|  |  |
| --- | --- |
| Roll. No. A016 | Name: Varun Khadayate |
| Class B.Tech CsBs | Batch: 1 |
| Date of Experiment: 08-10-2022 | Subject: Cryptology |

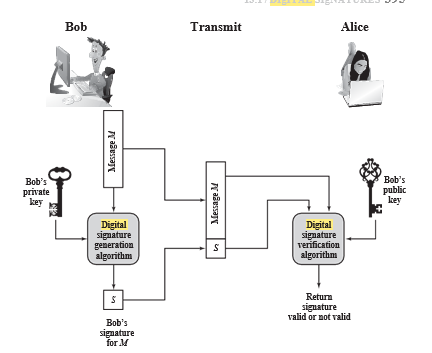
# Aim

To implement Digital Signature using <jwt.io> website

# Theory

The most important development from the work on public-key cryptography is the digital signature. The digital signature provides a set of security capabilities that would be difficult to implement in any other way.

Figure below is a generic model of the process of making and using digital sig- natures. Bob can sign a message using a digital signature generation algorithm. The inputs to the algorithm are the message and Bob’s private key. Any other user, say Alice, can verify the signature using a verification algorithm, whose inputs are the message, the signature, and Bob’s public key. In simplified terms, the essence of the digital signature mechanism is shown in Figure 13.2. This repeats the logic shown in Figure 11.4. A worked-out example, using RSA, is available at this book’s Premium Content Web site.

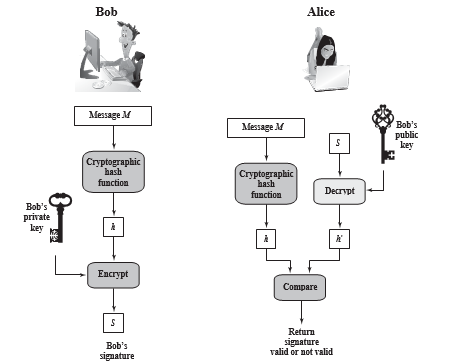


Properties

Message authentication protects two parties who exchange messages from any third party. However, it does not protect the two parties against each other. Several forms of dispute between the two are possible.

For example, suppose that John sends an authenticated message to Mary, using one of the schemes of Figure 12.1. Consider the following disputes that could arise.

1. Mary may forge a different message and claim that it came from John. Mary would simply have to create a message and append an authentication code using the key that John and Mary share.
2. John can deny sending the message. Because it is possible for Mary to forge a message, there is no way to prove that John did in fact send the message.



Both scenarios are of legitimate concern. Here is an example of the first scenario: An electronic funds transfer takes place, and the receiver increases the amount of funds transferred and claims that the larger amount had arrived from the sender. An example of the second scenario is that an electronic mail message contains instructions to a stockbroker for a transaction that subsequently turns out badly. The sender pretends that the message was never sent.

In situations where there is not complete trust between sender and receiver, something more than authentication is needed. The most attractive solution to this problem is the digital signature. The digital signature must have the following properties:

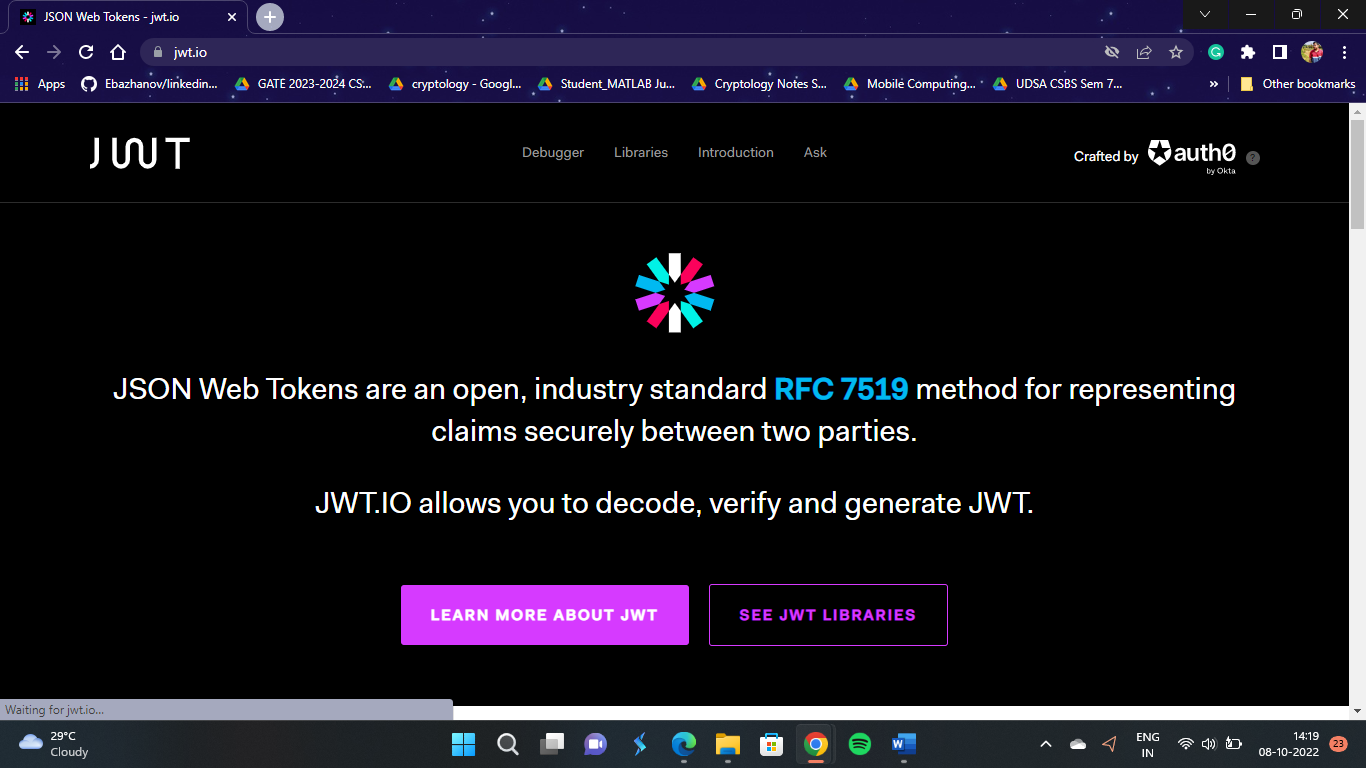
* It must verify the author and the date and time of the signature.
* It must authenticate the contents at the time of the signature.
* It must be verifiable by third parties, to resolve disputes.

Thus, the digital signature function includes the authentication function.

# Steps

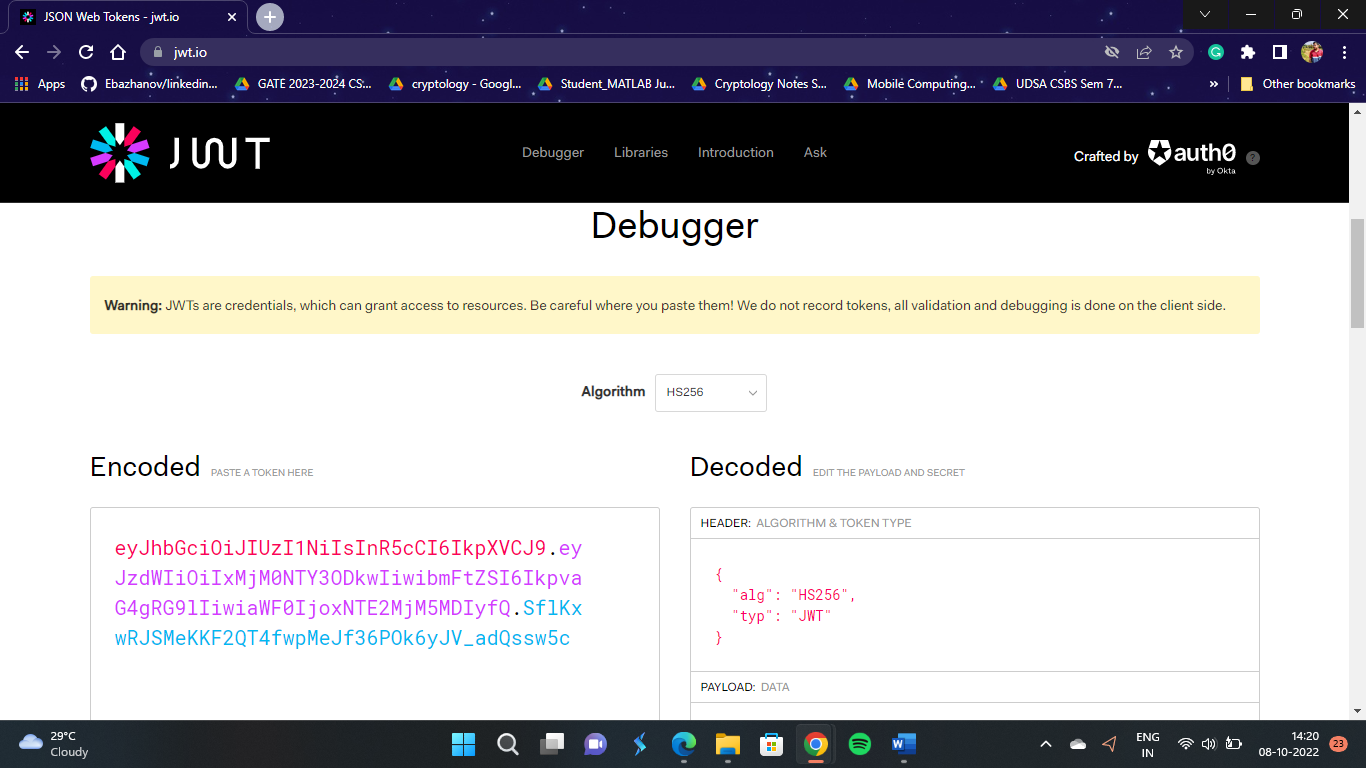
## Step-1

Open jwt.io website



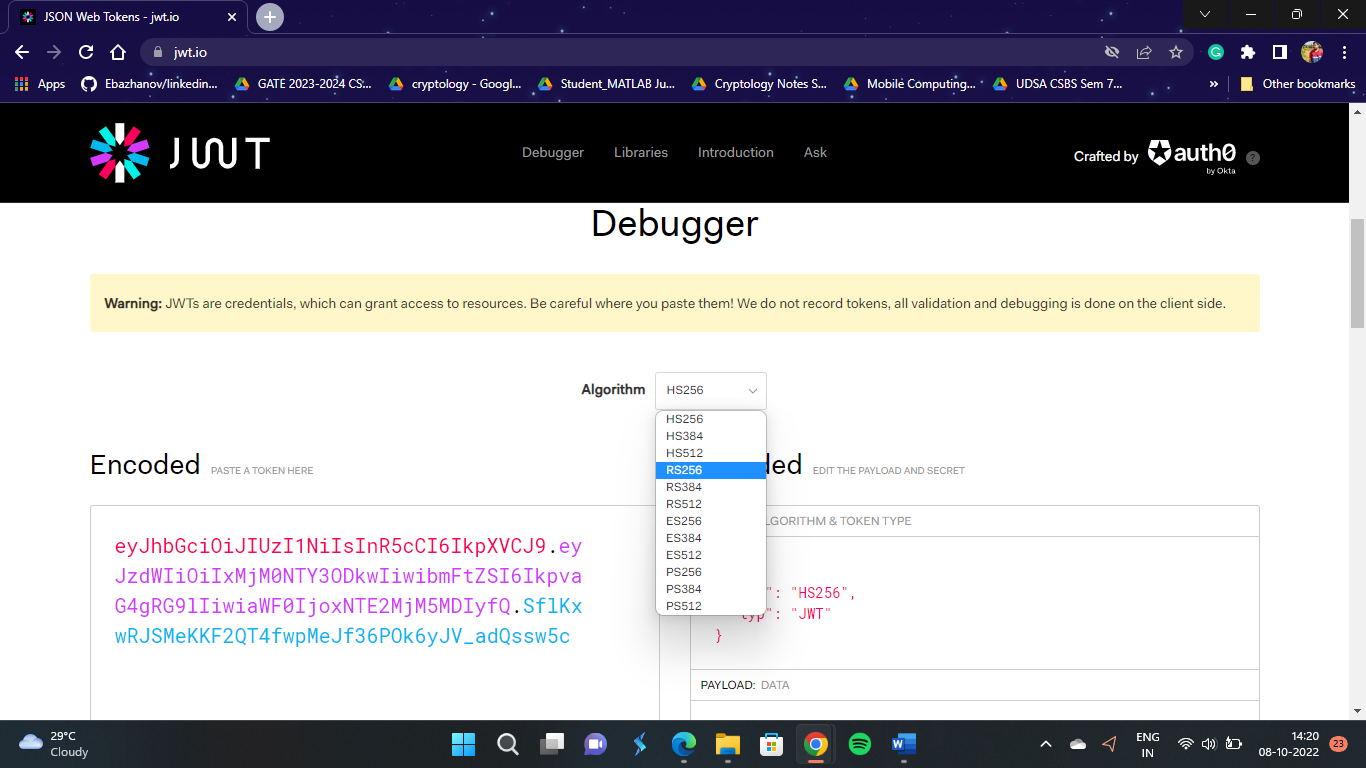
## Step-2

Go Below here at **Debugger**



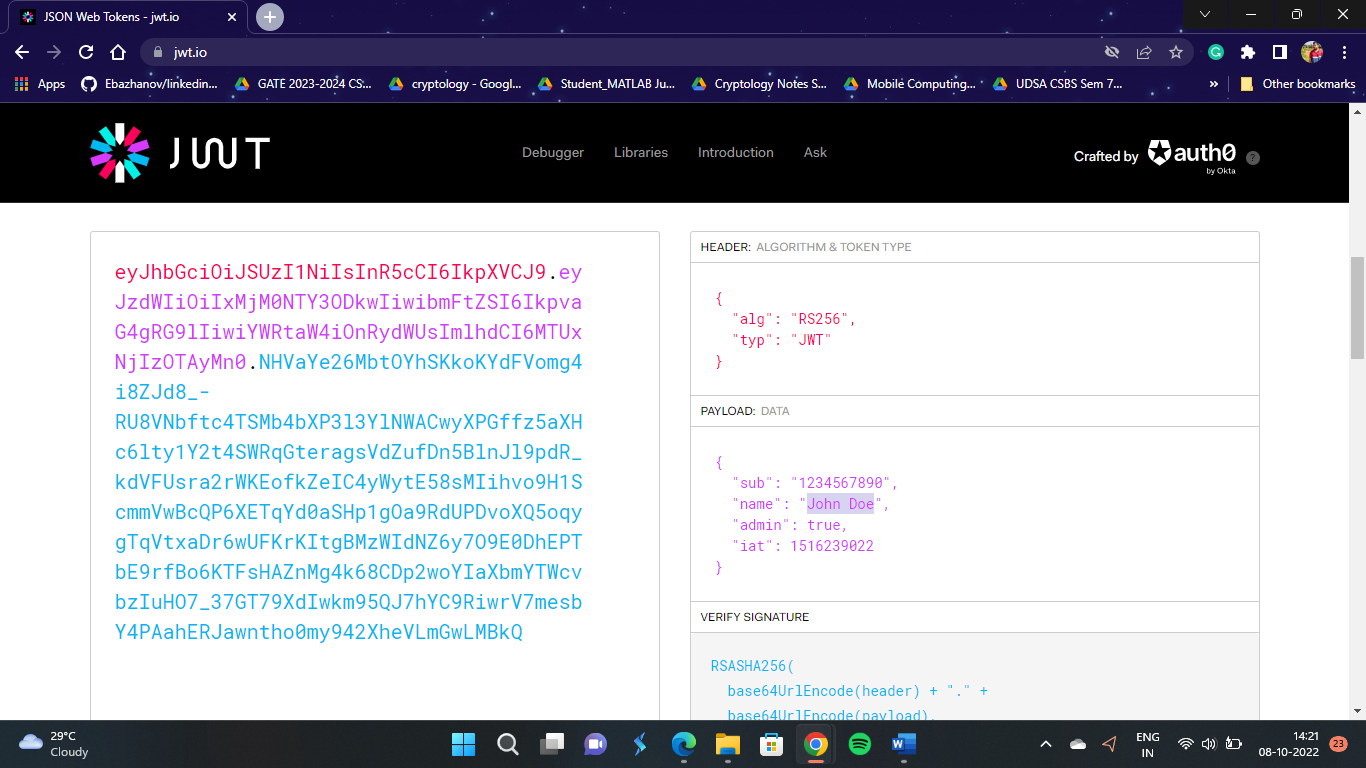
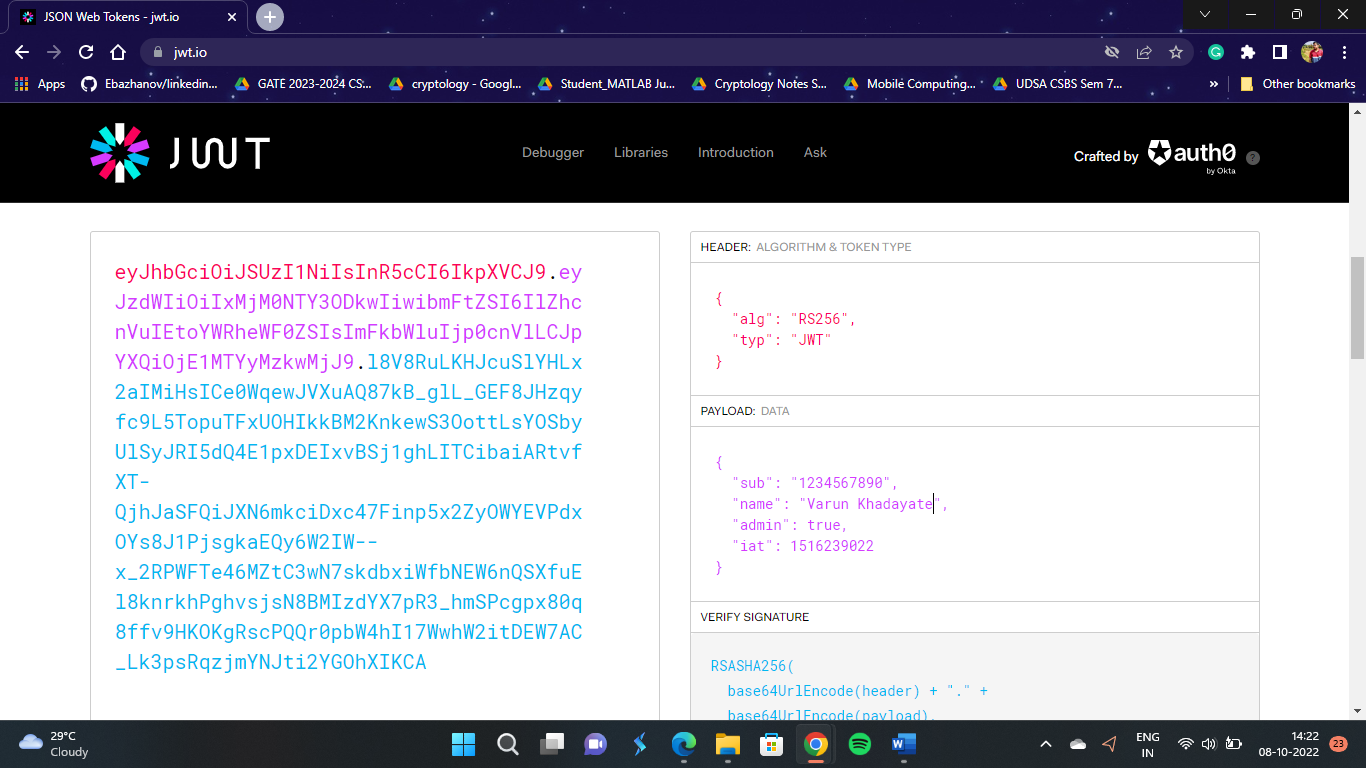
## Step-3

In Algorithm section select RS256



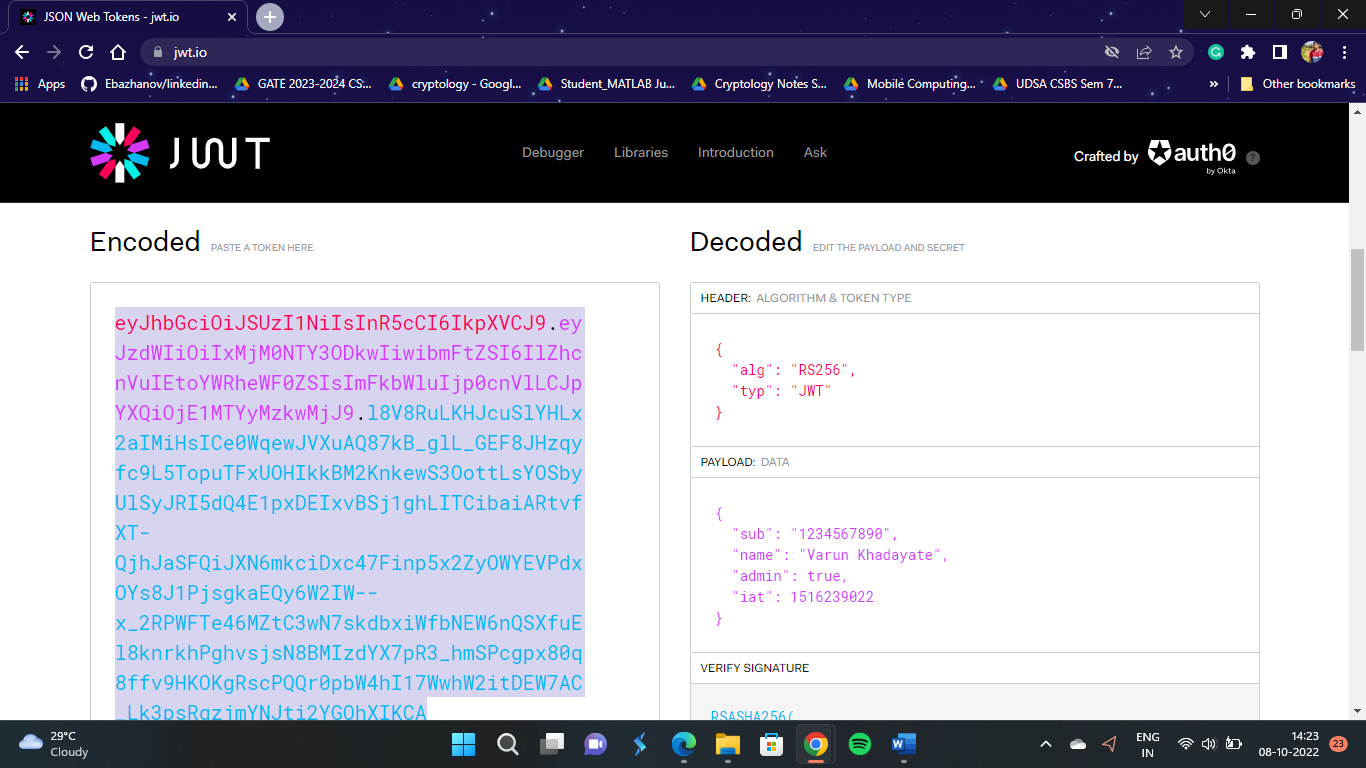
## Step-4

Here at Payload: DATA in the “name” section instead of “John Doe” keep it as “Varun Khadayate”.

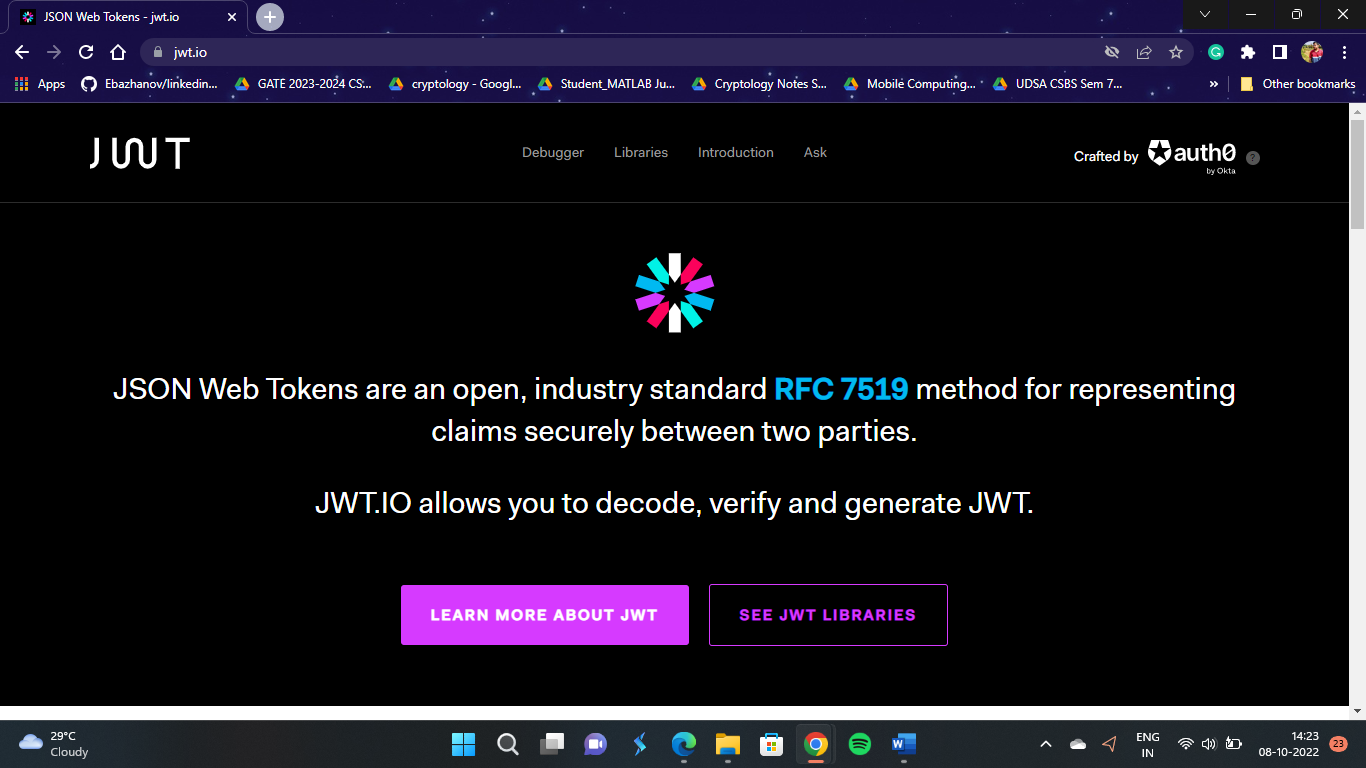
## Step-5

Copy the Encoded Section text



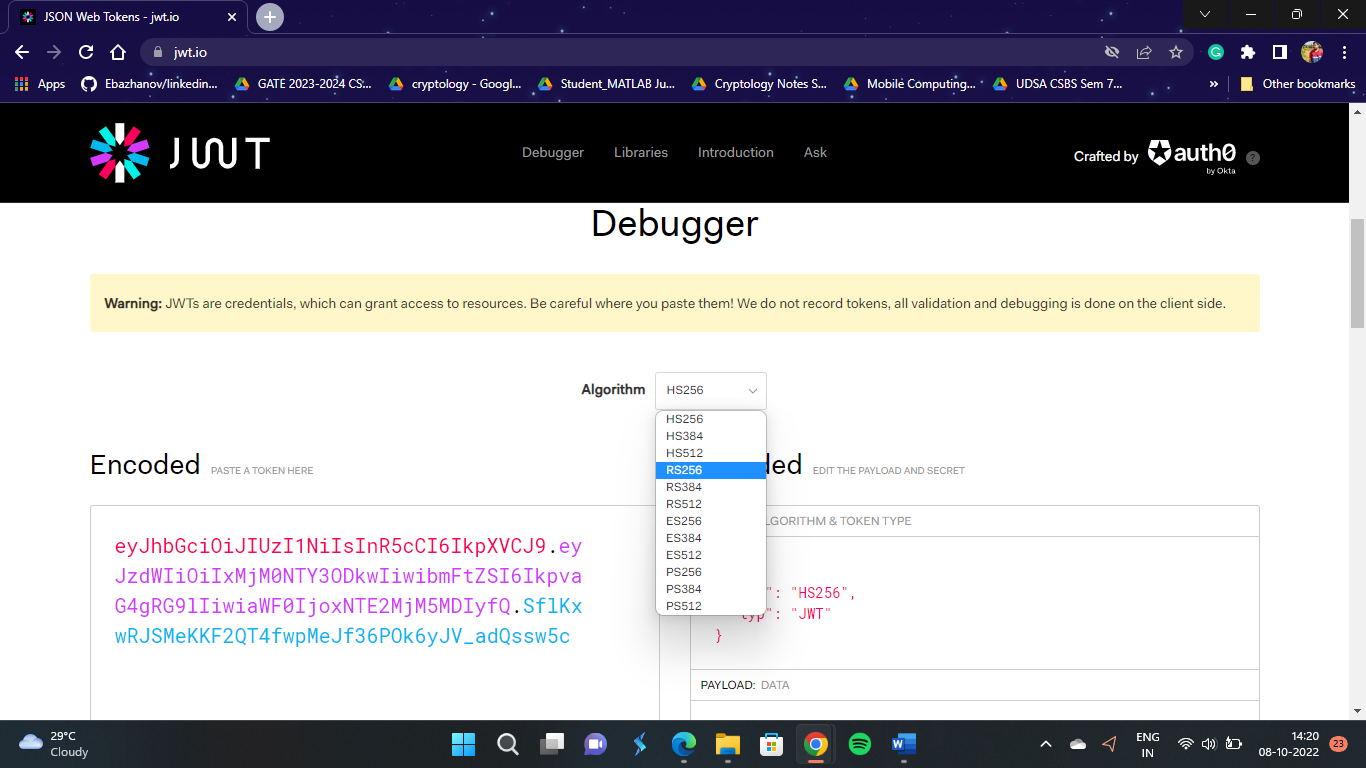
## Step-6

Refresh the page



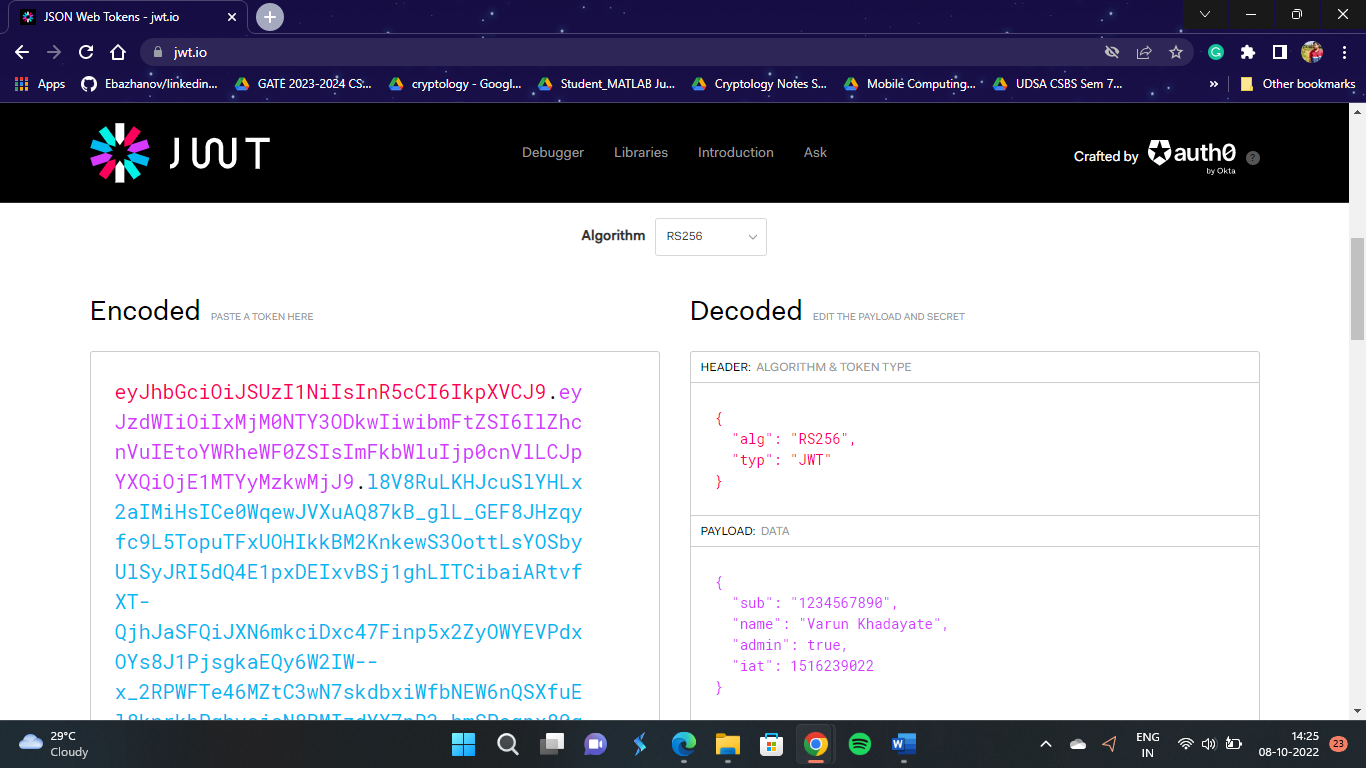
## Step-7

In Algorithm section select RS256



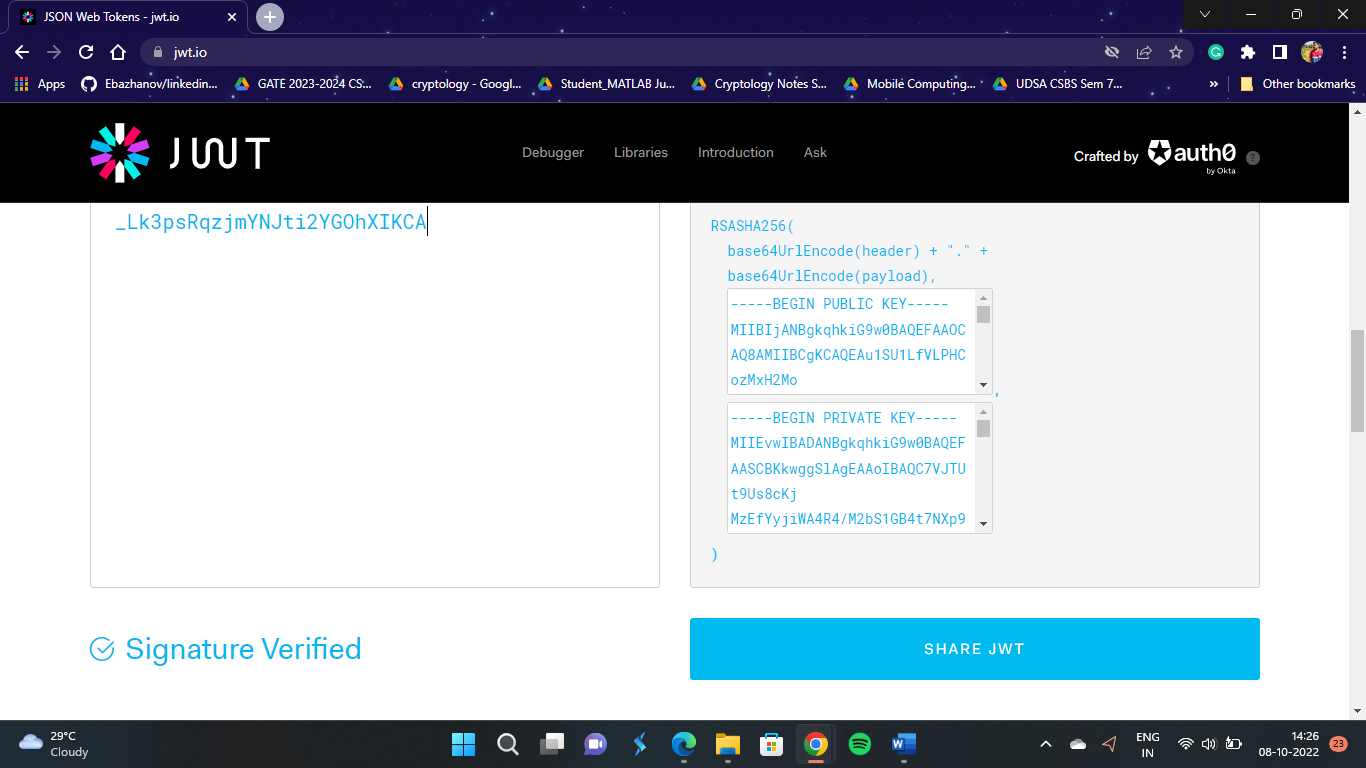
## Step-8

In the Encoded section paste the copied encoded text from previous steps



## Step-9

After pasting the text you will get the signature verified text below



# Conclusion

Hence we were able to perform the experiment successfully