Analisis Perbandingan Kyber (*Post Quantum*) dan ECDH Dalam Pembentukan Kunci

Muhammad Roihan 13522152
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13522152@std.stei.itb.ac.id

***Abstract*—Makalah ini menganalisis perbandingan antara algoritma *key exchange* klasik ECDH dan algoritma post-quantum Kyber dalam konteks pembentukan kunci. Penelitian dilakukan melalui *benchmarking* terhadap berbagai varian dari kedua algoritma: P-256 dan P-384 untuk ECDH, serta Kyber-512, Kyber-768, dan Kyber-1024 untuk Kyber. Hasil menunjukkan bahwa ECDH unggul signifikan dalam aspek performa dan penggunaan memori, dibandingkan Kyber. Namun, ancaman komputer kuantum mengubah paradigma keamanan: ECDH sangat rentan terhadap Shor's algorithm yang dapat memecahkan *Elliptic Curve Discrete Logarithm Problem* dalam waktu polinomial, sementara Kyber tetap tahan terhadap serangan kuantum berkat fondasi matematika *Module Learning With Errors* (MLWE). Trade-off ini menunjukkan bahwa meskipun ECDH masih optimal untuk aplikasi saat ini, transisi ke algoritma post-quantum seperti Kyber menjadi keharusan untuk mengantisipasi era *quantum computing*.**

Keywords*—ECDH, Kyber, *Post-Quantum Cryptography*, *Key Exchange*, *Quantum Resistance*, MLWE, *Benchmarking

I. PENDAHULUAN

Keamanan komunikasi digital modern bergantung pada protokol *key exchange* yang memungkinkan dua pihak membentuk *shared secret* melalui kanal publik yang tidak aman. Selama beberapa dekade terakhir, Elliptic Curve Diffie-Hellman (ECDH) telah menjadi standar industri untuk *key exchange* karena efisiensinya yang superior dibanding RSA tradisional. ECDH mendasarkan keamanannya pada kesulitan matematis *Elliptic Curve Discrete Logarithm Problem* (ECDLP), yang diyakini tidak dapat dipecahkan secara efisien oleh komputer klasik.

Namun, perkembangan teknologi *quantum computing* menghadirkan ancaman terhadap fondasi kriptografi modern. Pada tahun 1994, Peter Shor memperkenalkan algoritma kuantum yang dapat memecahkan *integer factorization* dan *discrete logarithm problem* dalam waktu polinomial. Ini berarti semua sistem kriptografi berbasis ECDH, RSA, dan DSA akan menjadi *obsolete* ketika quantum computer dengan kapasitas memadai tersedia. Meskipun *quantum computer* dengan kemampuan *cryptographically-relevant* belum ada saat ini, prinsip "harvest now, decrypt later" mengharuskan organisasi mulai mengadopsi kriptografi post-quantum untuk melindungi data sensitif jangka panjang.

Transisi dari ECDH ke Kyber bukan tanpa tantangan. Algoritma post-quantum umumnya memiliki *computational overhead* yang lebih tinggi, ukuran kunci yang lebih besar, dan *memory footprint* yang signifikan dibanding kriptografi klasik. Bagi industri yang telah mengoptimalkan infrastruktur mereka untuk ECDH, pertanyaan kritis muncul: apakah trade-off performa ini dapat diterima? Bagaimana dampaknya terhadap user experience, throughput sistem, dan bandwidth requirements?

Makalah ini bertujuan menjawab pertanyaan tersebut melalui analisis yang membandingkan ECDH dan Kyber dari berbagai dimensi: execution time, throughput, *memory usage*, *key sizes*, *bandwidth requirements*, dan *security characteristics* terhadap *classical* maupun *quantum attacks*. Benchmarking dilakukan terhadap multiple security levels untuk memberikan perbandingan yang adil: P-256 dan P-384 untuk ECDH serta Kyber-512, Kyber-768, dan Kyber-1024 untuk *post-quantum* KEM.

Hasil analisis ini diharapkan dapat memberikan insight praktis bagi praktisi keamanan, *system architects*, dan *decision makers* dalam merencanakan strategi migrasi ke post-quantum cryptography, dengan memahami konsekuensi teknis dan operational dari transisi ini.

II. LANDASAN TEORI

A. ECDH

Elliptic Curve Diffie-Hellman (ECDH) adalah protokol *key agreement* yang memungkinkan dua pihak untuk membentuk *shared secret* melalui kanal publik tanpa *prior shared information*. ECDH merupakan varian dari protokol Diffie-Hellman klasik yang menggunakan *elliptic curve cryptography* untuk mencapai security level yang sama dengan ukuran kunci yang jauh lebih kecil. ECDH beroperasi pada grup titik-titik dari *elliptic curve* yang didefinisikan pada *finite field*. Sebuah *elliptic curve* atas *finite field* didefinisikan dengan persamaan:

$$y^2 = x^3 + ax + b \pmod p$$

Alur ECDH :

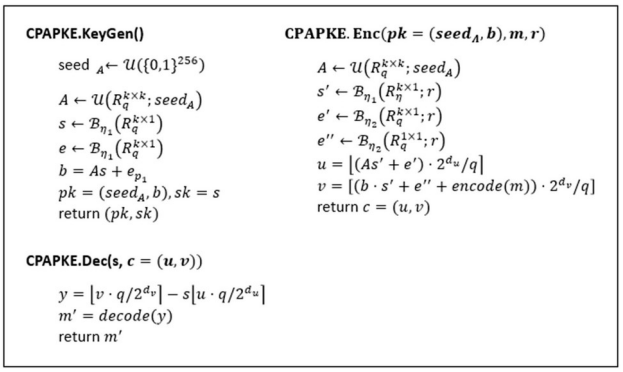
- Kedua pihak (Alice dan Bob) sepakat menggunakan elliptic curve E dan base point G dengan order n
- Alice memilih private key a (random integer) dan menghitung public key A = aG
- Bob memilih *private key* b (random integer) dan menghitung *public key* B = bG
- Alice dan Bob bertukar *public keys*
- Alice menghitung *shared key* sebagai aB
- Bob menghitung *shared key* sebagai bA

Keamanan ECDH bergantung pada Elliptic Curve Discrete Logarithm Problem (ECDLP): given points G dan Q = kG, sangat sulit secara komputasional untuk menemukan k.

B. Kyber

CRYSTALS-Kyber adalah *Key Encapsulation Mechanism* (KEM) berbasis lattice yang dirancang untuk tahan terhadap serangan quantum computer. Berbeda dengan key exchange tradisional seperti ECDH yang bersifat interaktif (kedua pihak berkontribusi pada s*hared secret*), KEM adalah protokol non-interaktif dimana satu pihak (*encapsulator*) mengenkapsulasi *random symmetric key* menggunakan *public key* pihak lain, menghasilkan ciphertext yang hanya bisa didekapsulasi oleh pemilik private key.

Protokol Kyber KEM:



Gambar 2.1 Protokol Kyber

Sumber :

https://www.mdpi.com/cryptography/cryptography-08-00015/article_deploy/html/images/cryptography-08-00015-g001.png

C. Pembentukan Kunci

Pembentukan kunci adalah proses kriptografi dimana dua atau lebih pihak membentuk *shared cryptographic key* yang kemudian digunakan untuk *symmetric encryption* atau *message authentication*. Terdapat dua kategori utama:

1. *Key Agreement* (ECDH)
   - Kedua pihak berkontribusi secara aktif pada pembentukan shared secret
   - *Compromise* terhadap *long-term keys* tidak mengompromikan *past session keys*
   - Memerlukan dua round-trip (public key exchange dari kedua pihak)
2. *Key Encapsulation* (Kyber)
   - Satu pihak mengenkapsulasi random key menggunakan public key pihak lain
   - Lebih efisien dalam komunikasi (satu round-trip)

D. Shor's Algorithm

Shor's algorithm, yang diperkenalkan oleh Peter Shor pada tahun 1994, adalah algoritma kuantum yang secara fundamental mengancam keamanan kriptografi modern. Algoritma ini dapat memecahkan dua masalah matematis yang menjadi fondasi sebagian besar sistem kriptografi: *integer factorization* dan *discrete logarithm problem* dalam waktu polinomial.

Shor's algorithm memanfaatkan dua fenomena kuantum fundamental:

1. ***Quantum superposition***: Kemampuan qubit untuk berada dalam multiple states secara simultan
2. ***Quantum Fourier Transform* (QFT)**: Transformasi yang efisien untuk menemukan periodisitas dalam fungsi

Untuk memecahkan integer factorization dari bilangan N, Shor's algorithm bekerja sebagai berikut:

- Pilih bilangan random a dimana 1 < a < N dan gcd(a, N) = 1
- Temukan period r dari fungsi f(x) = a^x mod N menggunakan QFT
- Jika r genap dan a^(r/2) ≠ -1 mod N, maka:
  - Faktor pertama: gcd(a^(r/2) - 1, N)
  - Faktor kedua: gcd(a^(r/2) + 1, N)

III. ANALISIS DAN PEMBAHASAN

A. Penjelasan Implementasi

Bagian ini melakukan analisis antara algoritma *key exchange* klasik (ECDH) dan *post-quantum* (Kyber) untuk mengevaluasi performa, efisiensi, dan karakteristik keamanan masing-masing. Implementasi dilakukan dengan metodologi *benchmarking* yang meliputi:

1. **Pengujian Multiple Variants**: Menguji beberapa varian dari masing-masing algoritma (P-256, P-384 untuk ECDH dan Kyber-512, Kyber-768, Kyber-1024) untuk mendapatkan perbandingan yang adil pada berbagai *security level*.
2. ***Benchmarking***: Setiap operasi dijalankan sebanyak 1000 iterasi dengan 100 *warmup runs* untuk menghilangkan *noise* dan mendapatkan hasil yang akurat. Statistik yang dikumpulkan meliputi mean, median, standard deviation, minimum, dan maximum time.
3. **Verifikasi *Correctness***: Setiap implementasi diverifikasi untuk memastikan bahwa *key*

exchange menghasilkan *shared secret* yang identik antara kedua pihak (Alice dan Bob untuk ECDH dan *encapsulation*/*decapsulation* untuk Kyber).

4. **Pengukuran Tambahan:** Melakukan pengukuran terhadap waktu eksekusi, *throughput* (*operations per second*), penggunaan memori, ukuran kunci, dan *bandwidth requirements* untuk memberikan gambaran lengkap tentang karakteristik masing-masing algoritma.

B. Library

1. Cryptography

Library cryptography adalah library kriptografi standar untuk Python yang menyediakan implementasi berbagai algoritma kriptografi termasuk ECDH (Elliptic Curve Diffie-Hellman).
Kegunaan dalam program:

- Mengimplementasikan *key generation* untuk *elliptic curve*
- Melakukan operasi *key exchange* menggunakan protokol ECDH
- Menyediakan berbagai kurva eliptik standar (SECP256R1, SECP384R1)

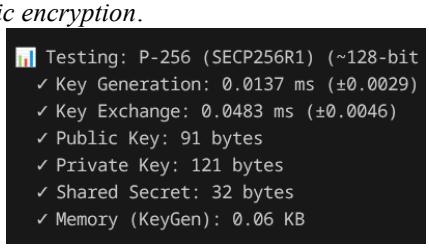
2. kyber_py

Library kyber_py adalah implementasi Python dari algoritma Crystal-Kyber, salah satu pemenang kompetisi NIST Post-Quantum Cryptography Standardization.
Kegunaan dalam program:

- Implementasi Key Encapsulation Mechanism (KEM) menggunakan Kyber
- Menyediakan tiga varian security level: Kyber512, Kyber768, Kyber1024
- Melakukan operasi keygen, *encapsulation*, dan *decapsulation*

C. Analisis Hasil P-256

P-256 atau SECP256R1 adalah *elliptic curve* yang didefinisikan dalam standar NIST FIPS 186-4. Kurva ini beroperasi pada finite field dengan ukuran 256-bit dan menyediakan *security level* setara dengan 128-bit *symmetric encryption*.



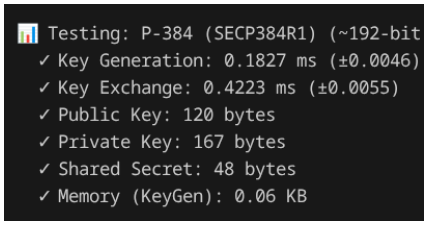
Gambar 3.1 *Hasil Pengujian P-256*
Sumber : Dokumen Penulis

Analisis:

- **Performa** : P-256 menunjukkan performa yang sangat baik dengan *key generation* hanya 0.0137 ms dan *key exchange* 0.0483 ms. Standard deviation yang rendah (±0.0029 dan ±0.0046) mengindikasikan konsistensi yang tinggi.
- **Ukuran Kunci**: *Public key* sebesar 91 bytes dan *private key* 121 bytes sangat efisien untuk transmisi dan penyimpanan. *Shared secret* 32 bytes (256 bits) sesuai dengan security level 128-bit.
- **Memori**: Penggunaan memori hanya 0.06 KB menunjukkan P-256 sangat cocok untuk sistem dengan *resource* terbatas.

D. Analisis Hasil P-384

P-384 atau SECP384R1 adalah *elliptic curve* dengan ukuran 384-bit yang menyediakan *security level* setara dengan 192-bit *symmetric encryption*



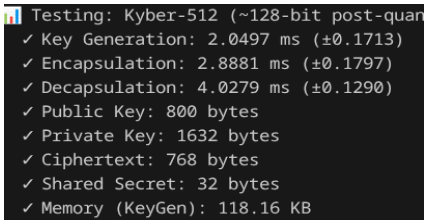
Gambar 3.2 *Hasil Pengujian P-384*
Sumber : Dokumen Penulis

Analisis:

- **Performa**: P-384 memerlukan waktu 13x lebih lama untuk *key generation* (0.1827 ms vs 0.0137 ms) dan 9x lebih lama untuk *key exchange* (0.4223 ms vs 0.0483 ms) dibanding P-256. Ini adalah trade-off yang diharapkan untuk mendapatkan *security level* 192-bit.
- **Ukuran Kunci:** *Public key* (120 bytes) dan *private key* (167 bytes) lebih besar 32% dan 38% dibanding P-256. *Shared secret* 48 bytes (384 bits) memberikan margin keamanan yang lebih besar.
- **Memori**: Penggunaan memori hanya 0.06 KB menunjukkan P-384 sangat cocok untuk sistem dengan *resource* terbatas.
- **Rekomendasi Penggunaan**: P-384 cocok untuk aplikasi dengan *requirement security* sangat tinggi dan tidak terlalu sensitif terhadap *latency*.

E. Analisis Hasil Kyber-512

Kyber-512 adalah varian Kyber dengan parameter *security level* 1, yang menyediakan keamanan setara dengan AES-128.



Gambar 3.3 *Hasil Pengujian Kyber-512*
Sumber : Dokumen Penulis

Analisis:

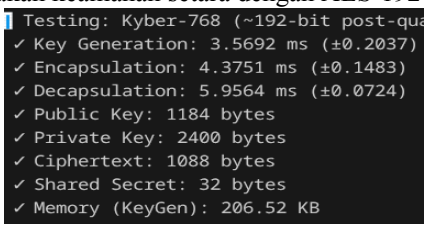
- **Performa**: Kyber-512 memerlukan waktu 150x lebih lama untuk *key generation* (2.05 ms vs 0.0137 ms) dan 60x lebih lama untuk operasi

key exchange dibanding P-256. Ini adalah *cost* yang harus dibayar untuk mendapatkan *quantum resistance*.

- **Total Handshake Time**: Total waktu untuk complete handshake (keygen + encaps + decaps) adalah 8.97 ms, yang masih *acceptable* [1].
- **Ukuran Kunci**: *Public key* 800 bytes (9x lebih besar dari P-256) dan *private key* 1632 bytes (13x lebih besar). Ciphertext 768 bytes harus ditransmisikan sebagai bagian dari handshake, menambah *overhead bandwidth*.
- **Memori**: Penggunaan memori 118.16 KB jauh lebih besar dibanding P-256 (0.06 KB), hampir 2000x lipat. Ini menjadi pertimbangan penting untuk *devices* dengan RAM terbatas.
- **Standard Deviation**: Variasi waktu eksekusi (±0.17-0.18 ms) lebih tinggi dibanding ECDH, mengindikasikan operasi ini kemungkinan lebih rentan terhadap *system noise.*

F. Analisis Hasil Kyber-768

Kyber-768 adalah varian dengan *security level* 3, menyediakan keamanan setara dengan AES-192



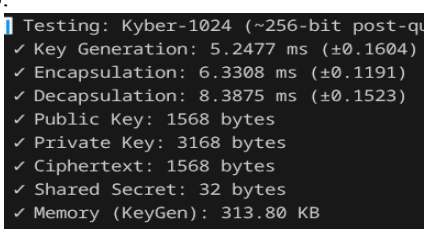
Gambar 3.4 *Hasil Pengujian Kyber-768*
Sumber : Dokumen Penulis

Analisis:

- **Performa**: Kyber-768 1.7x lebih lambat dari Kyber-512 dengan *key generation* 3.57 ms dan total handshake 13.9 ms. Peningkatan waktu ini *reasonable* untuk mendapatkan 64-bit a*dditional security*.
- **Ukuran Kunci**: *Public key* 1184 bytes dan ciphertext 1088 bytes menghasilkan total bandwidth 2272 bytes per handshake, yaitu 12.5x lebih besar dari P-256.
- **Memori**: 206.52 KB memory footprint menjadi concern untuk sistem dengan *resource* terbatas.

G. Analisis Hasil Kyber-1024

Kyber-1024 adalah varian dengan *security level* tertinggi (level 5), menyediakan keamanan setara dengan AES-256.

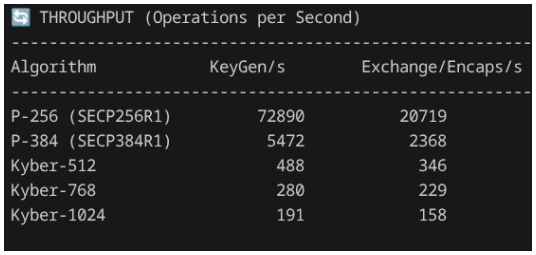


Gambar 3.5 *Hasil Pengujian P-256*
Sumber : Dokumen Penulis

Analisis:

- **Performa:** Kyber-1024 adalah yang paling lambat dengan total *handshake time* 19.97 ms, 160x lebih lambat dari P-256. Ini adalah *extreme security measure* untuk kebanyakan aplikasi.
- **Ukuran Kunci**: *Private key* 3168 bytes memerlukan *secure storage* yang significant. Untuk aplikasi dengan ribuan kunci, s*torage requirement* menjadi non-trivial.
- **Memory Intensive**: 313.80 KB *memory usage* menjadi terlalu tinggi untuk kebanyakan sistem.

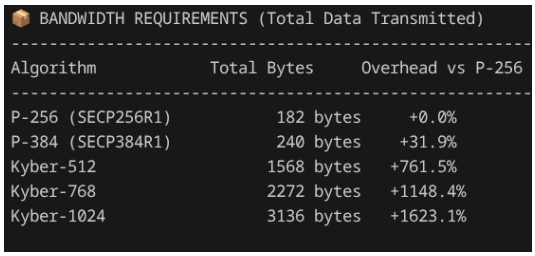
H. Perbandingan Throughput



Gambar 3.6 *Hasil Pengujian Throughput*
Sumber : Dokumen Penulis

- **ECDH Dominan dalam Throughput**: P-256 menunjukkan throughput yang sangat bagus dengan 72,890 *key generations per second* dan 20,719 key exchanges per second. Ini berarti P-256 dapat menangani puluhan ribu TLS *handshakes per second*, sangat cocok untuk high-traffic web servers.
- **P-384 *Performance Drop***: P-384 mengalami penurunan *throughput* signifikan menjadi 5,472 *keygen*/s (13x lebih lambat) dan 2,368 *exchange*/s (9x lebih lambat). Penurunan ini proporsional dengan penambahan kompleksitas komputasi.
- **Kyber *Throughput Characteristics***:
  - **Kyber-512**: 488 keygen/s dan 346 encaps/s adalah 150x dan 60x lebih lambat dari P-256
  - **Kyber-768**: 280 keygen/s (260x lebih lambat dari P-256)
  - **Kyber-1024**: 191 keygen/s (380x lebih lambat dari P-256)
- ***Real-world Context***: Meskipun *throughput* Kyber jauh lebih rendah, untuk *most client-server applications* (web browsing, API calls, email), latency tambahan 10-20ms tidak terasa oleh user. Namun untuk *high-frequency trading* atau *real-time gaming*, *latency* ini bisa signifikan.

H. Perbandingan Bandwidth Requirements



Gambar 3.6 *Hasil Pengujian Throughput*
Sumber : Dokumen Penulis

- **ECDH Bandwidth**:
  - P-256: 182 bytes (91 bytes × 2 public keys) sangat *compact*
  - P-384: 240 bytes hanya 32% *overhead* untuk security increase
  - ECDH optimal untuk b*andwidth-constrained environments*
- **Kyber Bandwidth**:
  - **Kyber-512**: 1568 bytes = 800 (public key) + 768 (ciphertext)
  - **Kyber-768**: 2272 bytes = 1184 + 1088
  - **Kyber-1024**: 3136 bytes = 1568 + 1568
- **Hybrid *Approach Impact***: Jika menggunakan hybrid (ECDH + Kyber), total bandwidth menjadi:
  - P-256 + Kyber-768 = 182 + 2272 = 2454 bytes
  - Masih *acceptable* untuk *most applications*
- **Compression Opportunity**: Public keys dan ciphertext tidak *compressible*. Jadi b*andwidth overhead* adalah *fixed cost* yang tidak bisa dihindari.

I. Analisis Keamanan Komputer Klasik

ECDH (P-256 dan P-384):

- **P-256**: Menyediakan ~128-bit *classical security level* berdasarkan kesulitan Elliptic Curve Discrete Logarithm Problem (ECDLP). *Security level* ini dianggap sangat aman dengan margin keamanan yang *comfortable*. Untuk memecahkan P-256, attacker perlu ~2^128 operasi, yang secara praktis *impossible* dengan teknologi *classical computing.*
- **P-384**: Menyediakan ~192-bit *security level*, memberikan margin keamanan yang *extremely high*. *Attacker perlu* ~2^192 operasi, sama halnya dengan P-256 ini masih *impossible* untuk teknologi sekarang.

Kyber:

- **Kyber-512, 768, 1024**: Semua varian Kyber menyediakan 128-bit, 192-bit, dan 256-bit *classical security level respectively*. Security mereka berdasarkan *Module Learning With Errors* (MLWE) *problem*, yang diyakini sulit untuk *classical computers*.

I. Analisis Keamanan Komputer Kuantum

Salah satu ancaman jika menggunakan komputer kuantum adalah menggunakan Shor's algorithm dapat memecahkan *Discrete Logarithm Problem* dan *Integer Factorization* dalam *polynomial time*. Ini berarti:
ECDH (P-256 dan P-384):

- **SANGAT RENTAN** terhadap *quantum computers*
- P-256 (128-bit security) dapat dipecahkan dengan ~2,330 logical qubits [2]
- P-384 (192-bit security) dapat dipecahkan dengan ~4,719 logical qubits
- Shor's algorithm *reduces security* dari 2^128 *operations* menjadi *polynomial time*

Kyber (Semua Varian):

- **TAHAN** terhadap algoritma kuantum saat ini.
- Berdasarkan Learning With Errors (LWE) problem yang tidak dapat dipecahkan secara efisien oleh *quantum computer*

IV. KESIMPULAN

Berdasarkan analisis yang telah dilakukan, terdapat beberapa kesimpulan yang bisa ditarik:

- ECDH, khususnya P-256, menunjukkan superioritas yang signifikan dengan *throughput* 72,890 *key generations per second* dan *total handshake time* hanya 0.062 ms lebih dari 150 kali lebih cepat dibanding Kyber-512. Efisiensi memori ECDH (0.06 KB) dan bandwidth requirement yang minimal (182 bytes untuk P-256) menjadikannya solusi ideal untuk *resource constrained devices*
- Keunggulan performa ECDH menjadi tidak relevan menghadapi jika ancaman *quantum computing*. Shor's algorithm dapat memecahkan ECDLP dalam *polynomial time* dengan hanya ~2,330 *logical qubits* untuk P-256 dan ~4,719 *logical qubits* untuk P-384
- Sebaliknya, Kyber mempertahankan keamanannya terhadap *quantum attacks* karena fondasi matematika Module-LWE tidak dapat dipecahkan secara efisien oleh *quantum algorithms* yang diketahui saat ini. Security margin yang diberikan oleh Kyber-768 (setara AES-192) atau Kyber-1024 (setara AES-256) memberikan *confidence* level tinggi untuk data *protection* jangka panjang.

V. UCAPAN TERIMA KASIH

Dengan penuh rasa syukur, penulis ingin mengucapkan puji dan syukur kepada Tuhan Yang Maha Esa atas segala rahmat, karunia, serta taufik dan hidayah-Nya, yang telah memandu dan memberikan keberkahan sehingga penulis berhasil menyelesaikan makalah berjudul "Analisis perbandingan Kyber (*Post Quantum*) dan ECDH dalam pembentukan kunci". Makalah ini disusun sebagai bagian dari tugas mata kuliah Kriptografi

IF4020.

Penulis juga mengucapkan terima kasih yang setinggi tingginya kepada Dr. Ir. Rinaldi, M.T., sebagai dosen pengajar dalam mata kuliah Kriptografi IF4020. Terima kasih atas dedikasi, bimbingan, dan pengajaran yang telah diberikan selama satu semester ini, memberikan kontribusi besar dalam pembentukan pemahaman kami sebagai mahasiswa. Penulis juga beterimakasih kepada seluruh pihak yang telah berkontribusi baik secara langsung maupun tidak langsung terhadap kelancaran penulisan makalah ini yang namanya tidak bisa disebutkan satu persatu. Penulis juga ingin menyampaikan permohonan maaf apabila terdapat kesalahan dalam penulisan makalah ini. Semoga makalah ini dapat bermanfaat dan memberikan sumbangan positif dalam pemahaman mata kuliah Kriptografi IF4020.

REFERENCES

[1] Jakob Nielsen, "Response Times: The 3 Important Limits", https://www.nngroup.com/articles/response-times-3-important-limits/ diakses pada tanggal 12 Januari 2025.

[2] Martin Roetteler et al, "Quantum resource estimates for computing elliptic curve discrete logarithms", https://doi.org/10.48550/arXiv.1706.06752 diakses pada tanggal 12 Januari 2025.

[3] Roberto Avanzi et al, "CRYSTALS-Kyber Algorithm Specifications And Supporting Documentation", https://pq-crystals.org/kyber/data/kyber-specification-round3-20210804.pdf diakses pada tanggal 12 Januari 2025.

[4] Peter W. Shor, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer", https://doi.org/10.1137/S0097539795293172 diakses pada tanggal 12 Januari 2025.

[5] Pierre Philip du Preez, "Understanding EC Diffie-Hellman", https://medium.com/swlh/understanding-ec-diffie-hellman-9c07be338d4a diakses pada tanggal 12 Januari 2025.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 12 Januari 2025

Muhammad Roihan 13522152