Hacking JWT tokens for fun and Profit

By Neha Gupta

Introduction

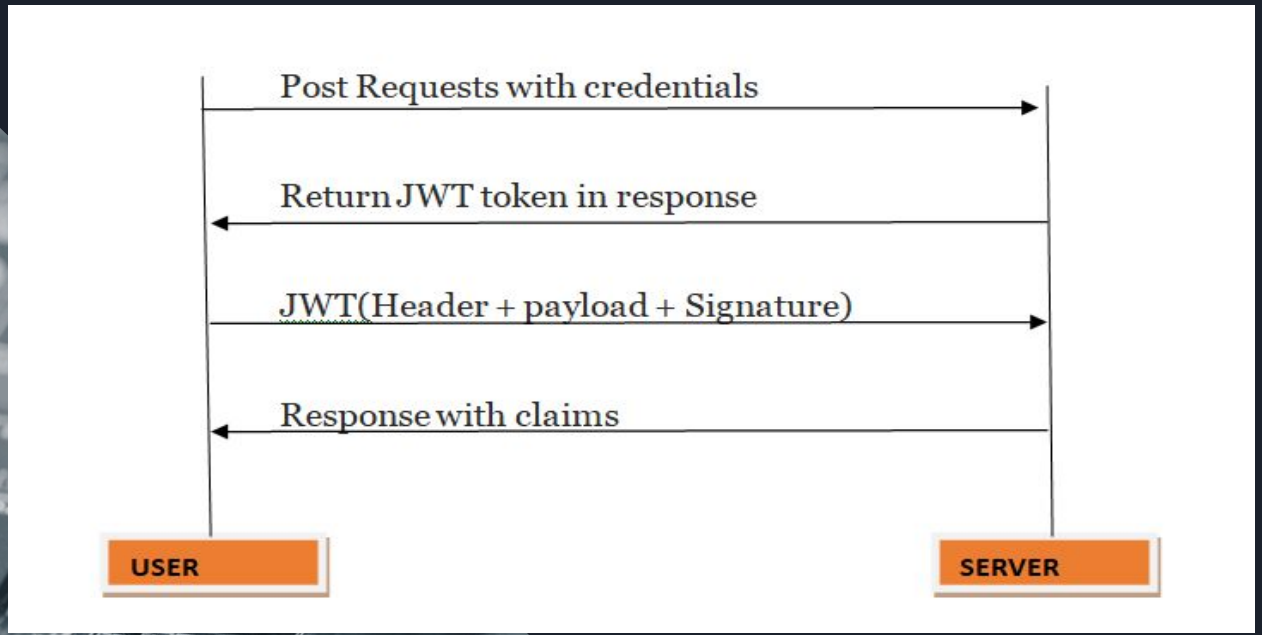
This article provides information about how you can hack JWT tokens for fun and profit.

We will also explore the the exploitation of it and how to crack the secret of a JWT token.

Understanding JWT

JWT provides a good way of securely transmitting the information between two parties and these information can be signed so both the parties will have the confirmation that the information is not tampered. Additionally the JWT tokens provides you a way to verify that the information inside it has not been tampered.

Information transfer through JWT tokens



Structure of JWT token

There are three parts of a JWT token

1. Header
2. Payload
3. Signature

These parts are separated by a dot(.) with each other

A JWT Token

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c

You can see that there are three dots in this token so these represents three parts of the token. Let's explore them in detail

Header

The portion till first dot(.) is known as the header of JWT token. This part of JWT is encoded in base64 and you can decode this part to get the value. Let decode the first part of our token

On decoding

{"alg":"HS256","typ":"JWT"}

So this part of JWT actually contains the algorithm used and the type. The algorithm is of different types such as HS256 which is used above and others are H512 and H384.

Payload

As the name suggests payload contains the data you want to transmit with the JWT token. Lets try to decode it. The Payload portion starts from first dot(.) and ends with the second one.

In our case the payload is this

eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ

Lets try to decode this as it is also encoded in the base64.

{"sub":"1234567890","name":"John Doe","iat":1516239022}

So it contains the parameter such as name and other information you want to send with the JWT token

Signature

This is the part which is used to verify whether the token is trustworthy or not. This part is not decodable in base64 as it encrypted with a secret key.

If you tamper all other parts of the token you need the secret key to sign the token again until then the token will not be accepted by the website because it will perform verification on the signature field and if found tampered token will not be accepted.

This field often uses RSA256 and HS256 algorithm.

Exploiting JWT tokens for fun and profit

So now the question arise how we can actually exploit this implementation.

Lets discuss on this

1. Since the information is saved in the plaintext and if the communication is not being done on the HTTPS and MITM attack can leak the sensitive information to the attacker.
2. If a weak secret is being used then it will be easy for an attacker to bruteforce the secret after that the secret can be used to modify the token.

    Let suppose application is carrying out authentication based on the username and you are able to modify the username and then sign the token again with the secret key, then you can takeover account of anyone on that website after getting the username.

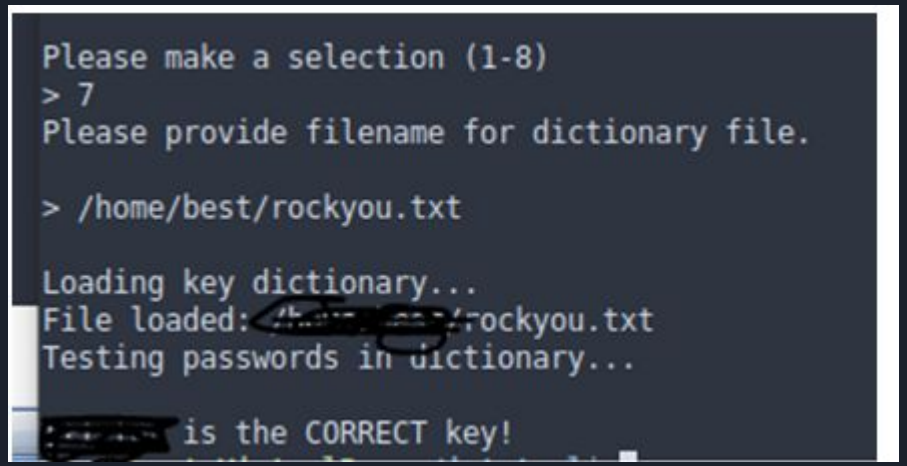
How to Brute Force the JWT token for the Secret

In order to find out the secret from a JWT token you can use this python tool

https://github.com/ticarpi/jwt_tool

This will ask for the JWT token and a wordlist and then it start bruteforcing the secret. All the lucks now depend on the wordlist.

Output From the Tool



References

https://jwt.io/

https://portswigger.net/daily-swig/jwt-heartbreaker-offers-remedy-for-weak-json-web-tokens

https://github.com/ticarpi/jwt_tool

https://gupta-bless.medium.com/jwt-usage-and-exploitation-56d9db92cf65