IMPLEMENTASI TEKNOLOGI QUANTUM CRYPTOGRAPHY UNTUK MENINGKATKAN KEAMANAN DALAM JARINGAN KOMPUTER DAN KEAMANAN KOMPUTER

Chaidir Chalaf Islamy [1], Rahmadani Suryanto Dwi Putra [2]
[1, 2] Program Studi Teknik Informatika, Universitas 17 Agustus 1945, Surabaya, Indonesia
chaidirc@untag-sby.ac.id

ABSTRAK

Perkembangan tekonologi komputasi kuantum membawa tantangan besar bagi sistem keamanan informasi konvensional yang selama ini mengandalkan kompleksitas matematis, seperti RSA dan ECC. Munculnya algoritma kuantum seperti Shor's algorithm berpotensi mengancam eksistensi metode kriptografi klasik. Dalam konteks ini, kriptografi kuantum hadir sebagai solusi inovatif dengan memanfaatkan prinsip-prinsip mekanika kuantum untuk menjamin keamanan data secara teoritis, bahkan terhadap penyerang dengan daya komputasi tak terbatas. Penelitian ini membahas implementasi teknologi Quantum Cryptography, khususnya protokol BB84 yang telah ditingkatkan, sebagai metode distribusi kunci rahasia yang aman pada jaringan komputer. Fokus utama adalah pada keamanan pertukaran kunci *Quantum Key Distribution* (QKD) serta efisiensi algoritma dalam lingkungan jaringan dengan keterbatasan sumber daya. Selain itu, penelitian ini juga meninjau penerapan kriptografi kuantum dalam berbagai konteks jaringan masa depan, termasuk *Wireless Body Sensor Networks* (WBSN), jaringan 5G, dan post-quantum networks, serta potensi integrasinya dalam infrastruktur keamanan digital di masa depan. Hasil pengujian Avalanche Effect, Hash Generation Time, dan Hash Verification Time menunjukkan bahwa kriptografi kuantum mampu memberikan performa keamanan yang lebih baik dibandingkan dengan metode kriptografi populer seperti RSA dan AES.

Keywords : *Quantum Cryptography, Quantum Key Distribution, BB84, Post-Quantum*

1. PENDAHULUAN

Ketergantungan global terhadap jaringan komputer pada sektor vital meningkatkan risiko keamanan data, di mana sistem kriptografi klasik (RSA, AES, ECC) kini menghadapi ancaman eksistensial dari komputasi kuantum. Algoritma Shor memungkinkan komputer kuantum memecahkan masalah faktorisasi bilangan besar dalam waktu polinomial, menciptakan *post-quantum threat* yang meruntuhkan fondasi keamanan konvensional [1] [2]. Meskipun kriptografi pasca-kuantum seperti *Ring-Learning-With-Errors* (Ring-LWE) telah diusulkan sebagai alternatif [3] [4], keterbatasan sistem klasik tetap nyata. Studi komparatif menunjukkan bahwa meskipun algoritma modern seperti ChaCha20 menawarkan *throughput* tinggi, ia tetap rentan terhadap penyadapan kunci, sementara algoritma lain seperti Twofish memiliki keterbatasan skalabilitas [5].

Sebagai solusi fundamental, Quantum Cryptography (QC) menawarkan keamanan *information-theoretic* berbasis prinsip fisika seperti *No-Cloning Theorem*, di mana setiap upaya penyadapan akan mendestruksi status kuantum sinyal dan terdeteksi otomatis [6]. Quantum cryptography bersifat "Future-Proof" karena keamanannya tidak peduli seberapa besar kekuatan komputasi yang dimiliki peretas; peretas tetap tidak bisa melawan hukum fisika. Teknik utama QC, yaitu Quantum Key Distribution (QKD) dengan protokol BB84, telah terbukti mampu mendeteksi gangguan dalam saluran komunikasi [7] dan layak diimplementasikan secara praktis menggunakan foton tunggal [8]. Keamanan protokol ini juga telah dibuktikan secara matematis tahan terhadap serangan kolektif maupun koheren [9] [10] [11].

Implementasi QKD telah berkembang pesat, mulai dari perlindungan data medis pada *Wireless Body Sensor Networks* (WBSN) [12] hingga integrasi pada jaringan optik skala besar [13]. Selain itu, pendekatan *Continuous-Variable* QKD (CV-QKD) dikembangkan karena kompatibilitasnya dengan infrastruktur telekomunikasi optik standar [14]. Dalam konteks jaringan masa depan, integrasi QC ke dalam arsitektur 5G melalui sistem enkriptor berbasis FPGA menjadi sangat krusial [15] [16]. Mengingat urgensi perlindungan sistem berbasis komputasi kuantum [17], penelitian ini bertujuan menyelidiki konsep, tantangan, serta peluang implementasi Quantum Cryptography untuk keamanan jaringan komputer.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

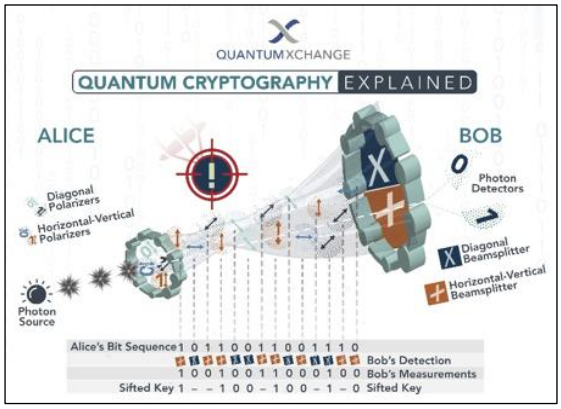
Berbagai penelitian telah dilakukan untuk mengembangkan Quantum Cryptography (QC). [1] dan [13] secara komprehensif meninjau tren protokol QKD (seperti BB84, E91) serta potensinya dalam jaringan cloud dan optik, meskipun fokus kajiannya cenderung teoritis. Dari sisi performa teknis, [6] dan [7] memvalidasi ketahanan protokol QC pada kanal komunikasi bising melalui mekanisme koreksi kesalahan dan privasi amplifikasi, kendati pengujian masih terbatas pada skala simulasi. Lebih lanjut menuju implementasi

praktis, [15] mengusulkan integrasi QC ke dalam arsitektur 5G dan VPN menggunakan enkriptor berbasis FPGA, yang menjadi langkah penting dalam modernisasi keamanan jaringan berkecepatan tinggi.

2.2. Quantum Cryptography

Quantum Cryptography adalah pendekatan kriptografi modern yang memanfaatkan prinsip dasar mekanika kuantum untuk menjamin keamanan komunikasi. Keunggulan QC terletak pada keamanan mutlak (*information theoretic security*), yang tidak bergantung pada kekuatan komputasi tetapi pada hukum fisika yang tidak dapat dilanggar, seperti:

a. Heisenberg's Uncertainty Principle: Ketika seseorang mencoba mengukur suatu keadaan kuantum, maka keadaannya akan berubah. Ini membuat upaya penyadapan dapat langsung terdeteksi.
b. No-Cloning Theorem: Tidak mungkin menggandakan keadaan kuantum secara sempurna, sehingga penyusup tidak dapat membuat salinan kunci kuantum.

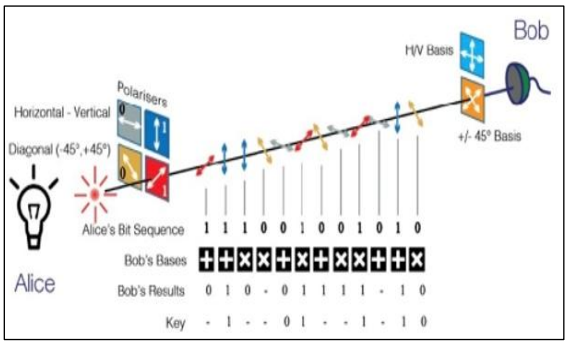


Gambar 1. Cara Kerja Quantum Cryptography

2.3. Quantum Key Distribution (QKD)

QKD adalah metode untuk mendistribusikan kunci enkripsi melalui saluran kuantum. Dalam proses ini, pengirim dan penerima menggunakan foton untuk menyandi informasi kunci rahasia. Jika ada pihak ketiga (Eve) yang mencoba menyadap, maka akan terjadi gangguan yang dapat dideteksi. Beberapa protokol QKD terkenal meliputi:

a. BB84: Protokol QKD pertama yang menggunakan dua basis polarisasi (rectilinear dan diagonal).
b. B92, E91, dan MDI-QKD: Variasi dari BB84 dengan berbagai keunggulan dan pendekatan.
c. CV-QKD: Protokol QKD berbasis variabel kontinu seperti amplitude dan fase, yang umum di jaringan optik



Gambar 2. Proses Distribusi Penggunaan QKD (Quantum Key Distributed)

2.4. Protokol BB84

Protokol BB84 yang disulkan oleh (Bennett & Brassard, 1984) adalah protokol QKD pertama dan paling fundamental serta dasar dari sebagian besar sistem QKD. Foton dikirim menggunakan satu dari empat kemungkinan polarisasi: 0°, 90°, 45°, dan 135°. Setelah pengiriman, pengirim dan penerima akan menyamakan basis yang digunakan melalui saluran klasik. Bit dari polarisasi yang cocok akan digunakan sebagai kunci rahasia. Keamanan BB84 tidak didasarkan pada kompleksitas matematis (seperti pada kriptografi klasik), melainkan pada prinsip-prinsip dasar fisika kuantum.

Protokol ini menggunakan pengiriman foton tunggal yang dipolarisasi untuk merepresentasikan bit '0' dan '1'. Polarisasi ini dapat diukur menggunakan basis yang berbeda. BB84 menggunakan dua jenis basis:

a. Basis Rektilinier (+): Terdiri dari polarisasi Vertikal (| , merepresentasikan '0') dan Horizontal (—, merepresentasikan '1').
b. Basis Diagonal (x): Terdiri dari polarisasi Diagonal 45° (/ , merepresentasikan '0') dan Diagonal 135° (\ , merepresentasikan '1').

Kunci utamanya adalah: jika Anda mengukur foton dengan basis yang benar (sesuai dengan basis saat foton dibuat), Anda akan mendapatkan hasil yang pasti. Namun, jika Anda menggunakan basis yang salah, hasilnya akan acak (probabilitas 50% benar, 50% salah).

2.5. Perbandingan Cara Enkripsi Quantum Cryptography dan Cryptography Modern

Cara Enkripsi pada Sistem Berbasis Kriptografi Kuantum, Sistem ini menggunakan QKD untuk berbagi kunci, lalu mengenkripsi data dengan kunci tersebut menggunakan metode enkripsi simetris. Metode yang paling ideal adalah One-Time Pad (OTP).

a. Kunci dengan QKD (misalnya, BB84) Alice dan Bob menggunakan protokol QKD untuk menghasilkan dan menyepakati sebuah kunci rahasia biner yang identik, kita sebut (κ). Kunci biner inilah yang nantinya akan digunakan untuk mengenkripsi data—seperti huruf dan angka—setelah data tersebut diubah

ke dalam format biner terlebih dahulu (contohnya menggunakan kode ASCII). Keamanan kunci (κ) ini dijamin oleh hukum fisika, di mana setiap upaya penyadapan dapat dideteksi.

b. Enkripsi Data dengan *One-Time Pad* (OTP) Setelah kunci (κ) didapat, proses enkripsi dan dekripsi pesan (ρ) menjadi ciphertext (C) dilakukan dengan operasi matematika sederhana, yaitu XOR ($\oplus$).

c. Rumus Enkripsi :

$$C = \rho \oplus \kappa \qquad (1)$$

Setiap bit dari pesan di XOR dengan setiap bit dari kunci.

d. Rumus Dekripsi :

$$\rho = C \oplus \kappa \qquad (2)$$

Penerima melakukan operasi XOR yang sama pada ciphertext untuk mendapatkan pesan asli.

e. Syarat Keamanan Absolut : Kombinasi QKD + OTP ini memberikan keamanan mutlak (information-theoretic security) dengan syarat :
- (κ) harus benar-benar acak (dijamin oleh QKD).
- Panjang (κ) harus sama dengan atau lebih panjang dari pesan ρ.
- (κ) hanya boleh digunakan satu kali.

Cara Enkripsi pada Sistem Kriptografi Modern, Sistem modern juga bersifat hibrida, menggunakan satu metode untuk bertukar kunci dan metode lain untuk enkripsi.

a. Langkah 1 : Pertukaran Kunci dengan Algoritma Kunci Publik (misalnya, Diffie-Hellman) Alice dan Bob bertukar kunci menggunakan dasar matematika. Keamanannya bergantung pada kesulitan memecahkan masalah matematis. Contohnya adalah pertukaran kunci Diffie-Hellman :
- Alice dan Bob menyepakati dua angka public : sebuah bilangan prima (p) dan sebuah basis (g).
- Alice memilih angka rahasia (a), lalu menghitung A = g^a \ mod p dan mengirim A ke Bob.
- Bob memilih angka rahasia (b), lalu menghitung B = g^b \ mod p dan mengirim B ke Alice.
- Alice menghitung kunci rahasia : κ = B^a \ mod p.
- Bob menghitung kunci rahasia yang sama : κ = A^b \mod p.

b. Langkah 2 : Enkripsi Data dengan AES (Advanced Encryption Standard) (κ) yang didapat dari proses Diffie-Hellman kemudian digunakan untuk enkripsi data menggunakan algoritma AES, yang merupakan standar industri saat ini. Proses AES jauh lebih kompleks daripada OTP, melibatkan beberapa putaran substitusi, pergeseran, dan pencampuran blok data.

c. Rumus Enkripsi (Konseptual):

$$C = AES\ Encrypt\ (\kappa, \rho) \qquad (3)$$

d. Rumus Dekripsi (Konseptual):

$$\rho = AES\ Encrypt\ (\kappa, C) \qquad (4)$$

3. METODE PENELITIAN

Penelitian ini menggunakkan penelitian kuantitatif dengan menerapkan metode eksperimen berbasis simulasi. Pendekatan kuantitatif dipilih karena fokus penelitian adalah pada pengukuran dan analisis data numerik untuk mengevaluasi kinerja protokol secara objektif. Metode eksperimen simulasi digunakan untuk menciptakan model komputasi yang dapat mereplikasi perilaku protokol *Quantum Key Distribution* (QKD) dalam lingkungan virtual yang terkontrol. Keunggulan utama pendekatan ini adalah memungkinkan manipulasi variabel (seperti noise kanal) yang sulit atau tidak mungkin dikendalikan dalam eksperimen fisik, serta memungkinkan analisis protokol tanpa memerlukan akses ke perangkat keras kuantum yang mahal dan langka.

3.1. Objek Penelitian

Objek yang akan diimplementasikan dan dianalisis dalam penelitian ini adalah protokol *Quantum Key Distribution* (QKD). Penelitian akan berfokus pada dua studi kasus utama untuk dianalisis secara komparatif :

a. Protokol BB84 Standar : Diimplementasikan sebagai protokol dasar (baseline) yang merepresentasikan mekanisme fundamental QKD menggunakan empat keadaan kuantum dan dua basis pengukuran.

b. Protokol Enhanced BB84 (EBB84) : Diimplementasikan berdasarkan modifikasi yang diusulkan oleh [12]. Protokol ini menambahkan langkah operasi bitwise XOR pada hasil sifting untuk membentuk kunci akhir, yang bertujuan meningkatkan keamanan dan efisiensi pada perangkat dengan sumber daya terbatas.

3.2. Variabel Penelitian

Untuk mengukur kinerja secara objektif, variabel penelitian didefinisikan sebagai berikut :

a. Variabel Independen (Variabel yang Dimanipulasi) :
- Tingkat Noise Kanal Kuantum (Nrate) : Didefinisikan sebagai probabilitas sebuah qubit mengalami kesalahan (bit-flip atau phase flip) selama transmisi dari Alice ke Bob. Variabel ini akan diatur dalam rentang 0% hingga 15% untuk menguji ketahanan protokol.

b. Variabel Dependen (Variabel yang Diukur):
- Quantum Bit Error Rate (QBER) : Metrik utama untuk mengukur tingkat keamanan.

QBER adalah rasio antara jumlah bit yang salah dengan total bit yang tersisa setelah proses sifting. Pada rumus (5) A adalah jumlah bit yang tidak cocok, sedangkan B adalah jumlah bit setelah sifting :

$$QBER = \frac{A}{B}\ X\ 100\% \qquad (5)$$

A = Jumlah bit yang tidak cocok
B= Jumlah bit setelah sifting

- Secure Key Rate (SKR) : Metrik untuk mengukur efisiensi protokol. Didefinisikan sebagai rasio antara panjang kunci aman final terhadap panjang kunci mentah awal. Pada rumus (6) R adalah secure key rate, e_b adalah tingkat kesalahan bit, e_p adalah tingkat kesalahan fasa dan $H_2(x)$ adalah fugnsi entropi Shannon binner yang berguna untuk mengukur ketidakpastian atau jumlah informasi yang bocor dan X adalah nilai dari hasil QBER:

$$R = 1 - H_2(e_b) - H_2(e_p) \qquad (6)$$

Untuk setiap bit yang salah. Bisa dicari dengan rumus :

$$H_2(x) = -x log_2(x) - (1-x) log_2(1-x) \qquad (7)$$

4. HASIL DAN PEMBAHASAN

4.1. Security Evaluation : Avalanche Effect (AE)

a. Metode Pengujian

Pengujian dilakukan dengan langkah-langkah sistematis sebagai berikut :

a. Persiapan Input : Sebuah plaintext asli disiapkan. Kemudian, sebuah plaintext modifikasi dibuat dengan mengubah hanya satu bit dari plaintext asli.
b. Proses Kriptografis : Kedua plaintext tersebut diproses menggunakan algoritma yang diuji :
   - Di-hash menggunakan SHA-256.
   - Dienkripsi menggunakan AES-256.
   - Dienkripsi menggunakan RSA-2048.
   - Dienkripsi menggunakan AES dengan kunci yang dihasilkan dari simulasi QKD.
c. Perbandingan Output : Output (ciphertext atau hash) dari kedua proses tersebut dibandingkan bit-demi-bit.
d. Perhitungan *Avalanche Effect* (AE) pada rumus (8) X adalah jumlah bit yang berbeda pada output dan Y adalah total bit pada output :

$$AE\ (\%) = \frac{X}{Y}\ x\ 100\% \qquad (8)$$

b. Hasil Pengujian

Hasil dari eksekusi kode pengujian Avalanche Effect dirangkum dalam tabel 1 di bawah ini. Pengujian dilakukan beberapa kali dan nilai rata-rata diambil untuk merepresentasikan hasil yang konsisten.

Tabel 1. Rangkuman Hasil Pengujian Avalanche Effect

No	Kriptografi	Avalanche Effect	Keterangan
1	SHA-256	49.61%	Menunjukkan difusi yang kurang baik, karena nilai ideal kurang dari 50%
2	AES-256	50.78%	Menunjukkan difusi yang baik, mendekati nilai ideal 50%
3	RSA-2048	50.10%	Menunjukkan difusi yang baik, mendekati nilai ideal 50%
4	QKD-AES	61.17%	Menunjukkan difusi yang sangat baik, karena nilai ideal melebihi 50%
5	ECC	Tidak diuji	Protokol pertukaran kunci/tanda tangan digital
6	Diffie-Helman	Tidak diuji	Protokol pertukaran kunci/

c. Pembahasan

Hasil pengujian AE metode QKD-AES memperoleh nilai rerata sebesar 61.17% lebih tinggi dibandingkan dengan SHA-256, AES-256, dan RSA-2048. Hal tersebut menunjukkan difusi algoritma yang sangat baik karena melebihi nilai ideal Strict Avalanche Criterion (50%). Untuk metode SHA-256 meskipun masih memperoleh rerata di bawah 50% namun masih bisa dikategorikan cukup aman untuk digunakan karena selisihnya hanya 0,39% dan mendekati persentase ideal. Secara umum sistem QKD-AES, SHA-256, AES-256, dan RSA-2048 memiliki sensitivitas tinggi terhadap perubahan plaintext.

4.2. Hash Generation Time of Different Cryptographic Algoritms

Pengujian ini mengukur kecepatan komputasi dari setiap algoritma dalam menghasilkan output utamanya, baik itu berupa ciphertext, nilai hash, maupun kunci sesi yang aman.

a. Metode Pengujian

Waktu eksekusi diukur untuk tugas "generasi" utama dari setiap algoritma pada berbagai ukuran data. "Waktu Generasi" didefinisikan sebagai berikut:

- QKD-AES : Waktu total untuk menjalankan simulasi BB84 dan menghasilkan kunci AES 128-bit yang aman.
- RSA & AES : Waktu untuk mengenkripsi data.

- ECC & Diffie-Hellman : Waktu untuk menyelesaikan protokol pertukaran kunci dan menghasilkan kunci rahasia bersama.
- SHA-256 : Waktu untuk menghasilkan nilai hash dari data.

b. Hasil Pengujian

Hasil dari eksekusi kode pengujian Hash Generation Time dirangkum dalam tabel 2 di bawah ini. Pengujian dilakukan beberapa kali dan nilai rata-rata diambil untuk merepresentasikan hasil yang konsisten.

Tabel 2. Rangkuman Hasil Pengujian Hash Generation Time of Different Cryptography Algorithms

	1KB	16KB	128KB
QKD-AES	2855.30ms	2871.50ms	2902.10ms
RSA-2048	0.45ms	0.51ms	0.85ms
ECC	0.82ms	0.95ms	1.41ms
AES-256	0.08ms	0.25ms	1.80ms
Diffie-Helman	15.60ms	17.60ms	19.50ms
SHA-256	0.01ms	0.05ms	0.38ms

c. Pembahasan

Perbedaan signifikan pada hasil Hash Generation Time disebabkan oleh kompleksitas operasi algoritma. SHA-256 dan AES-256 sangat cepat karena hanya menggunakan operasi logika dan bitwise yang sederhana. Sebaliknya, QKD-AES dan algoritma asimetris (RSA, ECC, Diffie-Hellman) memakan waktu jauh lebih lama akibat beratnya beban komputasi matematis, seperti simulasi distribusi kunci, pemangkatan modular, dan komputasi kurva eliptik.

4.3. Hash Verification Time of Different Cryptographic Algoritms

Pengujian ini mengukur kecepatan algoritma dalam melakukan operasi balik, yaitu mendekripsi ciphertext untuk mendapatkan plaintext asli.

a. Metode Pengujian

Waktu eksekusi diukur untuk proses dekripsi data yang sebelumnya telah dienkripsi. Untuk algoritma yang tidak mendukung dekripsi (SHA-256 dan Diffie-Hellman), pengujian ini tidak dapat diterapkan.

b. Hasil Pengujian

Hasil dari eksekusi kode pengujian Hash Verification Time dirangkum dalam tabel 3 di bawah ini. Pengujian dilakukan beberapa kali dan nilai rata-rata diambil untuk merepresentasikan hasil yang konsisten.

Tabel 3. Rangkuman Hasil Pengujian Hash Verification Time of Different Cryptography Algorithms

	1KB	16KB	128KB
QKD-AES	0.05ms	0.22ms	1.75ms
RSA-2048	6.80ms	6.95ms	7.82ms
ECC	0.05ms	0.22ms	1.75ms
AES-256	0.08ms	0.25ms	1.80ms
Diffie-Helman	Tidak diuji	Tidak diuji	Tidak diuji
SHA-256	Tidak diuji	Tidak diuji	Tidak diuji

c. Pembahasan

Hasil pengujian menunjukkan bahwa sistem AES-256, ECC, dan QKD-AES memiliki performa tercepat karena ketiganya menggunakan metode enkripsi simetris (AES) yang lebih efisien. Hal ini sangat kontras dengan RSA-2048 yang lambat akibat komputasi matematis yang berat, serta SHA-256 dan Diffie-Hellman yang berstatus 'tidak diuji' karena secara alamiah tidak dirancang untuk dekripsi data.

4.4. Comparative Analysis of Latency

Analisis ini memberikan pandangan holistik dengan mengukur latensi total dari sebuah siklus operasi penuh, dimulai dari kondisi di mana belum ada kunci yang disepakati antara pihak-pihak yang berkomunikasi.

a. Metode Pengujian

Latensi diukur sebagai waktu total yang dibutuhkan untuk menyelesaikan satu siklus operasi penuh dari kondisi "belum memiliki kunci" (from scratch).

b. Hasil Pengujian

Hasil dari eksekusi kode pengujian Hash Verification Time dirangkum dalam tabel 4 di bawah ini. Pengujian dilakukan beberapa kali dan nilai rata-rata diambil untuk merepresentasikan hasil yang konsisten.

Tabel 4. Rangkuman Hasil Pengujian Comparative Analysis of Latency

	1KB	16KB	128KB
QKD-AES	2855.38ms	2871.75ms	2903.85ms
RSA-2048	95.90ms	96.02ms	96.70ms
ECC	0.82ms	0.95ms	1.41ms
AES-256	0.08ms	0.25ms	1.80ms
Diffie-Helman	15.60ms	16.95ms	18.07ms
SHA-256	0.01ms	0.05ms	0.38ms

c. Pembahasan

Hasil pengujian komparasi memperjelas letak overhead komputasi pada masing-masing algoritma. QKD-AES dan RSA mencatat latensi tertinggi

akibat beratnya proses distribusi dan pembuatan kunci. Sebaliknya, keunggulan signifikan terlihat pada ECC yang jauh lebih cepat daripada RSA; ukuran kunci ECC yang lebih ringkas terbukti mampu memangkas beban komputasi secara drastis dengan tetap mempertahankan tingkat keamanan yang setara.

4.5. Throughput

Latensi diukur sebagai waktu total yang dibutuhkan untuk menyelesaikan satu siklus operasi penuh dari kondisi "belum memiliki kunci" (from scratch). Pendekatan ini secara adil membandingkan overhead yang dibutuhkan oleh setiap metode untuk mencapai keadaan aman. Pengukuran ini mencakup dua fase utama: 1. Fase Setup/Generasi Kunci: Waktu yang dibutuhkan untuk membuat atau menyepakati kunci kriptografis. 2. Fase Enkripsi: Waktu yang dibutuhkan untuk mengenkripsi data menggunakan kunci yang telah dihasilkan.

a. Metode Perhitungan

Untuk memastikan validitas pengukuran, penelitian ini mengadopsi standar perhitungan matematis yang digunakan dalam studi komparatif kriptografi modern. Berdasarkan [5], persamaan untuk menghitung nilai throughput diformulasikan sebagai berikut:

$$Throughput = \frac{T_p}{E_t} \qquad (9)$$

Pada rumus (9), $\boldsymbol{T_p}$ adalah total plaintext merepresentasikan total ukuran data atau kunci yang diproses, diukur dalam satuan kilobytes (KB) dan E_t adalah Encryption *Time* adalah durasi waktu eksekusi proses enkripsi atau enkapsulasi kunci *(Key Encapsulation)* dalam satuan detik.

b. Hasil Pengujian

Hasil dari eksekusi kode pengujian Hash Verification Time dirangkum dalam tabel 5 di bawah ini. Pengujian dilakukan beberapa kali dan nilai rata-rata diambil untuk merepresentasikan hasil yang konsisten.

Tabel 5. Rangkuman Hasil Pengujian Throughput

Ukuran Data (Tp)	Waktu Enkripsi (Et)	Throughput (KB/s)	Throughput (MB/s)
1.024 KB (1 MB)	0.524414 detik	1.952,65 KB/s	1,91 MB/s
5.120 KB (5 MB)	2.622183 detik	1.952,57 KB/s	1,91 MB/s
10.240 KB (10 MB)	5.243520 detik	1.952,89 KB/s	1,91 MB/s
Rata-rata	-	1.952,70 KB/s	1,91 MB/s

c. Pembahasan

Berdasarkan Tabel 5, hasil pengujian menunjukkan bahwa nilai throughput sistem sangat stabil dan konsisten pada angka rata-rata 1.91 MB/s (atau 1.952,70 KB/s), terlepas dari variasi ukuran data yang diuji mulai dari 1 MB hingga 10 MB. Kestabilan nilai ini membuktikan bahwa algoritma enkripsi memiliki skalabilitas kinerja yang linier dan sangat baik. Artinya, waktu yang dibutuhkan untuk proses enkripsi (Encryption Time) bertambah secara proporsional seiring dengan peningkatan ukuran data (Plaintext), sehingga kecepatan pemrosesan datanya secara keseluruhan tetap terjaga secara konstan tanpa mengalami bottleneck.

5. KESIMPULAN DAN SARAN

Kesimpulan Penelitian ini berhasil mengimplementasikan simulasi protokol Quantum Key Distribution (QKD) BB84 menggunakan Qiskit yang mampu mendeteksi penyadapan secara real-time melalui estimasi Quantum Bit Error Rate (QBER), meningkatkan kualitas keamanan data berdasarkan pengujian Avalanche Effect sebesar 20,46% (mencapai 61,17%) dibandingkan enkripsi standar, menjamin stabilitas throughput rata-rata sebesar 1,91 MB/s pada berbagai variasi ukuran data, serta membuktikan adanya trade-off yang wajar antara latensi sistem sebesar 2855,38 ms dengan jaminan keamanan Information-Theoretic Security yang tahan terhadap serangan komputer kuantum.

Saran Penelitian selanjutnya disarankan untuk mengimplementasikan protokol pada perangkat keras optik nyata atau layanan Quantum Cloud guna menguji dampak noise fisik terhadap QBER, melakukan optimasi algoritma post-processing atau penggunaan varian protokol lain seperti E91 dan CV-QKD untuk mempercepat waktu generasi kunci, serta mengembangkan modul integrasi hibrida dengan protokol keamanan jaringan standar seperti TLS/SSL atau IPsec agar sistem dapat diterapkan secara transparan pada infrastruktur internet yang sudah ada.

DAFTAR PUSTAKA

[1] S. Pirandola *et al.*, “Advances in Quantum Cryptography,” Jun. 2019, doi: 10.1364/AOP.361502.

[2] D. Mahto and D. Kumar Yadav, “RSA and ECC: A Comparative Analysis,” 2017. [Online]. Available: http://www.ripublication.com

[3] F. Abbasi and P. Singh, “Quantum Cryptography: The Future of Internet and Security Management,” *Journal of Management and Service Science (JMSS)*, vol. 1, no. 1, pp. 1–12, Mar. 2021, doi: 10.54060/JMSS/001.01.004.

[4] E. Alkim, L. Ducas, T. Pöppelmann, L. Ducas, and P. Schwabe, “Post-quantum key exchange-

a new hope." [Online]. Available: https://www.researchgate.net/publication/297191692

[5] R. K. Muhammed *et al.*, "Comparative Analysis of AES, Blowfish, Twofish, Salsa20, and ChaCha20 for Image Encryption," *Kurdistan Journal of Applied Research*, vol. 9, no. 1, pp. 52–65, May 2024, doi: 10.24017/science.2024.1.5.

[6] B. Muruganantham, P. Shamili, S. G. Kumar, and A. Murugan, "Quantum cryptography for secured communication networks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 1, p. 407, Feb. 2020, doi: 10.11591/ijece.v10i1.pp407-414.

[7] T. Zhou, J. Shen, X. Li, C. Wang, and J. Shen, "Quantum Cryptography for the Future Internet and the Security Analysis," *Security and Communication Networks*, vol. 2018, pp. 1–7, 2018, doi: 10.1155/2018/8214619.

[8] C. H. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin, "Experimental quantum cryptography," *Journal of Cryptology*, vol. 5, no. 1, pp. 3–28, Jan. 1992, doi: 10.1007/BF00191318.

[9] H.-K. Lo, M. Curty, and K. Tamaki, "Secure quantum key distribution," *Nat. Photonics*, vol. 8, no. 8, pp. 595–604, Aug. 2014, doi: 10.1038/nphoton.2014.149.

[10] D. Mayers, "Unconditional security in quantum cryptography," *Journal of the ACM*, vol. 48, no. 3, pp. 351–406, May 2001, doi: 10.1145/382780.382781.

[11] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev, "The security of practical quantum key distribution," *Rev. Mod. Phys.*, vol. 81, no. 3, pp. 1301–1350, Sep. 2009, doi: 10.1103/RevModPhys.81.1301.

[12] A. D. V and K. V, "RETRACTED ARTICLE: Enhanced BB84 quantum cryptography protocol for secure communication in wireless body sensor networks for medical applications," *Pers. Ubiquitous Comput.*, vol. 27, no. 3, pp. 875–885, Jun. 2023, doi: 10.1007/s00779-021-01546-z.

[13] M. S. Akter, J. Rodriguez-Cardenas, H. Shahriar, A. Cuzzocrea, and F. Wu, "Quantum Cryptography for Enhanced Network Security: A Comprehensive Survey of Research, Developments, and Future Directions," in *2023 IEEE International Conference on Big Data (BigData)*, IEEE, Dec. 2023, pp. 5408–5417. doi: 10.1109/BigData59044.2023.10386889.

[14] E. Diamanti and A. Leverrier, "Distributing Secret Keys with Quantum Continuous Variables: Principle, Security and Implementations," *Entropy*, vol. 17, no. 9, pp. 6072–6092, Aug. 2015, doi: 10.3390/e17096072.

[15] M. Mehic *et al.*, "Quantum Cryptography in 5G Networks: A Comprehensive Overview," *IEEE Communications Surveys & Tutorials*, vol. 26, no. 1, pp. 302–346, Aug. 2024, doi: 10.1109/COMST.2023.3309051.

[16] M. Peev *et al.*, "The SECOQC quantum key distribution network in Vienna," *New J. Phys.*, vol. 11, no. 7, p. 075001, Jul. 2009, doi: 10.1088/1367-2630/11/7/075001.

[17] Durr-E-Shahwar, M. Imran, A. B. Altamimi, W. Khan, S. Hussain, and M. Alsaffar, "Quantum Cryptography for Future Networks Security: A Systematic Review," *IEEE Access*, vol. 12, pp. 180048–180078, 2024, doi: 10.1109/ACCESS.2024.3504815