**LAB REPORT: RSA-OAEP Cipher using CryptoPP**

|  |  |  |
| --- | --- | --- |
| **Instructor** | **Đỗ Thị Phương Uyên** | |
| **Student 1:** | Tào Minh Đức | 23520315 |
| **Student 2:** | Mai Nguyễn Phúc Minh | 23520930 |

LAB 3 REPORT’S TABLE OF CONTENTS

[**1.** **Hardware resource** 2](#_Toc198556849)

[**2.** **Input testcase** 2](#_Toc198556850)

[**3.** **RSA-OAEP (Windows System)** 2](#_Toc198556851)

[**4.** **RSA-OAEP (Linux System)** 3](#_Toc198556852)

[**5.** **Sample images** 4](#_Toc198556853)

[**6.** **Conclusion** 5](#_Toc198556854)

1. **Hardware resource**

|  |  |
| --- | --- |
| **Device:** | Lenovo Gaming Legion 5 15IAH7H |
| **Chip:** | Intel Core i5 12500H   * Cores: 12 * P-core: 4 * E-core: 8 * Logical processor: 16 |
| **Ram & Memory:** | DDR5-4800 – 16GB (RAM)  512 GB SSD x2 |
| **Operating Systems:** | Window 11  Ubuntu |

1. **Input testcase**

* Making a executed program to automatically generate a random input with 6 different testcase:
  + 100 bytes input
  + 200 bytes input
  + 300 bytes input
  + 1 KB input
  + 10 KB input
  + 1 MB input
* **Note:** These testcase are generated randomly based on the program **makingtextcase.exe**

1. **RSA-OAEP (Windows System)**

* **Key using throughout all files:** *public.pem* (for public key) and *private.pem* (for private key)
* **Key size:** 3072
* **File encrypted:** *cipher.bin*
* **File plaintext after decrypted:** *output.txt || decrypted.txt*
* **Usage format:**

+ D:\CRYPTO\LAB\rsaoaep.exe gen <keysize> <format> <privateKeyFile> <publicKeyFile> (*e.g:* .\rsaoecp.exe gen 3072 PEM private.pem public.pem)

+ D:\CRYPTO\LAB\rsaoaep.exe enc <format> <publicKeyFile> <plainFile> <cipherFile> (*e.g:* .\rsaoecp.exe enc PEM public.pem random\_1K.txt cipher.bin)

+D:\CRYPTO\LAB\rsaoaep.exe dec <format> <privateKeyFile> <plainFile> <cipherFile> *(e.g:* .\rsaoecp.exe dec PEM private.pem output.txt cipher.bin)

* **Abbreviations:** TT (Total Time), AT (Average Time)
* **Time counter:** Mili second (ms)
* **Execution Time (average of 10000 executions):**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **100B** | **200B** | **300B** | **1KB** | **10KB** | **1MB** |
| **Encrytion** | TT: 3005  AT: 0.3005 | TT: 4013  AT: 0.4013 | TT: 4381  AT: 0.4381 | TT: 14329  AT: 1.4329 | TT: 38988  AT: 3.5988 | TT: 2223489  AT: 222.3489 |
| **Decryption** | TT: 944064  AT: 94.4064 | TT: 1.25673e+06  AT: 125.673 | TT: 1.32358e+06  AT: 132.358 | TT: 1130723  AT: 113.0723 | TT: 2768786  AT: 276.8796 | TT: 169267852  AT: 16926.7852 |

* **Note:** The execution time on big files (10KB and 1MB) would be bigger due to the complexity of computing.

1. **RSA-OAEP (Linux System)**

* **Key using throughout all files:** *public.pem* (for public key) and *private.pem* (for private key)
* **Key size:** 3072
* **File encrypted:** *cipher.bin*
* **File plaintext after decrypted:** *output.txt || decrypted.txt*
* **Usage format:**

+ ./rsaoaep.exe gen <keysize> <format> <privateKeyFile> <publicKeyFile> (*e.g:* .\rsaoecp.exe gen 3072 PEM private.pem public.pem)

+ ./rsaoaep.exe enc <format> <publicKeyFile> <plainFile> <cipherFile> (*e.g:* .\rsaoecp.exe enc PEM public.pem random\_1K.txt cipher.bin)

+ ./rsaoaep.exe dec <format> <privateKeyFile> <plainFile> <cipherFile> *(e.g:* .\rsaoecp.exe dec PEM private.pem output.txt cipher.bin)

* **Abbreviations:** TT (Total Time), AT (Average Time)
* **Time counter:** Mili second (ms)
* **Execution Time (average of 10000 executions):**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **100B** | **200B** | **300B** | **1KB** | **10KB** | **1MB** |
| **Encrytion** | TT: 2573  AT: 0.2573 | TT: 3022  AT: 0.3022 | TT: 3094  AT: 0.3094 | TT: 1914  AT: 0.1914 | TT: 14940  AT: 1.494 | TT: 1.49644e+06  AT: 149.644 |
| **Decryption** | TT: 519651  AT: 51,9651 | TT: 519974  AT: 51.9974 | TT: 520512  AT: 52.0512 | TT: --  AT: -- | TT: --  AT: -- | TT: --  AT: -- |

* **Note:** The execution time on big files (1KB, 10KB and 1MB) would be bigger due to the complexity of computing.

1. **Sample images**

* **Generate public/private key:**

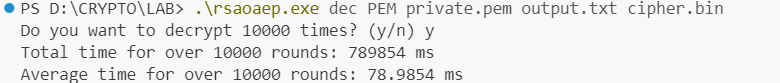
****

* **Encrypted text:**

**A screenshot of a computer code

AI-generated content may be incorrect.**

* **Decrypted text:**

****

1. **Conclusion**

* The execution time of RSA-OAEP appears to be lower than AES in Labs 1 and 2. This is likely due to RSA-OAEP relying heavily on expensive mathematical operations involving large numbers (modular exponentiation).
* Execution times on a Linux system (over 10,000 iterations) were consistently faster than those on a Windows system. As file sizes increase, the encryption and decryption times using RSA also increase noticeably.
* RSA, as an asymmetric encryption algorithm (using a public-private key pair), is not efficient for encrypting large files due to its high computational overhead. For larger files (e.g., around 1MB), AES is a more suitable choice due to its simpler algorithm and significantly lower computation time compared to RSA.
* RSA is best used to securely exchange symmetric keys, which can then be used by AES for efficient bulk data encryption. This hybrid approach combines the strengths of both algorithms to ensure both data security and performance.