec 2021 22:07:52 GMT
x-amzn-requestid: 26f490cf-81a6-44f9-bf90-f1d269a46502

{"Credentials":{"AccessKeyId":"[redacted]","Expiration":[redacted],"SecretKey":"[redacted]
```

aws cognito-identity get-credentials-for-identity --identity-id {redacted} --region {redacted}

Tools

- enumerate-iam
- cloud-service-enum
- pacu

IAM PrivSec

- Misconfigured Trust Policy
- Pass Role
- Overly Permissive Policy
- Dangerous Policy Combination

Tools

- cloudfox
- purplepanda
- scountsuite
- enumerate-iam
- cloud-service-enum
- pacu

Some Overly IAM Permissive Policy

- iam:PutGroupPolicy
- iam:CreatePolicyVersion
- iam:PutRolePolicy
- iam:SetDefaultPolicyVersion
- iam:PutUserPolicy
- iam:AddUserToGroup
- iam:AttachGroupPolicy
- iam:CreateLoginProfile
- iam:AttachRolePolicy
- iam:UpdateLoginProfile
- iam:AttachUserPolicy
- iam:CreateAccessKey

S3 Bucket

Attack Vectors

- Full anonymous access
- Arbitrary file listing
- Arbitrary file upload and exposure
- Arbitrary read/writes of objects
- Reveals ACP/ACL
- Bucket takeover
- Data exfiltration via abusing S3 replication service (Insufficient logging)
- Abuse RedShift Copy Command

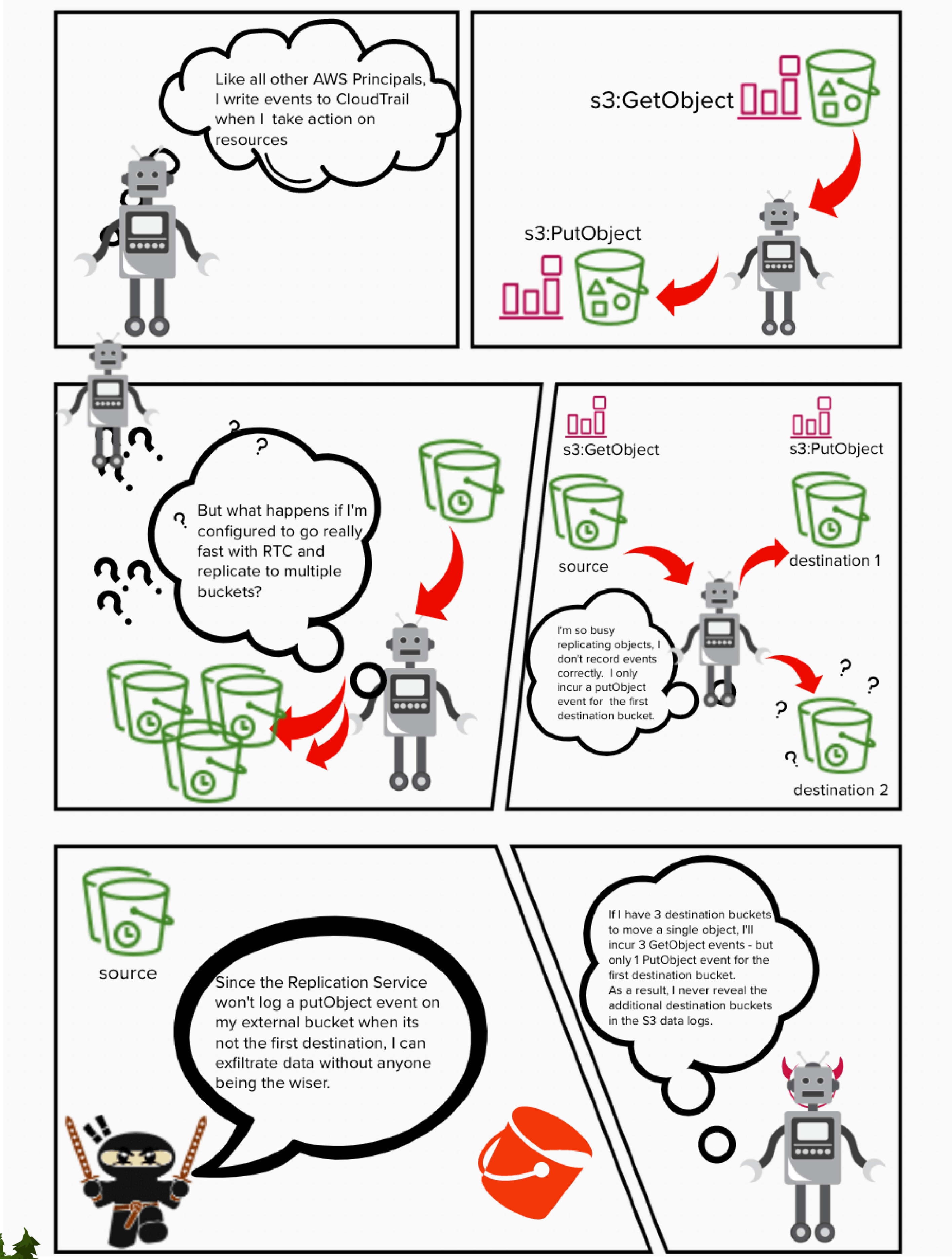
Risks

- Sensitive data (credentials, keys, backup, source code, PII/PHI etc.) disclosure.
- Backdooring / Persistence access via malicious file upload or over-writing file.
- Malware injection through overwriting cloudformation template hosted on S3.

Tools

- red-bucket
- s3enum
- S3Scanner

Abuse S3 Replication Service


Like all other AWS Principals, I write events to CloudTrail when I take action on resources
s3:GetObject
s3:PutObject
But what happens if I'm configured to go really fast with RTC and replicate to multiple buckets?
s3:GetObject
s3:PutObject
source
destination 1
I'm so busy replicating objects, I don't record events correctly. I only incur a putObject event for the first destination bucket.
destination 2
source
Since the Replication Service won't log a putObject event on my external bucket when its not the first destination, I can exfiltrate data without anyone being the wiser.
If I have 3 destination buckets to move a single object, I'll incur 3 GetObject events - but only 1 PutObject event for the the first destination bucket.
As a result, I never reveal the additional destination buckets in the S3 data logs.

S3 Replication Rule - IAM Policy

```
{
 "Version": "2012-10-17",
 "Id": "S3-Console-Replication-Policy",
 "Statement": [
  {
   "Action": [
    "s3:ListGetBucket",
    "s3:GetReplicationConfiguration",
    "s3:GetObjectVersionAcl",
    "s3:Replicate",
   ],
   "Effect": "Allow",
   "Resource": [
    "*"
   ]
  }
 ]
}
```

https://www.vectra.ai/blogpost/abusing-the-replicator-silently-exfiltrating-data-with-the-aws-s3-replication-service

Compliance and Monitoring

- HIPAA
- SOC 2
- CIS and Other Benchmark
- Subdomain Takeover

Tools

- prowler
- scoutsuite
- electriceye
- domain-protect
- pacbot
- cloud custodian
- steampipe

Thank you!