S3 Security:
Attack & Defense

Nitin Sharma
LinkedIn: linkedin.com/in/nitins87
Quora: quora.com/profile/NitinS-1
Blog: 4hathacker.in

Contents

$ whoami

$ Introduction: AWS S3

$ AWS S3: (#Attack)/Breach Scenarios

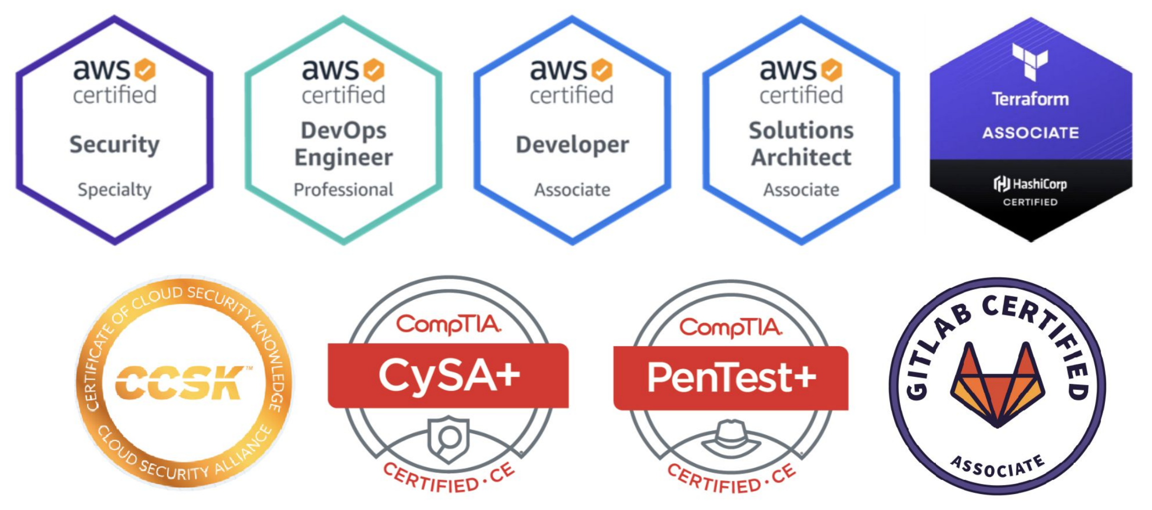
$ Introduction: Threat Modeling

$ Threat Modeling: AWS S3 (#Defense)

$ AWS S3: Security Best Practices

$ whoami

- CyberSecurity and DevSecOps professional experienced in Cloud Security, Container Security and DevOps Research.
- Certifications:

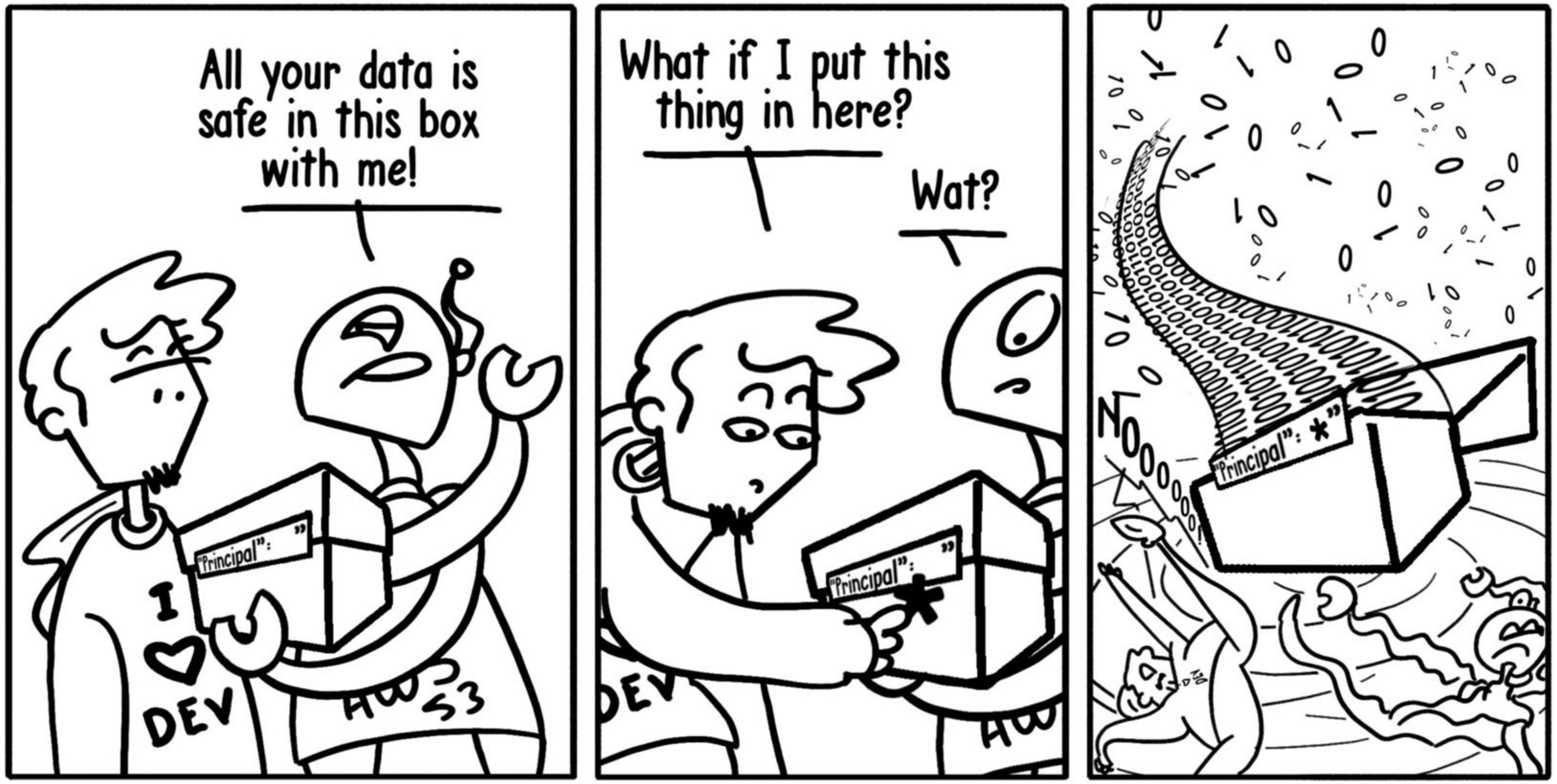


- Published author for "Securing Docker - The Attack & Defense Ways" book under CyberSecrets Publication
- Half Marathon runner, Cyclist and Fitness Enthusiast
- Helping out beginners in Cloud, DevOps and CyberSec at Quora

$ whoami

Finding mistakes in the past…

Actually... some big mistakes!!!



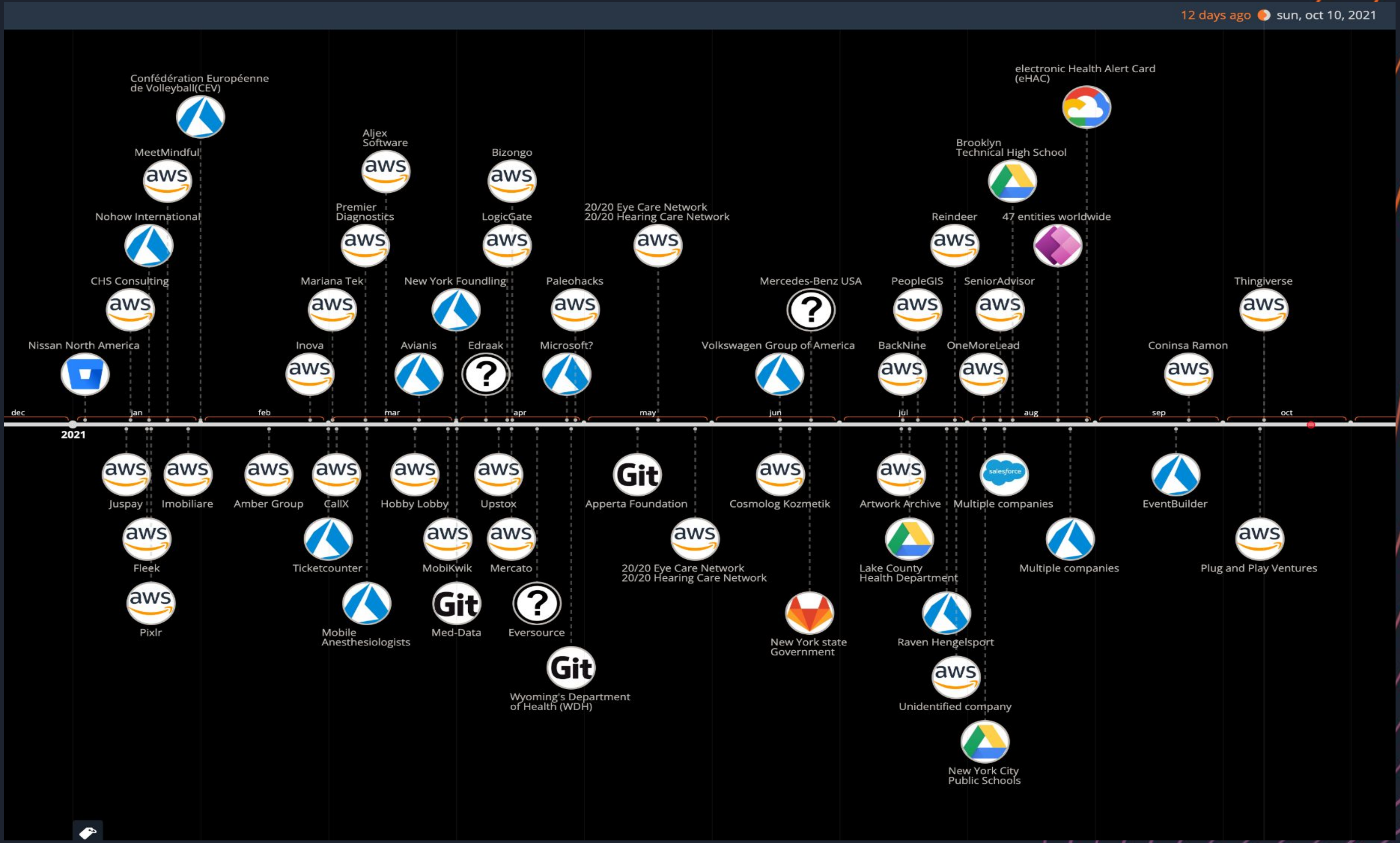
© TotalCloud Inc. | https://TotalCloud.io

$ Introduction: AWS S3

- S3 stands for Simple Storage Service
- S3 Bucket: A container for uploaded data which must have a unique name.
- S3 Object: Item stored within a bucket which is identified by key and version ID.
- S3 Policy: JSON Based Access control statement.
- S3 Bucket ACLs:sub-resource that’s attached to every S3 bucket and object. It defines which AWS accounts or groups are granted access and the type of access. When you create a bucket or an object, Amazon S3 creates a default ACL that grants the resource owner full control over the resource.

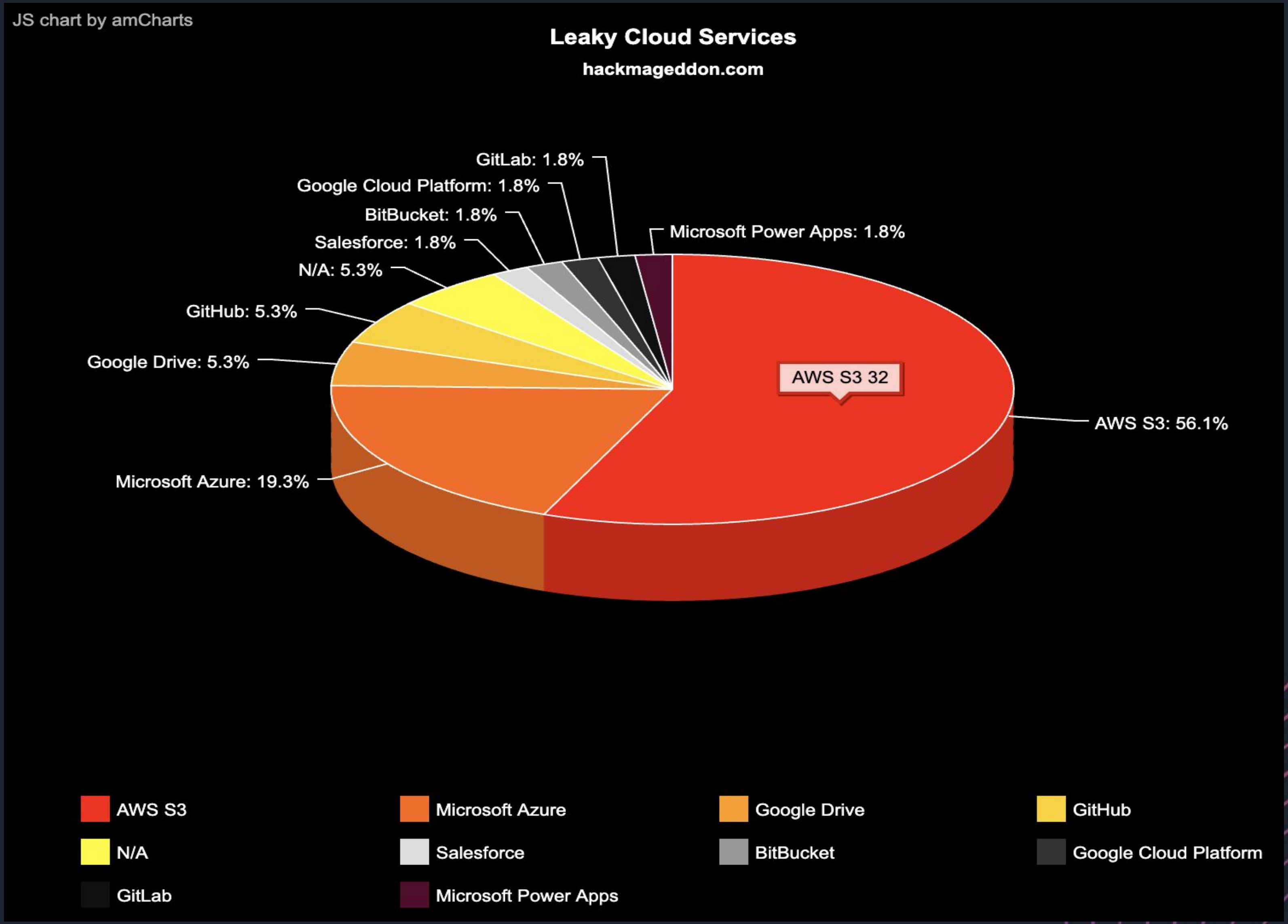
aws COMMUNITY DAY SOUTH ASIA VIRTUAL EDITION 2021

$ 2021: Leaky Bucket Breaches



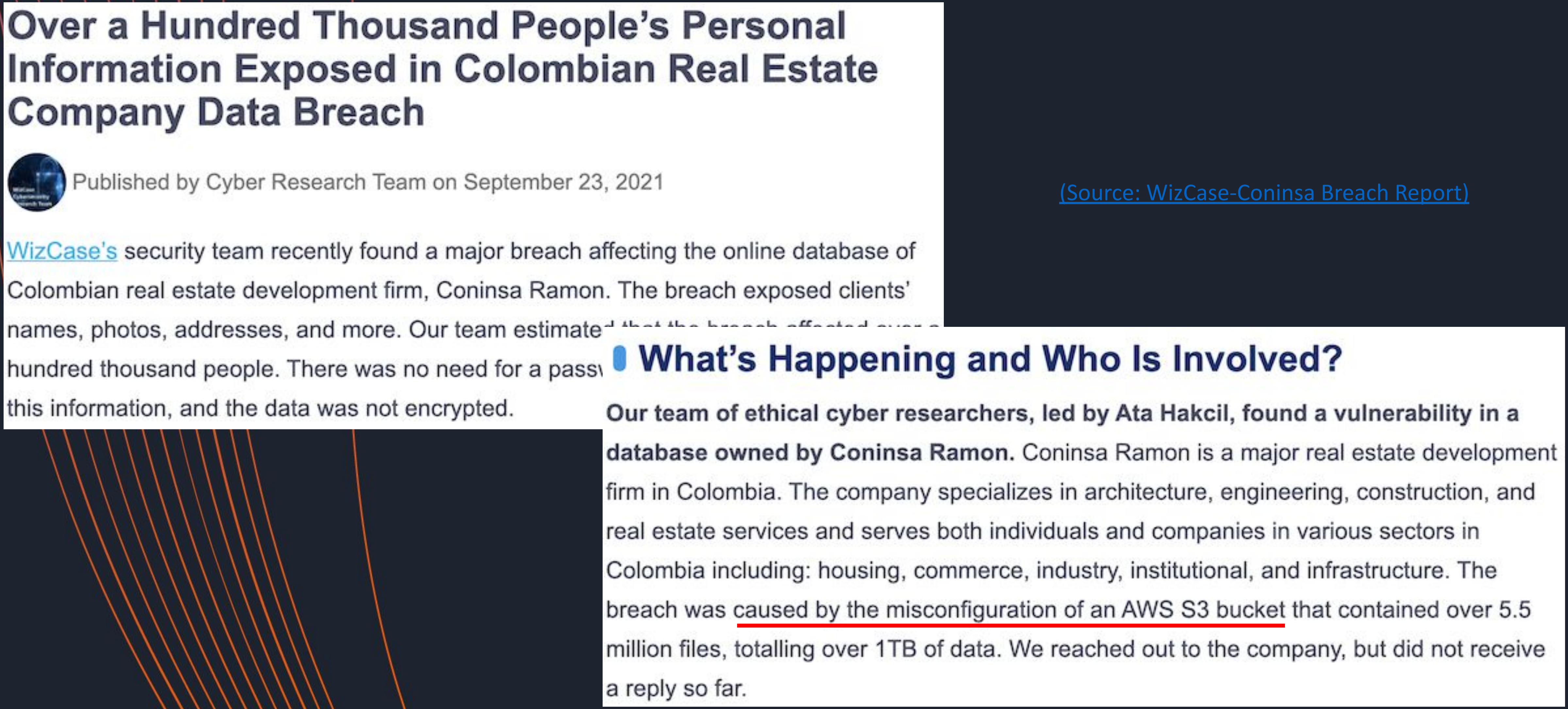
(Source: Hackmageddon)

$ 2021: AWS S3 Breaches



$ AWS S3 Attack/Breach Scenarios

➔ Scenario 1: Misconfigured S3 Bucket with Customer Data



Over a Hundred Thousand People's Personal Information Exposed in Colombian Real Estate Company Data Breach

Published by Cyber Research Team on September 23, 2021

WizCase's security team recently found a major breach affecting the online database of Colombian real estate development firm, Coninsa Ramon. The breach exposed clients' names, photos, addresses, and more. Our team estimate hundred thousand people. There was no need for a pass this information, and the data was not encrypted.

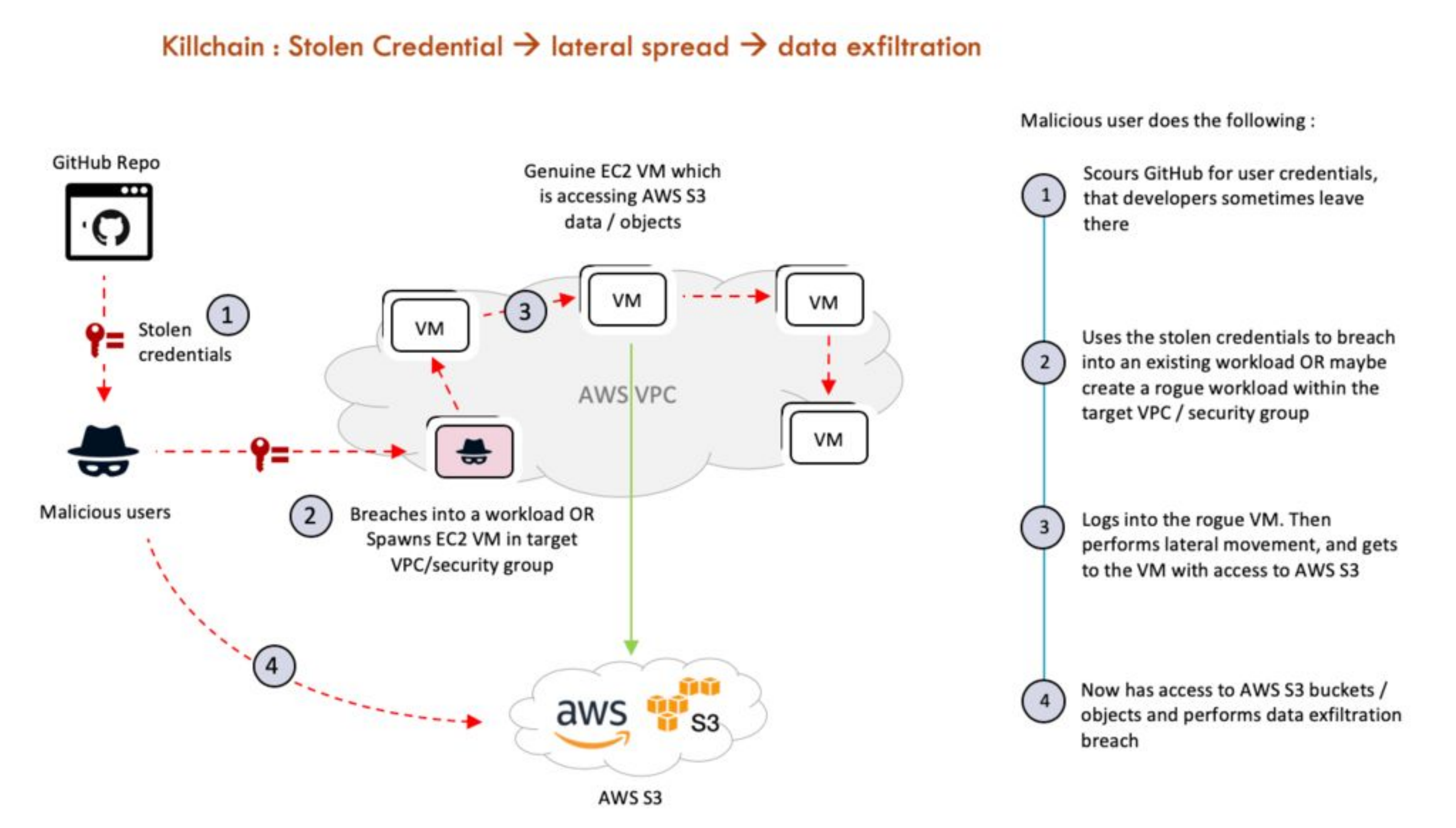
What's Happening and Who Is Involved?

Our team of ethical cyber researchers, led by Ata Hakcil, found a vulnerability in a database owned by Coninsa Ramon. Coninsa Ramon is a major real estate development firm in Colombia. The company specializes in architecture, engineering, construction, and real estate services and serves both individuals and companies in various sectors in Colombia including: housing, commerce, industry, institutional, and infrastructure. The breach was caused by the misconfiguration of an AWS S3 bucket that contained over 5.5 million files, totalling over 1TB of data. We reached out to the company, but did not receive a reply so far.

(Source: WizCase-Coninsa Breach Report)

$ AWS S3 Attack/Breach Scenarios

- ➔ Scenario 2:GitHub code committed with AWS Access Keys associated with S3 permissions to Customer Bucket



Killchain : Stolen Credential → lateral spread → data exfiltration

GitHub Repo
1
Stolen credentials
Malicious users
2
Breaches into a workload OR Spawns EC2 VM in target VPC/security group
Genuine EC2 VM which is accessing AWS S3 data / objects
VM
3
VM
VM
VM
AWS VPC
4
aws
S3
AWS S3

Malicious user does the following :

1 Scours GitHub for user credentials, that developers sometimes leave there

2 Uses the stolen credentials to breach into an existing workload OR maybe create a rogue workload within the target VPC / security group

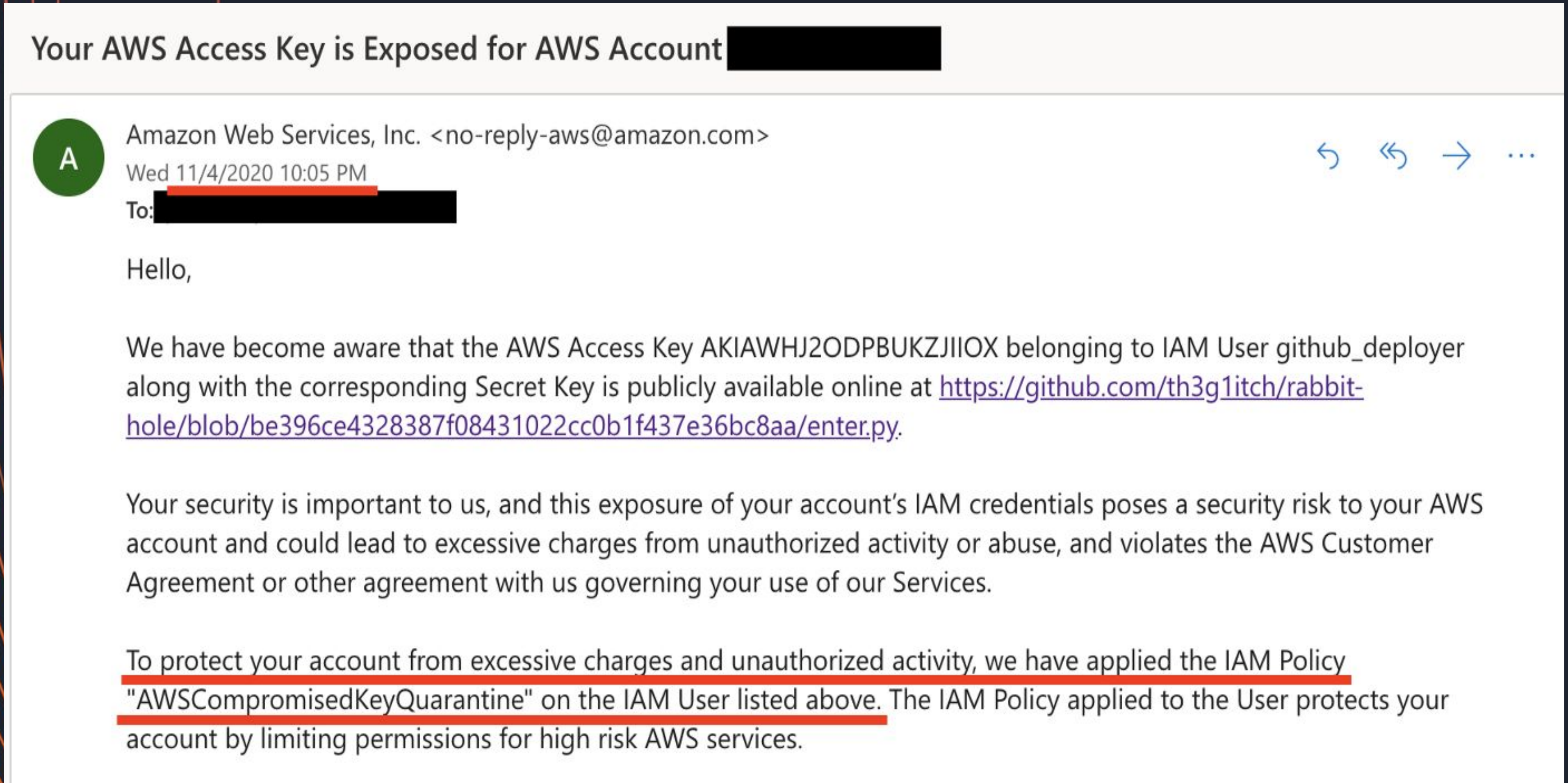
3 Logs into the rogue VM. Then performs lateral movement, and gets to the VM with access to AWS S3

4 Now has access to AWS S3 buckets / objects and performs data exfiltration breach

(Source: Mesh7 Blog)

$ AWS S3 Attack/Breach Scenarios

➔ Scenario 2:GitHub code committed with AWS Access Keys associated with S3 permissions to Customer Bucket



Your AWS Access Key is Exposed for AWS Account [redacted]

Amazon Web Services, Inc. <no-reply-aws@amazon.com>
Wed 11/4/2020 10:05 PM
To: [redacted]

Hello,

We have become aware that the AWS Access Key AKIAWHJ2ODPBUKZJIIOX belonging to IAM User github_deployer along with the corresponding Secret Key is publicly available online at https://github.com/th3g1itch/rabbit-hole/blob/be396ce4328387f08431022cc0b1f437e36bc8aa/enter.py.

Your security is important to us, and this exposure of your account's IAM credentials poses a security risk to your AWS account and could lead to excessive charges from unauthorized activity or abuse, and violates the AWS Customer Agreement or other agreement with us governing your use of our Services.

To protect your account from excessive charges and unauthorized activity, we have applied the IAM Policy "AWSCompromisedKeyQuarantine" on the IAM User listed above. The IAM Policy applied to the User protects your account by limiting permissions for high risk AWS services.

(Source: Pawel Rzepa Medium Blog)

$ AWS S3 Attack/Breach Scenarios

➔ Scenario 3:Attackers encrypting S3 buckets in victim AWS account with their own KMS Key (S3 Ransomwar Scenario)

- Attacker creates a KMS key in their own "personal" AWS account (or another compromised account) and provides "the world" access to use that KMS key for encryption.
- Attacker identifies a target S3 bucket and gains write level access to it.
- Attacker checks the configuration of the bucket to determine if it is able to be targeted by ransomware.
- Attacker uses the AWS API to replace each object in a bucket with a new copy of itself (encrypted with KMS key)
- Attacker schedules the deletion of the KMS key
- Attacker uploads a final file such as "ransom-note.txt" without encryption

(Source: Rhino Security Labs)

$ AWS S3 Attack/Breach Scenarios

➔ Scenario Nth:

There could be ‘N’ different such scenarios to understand and then tackle the AWS S3 Attackers.

Hence, an efficient way to address the attacker concerns is needed !

Threat Modeling...

$ Introduction: Threat Modeling

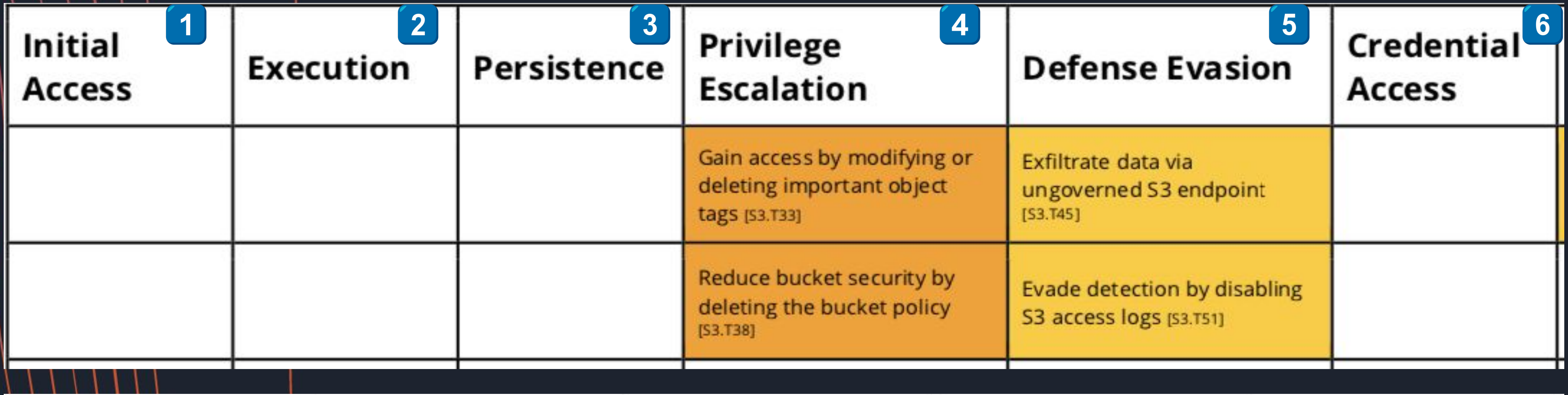
aws COMMUNITY DAY SOUTH ASIA VIRTUAL EDITION 2021

- ➔ **Threat**: Any circumstance or event with the potential to harm an information system through unauthorized access, destruction, disclosure, modification of data, and/or denial of service.

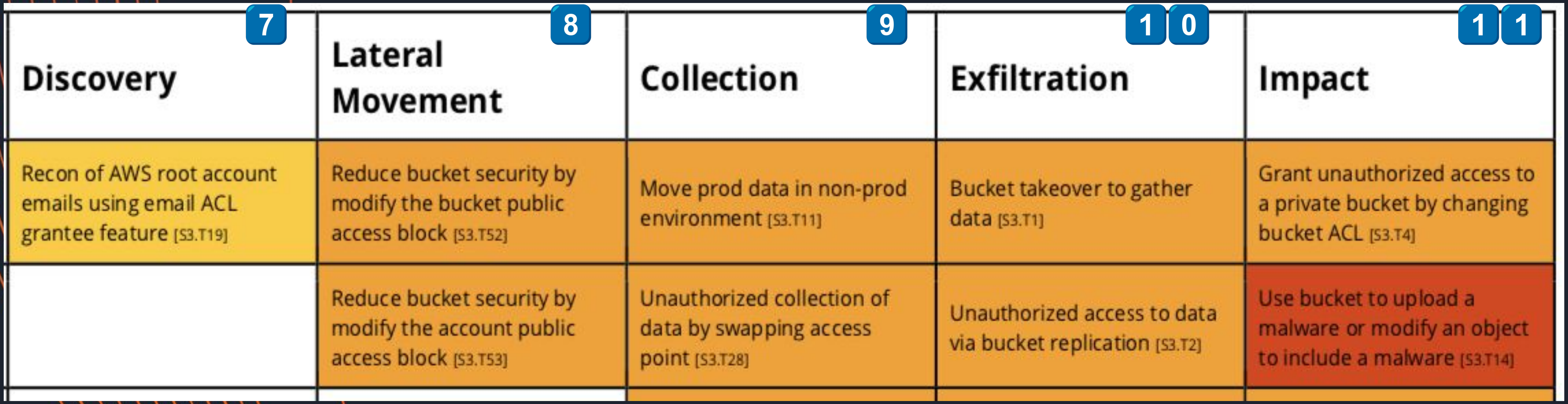
- ➔ Core steps for threat modeling[**Design Time Activity**]:

1. Identify assets, actors, entry points, components, use cases, and trust levels, and include these in a design diagram.
2. Identify a list of threats.
3. Per threat, identify mitigations, which may include security control implementations.
4. Create and review a risk matrix to determine if the threat is adequately mitigated.

$ MITRE ATT&CK MATRIX FOR AWS S3



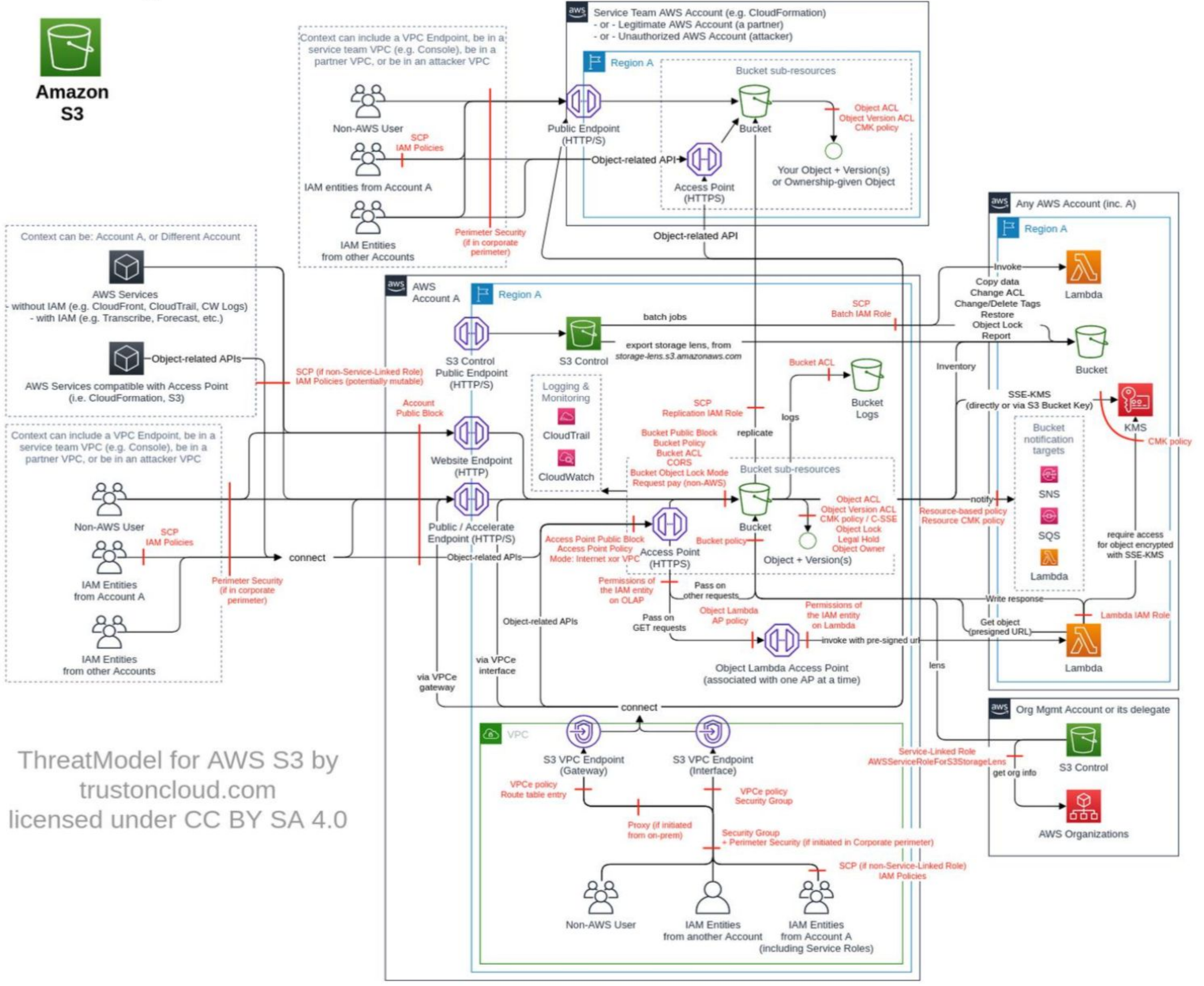
Initial Access 1	Execution 2	Persistence 3	Privilege Escalation 4	Defense Evasion 5	Credential Access 6
			Gain access by modifying or deleting important object tags [S3.T33]	Exfiltrate data via ungoverned S3 endpoint [S3.T45]	
			Reduce bucket security by deleting the bucket policy [S3.T38]	Evade detection by disabling S3 access logs [S3.T51]	



Discovery 7	Lateral Movement 8	Collection 9	Exfiltration 10	Impact 11
Recon of AWS root account emails using email ACL grantee feature [S3.T19]	Reduce bucket security by modify the bucket public access block [S3.T52]	Move prod data in non-prod environment [S3.T11]	Bucket takeover to gather data [S3.T1]	Grant unauthorized access to a private bucket by changing bucket ACL [S3.T4]
	Reduce bucket security by modify the account public access block [S3.T53]	Unauthorized collection of data by swapping access point [S3.T28]	Unauthorized access to data via bucket replication [S3.T2]	Use bucket to upload a malware or modify an object to include a malware [S3.T14]

$ Threat Modeling: AWS S3

Data Flow Diagram


Amazon S3
Service Team AWS Account (e.g. CloudFormation)
- or - Legitimate AWS Account (a partner)
- or - Unauthorized AWS Account (attacker)
Region A
Bucket sub-resources
Context can include a VPC Endpoint, be in a service team VPC (e.g. Console), be in a partner VPC, or be in an attacker VPC
Non-AWS User
SCP
IAM Policies
IAM entities from Account A
IAM Entities from other Accounts
Perimeter Security (if in corporate perimeter)
Public Endpoint (HTTP/S)
Bucket
Object ACL
Object Version ACL
CMK policy
Your Object + Version(s) or Ownership-given Object
Object-related API
Access Point (HTTPS)
Object-related API
Context can be: Account A, or Different Account
AWS Services
- without IAM (e.g. CloudFront, CloudTrail, CW Logs)
- with IAM (e.g. Transcribe, Forecast, etc.)
Object-related APIs
AWS Services compatible with Access Point (i.e. CloudFormation, S3)
SCP (if non-Service-Linked Role)
IAM Policies (potentially mutable)
AWS Account A
Region A
S3 Control Public Endpoint (HTTP/S)
S3 Control
batch jobs
export storage lens, from storage-lens.s3.amazonaws.com
SCP
Batch IAM Role
Account Public Block
Logging & Monitoring
CloudTrail
CloudWatch
Website Endpoint (HTTP)
Bucket ACL
Bucket Logs
logs
SCP
Replication IAM Role
replicate
Bucket Public Block
Bucket Policy
Bucket ACL
CORS
Bucket Object Lock Mode
Request pay (non-AWS)
Bucket sub-resources
Object ACL
Object Version ACL
CMK policy / C-SSE
Object Lock
Legal Hold
Object Owner
Bucket policy
Object + Version(s)
Context can include a VPC Endpoint, be in a service team VPC (e.g. Console), be in a partner VPC, or be in an attacker VPC
Non-AWS User
SCP
IAM Policies
IAM Entities from Account A
IAM Entities from other Accounts
Perimeter Security (if in corporate perimeter)
connect
Public / Accelerate Endpoint (HTTP/S)
Object-related APIs
Access Point Public Block
Access Point Policy
Mode: Internet xor VPC
Access Point (HTTPS)
Permissions of the IAM entity on OLAP
Pass on other requests
Pass on GET requests
Object Lambda AP policy
Permissions of the IAM entity on Lambda
invoke with pre-signed url
Object Lambda Access Point (associated with one AP at a time)
Object-related APIs
via VPCe gateway
via VPCe interface
connect
VPC
S3 VPC Endpoint (Gateway)
S3 VPC Endpoint (Interface)
VPCe policy
Route table entry
VPCe policy
Security Group
Proxy (if initiated from on-prem)
Security Group
+ Perimeter Security (if initiated in Corporate perimeter)
SCP (if non-Service-Linked Role)
IAM Policies
Non-AWS User
IAM Entities from another Account
IAM Entities from Account A (including Service Roles)
Any AWS Account (inc. A)
Region A
Invoke
Lambda
Copy data
Change ACL
Change/Delete Tags
Restore
Object Lock
Report
Bucket
Inventory
SSE-KMS (directly or via S3 Bucket Key)
KMS
CMK policy
Bucket notification targets
SNS
SQS
Lambda
notify
Resource-based policy
Resource CMK policy
require access for object encrypted with SSE-KMS
Write response
Lambda IAM Role
Get object (presigned URL)
Lambda
lens
Org Mgmt Account or its delegate
S3 Control
Service-Linked Role
AWSServiceRoleForS3StorageLens
get org info
AWS Organizations

$ Threat Modeling: AWS S3

→ Pick one Feature of S3 (Involved in your DFD)

→ Identify Associated IAM/Bucket Actions

→ Create Threat List & Map it to MITRE ATT&CK

→ Set Security Control Objectives and Priorities

→ Testing & Validation (#Defend)

$ Threat Modeling: AWS S3

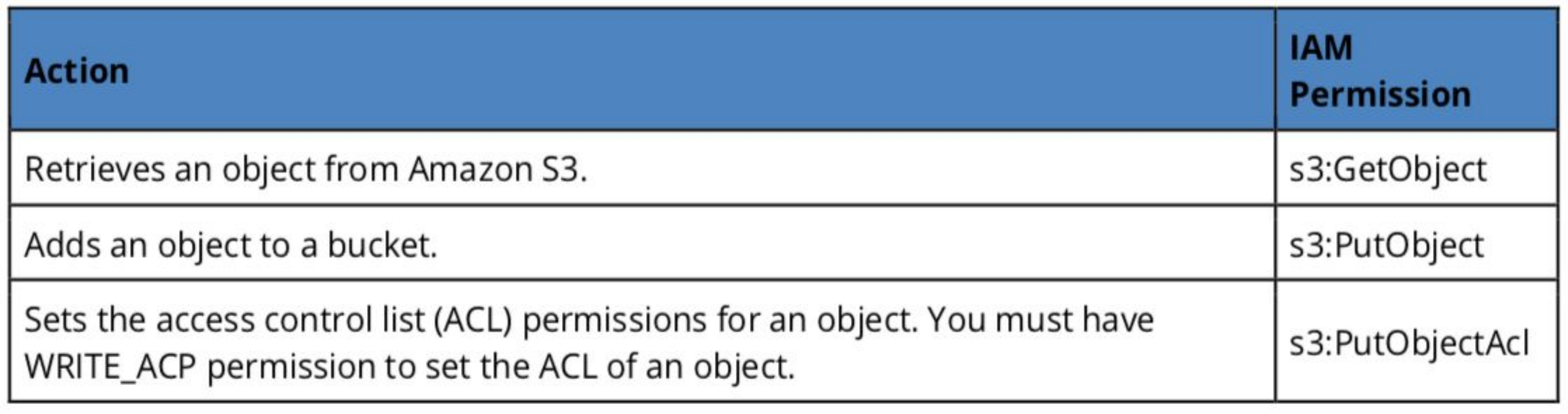
→ Pick one Feature of S3 (Involved in your DFD)

Object Upload/Download

You can upload and download virtually any number of objects to an external S3 bucket you are authorized to.

$ Threat Modeling: AWS S3

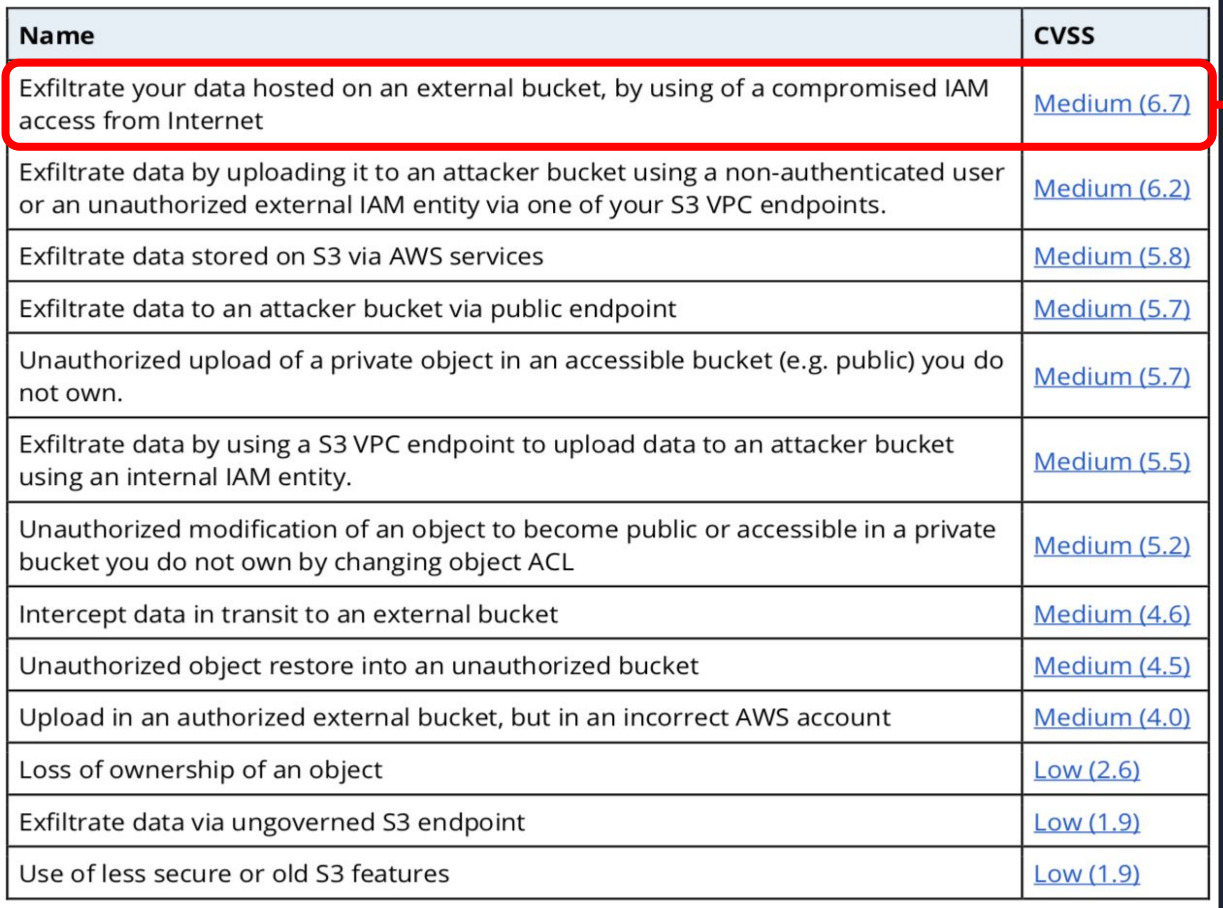
→ Identify Associated IAM/Bucket Actions



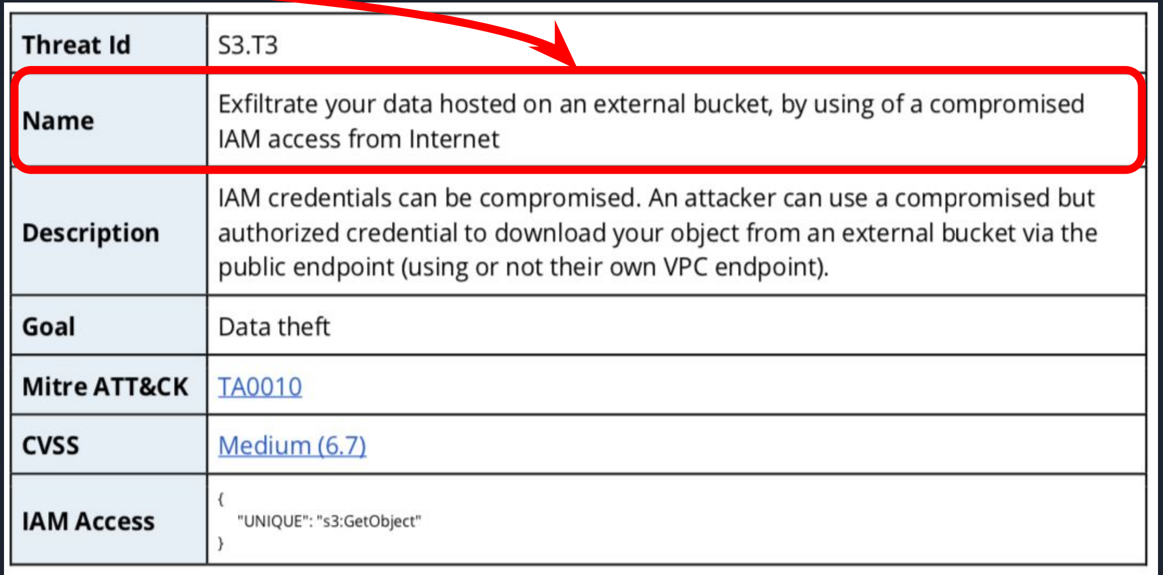
Action	IAM Permission
Retrieves an object from Amazon S3.	s3:GetObject
Adds an object to a bucket.	s3:PutObject
Sets the access control list (ACL) permissions for an object. You must have WRITE_ACP permission to set the ACL of an object.	s3:PutObjectAcl

$ Threat Modeling: AWS S3

→ Create Threat List & Map it to MITRE ATT&CK



Name	CVSS
Exfiltrate your data hosted on an external bucket, by using of a compromised IAM access from Internet	Medium (6.7)
Exfiltrate data by uploading it to an attacker bucket using a non-authenticated user or an unauthorized external IAM entity via one of your S3 VPC endpoints.	Medium (6.2)
Exfiltrate data stored on S3 via AWS services	Medium (5.8)
Exfiltrate data to an attacker bucket via public endpoint	Medium (5.7)
Unauthorized upload of a private object in an accessible bucket (e.g. public) you do not own.	Medium (5.7)
Exfiltrate data by using a S3 VPC endpoint to upload data to an attacker bucket using an internal IAM entity.	Medium (5.5)
Unauthorized modification of an object to become public or accessible in a private bucket you do not own by changing object ACL	Medium (5.2)
Intercept data in transit to an external bucket	Medium (4.6)
Unauthorized object restore into an unauthorized bucket	Medium (4.5)
Upload in an authorized external bucket, but in an incorrect AWS account	Medium (4.0)
Loss of ownership of an object	Low (2.6)
Exfiltrate data via ungoverned S3 endpoint	Low (1.9)
Use of less secure or old S3 features	Low (1.9)

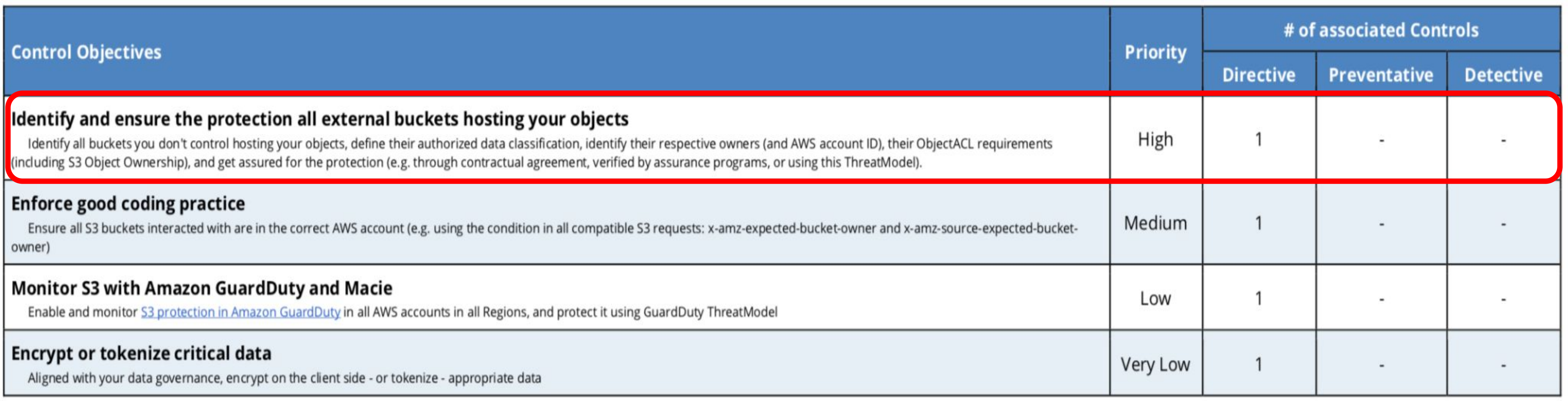


Threat Id	S3.T3
Name	Exfiltrate your data hosted on an external bucket, by using of a compromised IAM access from Internet
Description	IAM credentials can be compromised. An attacker can use a compromised but authorized credential to download your object from an external bucket via the public endpoint (using or not their own VPC endpoint).
Goal	Data theft
Mitre ATT&CK	TA0010
CVSS	Medium (6.7)

IAM Access	{
	"UNIQUE": "s3:GetObject"
	}

$ Threat Modeling: AWS S3

→ Set Security Control Objectives and Priorities

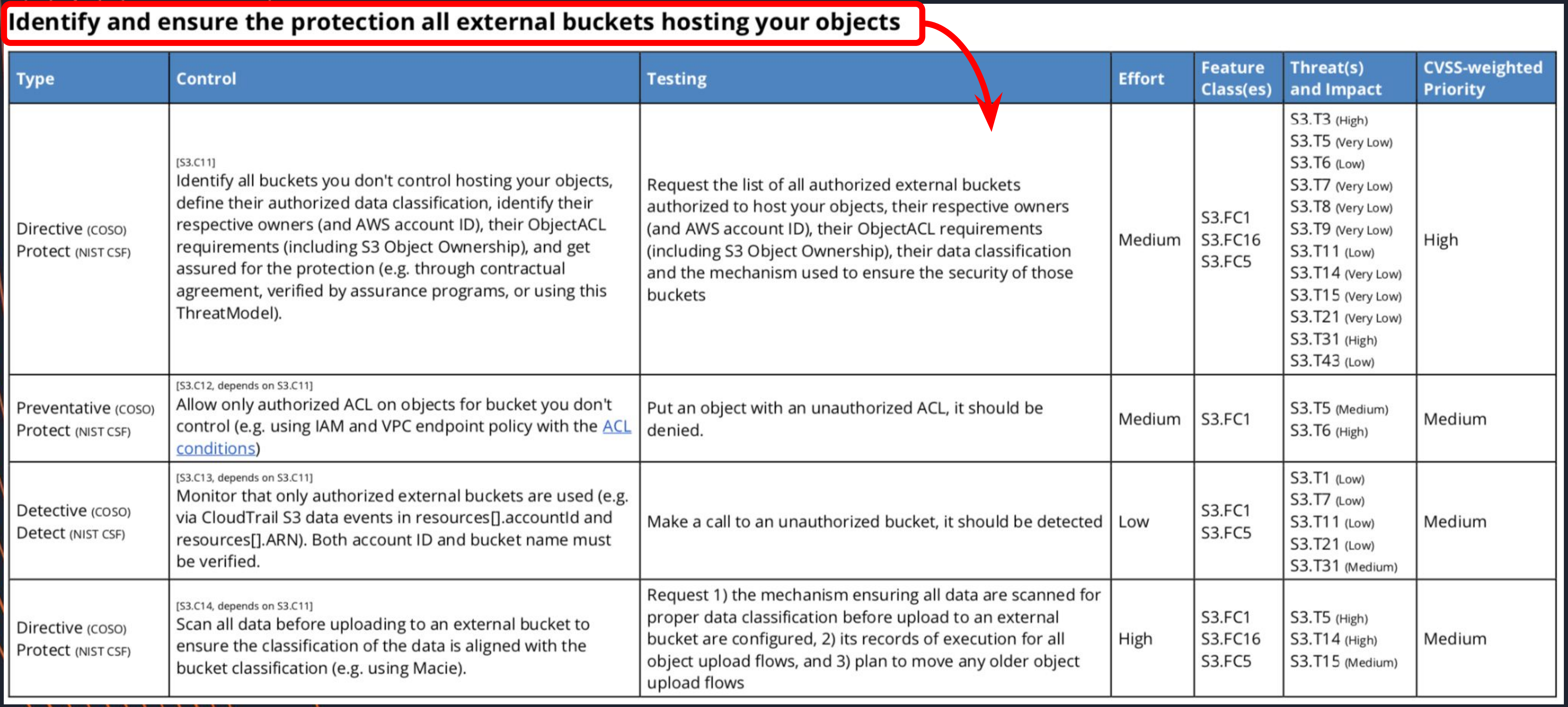


Control Objectives	Priority	# of associated Controls		
		Directive	Preventative	Detective
Identify and ensure the protection all external buckets hosting your objects Identify all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured for the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).	High	1	-	-
Enforce good coding practice Ensure all S3 buckets interacted with are in the correct AWS account (e.g. using the condition in all compatible S3 requests: x-amz-expected-bucket-owner and x-amz-source-expected-bucket-owner)	Medium	1	-	-
Monitor S3 with Amazon GuardDuty and Macie Enable and monitor S3 protection in Amazon GuardDuty in all AWS accounts in all Regions, and protect it using GuardDuty ThreatModel	Low	1	-	-
Encrypt or tokenize critical data Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data	Very Low	1	-	-

$ Threat Modeling: AWS S3

aws
COMMUNITY DAY
SOUTH ASIA
VIRTUAL EDITION 2021

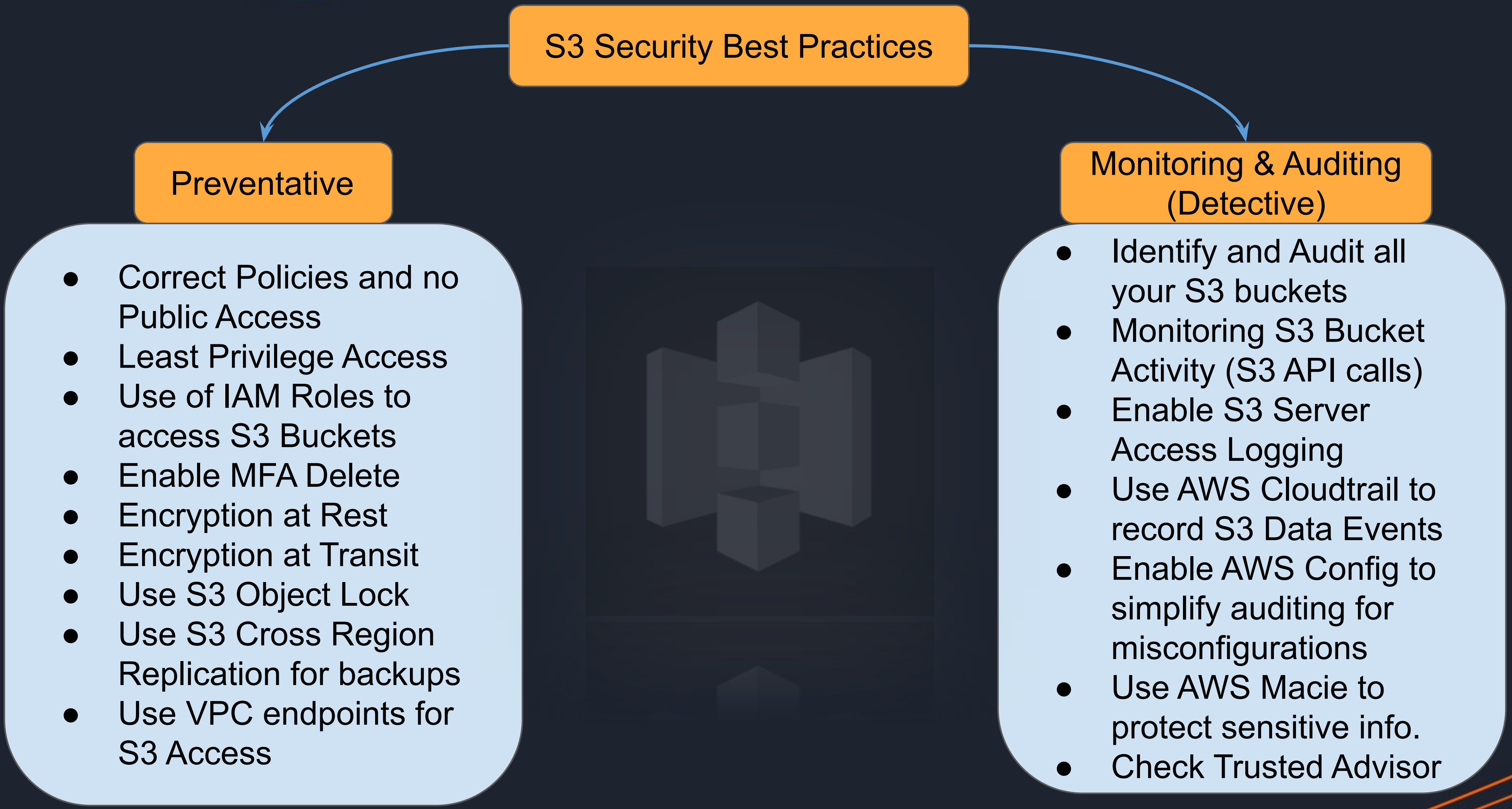
→ Testing & Validation (#Defend)



Identify and ensure the protection all external buckets hosting your objects

Type	Control	Testing	Effort	Feature Class(es)	Threat(s) and Impact	CVSS-weighted Priority
Directive (COSO) Protect (NIST CSF)	[S3.C11] Identify all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured for the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).	Request the list of all authorized external buckets authorized to host your objects, their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), their data classification and the mechanism used to ensure the security of those buckets	Medium	S3.FC1 S3.FC16 S3.FC5	S3.T3 (High) S3.T5 (Very Low) S3.T6 (Low) S3.T7 (Very Low) S3.T8 (Very Low) S3.T9 (Very Low) S3.T11 (Low) S3.T14 (Very Low) S3.T15 (Very Low) S3.T21 (Very Low) S3.T31 (High) S3.T43 (Low)	High
Preventative (COSO) Protect (NIST CSF)	[S3.C12, depends on S3.C11] Allow only authorized ACL on objects for bucket you don't control (e.g. using IAM and VPC endpoint policy with the ACL conditions)	Put an object with an unauthorized ACL, it should be denied.	Medium	S3.FC1	S3.T5 (Medium) S3.T6 (High)	Medium
Detective (COSO) Detect (NIST CSF)	[S3.C13, depends on S3.C11] Monitor that only authorized external buckets are used (e.g. via CloudTrail S3 data events in resources[].accountId and resources[].ARN). Both account ID and bucket name must be verified.	Make a call to an unauthorized bucket, it should be detected	Low	S3.FC1 S3.FC5	S3.T1 (Low) S3.T7 (Low) S3.T11 (Low) S3.T21 (Low) S3.T31 (Medium)	Medium
Directive (COSO) Protect (NIST CSF)	[S3.C14, depends on S3.C11] Scan all data before uploading to an external bucket to ensure the classification of the data is aligned with the bucket classification (e.g. using Macie).	Request 1) the mechanism ensuring all data are scanned for proper data classification before upload to an external bucket are configured, 2) its records of execution for all object upload flows, and 3) plan to move any older object upload flows	High	S3.FC1 S3.FC16 S3.FC5	S3.T5 (High) S3.T14 (High) S3.T15 (Medium)	Medium

$ AWS S3 Security Best Practices



S3 Security Best Practices

Preventative

- Correct Policies and no Public Access
- Least Privilege Access
- Use of IAM Roles to access S3 Buckets
- Enable MFA Delete
- Encryption at Rest
- Encryption at Transit
- Use S3 Object Lock
- Use S3 Cross Region Replication for backups
- Use VPC endpoints for S3 Access

Monitoring & Auditing (Detective)

- Identify and Audit all your S3 buckets
- Monitoring S3 Bucket Activity (S3 API calls)
- Enable S3 Server Access Logging
- Use AWS Cloudtrail to record S3 Data Events
- Enable AWS Config to simplify auditing for misconfigurations
- Use AWS Macie to protect sensitive info.
- Check Trusted Advisor

aws
User Group
Member

AWS
community
builders

Thank You...