23CSE331
CRYPTOGRAPHY

MOBILE COMMUNICATION SECURITY

Why Mobile Communication Security?

- The topic aligns with both academic interests and professional aspirations.
- It integrates network security, cryptography, IoT, and emerging technologies, providing rich research potential.
- The topic bridges multiple domains, allowing us to study both theoretical and practical aspects of security.
- Emerging technologies like IoT and 5G/6G add new layers of security complexity.
- Our goal is to understand how advanced cryptographic solutions can strengthen privacy, trust, and reliability in mobile communication.

Mobile IoT = Massive Communication + Weak Security

- In the Internet of Things (IoT), billions of small devices (phones, wearables, sensors, cars, etc.) communicate wirelessly.
- These devices often:
  - have limited computing power and memory,
  - send sensitive data (like health, payment, or location),
  - and rely on lightweight encryption (like RSA or ECC).
- Works fine today.
- But vulnerable in the quantum future — because quantum computers can break RSA, ECC, and Diffie-Hellman easily.
- So, mobile IoT networks become the first big victims of quantum attacks, since they're everywhere and use these classical protocols.

How to secure mobile IoT communication in the quantum era

- QKD (Quantum Key Distribution) – a physics-based way to securely share keys.
- PQC (Post-Quantum Cryptography) – classical algorithms designed to resist quantum attacks.
- AES-256-GCM – a symmetric cipher that remains strong even against quantum computers.
- So, "quantum" is not about making the IoT device quantum – it's about protecting the IoT devices from quantum-based attackers.

Quantum-Powered Security for Mobile Applications

- Mobile payment apps handle sensitive financial and personal data.
- Require strong authentication, secure transactions, and privacy preservation.
- Traditional cryptography (RSA, ECC) is at risk from emerging quantum computers.

Quantum Algorithms in Mobile Applications Security.

1. Quantum Key Distribution (QKD)

- Enables ultra-secure key exchange using photons.
- Detects eavesdropping instantly – ensures unbreakable encryption.

Quantum-Powered Security for Mobile Applications

2.CRYSTALS-Dilithium

- Post-quantum key exchange algorithm
- Provides fast, secure encryption for mobile transactions..

3.Quantum Random Number Generation (QRNG)

- Generates true random keys and tokens using quantum entropy.
- Eliminates predictable OTPs and session keys.

4.Quantum Key Agreement (QKA)

- Both users jointly generate encryption keys.
- Strengthens peer-to-peer and wallet communication security.

Quantum-based Algorithms in LIGHWEIGHT CRYPTOGRAPHY

1. Ascon (Lightweight Cipher)
   - Ascon works well on small devices like IoT sensors and smart cards.
   - It is strong against quantum attacks and other security threats.
2. CRYSTALS-Dilithium (Digital Signatures)
   - It uses lattice math to create a secure, verifiable digital stamp.It's highly efficient for mobile devices to confirm a message is authentic.

Quantum-based Algorithms in LIGHWEIGHT CRYPTOGRAPHY

3. CRYSTALS-Kyber (Key Exchange)

- This algorithm is incredibly power and memory efficient for
- Also uses lattice math, like Dilithium, but for securely sharing keys instead of signing.
- Based on hard mathematical problems that quantum computers cannot solve efficiently.
- Fast enough to run on smartphones and IoT devices.

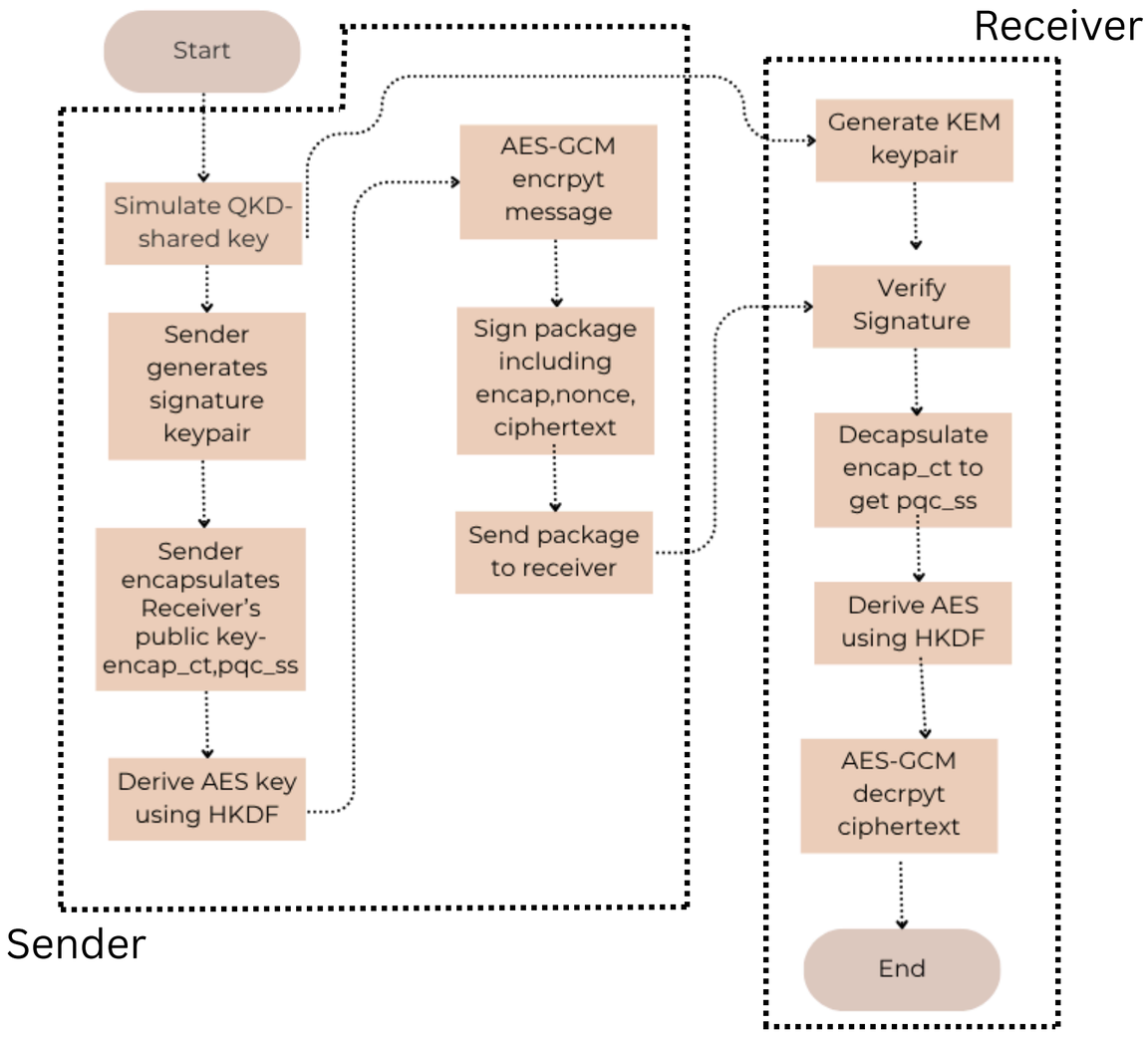
5G/6G COMMUNICATION

- Quantum computing is a threat current cryptography like RSA and AES. Post-Quantum Cryptography (PQC) offers quantum-resistant solutions but faces challenges. These risks are critical for 6G applications requiring secure, low-latency communication.
- Key solutions include:
  - PQC algorithms for encryption and signatures.
  - Quantum Key Distribution (QKD) for secure key exchange.
  - Hybrid cryptography combining PQC with AES-256-GCM.
- Challenges remain in performance, interoperability, key management, and standards.
- Future work focuses on optimizing hybrid schemes.

Benefits

- Provides confidentiality (PQC + QKD).
- Adds authenticity and integrity with digital signatures (Dilithium-like), allowing verification of sender and detection of tampering.
- Explicit sender authentication and message integrity via signatures.
- More robust to man-in-the-middle and forgery attacks.

Flowchart


Receiver
Start
Simulate QKD-shared key
Sender generates signature keypair
Sender encapsulates Receiver's public key-encap_ct,pqc_ss
Derive AES key using HKDF
AES-GCM encrpyt message
Sign package including encap,nonce, ciphertext
Send package to receiver
Generate KEM keypair
Verify Signature
Decapsulate encap_ct to get pqc_ss
Derive AES using HKDF
AES-GCM decrpyt ciphertext
End
Sender

Implementation

Comparison

Paper Implementation

- Implements post-quantum crytpography for key encapsulation.
- Uses quantum-derived key for symmetric encryption.
- No digital signature (authentication) scheme is implemented in the described algorithm.

Code Implementation

- Implements post-quantum KEM (Mock Kyber) for key establishment.
- Simulates QKD for shared secret generation.
- Adds a digital signature algorithm (Mock Dilithium) to sign encrypted messages, enhancing integrity and authenticity.

THANK YOU