Problem 1 (2 pts)

Question: Explain, in your own words, why the dictionary meaning of cryptography as the "art of writing and solving codes" is inaccurate today.

Part (a): Accuracy of the Traditional Definition

The traditional dictionary definition of cryptography as the "art of writing and solving codes" is somewhat inaccurate in today's context due to the evolution and expansion of the field. While cryptography indeed involves encoding and decoding information to secure it, it has grown far beyond just codes. Here's why this definition is no longer entirely accurate:

a) **Broader Scope:** Cryptography today encompasses various techniques for ensuring the confidentiality, integrity, and authenticity of data, not just limited to codes or ciphers. It involves cryptographic algorithms, protocols, and mathematical techniques that go beyond simple code-making and code-breaking [1].

b) **Digital Environment:** In modern cryptography, the focus has shifted to securing digital data and communication, not just physical documents or messages. Cryptography is used extensively in securing online transactions, communication over the internet, protecting digital identities, and more [2].

c) **Encryption:** Cryptography now heavily relies on encryption techniques, which involve complex mathematical algorithms and not just codes. Encryption transforms data into a format that is unreadable without the proper decryption key, providing a higher level of security than traditional codes [3].

d) **Authentication and Integrity:** Cryptography is used for purposes other than just concealing the content of a message. It also ensures the authenticity of data (verifying the source) and integrity (ensuring data has not been tampered with), which goes beyond the concept of code-making [3].

e) **Advanced Cryptographic Concepts:** Modern cryptography incorporates advanced concepts like public-key cryptography, digital signatures, secure key exchange, and cryptographic hashing, which are not captured by the traditional definition [5].

In summary, while the traditional definition of cryptography as the "art of writing and solving codes" was accurate in its historical context, it no longer adequately describes the multifaceted nature of cryptography in the digital age. Cryptography today involves a wide array of mathematical and technological tools to ensure data security and privacy in various applications, making it much more than just code-making and code-breaking.

References

[1] Stallings, W. (2017). Cryptography and Network Security: Principles and Practice. Pearson.

[2] Schneier, B. (2015). Applied Cryptography: Protocols, Algorithms, and Source Code in C. Wiley.

[3] Katz, J., Lindell, Y. (2014). Introduction to Modern Cryptography: Principles and Protocols. CRC Press.

[4] Ferguson, N., Schneier, B., Kohno, T. (2010). Cryptography Engineering: Design Principles and Practical Applications. Wiley.

[5] Paar, C., Pelzl, J. (2010). Understanding Cryptography: A Textbook for Students and Practitioners. Springer.

Problem 2 (2 pts)

Question: Explain the three principles of modern cryptography.

Part (a): Principles of Modern Cryptography

Modern cryptography is built upon three fundamental principles that form the basis for secure communication and data protection. These principles are:

a) **Confidentiality:** Confidentiality is the principle of ensuring that information remains private and hidden from unauthorized access. In modern cryptography, this is achieved through techniques like encryption, which transform plaintext data into ciphertext that can only be deciphered by those with the proper decryption key. Confidentiality ensures that even if an adversary intercepts the data, they cannot understand its content without the appropriate decryption key [1].

b) **Integrity:** Integrity ensures that data remains unchanged and uncorrupted during transmission or storage. Cryptographic methods such as hash functions are used to generate fixed-length checksums or hash values of data. By comparing these hash values before and after transmission, one can detect any unauthorized alterations. This principle helps in ensuring that data has not been tampered with or modified by malicious actors [2].

c) **Authentication:** Authentication is the process of verifying the identity of users, devices, or entities in a communication system. Modern cryptography employs techniques like digital signatures and public-key infrastructure (PKI) to provide strong authentication. Digital signatures allow the recipient to verify the sender's identity and the integrity of the message, while PKI establishes trust through the use of digital certificates issued by trusted authorities [3].

These three principles collectively ensure that data remains confidential, unaltered, and authenticated during communication or storage, forming the foundation of secure modern cryptography.

References

[1] Stinson, D. R. (2005). Cryptography: Theory and Practice. CRC Press.

[2] Schneier, B. (2015). Applied Cryptography: Protocols, Algorithms, and Source Code in C. Wiley.

[3] Ferguson, N., Schneier, B., Kohno, T. (2010). Cryptography Engineering: Design Principles and Practical Applications. Wiley.

Problem 3 (15 pts)

Part (a): Construction of a Perfectly-Secret Encryption (3 pts)

A perfectly-secret encryption scheme is achieved using a one-time pad. In this scheme, both the sender and the receiver share a pad of random bits (the same length as the plaintext). To encrypt a message, the sender performs a bitwise XOR operation between the plaintext and the random pad bits, and to decrypt, the receiver XORs the ciphertext with the same pad bits. Since each pad bit is used only once and is completely random, this scheme provides perfect secrecy as it produces ciphertexts that are indistinguishable from random noise.

Part (b): Security Definition of Perfectly-Secret Encryption (3 pts)

Perfect secrecy, also known as Shannon secrecy, is a property of an encryption scheme where the ciphertext reveals no information about the plaintext, no matter how much ciphertext an adversary observes. Formally, an encryption scheme is perfectly secret if, for every possible plaintext m and every possible ciphertext c, the following holds:

$$Pr(M = m|C = c) = Pr(M = m)$$

This means that the conditional probability of a particular plaintext given the ciphertext is the same as the unconditional probability of that plaintext. In other words, knowing the ciphertext does not provide any information about the plaintext.

Part (c): Perfect Indistinguishability (3 pts)

Perfect indistinguishability refers to the property of an encryption scheme where two ciphertexts, generated from two different plaintexts of the same length, are statistically indistinguishable. Formally, an encryption scheme is said to have perfect indistinguishability if, for any two plaintexts m_0 and m_1 of the same length, and for any ciphertext c of the same length:

$$Pr(E(k, m_0) = c) = Pr(E(k, m_1) = c)$$

In simple terms, an adversary cannot tell whether a given ciphertext corresponds to m_0 or m_1 when both are equally likely plaintexts.

Part (d): Adversarial Indistinguishability (3 pts)

Adversarial indistinguishability is a related concept that focuses on the ability of an adversary to distinguish between two ciphertexts generated from different plaintexts. An encryption scheme has adversarial indistinguishability if, for any efficient adversary $\mathcal{A}$, the following holds:

$$|Pr[\mathcal{A}(E(k, m_0)) = 1] - Pr[\mathcal{A}(E(k, m_1)) = 1]| \leq negligible$$

In this definition, $\mathcal{A}$ is an algorithm representing the adversary's ability to distinguish between the two ciphertexts. The difference in probabilities should be negligible, meaning that the adversary cannot significantly distinguish between the two ciphertexts.

Part (e): Insecurity of the One-Time Pad Scheme Under Two Messages (3 pts)

The one-time pad scheme is perfectly secure when each key is used only once. However, it becomes insecure when the same key is used for encrypting two different messages of the same length.

Let's consider two messages: m_0 and m_1, both of the same length, and a key k. When we encrypt m_0 and m_1 using the same key k, we obtain two ciphertexts: $c_0 = m_0 \oplus k$ and $c_1 = m_1 \oplus k$. Now, if an adversary intercepts c_0 and c_1, they can calculate $c_0 \oplus c_1$:

$$c_0 \oplus c_1 = (m_0 \oplus k) \oplus (m_1 \oplus k) = m_0 \oplus m_1$$

The adversary has successfully obtained the XOR of the two plaintexts, which reveals information about both m_0 and m_1. This demonstrates that the one-time pad scheme is insecure when the same key is used for two different messages.

References

[1] Shannon, C. E. (1949). Communication theory of secrecy systems. *Bell System Technical Journal, 28*(4), 656-715.

Problem 4 (21 pts)

Part (a): Meaning of PPT in Adversarial Indistinguishability (3 pts)

In the context of adversarial indistinguishability, PPT stands for "Probabilistic Polynomial Time." It refers to algorithms that run in polynomial time and may include probabilistic elements. Adversarial indistinguishability typically considers the behavior of adversaries that are probabilistic polynomial-time algorithms. These are algorithms that can efficiently run and make probabilistic choices during their execution.

Part (b): PrivKeavA,Q for Q = (Gen, Enc, Dec) (3 pts)

PrivKeavA,Q represents the advantage of an adversary A in distinguishing between two encryption schemes Q when given access to an oracle for Q. Specifically, Q = (Gen, Enc, Dec) consists of the following:

- Gen: The key generation algorithm. - Enc: The encryption algorithm. - Dec: The decryption algorithm.

Pr[PrivKeavA,Q = 1] represents the probability that the adversary A, when given access to the oracle for Q, can correctly distinguish between two encryptions produced by Q. In other words, it measures the advantage of A in breaking the security of Q.

Part (c): Properties of Pseudorandom Generators (3 pts)

Two properties of pseudorandom generators are:

1. Expansion: Pseudorandom generators expand a short random seed (with a fixed length) into a longer pseudorandom sequence. This property ensures that the generated pseudorandom output is significantly longer than the seed, providing a large amount of randomness.

2. Indistinguishability: Pseudorandom generators aim to produce sequences that are computationally indistinguishable from truly random sequences. This means that an efficient algorithm should not be able to distinguish between the output of the pseudorandom generator and true random data, even with access to unlimited computational resources.

Part (d): Construct Q Indistinguishable in the Presence of an Eavesdropper (3 pts)

To construct Q = (Gen, Enc, Dec) that is indistinguishable in the presence of an eavesdropper, we can use a cryptographic pseudorandom generator (PRG). The encryption scheme can be defined as follows:

- Gen: Generate a short random seed s. - Enc: Use the PRG to expand the seed into a longer pseudorandom sequence $r = PRG(s)$. - Encrypt the message m as $c = m \oplus r$. - Output the ciphertext c.

This encryption scheme ensures that the ciphertext c appears random and indistinguishable from truly random data. Even with knowledge of s, an eavesdropper cannot distinguish between the output of the PRG and truly random data.

Part (e): Privacy-Key Encryption Schemes and Indistinguishable Multiple Encryptions (3 pts)

It is possible to have privacy-key encryption schemes that have indistinguishable encryptions in the presence of an eavesdropper but do not have indistinguishable multiple encryptions. This is because the security properties of an encryption scheme can vary depending on the specific scenario and threat model.

For example, consider an encryption scheme that uses a random one-time pad for each message. In this case, each encryption is indistinguishable from random noise, even in the presence of an eavesdropper. However, if the same key is used to encrypt multiple messages, an adversary can potentially perform statistical analysis to distinguish between the encrypted messages, as the same key is used for all of them.

Part (f): Properties of Variable Length PRG (3 pts)

Three properties of variable-length pseudorandom generators (PRGs) and their importance:

1. Expansion: Variable-length PRGs expand a short seed into a variable-length pseudorandom sequence. This property is essential because it allows generating pseudorandom data of arbitrary length from a fixed-size seed, making it suitable for various cryptographic applications where different data lengths are required.

2. Pseudorandomness: Like fixed-length PRGs, variable-length PRGs aim to produce output that is computationally indistinguishable from true random data. This ensures that the generated data appears random and unpredictable, providing a foundation for secure encryption and other cryptographic operations.

3. Efficiency: Variable-length PRGs should be computationally efficient, meaning that they should generate pseudorandom sequences quickly. Efficiency is crucial for practical cryptographic applications, where real-time performance is often required.

Each of these properties contributes to the overall security and practicality of variable-length PRGs in cryptographic systems.

Part (g): CPA and CCA Attack Scenarios (3 pts)

1. CPA (Chosen-Plaintext Attack): In the CPA attack scenario, an adversary can choose plaintext messages and obtain their corresponding ciphertexts from an encryption oracle. The goal is to distinguish between two ciphertexts produced for two different chosen plaintexts. For example, if an adversary chooses two plaintexts, "yes" and "no," and receives their ciphertexts, the adversary aims to determine which ciphertext corresponds to "yes" and which one corresponds to "no." This scenario is possible when an encryption scheme does not provide semantic security.

2. CCA (Chosen-Ciphertext Attack): In the CCA attack scenario, an adversary can choose ciphertexts and obtain their corresponding plaintexts from a decryption oracle. The adversary aims to perform various operations with the oracle's help to obtain information about the secret key or plaintexts encrypted by others. For example, an adversary may choose ciphertexts and request their decryption, then use the obtained plaintexts to gain further knowledge about the encryption scheme. This scenario is possible when an encryption scheme is vulnerable to adaptive chosen-ciphertext attacks.

These attack scenarios demonstrate the importance of designing encryption schemes that are secure against various forms of adversary interactions, including chosen-plaintext and chosen-ciphertext attacks.

References

[1] Goldwasser, S., Bellare, M. (2008). Lecture Notes on Cryptography. Retrieved from https://www.cs.princeton.edu/ sudhakar/courses/cos433/cos433.pdf