CS459/698 Privacy, Cryptography, Network and Data Security
Assignment One

Please use Piazza for questions and clarifications.

Assignment due date: Sept 30, 2025 at 3:00 pm

Total Marks: 45 (25 written + 20 programming)

TA: Anais Huang

TA Office hours: Mondays 10:00 – 11:00 am, DC 3333A (CrySP Lab office hours room)

- Submissions are to be uploaded to the appropriate LEARN dropbox as a zipped folder containing both the written and programming components.
- All questions (about the programming and written parts) should be asked privately on Piazza under Assignment 1.
- The TA may decide to make responses to your private questions public so that everyone benefits from knowing the same information. Questions where you need to post partial solutions or questions that describe the locations of vulnerabilities or code to exploit those vulnerabilities will be sanitized before the TA provides a public response.

1 Written (25 Marks)

1. **(12 marks, 4 for each part)** Identify the plaintext for each of the following ciphers. You **must** include an explanation of each step and analysis you took to arrive at this plaintext to receive credit for the solution. Note that saying you inputted the ciphertext into a cipher solver does not count for explaining analysis. The ciphertext is everything between quotation marks, but does not include the quotation marks. Each plaintext only contains the letters A–Z (numbers, spaces, punctuation, etc. are removed). You may use your knowledge of the original text to help you find the original passage. You may need to look up information about these ciphers.

   Sample Explanation: Recall the many-time pad attack from class. If you were to explain it as you will for questions 1(a-c), it could look something like the following.

   - First, XOR the two ciphertexts together, this will result in an output corresponding to the two messages XORd together
   - At this point, you know that the values that have been XORd are non-uniform (they correspond to English text). This limits the space of possibilities to components from the English language. Start by looking for expected values (possibly a space), if the space XORd with another value produces a legitimate character, that is likely the correct character. Proceed through common/likely English characters until the output is either i) a coherent English phrase or ii) close enough to guess the phrase.
   - The solution can be "verified" by XORing the messages with their original ciphertexts; if the output is the same value, this is the key.

   Ciphertexts:

   (a) Ciphertext: "OLWVRLJGERBLVKVRCWVPEVCGGSRDPOVEOJCFFGPBCOPJM
   RPMILPBLCUWJDFJAKDODCGGSRDRFPBPJDRPMEPXPEDCGRIPOLBJMAG
   PBOPMUPMOVWVRORKDROBJJFVWCOVPEVCGGSIVIJDGEGPQVOLVBJJFV
   WCOPJMOJYVYCRVEJMKDODCGBJMRVMOYDOOLVXVOJFJIVWOLPRKVB
   LCMPRKUPXVROJVCBLKVKYVWBCMFCWCGSTVOLVCBOPXPOPVRJAOLV
   UWJDF"

   - Cipher is: substitution cipher
   - Plaintext source: "How to Share a Secret" [1]

   (b) Ciphertext: "CTTNWOEEFCKFQDECCSLFAKXZQNNYRZVCAIGRHCIJGCNC
   WOCFHTAPDPFCKCDPMDRWTALEFPGKWRXTGOLRVDNCWIKBGYLPHP
   TGTEOTCPWICNWZAXLFKCXDOMIEQTKPDZEKGDBYDPFCKCDPMKVFR
   EKEWZWRTEVZBNMUGRXOPPXKJILTGNYVKSGZHVHUTELDSYAZVHGP
   JZVKJEEPGNGFPCEFRDRXVHTEHCIVPTKZDTSWVHXZIOTLVDBDHMMS
   WTBZBJJJVAGOOMHZBEWCGVOVAGXYSMEKKOGTGVPNCYLLZHSJVMT
   IWHECCNWEVZSLVPNEGVVVJAKOHJJRETHCWAJRETHCWIKZPGXYSME
   CKSALFY"

   - Cipher is: Vigenère cipher
   - Plaintext source: "Ron was wrong, Whit is right" [2]

[1] https://dl.acm.org/doi/pdf/10.1145/359168.359176
[2] https://eprint.iacr.org/2012/064.pdf

(c) Ciphertext: "CBGREDISGERZLYLETGHEDWIRUIRFGTAYRIHIAGEAQHYWG
GFHBGITSYRVGTAEHIYRFTIVEHGEDISGSERKEWGAOTGRYYRGIAETYO
RZERZCIHBHBGGJSGFHIYRYNEBIZZGRHEFGTGSYTZGTABGSERLGTGEA
YRELDQAOTGHBEHRYYRGGDAGCIDDBGETHBGSYRVGTAEHIYRNOTHB
GTHBGYRDQGVIZGRSGERQYRGSERYLHEIRYNHBGSYRVGTAEHIYRIALY
LACYTZELYOHCBEHBEFFGRGZ"

- Cipher is: affine cipher
- Plaintext source: "Off-the-Record Communication, or, Why Not To Use PGP" [3]

2. **(6 marks, 2 for each part)** Read Lily's story. For each scenario below, argue why it is a privacy issue, a security issue (identify which aspect of the CIA triad is violated), or both.

   (a) Lily works at a large company where employees bring their own devices to work. One day, when she started her laptop and tried to use the company's database management system (DBMS), a window popped up:
   "Pay XXX amount of money to restore your system, or we will make all your files public."
   She heard her colleagues complaining about the same issue and realized that the company's DBMS had been taken down by ransomware.

   (b) Later, the IT department resolved the issue and identified the cause. To "maintain better security," the manager required all employees to install monitoring software on their laptops. Afterward, Lily and her colleagues complained about company policies in their private chat group. That same afternoon, everyone who criticized the company received an email warning them to "stop defaming the company."

   (c) Disturbed by the censorship, Lily collected evidence showing that the monitoring software was intercepting employees' private communications. She compressed the evidence into a file and shared it on a public forum, attaching a hash digest of the file. When journalist Luna downloaded the file, she noticed that the hash did not match.

3. **(3 marks)** Suppose we use the CBC mode of operation, but instead of a normal block cipher, our "encryption function" is simply an XOR with a fixed key K (the same K is reused for every block). Describe (as you did in problem 1) how you would go about finding the plaintext (without a provided key) for a ciphertext consisting of several blocks.

4. **(4 marks)** Suppose Alice writes a message and sends it to Bob authenticated with a MAC using a key only Alice and Bob know. (Alice and Bob are confident that only the two of them have the MAC key at the time that Bob receives the message.) After reading the message, Bob shows the message to Carol and gives her the key to verify the MAC tag. Indicate whether or not each character (Alice, Bob, and Carol) can be certain that Alice is the author of the message. Explain why they are certain or uncertain. Alice's explanation has been provided for you.

   Alice: Alice wrote the message herself, so she knows that she was the author.

[3] https://otr.cypherpunks.ca/otr-wpes.pdf

2 Programming (20 Marks)

You are free to use either Python or C++. If you want to use a different language please reach out to the TA. You must implement each question yourself from scratch. You may use the math and IO libraries from your language, but not any other library (no cryptography library). You may re-use code from previous questions to help solve subsequent questions. We have included a sample input for each question. The assignment will be graded on different but similarly formatted inputs of the same size.

1. **(10 marks)** From Lily's perspective, "reinventing the wheel" is a good way to learn — especially when it comes to cryptography. Recently, Lily studied the Diffie–Hellman key exchange and wrote her own program to share a secret with you. She encrypts her secret using the DH shared key. Your task is to act as a client and recover Lily's secret.

   Instructions:

   First, start Lily's server locally with:

   ```
   python3 q1-Server.py
   ```

   Write a client program that:

   (a) Connects to Lily's server.

   (b) Performs the Diffie–Hellman key exchange with the server.

   (c) Get the shared secret key.

   (d) Uses the derived key to decrypt Lily's encrypted secret.

   (e) Prints the recovered secret to standard output.

   Formatting:

   If you use Python, your code file should be "q1.py". It should be able to be run using the command "python3 q1.py".

   If you use C++, your source code file should be called "q1.cpp" and it must compile and be contained within a single file or no marks will be given.

   During grading, we will test your program on a different file with a different secret string.

2. **(10 marks)** In question 1, you saw the paper "Ron was wrong, Whit is right" `https://eprint.iacr.org/2012/064.pdf`. The paper studies a number of RSA public keys and determines how many offer no security. We have provided a file (*decoded_keys.txt*) filled with the modulus n for 50 RSA public keys (one per line), some of which offer no security[4]. Your job is to identify and break the vulnerable keys.
   Formatting:

[4]The original RSA public keys are contained in a file called "real_keys.txt". Each key contains e,n and some additional meta-data, encoded in PEM format. For simplicity we have decoded the keys for you and "decoded_keys.txt" contains only the modulus value n for each key (one per line).

If you use Python, your code file should be "q2.py". It should run on its own without any dependency besides basic Python library imports like math and random. It should be able to be run using the command "python3 q2.py".

If you use C++, your source code file should be called "q2.cpp" and it must compile and be contained within a single file or no marks will be given.

When run, it should read the keys in the provided "decoded_keys.txt" file, then identify and break the vulnerable keys. It should then write the results to a file "output_q2.txt" where each line contains 3 comma-separated values: first the index of the key in the file (starting at 0), followed by the two prime factors of the broken key (also comma-separated).

The sample "decoded_keys.txt" file we have provided has vulnerable keys at indices 14, 30, 34, 39, 46, and 47. We will evaluate your code on a different file of keys with the same format.