AMERICAN UNIVERSITY OF BEIRUT

Applied Cryptography

CMPS 297AD/396AI, Fall 2025

Instructor: Nadim Kobeissi **Website:** https://appliedcryptography.page

Problem Set 3: Asymmetric Cryptography

Instructions: This problem set covers topics in provable security from parts 1.7[a] and 1.8[b] of the course. Submit your solutions as a neatly formatted PDF. You are encouraged to collaborate with classmates in studying the material, but your submitted solutions must be your own work. For proofs, clearly state your assumptions, steps, and conclusions.

[a] https://appliedcryptography.page/slides/1-7.pdf
[b] https://appliedcryptography.page/slides/1-8.pdf

1 Cryptographic Hardness and Real-World Implications (20 points)

1.1 Breaking Cryptography: Attack Scenarios (10 points)

1. (5 points) **The Cryptographic Apocalypse Scenario:** Imagine you wake up tomorrow to headlines: *"Breakthrough Algorithm Solves P vs NP - Computer Scientists Prove P = NP!"*
   (a) As the Chief Security Officer of a major bank, write a crisis response memo outlining which systems fail immediately, which have grace periods, and what emergency measures you would implement.
   (b) Design an alternative security model for online banking that could work in a post-P=NP world. What assumptions would you rely on instead?
   (c) Analyze why NP-complete problems, despite being *"hard,"* wouldn't save us in this scenario. What's the fundamental difference between NP-complete hardness and cryptographic hardness?
2. (5 points) **The Weak DH Parameters Problem:** A security researcher discovers that a popular cryptographic library has been generating Diffie-Hellman parameters where the prime p satisfies $p - 1$ having many small factors, making 75% of generated groups vulnerable to Pohlig-Hellman attacks that reduce the discrete log problem to much smaller subgroups.
   (a) Evaluate whether this discovery completely breaks Diffie-Hellman or only partially weakens it. Consider both the mathematical impact and practical deployment consequences.
   (b) Design a strategy for systems using this library: should they immediately regenerate all parameters, implement parameter validation, or pursue a different approach?
   (c) Compare this scenario to a hypothetical breakthrough that reduces the discrete logarithm problem in prime-order groups by a factor of 2^{20} but still leaves it exponential. Which vulnerability would be more urgent to address and why?

1.2 Discrete Logarithm Security Architecture (10 points)

1. (5 points) **The Weak Parameter Disaster:** Your security audit discovers that a legacy system has been using $p = 2047$ (which factors as 23×89) for Diffie-Hellman key exchange, and the generator $g = 2$.
   (a) Analyze exactly why this parameter choice is catastrophically weak. Estimate how long it would take an attacker with a modern laptop to break this system.
   (b) Design an emergency response plan: how do you migrate users to secure parameters while maintaining service availability?

(c) Compare the security implications if the system had instead used a proper 2048-bit prime but with a generator that only generates a small subgroup.

2. (5 points) **Elliptic Curve vs. Finite Field Trade-off Analysis:** You're designing a cryptographic protocol for IoT devices with severe computational and bandwidth constraints.

   (a) Compare elliptic curve and finite field DLP for your use case: which offers better security per bit of key size, and which offers better computational performance?

   (b) Analyze why index calculus attacks work against finite fields but not elliptic curves. How does this fundamental difference affect your security margins?

   (c) Design a hybrid approach that uses both elliptic curves and finite fields strategically. When might this provide benefits over using just one?

2 Diffie-Hellman in Hostile Environments (20 points)

2.1 Attack and Defense Scenarios (10 points)

1. (5 points) **The Perfect Man-in-the-Middle:** An attacker has complete control over the network between Alice and Bob, can modify any message, and can initiate connections that appear to come from either party.

   (a) Design the most effective man-in-the-middle attack against unauthenticated Diffie-Hellman. Your attack should be undetectable to Alice and Bob during the key exchange.

   (b) Alice and Bob have never met but each has the other's public key fingerprint written on a piece of paper. Design an authentication protocol that defeats your attack using only these fingerprints.

   (c) Compare your fingerprint-based solution to certificate authorities and web-of-trust models. What are the usability and security trade-offs?

   (d) The attacker now has quantum capabilities. How does this change your attack and defense strategies?

2. (5 points) **The Paranoid Whistleblower Scenario:** A whistleblower needs to securely communicate with a journalist. They assume the government monitors all internet traffic, has compromised most Certificate Authorities, and can perform man-in-the-middle attacks on any connection.

   (a) Design a key exchange protocol for this scenario using only methods available to ordinary civilians (no specialized hardware or pre-shared secrets).

   (b) Analyze what happens if the government can also compromise one of their devices after the key exchange. How can you provide forward secrecy?

   (c) Compare your solution to existing tools like Tor, Signal, and SecureDrop. What additional protection does your design provide?

2.2 Protocol Design Challenge (10 points)

1. (10 points) **SSH Trust-on-First-Use Analysis:** Your organization wants to deploy SSH across 10,000 servers, but the current TOFU model creates security and usability problems at scale.

   (a) Analyze specific attack scenarios where the TOFU model fails in practice. When are users most vulnerable?

   (b) Design an improved authentication model that maintains SSH's simplicity while providing better security guarantees than pure TOFU.

   (c) Compare your solution to proposals like DNS-based SSH public key distribution (SSHFP records) and OAuth-based SSH certificates. What are the deployment challenges for each approach?

3 Elliptic Curve Security Engineering (30 points)

3.1 Curve Selection Under Pressure (15 points)

1. (5 points) **The Government Backdoor Controversy:** You're the security architect for a new messaging app. Cryptographers are debating whether NIST P-256 contains a government backdoor, while Curve25519 offers better security properties but less widespread hardware support.
   (a) Analyze the specific concerns about NIST curves: what would a backdoor look like, and how could it be exploited without breaking the underlying mathematical problems?
   (b) Design a risk assessment framework for choosing between P-256 and Curve25519. What factors should influence your decision?
   (c) Your legal team reports that several countries require NIST-compliant cryptography for government sales. How does this constraint affect your technical decision?
   (d) Propose a solution that addresses both the backdoor concerns and the compliance requirements. What compromises would you make?
2. (5 points) **The Invalid Curve Attack Scenario:** A security researcher discovers that your ECDH implementation doesn't validate input points, making it vulnerable to invalid curve attacks.
   (a) Design a specific attack exploiting this vulnerability. What information can an attacker extract, and how long would the attack take?
   (b) Analyze why this attack works: what mathematical properties of elliptic curves does it exploit?
   (c) Develop a comprehensive input validation strategy that prevents this attack class. What performance impact does your solution have?
   (d) Compare this vulnerability to other implementation mistakes like reusing nonces in ECDSA. Which class of error is more dangerous in practice?
3. (5 points) **Mobile Performance Optimization Challenge:** Your mobile app needs to perform thousands of ECDH operations per minute on low-end smartphones, but battery life and performance are critical concerns.
   (a) Compare the performance characteristics of different elliptic curves for your use case. Consider both computational cost and memory usage.
   (b) Design an optimization strategy that balances security and performance. Would you use precomputed tables, special curve forms, or other techniques?
   (c) Analyze the security implications of your optimizations: what new attack surfaces do they create?
   (d) Evaluate whether quantum resistance should influence your current design decisions, given the mobile hardware lifecycle.

3.2 Implementation Vulnerability Analysis (15 points)

1. (5 points) **The PlayStation 3 Forensics Challenge:** You're a digital forensics expert investigating cryptocurrency theft. You discover that the thief's wallet software reused nonces in ECDSA signatures, similar to the PlayStation 3 vulnerability.
   (a) Design a forensic analysis procedure to recover the private key from blockchain transaction signatures. What information do you need, and how would you process it?
   (b) Estimate how many transactions with reused nonces you would need to guarantee key recovery. How does this depend on the specific nonce reuse pattern?
   (c) Develop a tool to scan existing blockchains for this vulnerability. What would you look for, and how would you optimize the search?
   (d) Analyze the broader implications: if wallet software commonly had this bug, what percentage of cryptocurrency could be at risk?
2. (5 points) **Side-Channel Attack Laboratory:** You're tasked with testing an embedded device's ECDSA implementation for side-channel vulnerabilities.
   (a) Design a timing attack against variable-time scalar multiplication. What information would you measure, and how would you extract the private key?

(b) Develop countermeasures that maintain performance while resisting your attack. What constant-time techniques would you implement?
(c) Analyze power analysis attacks: how would an attacker use power consumption traces to recover cryptographic keys?
(d) Evaluate the trade-offs between security and performance for different countermeasures. Which threats should you prioritize defending against?

3. (5 points) **The Ed25519 Validation Crisis:** You discover that two widely-used Ed25519 libraries accept different signatures as valid for the same message and public key, breaking interoperability.

(a) Investigate what causes this inconsistency: what validation steps do different implementations handle differently?
(b) Analyze the security implications: could an attacker exploit these differences to create practical attacks?
(c) Design a test suite to identify which Ed25519 implementations are compatible with each other. What edge cases would you test?
(d) Propose a strategy for the cryptographic community to resolve this issue without breaking existing deployments.

4 Applied Cryptography Case Studies (30 points)

1. (10 points) **Key Exchange Protocol Design** You are designing a secure messaging application that needs to establish encrypted communication channels between users who have never communicated before. The application must work on mobile devices with limited computational resources and intermittent network connectivity.

(a) Design a complete key exchange protocol using the cryptographic primitives from lectures 1.7 and 1.8. Your design should address:
- Initial key establishment between strangers
- Authentication to prevent man-in-the-middle attacks
- Forward secrecy for long-term security
- Efficiency for mobile devices

(b) Analyze the security properties of your protocol. What attacks does it defend against, and what are its limitations?
(c) Discuss how your protocol would handle practical issues like key fingerprint verification and key rotation.

2. (10 points) **Cryptocurrency Signature Scheme Analysis** A new cryptocurrency project is choosing between ECDSA and Ed25519 for transaction signatures. The system requirements include:

- High transaction throughput (thousands of signatures per second)
- Long-term security (system should remain secure for decades)
- Compatibility with hardware wallets and mobile devices
- Deterministic transaction signing for reproducibility

Analyze this decision:

(a) Compare ECDSA and Ed25519 for each requirement above. Which algorithm better meets each criterion and why?
(b) Discuss the implications of signature malleability. How does this affect each algorithm and why might it matter for cryptocurrency applications?
(c) Analyze the quantum resistance of both options. What migration path would you recommend for long-term security?
(d) Consider the ecosystem effects: existing wallet software, hardware support, and developer familiarity. How do these practical factors influence the decision?
(e) Make a final recommendation with justification, considering both technical and practical factors.

3. (10 points) **Secure Communication System Architecture** You are architecting a secure communication system for a large organization (10,000+ employees) that needs to protect against both external attackers and potential insider threats. The system must support real-time messaging, file sharing, and voice calls. Design and analyze a complete solution:

   (a) Specify your cryptographic algorithm choices for:
   - Key exchange protocols
   - Digital signature schemes
   - Symmetric encryption algorithms
   - Hash functions and MACs

   (b) Describe your key management architecture. How do you bootstrap trust, distribute keys, and handle key rotation?

   (c) Analyze your system's security properties against various attack scenarios:
   - Network eavesdropping
   - Server compromise
   - Endpoint compromise
   - Insider attacks

   (d) Discuss the performance implications of your design choices and how you would optimize for a large-scale deployment.

   (e) Evaluate your system's compliance with modern security standards and its readiness for post-quantum cryptography migration.

Bonus Challenge (20 extra points): The transition to post-quantum cryptography will require replacing current elliptic curve systems with quantum-resistant alternatives. Research and analyze one of the following aspects of this transition:

1. **NIST Post-Quantum Standards**: Analyze the recently standardized ML-KEM and ML-DSA algorithms. How do their key sizes, performance characteristics, and security assumptions compare to current ECC systems?
2. **Hybrid Classical/Post-Quantum Systems**: Describe approaches for combining classical and post-quantum algorithms during the transition period. What are the benefits and challenges of hybrid systems?
3. **Migration Timeline and Challenges**: Analyze the practical challenges of migrating existing systems (browsers, mobile apps, IoT devices) from ECC to post-quantum cryptography. What factors determine the migration timeline?

Your answer should include: current standardization status, performance comparisons with existing systems, deployment challenges, and recommendations for practitioners preparing for the post-quantum transition. Check the Optional Readings under the topic listing for the Post-Quantum Cryptography on the course website for helpful references!