CS 432– Computer and Network Security – HW #1 – Due April 7, 2023, 21:00

NO HANDWRITING PLEASE
Written answers are to be in a single pdf file
Screenshots are to be embedded in pdf file
Source codes, if any, are in their native file types
Submit a single zip file if there are multiple files
Generic file naming format: username_lastname_othernames_HW1.xxx (if needed add Q1, Q2 after HW1)

1) (25 points) What are the smallest three *q* values of which 13 could be a primitive root?

You can write a small program for this. If you do so, submit the source code and a screenshot that shows the execution.

If you do not write a code for this, show all calculations.

You may find sites that do this job for you. Using them for crosschecking is OK, but copying the answer given by that sites as your answer will definitely be considered as plagiarism. And remember that I can also use AI Chatbots ☺.

2) (35 points) Suppose you use RSA and the public key is given as (263, 1746786788707). The ciphertext that was encrypted using this public key is 1661993860336. Break this key first to learn the primes p and q. Then calculate the private key and finally make the decryption. Give all results.

You probably need to write a program for this question since the numbers given are large for manual/calculator computations, but small enough for programming languages. However, the results to be calculated within the program could be large for some programming languages. I tried in Python 3 and it worked just fine. However, you may need to make research and use some fast algorithms (e.g. for modular inverse and modular exponentiation). If you write a program for this question, submit the source code and a screenshot that shows the execution.

If you do not write code for this, show all calculations; but as I mentioned above, this is not so feasible. I have not checked but if you find some websites that does important steps for you (e.g. factorization, finding modular inverse, modular exponentiation), you can only use them for crosscheck purposes; using such websites as main answer is not allowed.

Use of any external resources directly will be considered as plagiarism. For fast factorization, modular inverse and modular exponentiation, you can use algorithms that you find on the Internet but the codes must be your own codes. Again remember that we are also capable of using AI tools for crosscheck!

3) (20 points) Suppose you use AES in Counter Mode. Moreover, suppose you obtained the following three ciphertext blocks and you know that they are produced using the same key and counter values.

C_1 = `586EFEA0879BBE0B`
C_2 = `2EBEFD6874DCEE2F`
C_3 = `C284E0DC4F364811`

If you also know that the plaintext block that corresponds to the first ciphertext is the following, what are the other plaintext blocks?

P_1 = `FF7AB6C7303DFF4F`

All values are given in hexadecimal. Please give the results in hexadecimal as well. Explain the details of your solution approach as well.

4) (20 points) Suppose you use rotor machine, which is explained in pages 108 - 110 of 7E of our textbook, for encryption purposes. Assuming you use three rotors and the initial condition is given in slide 30 of crypto1.pptx, what is the encrypted version of the word EYT? Show your work. No program writing in this question.