Assessment on Security Algorithms

December 20, 2024
Max Marks: 100
Time: 3 Hours

Instructions:

- Attempt all questions. Each question carries equal marks.
- Show all workings for full credit.
- This paper tests advanced topics; innovative approaches are encouraged.
- Use formal mathematical notation wherever applicable.

Questions:

Q1. Symmetric and Asymmetric Encryption: Explain the concepts of symmetric and asymmetric encryption. Compare the two techniques in terms of key generation, encryption, decryption, and their computational complexity. Illustrate with examples of algorithms for each, such as DES, AES, RSA, and ElGamal. Furthermore, prove the security of AES under the assumption of a secure block cipher, and demonstrate how the RSA algorithm can be attacked using the chosen ciphertext attack.

Q2. Diffie-Hellman Key Exchange: Derive the Diffie-Hellman key exchange protocol. Show why it is vulnerable to a Man-in-the-Middle (MITM) attack and discuss how the use of public key certificates can mitigate such an attack. Provide a mathematical proof of the security of Diffie-Hellman in the random oracle model, and discuss the computational hardness assumptions behind its security.

Q3. RSA Algorithm: Given two large prime numbers $p = 101$ and $q = 103$, compute the RSA public and private keys. Prove that the decryption of a message $m = 54$ using the private key yields the original message. Next, demonstrate the attack on RSA using the chosen plaintext attack, and describe how the algorithm could be improved by using hybrid encryption techniques.

Q4. Elliptic Curve Cryptography (ECC): Explain the differences between RSA and ECC in terms of key size, efficiency, and security. Prove the elliptic curve discrete logarithm problem (ECDLP) and its importance in the security of ECC. Implement a basic elliptic curve point addition algorithm and provide an example of scalar multiplication on the curve $y^2 = x^3 + 2x + 3 \mod 97$.

Q5. Digital Signatures: Discuss the concept of digital signatures. Provide the mathematical formulation of the Digital Signature Algorithm (DSA) and prove its security under the random oracle model. Additionally, discuss the application of the RSA and ECC algorithms in digital signatures and analyze their security properties in the context of chosen message attacks.

Q6. Hash Functions and Cryptographic Hashing: Explain the role of cryptographic hash functions in digital forensics and blockchain technology. Discuss the Merkle-Damgård construction and prove the collision resistance of SHA-256. Given a message

M, calculate the SHA-256 hash of $M =$ `The quick brown fox jumps over the lazy dog.`, and discuss the potential vulnerabilities of hash functions in the context of the birthday paradox.

Q7. Secure Socket Layer (SSL) and Transport Layer Security (TLS): Explain the working of the SSL/TLS handshake protocol. Prove the security guarantees provided by SSL/TLS in the context of key exchange, data integrity, and confidentiality. Then, analyze how the POODLE attack compromises SSLv3 and explain why the forward secrecy feature was introduced in later versions of TLS.

Q8. Quantum Cryptography: Discuss the impact of quantum computing on classical cryptography. Explain how Shor's algorithm can factor large integers efficiently and break RSA encryption. Then, describe the working of Quantum Key Distribution (QKD) and prove how the BB84 protocol ensures secure key exchange, even in the presence of a quantum eavesdropper.

Q9. Advanced Encryption Standard (AES): Prove that AES is secure against brute-force attacks. Describe the full encryption process, including key expansion, substitution, permutation, and final XOR. Analyze the theoretical limits of AES with respect to the birthday bound and distinguish between the security of AES-128, AES-192, and AES-256.

Q10. Hybrid Cryptosystems: Design and analyze a hybrid cryptosystem combining RSA and AES for secure email communication. Prove the security of your design under the assumption that RSA is secure against chosen ciphertext attacks and that AES is semantically secure. Provide a detailed discussion of the performance trade-offs between symmetric and asymmetric encryption when used in this context.

***