Name : Shivam Agarwal
Roll No. : 2020123

Ans 2.b) I created a function called get_secret() which takes in 2 arguments, the payload(decoded) and the signature. It iterates over all lowercase-alphanumeric-length-5 of the key/secret and encodes the payload using it.
It then checks the signature passed on to the function with the one we obtained using the encoder and compares them.
If they are the same then it returns the key secret.

For changing the role, we first decode the token, then change the role of the payload like we do in dictionaries and then encode it again using the same key

Ans 2.c) Several methods can be used to prevent widespread damage are:

1. Use a long string as secret:
   In the above question we needed to permutate over 36^5 possibilities to get the secret(which already was very slow). So a lengthy secret can prevent possible cracking of the secret

2. Adding more symbols and uppercase characters as secret:
   If say only uppercase characters were also allowed in the above example then the time taken to break it would have been 15x greater

3. Changing/Updating secret regularly:
   Updating the secret regularly will limit the time available for cracking the secret

4. Making sure that the public keys of users available to us are not tampered with to prevent Man in the Middle attacks

5. Since payload of jwts can be decoded even without the need of the secret, one shouldnt send sensitive data using jwts and preferably use them over a secure connection.