APPLIED CRYPTOGRAPHY

Lab 5 : RSA - Public Key Encryption

Name: Siri N Shetty SRN: PES2UG22CS556

(use SEEDVM)

Task 1: Installation of tool

Expected Deliverables -

i) Output Screenshot on running 'rsa-tool' command

```
PES2UG22CS556:~$>rsa-tool
Usage:
  rsa-tool --generate <privateKeyFile> <publicKeyFile>
  rsa-tool --encrypt <inputFile> <outputFile> <publicKey>
  rsa-tool --decrypt <inputFile> <outputFile> <privateKey>
  rsa-tool --sign <inputFile> <signatureFile>
  rsa-tool --verify <inputFile> <signatureFile>
PES2UG22CS556:~$>
```

Task 2: Decrypting given cipher

Expected Deliverables -

i) Output Screenshot of output from command

ii) Text in decrypted.txt

```
PES2UG22CS556:~/.../Lab5_Files$>rsa-tool --decrypt secret.enc unecrypted.txt privateKeytask2.pem
File decrypted and written to unecrypted.txt
PES2UG22CS556:~/.../Lab5_Files$>cat unecrypted.txt
PES Innovation Lab Presents:
ROADSHOW 11.0

We are excited to announce the 11th iteration of our project exhibition, showcasing the work of o
ur summer interns who dedicated two months to their projects.

RR Campus: October 16, 2024, 3rd Floor, Be Block
EC Campus: October 23, 2024, Quadrangle

Make sure to join us and explore cutting-edge technology in networks, distributed systems, roboti
cs, LLMs, and much more!
PES2UG22CS556:~/.../Lab5_Files$>
```

Task 3 : Generation of Key-Pair

Expected Deliverables -

i) Output Screenshot of modifying key size in code

```
PES2UG22CS556:~/.../task3$>rsa-tool --generate changed-PrivateKey.pem changed-Publickey.pem
Key Pair Generated
PES2UG22CS556:~/.../task3$>wc -c changed-PrivateKey.pem
3294 changed-PrivateKey.pem
PES2UG22CS556:~/.../task3$>wc -c default_privateKey.pem
1706 default_privateKey.pem
PES2UG22CS556:~/.../task3$>
```

ii) Comparison of the 2 key sizes

DEFAULT_PRIVATEKEY=2048
CHANGED_PRIVATEKEY=4096

Task 4 : Encryption and Decryption

Expected Deliverables -

i) Output Screenshot for encrypting the file

```
PES2UG22CS556:~/.../siri$>rsa-tool
Usage:
  rsa-tool --generate <privateKeyFile> <publicKeyFile>
  rsa-tool --encrypt <inputFile> <outputFile> <publicKey>
  rsa-tool --decrypt <inputFile> <outputFile> <privateKey>
  rsa-tool --sign <inputFile> <signatureFile>
  rsa-tool --verify <inputFile> <signatureFile>
PES2UG22CS556:~/.../siri$>rsa-tool --generate PES2UG22CS556_privateKey.pem PES2UG22CS556_publicKey
.pem
Key Pair Generated
```

```
PES2UG22CS556:~/.../siri$>rsa-tool --encrypt pES2UG22CS556_MSG.TXT output.enc PES2UG22CS594_PUBLIC
KEY.PEM
File encrypted and written to output.enc
```

ii) Name and SRN of person you exchanged keys with

Sumeet Pai - PES2UG22CS594

iii) Output Screenshot for friend decrypting the encrypted file you sent

```
Sumeet_PES2UG22CS594:~/.../task4$>cat decrypted_message.txt
hello guys, this is siri, i am extremely cooked at this moment, please shower so
me blessings until i gradute! thankd and regards, siri

Sumeet_PES2UG22CS594:~/.../task4$>
```

iv) Output Screenshot for you decrypting encrypted file received from friend

```
PES2UG22CS556:~/.../siri$>rsa-tool --decrypt PES2UG22CS594_output.enc  PES2UG22CS594_MESSAGE.txt P
ES2UG22CS556_privateKey.pem
File decrypted and written to PES2UG22CS594_MESSAGE.txt
PES2UG22CS556:~/.../siri$>cat PES2UG22CS594_MESSAGE.txt
hello world this is a mesage to siri
pallaavi plwease ectsend thedeadline
ill kill myslef othwerwise
```

Task 5 : Real Scenario

Expected Deliverables -

i) Explanation of functions

AES encryption is usually used instead of RSA for the encryption of large files due to its speed. A small amount of data can, however be encrypted with RSA. This way of encrypting the data with the two algorithms has the advantage that each algorithm can be used for its greatest strength; the former applies AES for encrypting the data and the latter applies RSA for encrypting a small IV. This way large files are encrypted safely and efficiently. Generating public and private keys has taken care of through libraries or functions on cryptography.

Overall Submission

1) SRN_Lab_5_AC : pdf
2) SRN_Lab_5.zip : below file structure

```
SRN_Lab_5/
        |
        | - Task_4/
                | - input.txt (Plain text that you have encrypted)
                | - SRN_publicKey.pem (This is your friend's public Key, put
their SRN)
                | - SRN_privateKey.pem (This your private key, put your SRN)
                | - output.txt (Text that you decrypted)
        | - Task_5/
                | - realAlgo.ts
```