External Attack Surface for Initial Access in AWS Cloud

© All Rights Reserved CyberwarFare Labs

❖ CyberWarFare Labs

CW Labs is a renowned UK based Ed-tech company specializing in cybersecurity cyber range labs. They provide on-demand educational services and recognize the need for continuous adaptation to evolving threats client requirements.

The company has two primary divisions :

1. **Cyber Range Labs**
2. **Up-Skilling Platform**

© All Rights Reserved CyberwarFare Labs

About Speaker:

Parth Agrawal
(Security Intern @CWL)

Is a cloud security enthusiast with a keen interest in the intricacies of cloud services offered by AWS, Azure, and GCP. Possessing a comprehensive understanding of these platforms, they are particularly drawn to exploring Red Team methodologies. Interested in Red Team methodologies, focusing on vulnerability testing and detection across external attack surfaces.

© All Rights Reserved CyberwarFare Labs

Table of Contents

© All Rights Reserved CyberwarFare Labs

AWS Services

EC2 | S3 | ELB | RDS | EBS
Lambda | CloudFront | Cognito

© All Rights Reserved CyberwarFare Labs

Elastic Compute Cloud (EC2)

- EC2 provides globally distributed virtual machines known as Instances.
- These Instances are divided into different categories which are targeted towards specific workloads.
- In EC2, an Operating System (OS) is known as Amazon Machine Image (AMI).

© All Rights Reserved CyberwarFare Labs

Simple Storage Service (S3)

- S3 provides scalable object storage service.
- S3 is divided into multiple storage classes tailored for particular use cases.
- The data objects are stored inside storage containers dubbed as buckets.
- These bucket names are globally unique across all S3 buckets in a given partition (Standard Regions, China Regions, and AWS GovCloud-US).

© All Rights Reserved CyberwarFare Labs

Elastic Block Store (EBS)

- AWS EBS (Elastic Block Store) is a block storage service provided by Amazon Web Services (AWS) for use with Amazon Elastic Compute Cloud (EC2) instances.
- Key features of AWS EBS include:
  - Block Storage
  - Elasticity
  - Performance
  - Snapshots and Backups
  - Data Encryption
  - High Availability
  - Integration with AWS Services

© All Rights Reserved CyberwarFare Labs

Relational Database Service (RDS)

- AWS RDS (Relational Database Service) is a managed database service provided by Amazon Web Services (AWS).
- RDS supports various database engines, including MySQL, PostgreSQL, MariaDB, Oracle, SQL Server, and Amazon Aurora.
- Key features of AWS RDS include:
  - Automated Backups
  - Automated Scaling
  - High Availability
  - Security
  - Monitoring and Metrics
  - Database Engine Options

© All Rights Reserved CyberwarFare Labs

Elastic Load Balancing (ELB)

- ELB is load balancing service which scales resources and distributes traffic among different targets (Instances, IPs, Containers) in one or more Availability Zones.
- ELB provides following offerings for different use cases:-
  - Application Load Balancers: Operates on layer 7 with support for protocols like HTTP(s) & WebSockets
  - Classic Load Balancer: Operates on layer 4/7 while supporting HTTP(s), TCP, & SSL/TLS
  - Gateway Load Balancers: Operates on layer 3 while supporting IP protocol
  - Network Load Balancers: Operates on layer 4 with support for TCP & UDP

© All Rights Reserved CyberwarFare Labs

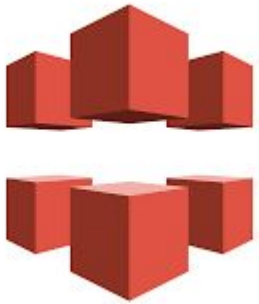
Cognito

- Cognito is a Customer Identity and Access Management service.
- It provides scalable user identity management with integrations with AWS services & custom solutions.
- Cognito is consist of two main components as follows:
  - User Pool: Directory of user information
  - Identity Pool: Access manager for different AWS resource access

© All Rights Reserved CyberwarFare Labs

CloudFront

- CloudFront is a globally distributed, fast & secure Content Delivery Network (CDN).
- It is commonly used to serve content (static/dynamic) from a location near to visitor.
- CloudFront provides seamless integration with AWS services & custom applications.
- It can be used to hide origin information.



© All Rights Reserved CyberwarFare Labs

Lambda

- AWS Lambda is a serverless computing service provided by Amazon Web Services (AWS). It allows you to run code without provisioning or managing servers.
- With Lambda, you can execute code in response to events triggered by other AWS services, HTTP requests via Amazon API Gateway, or custom events.
- Key aspects of AWS Lambda include:
  - Event-driven
  - Serverless
  - Supported Runtimes
  - Pay-per-use
  - Scalability
  - Integration

© All Rights Reserved CyberwarFare Labs

Public URLs

For Available Services

© All Rights Reserved CyberwarFare Labs

Services	Sample Public URL
EC2	http://<ec2-public-ip>
S3	https://<s3-bucket-name>.s3.amazonaws.com/<object-key> OR http://[bucketname].s3-website-[region].amazonaws.com/
CloudFront	https://<random_id>.cloudfront.net
SSO	https://<CompanyName>.awsapps.com/

© All Rights Reserved CyberwarFare Labs

Scenario 1: OSINT

RECON

© All Rights Reserved CyberwarFare Labs

EC2 Recon

Recon via Shodan:

```
org:Amazon ssl.cert.subject.cn:<ORGANIZATION_NAME>
```

```
net:<NETBLOCK_RANGE> ssl.cert.subject.cn:<ORGANIZATION_NAME>
```

```
org:Amazon hostname:<ORGANIZATION_NAME>
```

© All Rights Reserved CyberwarFare Labs

EC2 Recon

Recon via fofa:

```
body="AWS" && protocol="SSH" || protocol="RDP" && region="aws_region"
                    && port="22" || port="3389"
```

```
app="AWS" && protocol="HTTPS" && title="SSL Certificate" &&
                    region="aws_region"
```

```
app="AWS" && protocol="HTTP" && title="Apache" && region="aws_region"
```

© All Rights Reserved CyberwarFare Labs

S3 Recon

Recon via Shodan:

```
title:"AWS S3 Explorer"
```

```
http.title:"Amazon Cognito Developer Authentication Sample"
```

Recon via fofa:

```
app="amazon-AmazonS3"
```

```
body="ListBucketResult"
```

© All Rights Reserved CyberwarFare Labs

S3 Recon

CLI-based Recon:

- Cloud Enum:

```
./cloud_enum.py -k <KEYWORD> --disable-azure --disable-gcp
```

- S3 Scanner:

```
./s3scanner -bucket <KEYWORD> -enumerate -json
```

- BucketLoot:

```
./bucketloot <Target_URL>
```

© All Rights Reserved CyberwarFare Labs

S3 Recon

Web-based Recon:

- Bucket search:
  - https://osint.sh/buckets
  - https://buckets.grayhatwarfare.com
  - https://builtwith.com/
  - https://s3browser.com/
- Dorks:
  - GitHub Dorks:

```
"s3.amazonaws.com" org:<ORGANIZATION_NAME>
```

© All Rights Reserved CyberwarFare Labs

S3 Recon

Web-based Recon:

- Dorks:
  - More Google Dorks:

```
site:.s3.amazonaws.com "Company"
```

```
site:http://s3.amazonaws.com intitle:index.of.bucket “”
```

```
site:s3.amazonaws.com "index of /" s3
```

```
site:amazonaws.com filetype:xls password
```

```
site:*.s3.amazonaws.com ext:xls | ext:xlsx | ext:csv
password|passwd|pass user|username|uid|email
```

© All Rights Reserved CyberwarFare Labs

EBS Recon

Recon via fofa:

```
app="AWS" && body="ebs" && body="size:100" && region="aws_region"
```

```
app="AWS" && body="ebs" && body="volumeType:gp2" && region="aws_region"
```

```
app="AWS" && body="snapshot" && body="public:true" && region="aws_region"
```

```
app="AWS" && body="EC2" && region="aws_region"
```

```
app="AWS" && body="blockDeviceMapping"
```

© All Rights Reserved CyberwarFare Labs

EBS Recon

CLI-based Recon:

- Cloud Enum:

```
./cloud_enum.py -k ebs --disable-azure --disable-gcp
```

Web-based Recon:

- Dork:

```
"Elastic Block Store" OR "AWS EBS" site:github.com OR
    site:stackoverflow.com OR site:aws.amazon.com
```

© All Rights Reserved CyberwarFare Labs

RDS Recon

Recon via Shodan:

```
product:"postgreSQL" port:5432 org:"Amazon.com"
```

- PostgreSQL → 5432, MySQL → 3305, SQL Server → 1433

Recon via fofa:

```
app="AWS" && body="RDS" && region="aws_region"
```

```
app="AWS" && body="RDS" && body="MySQL" && region="aws_region"
```

```
app="AWS" && body="RDS" && body="public" && region="aws_region"
```

© All Rights Reserved CyberwarFare Labs

RDS Recon

Web-based Recon:

- Dorks:

```
"Relational Database Service" OR "AWS RDS" site:github.com OR
        site:stackoverflow.com OR site:aws.amazon.com
```

```
site:pastebin.com "rds.amazonaws.com" "u " pass OR password
```

CLI-based Recon:

- Cloud Enum:

```
./cloud_enum.py -k rds --disable-azure --disable-gcp
```

© All Rights Reserved CyberwarFare Labs

ELB Recon

Recon via Shodan:

```
cloud.provider:Amazon product:"AWS ELB"
```

```
Set-Cookie: AWSELB
```

```
Location: elb.amazonaws.com
```

Recon via Censys:

```
((services.http.response.headers: (key: "Set-Cookie" and
value.headers: "AWSELB")) and autonomous_system.name=`AMAZON-02`) and
labels=`<KEYWORD>`\
```

```
dns.names: "elb.amazonaws.com"
```

© All Rights Reserved CyberwarFare Labs

ELB Recon

Recon via fofa:

```
app="AWS" && body="ELB" && region="aws_region"
```

```
app="AWS" && body="ELB" && body="DNSName" && region="aws_region"
```

```
app="AWS" && body="ELB" && body="SSLCertificateId" && region="aws_region"
```

© All Rights Reserved CyberwarFare Labs

Cognito Recon

Recon via Shodan:

```
Location: aws.cognito.signin.user.admin
```

```
amazoncognito.com
```

Recon via Censys:

```
"aws.cognito.signin.user.admin"
```

© All Rights Reserved CyberwarFare Labs

Cognito Recon

Recon via fofa:

```
app="AWS" && body="Cognito" && body="UserPool"
```

```
app="AWS" && body="Cognito" && body="IdentityPool"
```

```
app="AWS" && body="Cognito" && body="DomainName"
```

© All Rights Reserved CyberwarFare Labs

CloudFront Recon

Recon via Shodan:

```
User-Agent: Amazon Cloudfront
```

Recon via Censys:

```
(services.http.response.headers.key: "X-Amz-Cf-Id") and
labels=`<KEYWORD>`
```

© All Rights Reserved CyberwarFare Labs

CloudFront Recon

Recon via fofa:

```
app="AWS" && body="CloudFront" && body="Distribution"
```

```
app="AWS" && body="CloudFront" && body="OriginDomainName"
```

```
app="AWS" && body="CloudFront" && body="SSLCertificateId"
```

© All Rights Reserved CyberwarFare Labs

Lambda Recon

Recon via fofa:

```
app="AWS" && body="Lambda" && body="Function"
```

```
app="AWS" && body="Lambda" && body="Runtime"
```

Web-based Recon:

- Dorks:
  - Github Dorks:

```
filename:serverless.yml "aws_lambda_function"
```

```
filename:credentials "aws_access_key_id"
"aws_secret_access_key"
```

```
filename:serverless.yml "iamRoleStatements"
```

© All Rights Reserved CyberwarFare Labs

Lambda Recon

Web-based Recon:

- Dorks:
  - Google Dorks:

```
site:aws.amazon.com "<KEYWORD>"
```

```
site:stackoverflow.com aws lambda
```

```
site:github.com "arn:aws:lambda"
```

© All Rights Reserved CyberwarFare Labs

Scenario 2: Unauthenticated Enumeration

`Enum`

© All Rights Reserved CyberwarFare Labs

EC2 Image Recon

CLI-based Recon:

➢ To identify publicly accessible Amazon Machine Image (AMI), Search AMI by ownerID, Search AMI by substr ("shared" in the example).

```
aws ec2 describe-images --executable-users all
```

```
aws ec2 describe-images --executable-users all --query
'Images[?contains(ImageLocation, `967541184254/`) == `true`]'
```

```
aws ec2 describe-images --executable-users all --query
'Images[?contains(ImageLocation, `shared`) == `true`]'
```

© All Rights Reserved CyberwarFare Labs

EC2 Image Recon

CLI-based Recon:

➢ To identify any publicly accessible Amazon Machine Image (AMI) available in the selected AWS cloud region.

```
aws ec2 describe-images --region us-east-1
        --owners self --output table
       --query 'Images[*].ImageId'
```

★ **OUTPUT**

```
-------------------------
|     DescribeImages     |
+------------------------+
|  ami-0abcd1234abcd1234  |
|  ami-01234abcd1234abcd  |
|  ami-0abcdabcdabcdabcd  |
+------------------------+
```

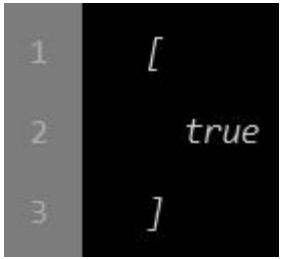
© All Rights Reserved CyberwarFare Labs

EC2 Image Recon

CLI-based Recon:

➢ To determine whether the selected image has public launch permissions.

```
aws ec2 describe-images --region us-east-1
    --image-ids ami-0abcd1234abcd1234
  --owners self --query 'Images[*].Public'
```



★ **OUTPUT**

➔ **“true”** means publicly shared and
“false” means publicly not shared

© All Rights Reserved CyberwarFare Labs

S3 Recon

CLI-based Recon:

➢ To list the names of all Amazon S3 buckets available in your AWS cloud account

```
aws s3api list-buckets --query 'Buckets[*].Name'
```

★ **OUTPUT**

```
[
    "cc-production-web-data",
    "cc-project5-audit-logs"
]
```

© All Rights Reserved CyberwarFare Labs

S3 Recon

CLI-based Recon:

➢ To describe the Access Control List (ACL) configuration set for the Everyone (public access) grantee, available for the selected S3 bucket

```
aws s3api get-bucket-acl --bucket
your-bucket-name
```

➔ **“Permission=FULL_CONTROL”** means the selected Amazon S3 bucket is publicly exposed to the Internet, therefore the bucket ACL configuration is not secure and compliant

★ **OUTPUT**

© All Rights Reserved CyberwarFare Labs

EBS Snapshot Recon

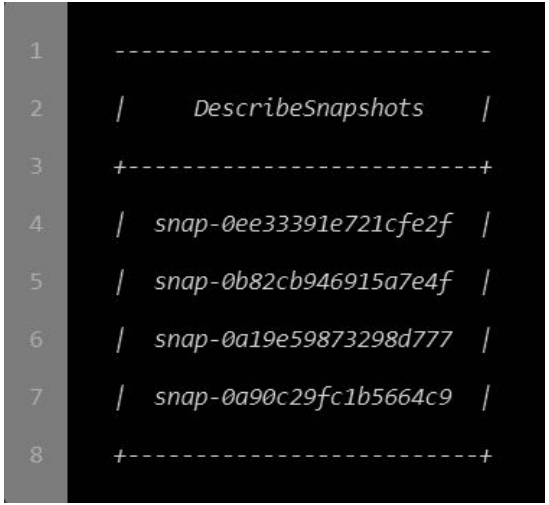
CLI-based Recon:

➢ To identify any publicly accessible EBS volume snapshots within your AWS account.

```
aws ec2 describe-snapshots
```

```
aws ec2 describe-snapshots
--restorable-by-user-ids all
```

```
aws ec2 describe-snapshots --region us-east-1
--owner-ids <owner/snapshot_id>
--filters Name=status,Values=completed
--output table --query 'Snapshots[*].SnapshotId'
```



★ **OUTPUT**

© All Rights Reserved CyberwarFare Labs

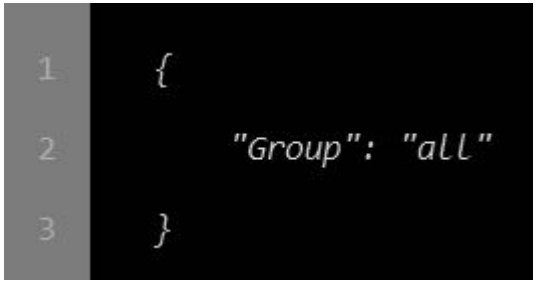
EBS Snapshot Recon

CLI-based Recon:

➢ To identify any publicly accessible EBS volume snapshots within your AWS account.

```
aws ec2 describe-snapshot-attribute --region
us-east-1 --snapshot-id snap-0ee33391e721cfe2f
        --attribute createVolumePermission
      --query 'CreateVolumePermissions[]'
```

➔ **“Group=all”** means selected snapshot is publicly accessible.



★ **OUTPUT**

© All Rights Reserved CyberwarFare Labs

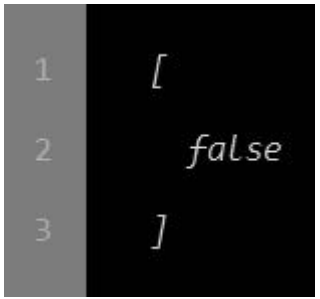
EBS Snapshot Recon

CLI-based Recon:

➢ To discover any unencrypted EBS volume snapshots present in your AWS account.

```
aws ec2 describe-snapshots --region us-east-1
    --snapshot-id snap-0b82cb946915a7e4f
      --query 'Snapshots[*].Encrypted'
```

➔ **“true”** for encrypted and **“false”** for unencrypted



★ **OUTPUT**

© All Rights Reserved CyberwarFare Labs

EBS Snapshot Recon

CLI-based Recon:

- To obtain zone and instance after obtaining public snapshot name

```
aws ec2 describe-instances --filters
Name=tag:Name,Values=<Machine_Name>
```

- To Create a new volume of it

```
aws ec2 create-volume --snapshot-id snap-0b82cb946915a7e4f
--availability-zone <ZONE>
```

© All Rights Reserved CyberwarFare Labs

EBS Snapshot Recon

CLI-based Recon:

- Attach to an EC2 instance

```
aws ec2 attach-volume --device /dev/sdh --instance-id
<INSTANCE_ID> --volume-id <VOLUME_ID>
```

- To view EC2 instance

```
aws ec2 describe-volumes
--filters Name=volume-id,Values=<VOLUME_ID>
```

© All Rights Reserved CyberwarFare Labs

RDS Snapshot Recon

CLI-based Recon:

➢ To identify whether Amazon RDS database snapshot have public accessibility.

```
aws ec2 describe-db-snapshots
```

```
aws rds describe-db-snapshots --snapshot-type
manual --include-public
```

- snapshot-type: manual, automated, awsbackup

```
aws rds describe-db-snapshots
--region us-east-1 --output table
--query 'DBInstances[*].DBInstanceIdentifier'
```

```
1  ------------------------------
2  |      DescribeDBSnapshots     |
3  +------------------------------+
4  |  cc-prod-mvp-snapshot        |
5  |  cc-dev-mvp-final-snapshot   |
6  |  cc-mysql-aurora-snapshot    |
7  +------------------------------+
```

★ **OUTPUT**

© All Rights Reserved CyberwarFare Labs

RDS Snapshot Recon

CLI-based Recon:

➢ To identify whether your Amazon RDS database snapshots have public accessibility.

```
aws rds describe-db-snapshots-attributes
--region tw-west-1
--db-snapshot-identifier cc-dev-mvp-final-snapshot
--query
'DBSnapshotAttributesResult.DBSnapshotAttributes'
```

➔ **"AttributeValues=all"** means selected Amazon RDS database snapshot is publicly accessible and available for any AWS account to copy or restore it.

```
{
    "AttributeName": "restore",
    "AttributeValues": [
        "all"
    ]
}
```

★ **OUTPUT**

© All Rights Reserved CyberwarFare Labs

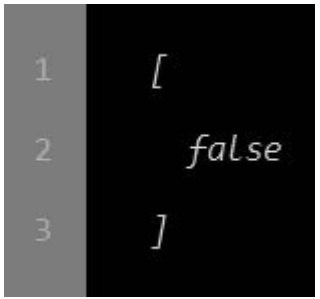
RDS Snapshot Recon

CLI-based Recon:

➢ To discover any unencrypted RDS database snapshots present in your AWS account.

```
aws rds describe-db-instances --region us-east-1
--db-instance-identifier cc-project5-mysql-database
    --query 'DBInstances[*].StorageEncrypted'
```

➔ **“true”** for encrypted and **“false”** for unencrypted



★ **OUTPUT**

© All Rights Reserved CyberwarFare Labs

RDS Snapshot Recon

CLI-based Recon:

- To restore Amazon RDS database snapshot in instance

```
aws rds restore-db-instance-from-db-snapshot --db-instance-identifier
      recoverdb --publicly-accessible --db-snapshot-identifier
  arn:aws:rds:us-east-1:159236164734:snapshot:globalbutterdbbackup
                    --availability-zone us-east-1b
```

- Once restored, try to access

```
  aws rds describe-db-instances --db-instance-identifier recoverdb
```

© All Rights Reserved CyberwarFare Labs

RDS Snapshot Recon

CLI-based Recon:

- To reset the master credentials

```
aws rds modify-db-instance --db-instance-identifier recoverdb
   --master-user-password NewPassword1 --apply-immediately
```

- To check, Instance is there

```
aws rds describe-db-instances
```

© All Rights Reserved CyberwarFare Labs

MCRTA Certification

AWS | Azure | GCP

➢ Who can opt for it

- ➔ Cyber Security Beginners / Professionals
- ➔ Security Analysts / Security Consultants / Security Engineers
- ➔ Anyone Interested in Cloud Security / Cloud Pentesting / Cloud Red Teaming Domains

© All Rights Reserved CyberwarFare Labs

Thank You

For Professional Red Team / Blue Team / Purple Team / Cloud Cyber Range labs / Trainings please contact

support@cyberwarfare.live

To know more about our offerings, please visit: **https://cyberwarfare.live**

© All Rights Reserved CyberwarFare Labs