TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PURWANCHAL CAMPUS

A MINOR PROJECT PROPOSAL ON UNIVERSAL FRAMEWORK FOR BENCHMARKING CRYPTOGRAPHIC ALGORITHMS

SUBMITTED BY:

AMOGH BHATTARAI (078BCT005)

ARPANA SHARMA GHIMIRE(078BCT016)

BIBISHA BASNET(078BCT022)

JENISHA SHRESTHA(078BCT039)

SUBMITTED TO:

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

PURWANCHAL CAMPUS

DHARAN, NEPAL

February 27, 2025

ACKNOWLEDGEMENT

We are deeply grateful to the Deputy Head of the Department Mr. Pukar Karki for his invaluable guidance, insightful suggestions and continuous support throughout the preparation of this project. We would like to express our sincere gratitude to our supervisor Mr. Binay Lal Shrestha for his constructive feedbacks that have been instrumental in shaping the direction of this research.

We also extend our gratitude to IOE, Purwanchal Campus and its faculty for providing a conducive learning environment and essential resources that contributed significantly to this project. Their sense of commitment to academic excellence as well as a call towards disseminating knowledge always inspire us. Their individual and collective guidance has significantly contributed towards the betterment and refinement of this project. Last but not least, we immensely appreciate our friends and family for their patience, motivation, and unwavering encouragement throughout this journey.

AMOGH BHATTARAI [PUR078BCT005]

ARPANA SHARMA GHIMIRE [PUR078BCT016]

BIBISHA BASNET [PUR078BCT022]

JENISHA SHRESTHA [PUR078BCT039]

ABSTRACT

Cryptographic algorithms play a crucial role in ensuring data security, confidentiality, and integrity across various digital media formats such as text, images, and videos. However, the performance of these algorithms can vary significantly depending on factors such as data type, size, encryption key length, processing power, and system resources. This project aims to develop a universal benchmarking framework to evaluate and compare cryptographic algorithms based on key performance metrics, including encryption and decryption time, throughput, latency, CPU cycles, memory consumption, energy efficiency, and avalanche effect.

The benchmarking process involves implementing and testing various symmetric and asymmetric encryption algorithms on diverse data types (text, images, and videos) across different hardware and software environments. The results will provide insights into the efficiency, scalability, and security strength of each algorithm under varying conditions. By analyzing these metrics, we aim to identify the most suitable cryptographic algorithms for different applications, optimizing both security and performance.

Additionally, the benchmarking framework developed in this study will consider hardware acceleration, such as the use of GPU processing and FPGA optimization, to enhance performance. It will serve as a standardized tool for future cryptographic performance evaluations and can be used for assessing the impact of cryptographic optimizations on real-world applications. The framework will also take into account various security levels to ensure that the chosen algorithm maintains a robust defense against known vulnerabilities while achieving the desired efficiency.

Keywords: encryption, GPU, FPGA, energy efficiency, hardware acceleration, scalability,

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Description
AES	Advanced Encryption Standard
SHA	Secure Hash Algorithm
RSA	Rivest–Shamir–Adleman
IS	Information Security

CHAPTER 1

INTRODUCTION

The term 'cryptos,' meaning 'hidden,' and 'graphene,' meaning 'to write,' form the basis for the origin of the word 'Cryptography. It is thus the science of securing communication by converting plain text into ciphertext, ensuring that only the intended recipient can understand the message. Cryptographic algorithms are the backbone of securing data in computer systems and communication networks. They are mathematical techniques designed to ensure confidentiality, integrity, authenticity, and non-repudiation of data. With the growing emphasis on data security in today's digital era, cryptographic algorithms play a crucial role in protecting sensitive information. However, different types of media files, such as text, images, audio, and video present unique challenges for encryption and decryption due to their varying sizes, structures, and processing demands. This project's significance is underscored by its potential to create a universal framework for benchmarking cryptographic algorithms on various media files, comparing their performance in terms of speed, resource usage, and scalability.

1.1 Background

The exchange of sensitive information through digital platforms, such as online banking, e-commerce, and telecommunications, has significantly increased. This has heightened the demand for cryptographic techniques that balance security and performance. While algorithms like AES and hashing techniques have been the gold standard for decades, emerging paradigms like lightweight and post-quantum cryptography are reshaping the cryptographic landscape. However, a standardized method to assess these algorithms across varying scenarios is currently lacking. Benchmarking the cryptographic algorithms serves not only as a performance monitor but also as a tool for discovering vulnerabilities, uncovering hidden inefficiencies, and inspiring innovation in cryptographic algorithms. It provides valuable insights into how algorithms perform under various conditions, aiding in informed decision-making for real-world implementations.

1.2 Problem Statement

A standardized way to evaluate cryptographic algorithms across different medias is missing. This absence hinders accurate assessment of algorithm performance and suitability in real-world situations. This makes it difficult for developers and network engineers to make informed decisions about cryptographic implementations. Our project can define a universal, standardized framework that consistently evaluates algorithms across multiple dimensions viz. execution time , memory usage , energy consumption , security strength , scalability with data size.

1.3 Gap Identification

Lack of Standardization Across Benchmarks: Existing benchmarking studies often focus on specific algorithms, use different metrics, and methodologies, which makes it hard to directly compare results across studies. For example, some focus on execution time, others on memory, but few combine these metrics comprehensively.

1.4 Motivation

Researchers often use different methodologies to evaluate cryptographic algorithms. A universal framework provides standardized metrics, facilitating more accurate comparisons. In addition to this, framework breaks down complex cryptographic concepts into measurable outputs, making them easier to grasp. A universal framework can serve as a bridge between academic research and real-world applications, fostering collaboration between researchers and industry professionals as well.

1.5 Objective

- **Primary Objective:** To develop a universal framework for benchmarking cryptographic algorithms across diverse media types including text, audio, images, and more, ensuring standardized evaluation and comparison.
- **Secondary Objective:** To develop a pipeline for all sort of media inputs that can be fed into existing algorithms.

CHAPTER 2

RELATED THEORY

In recent years, malicious assaults and information leakage have emerged as critical concerns in nearly every area of information and communication technology (ICT). Information security (IS) plays a pivotal role in safeguarding enterprise data, ensuring regulatory compliance, and maintaining competitive positions in the market. Cryptographic algorithms are at the heart of IS, offering robust mechanisms to protect sensitive information against cyber threats. In addressing the privacy and security needs of Industry 5.0 (the Fifth Industrial Revolution), the cryptographic techniques explore how cryptographic algorithms can enhance security policies and examine information security (IS) challenges and solutions.[1][2]

2.1 Classification of cryptographic techniques:

The cryptographic techniques are broadly classified into following types:

2.1.1 Symmetric Encryption:

It uses the same key for both encryption and decryption. Example: AES (Advanced Encryption Standard) and ChaCha20.

- **AES (Advanced Encryption Standard):** It is the widely used symmetric algorithm that operates on 128-bit blocks and supports 128, 192, or 256-bit keys and used in data encryption, secure communications, and VPNs. AES-256 will be essential for quantum resistance, as quantum computers could break AES-128.
- **ChaCha20:** is a stream cipher known for its simplicity and security. Designed for fast performance and strong encryption. Ideal for mobile devices and low-power IoT devices due to its efficiency.

2.1.2 Asymmetric Encryption:

It uses a pair of keys: a public key for encryption and a private key for decryption. Example: RSA(Rivest–Shamir–Adleman).

- **RSA(Rivest–Shamir–Adleman):** Widely used for secure key exchange and digital signatures, and ECC (Elliptic Curve Cryptography) – a more efficient alternative to RSA with smaller key sizes, often used in modern applications like secure communication and blockchain.

2.1.3 Hashing:

Hash functions generate a fixed-size hash value from input data, ensuring integrity. Example: SHA-256 (Secure Hash Algorithm 256-bit).

- **SHA-256 (Secure Hash Algorithm 256-bit):** A cryptographic hash function widely used for data verification.

Similarly, these algorithms are fundamental for ensuring data confidentiality, integrity, and authentication in various systems:

- **Playfair Cipher**:A classical encryption technique that uses a 5x5 matrix of letters to encrypt digraphs (pairs of letters), offering simplicity and historical significance.
- **Lightweight Cryptography:** Focuses on efficient cryptographic algorithms. Optimized for computation, memory, and power consumption. Essential for resource-constrained environments like IoT devices and embedded systems.
- **Post-quantum cryptography (PQC)** It refers to algorithms designed to be secure against quantum computers. Quantum computers could break existing systems like RSA and ECC, making the development of PQC crucial for future security.
- **Crystals-Kyber:** Lattice-based encryption algorithm designed to be quantum-resistant. Part of NIST's post-quantum cryptography standardization for public-key encryption.

- **Crystals-Dilithium:** Lattice-based quantum-resistant signature algorithm, used for digital signatures in secure communications and blockchain.
- **Falcon:** Lattice-based signature algorithm, quantum-resistant and efficient, a strong candidate for future quantum-resistant standards.
- **SPHINCS++:** Hash-based signature scheme offering strong quantum security. Designed to replace existing digital signature schemes in the post-quantum era.

CHAPTER 3

LITERATURE REVIEW

Z. A. Shaikh et al.: New Trend in Cryptographic IS for Industry 5.0: A Systematic Review(2024), presents a systematic review of cryptographic information security (IS) for Industry 5.0, addressing privacy protection, real-time access control, and secure lifecycle frameworks to meet regulatory and industrial goals. It highlights existing challenges in interconnectivity, control access, and cryptographic adoption while proposing solutions like re-encryption-enabled IS for future developments in the Industrial Internet of Things (IIoT). [1]

H. Kwon et al.: Evaluating KpqC Algorithm Submissions: Balanced and Clean Benchmarking Approach (2023), presents a performance evaluation of 16 KpqC algorithms using a unified benchmarking approach. By removing external library dependencies, the study ensures fair comparisons and provides standardized implementations for Post Quantum Cryptography, facilitating adoption and eliminating performance deviations caused by diverse environments.[2]

Chalkias et al.(2024), has explored thatcontinuous benchmarking has uncovered key performance patterns and security insights in cryptography.This paper presents a comprehensive benchmarking framework for cryptographic algorithms in the Rust library fastcrypto, uncovering security flaws, performance bottlenecks, and optimization opportunities in encryption and blockchain technologies. By enabling automated, continuous benchmarking, the framework identifies vulnerabilities, enhances algorithmic efficiency, and provides tools for auditing and innovation in cryptographic implementations.[3]

CHAPTER 4

METHODOLOGY

4.1 Overview

In this approach, we plan to break down the development of our framework into smaller, manageable parts called sprints where we can develop features and make changes based on ongoing feedback. It ensures that we can easily adapt to new requirements or adjustments, making the development process more dynamic and responsive.

Requirement Analysis

1. **Functional Requirements:**
    - Supports asymmetric, symmetric, and hashing algorithms.
    - Ability to measure the encryption/decryption parameters for benchmarking.
    - Accept user input and allow the user to view performance results.
2. **Non-functional Requirements:**
    - Smooth performance.
    - Easy usability.
    - Secured from unauthorized access.

4.2 Tools

- **Programming Language:** Python, for scripting and testing algorithms. Libraries like PyCryptodome are useful.
- **Cryptographic Library:** OpenSSL, widely used for implementing and testing cryptographic protocols.
- **PyCryptodome:** A Python library for cryptographic operations.
- **HTML:** Used to structure a web page and its content.
- **CSS:** Defines styles for web pages, including design, layout, and variations in display for different devices and screen sizes.

- **JavaScript (JS):** A lightweight, interpreted programming language with first-class functions, primarily used as the scripting language for web pages.

4.3 Benchmarking Techniques

Performance Metrics:

- **Throughput:** Measure the amount of data processed per unit time.
- **Latency:** Track how long it takes to complete encryption, decryption, or hashing tasks.
- **Resource Utilization:** Analyze CPU, memory, and power consumption.
- **Cross-Media Testing:** Benchmark cryptographic algorithms on various platforms (e.g., desktop CPU, mobile devices, GPU).

4.4 User-Defined Parameters

- **Key Size:** Users can choose different key lengths for algorithms (e.g., 128-bit, 256-bit for AES).
- **Data Size:** Options for input data size, such as small, medium, or large datasets.
- **Iterations:** Allow users to specify the number of iterations for hashing or encryption processes, enabling more customized performance testing.

4.5 Testing and Validation Techniques

Accuracy Testing: Ensure the encryption, decryption, and hashing functions produce correct results by comparing outputs with known accurate values.

4.6 Visualization Techniques

Performance Charts: Display benchmarking results with dynamic graphs such as bar charts or line graphs for better interpretation of performance metrics (e.g., time taken, throughput).

4.7 Deployment and Hosting Techniques

Web Hosting: Deploy the website on version control platform like Github.

4.8 System Block-Diagram

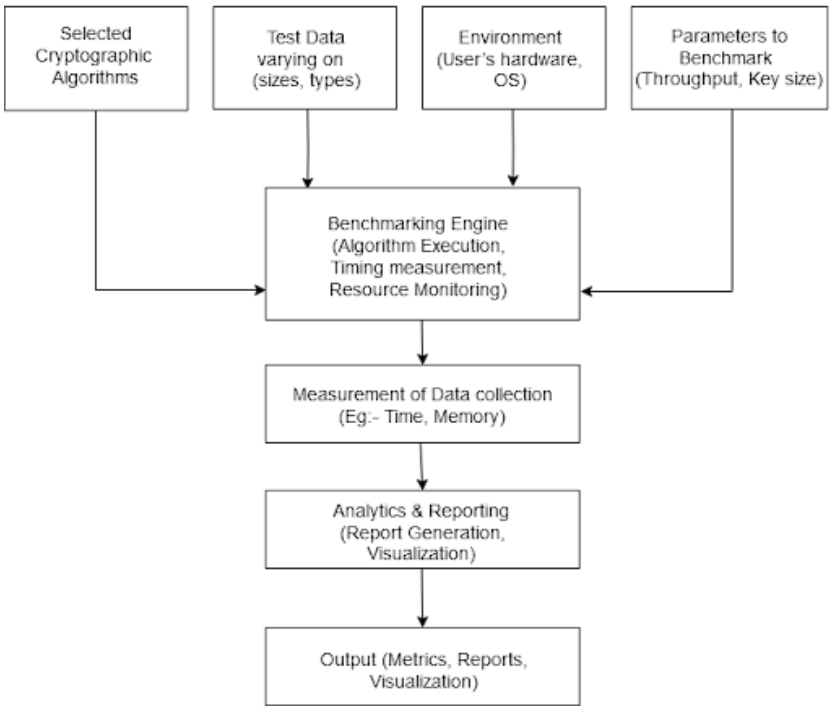

Figure 4.1: System block-diagram

4.9 Sequence Diagram

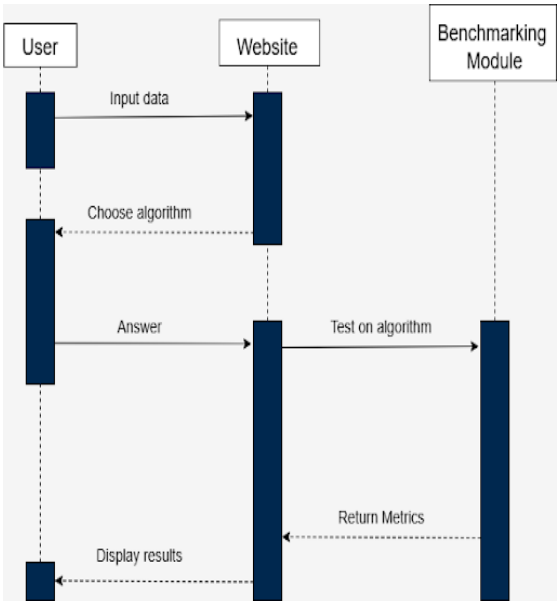

Figure 4.2: Sequence diagram

4.10 Use Case Diagram

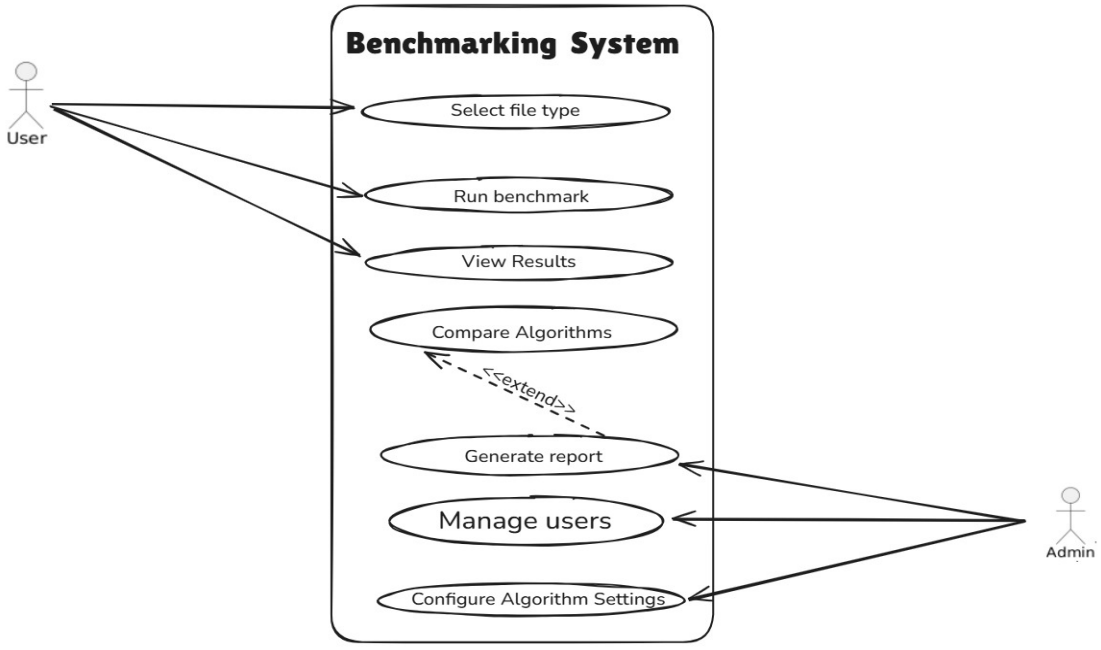

Figure 4.3: Use Case Diagram.

4.11 Class Diagram

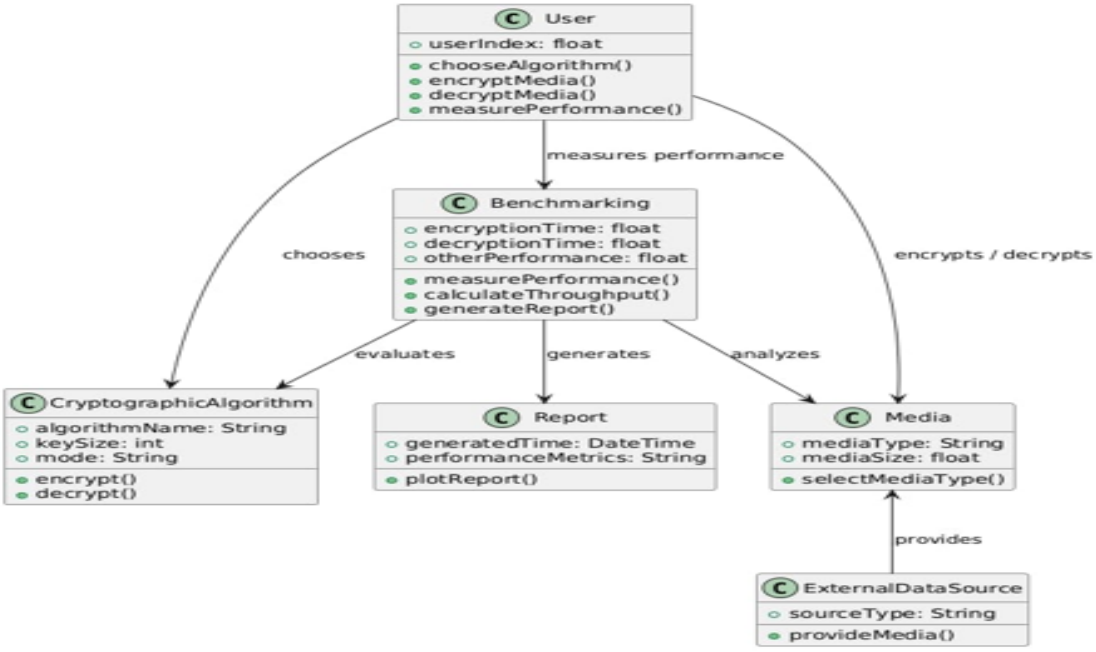

Figure 4.4: Class Diagram

4.12 Gantt Chart

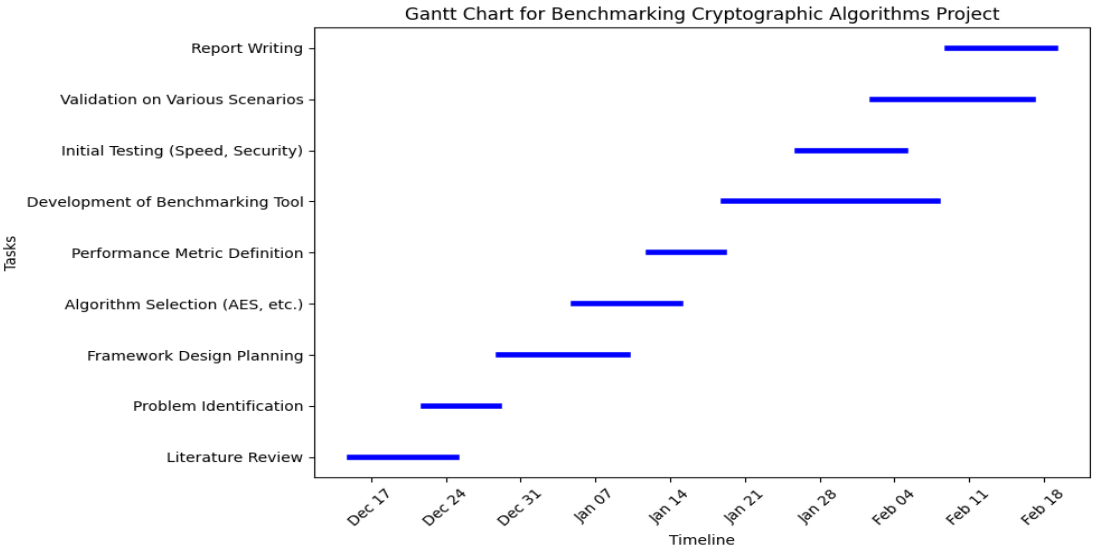

Figure 4.5: Gantt Chart.

CHAPTER 5

EXPECTED RESULTS

5.1 Expected interface

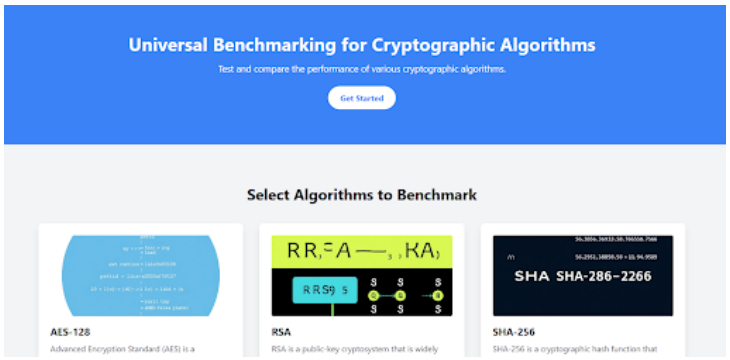

Figure 5.1: Expected interface

5.2 Expected Result

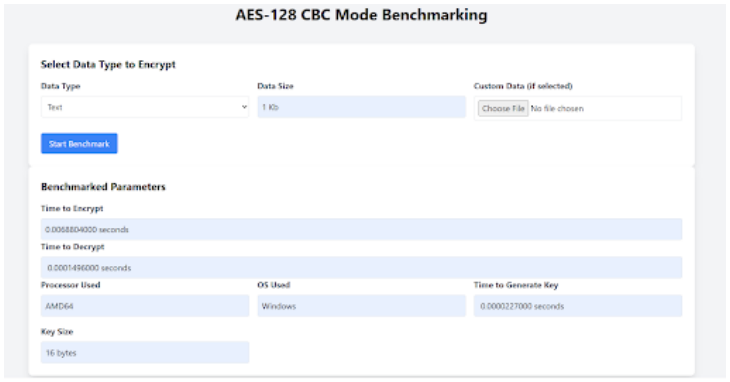

Figure 5.2: Expected result from the analysis.

5.3 Benchmark Table

Algorithm	Time(s)	Memory(kB)	File size(MB)
AES	0.01	1024	1
RSA	0.20	2048	1
SHA-256	0.005	512	1

Table 5.1: Benchmark Table

REFERENCES

[1] Z. A. Shaikh, F. Hajjej, Y. D. Uslu, S. Yuksel, H. Dinçer, R. Alroobaea, A. M. Baqasah, and U. Chinta, "A new trend in cryptographic information security for industry 5.0: A systematic review," *IEEE Access*, 2024.

[2] H. Kwon, M. Sim, G. Song, M. Lee, and H. Seo, "Evaluating kpqc algorithm submissions: Balanced and clean benchmarking approach," in *International Conference on Information Security Applications*. Springer, 2023, pp. 338–348.

[3] K. K. Chalkias, J. Lindstrøm, D. Maram, B. Riva, A. Roy, A. Sonnino, and J. Wang, "Fastcrypto: Pioneering cryptography via continuous benchmarking," in *Companion of the 15th ACM/SPEC International Conference on Performance Engineering*, 2024, pp. 227–234.