MARMARA UNIVERSITY

FACULTY OF ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

CSE 4057

INFORMATION SYSTEMS SECURITY

HOMEWORK-1 REPORT

Mikail Torun

150116021

Enes Garip

150116034

Mert İsmail Eği

150115025

AES

The Advanced Encryption Standard (AES, Rijndael) is a block cipher encryption and decryption algorithm, the most used encryption algorithm in the worldwide. The AES processes block of 128 bits using a secret key of 128, 192, or 256 bits. We are using 128 and 256 bit secret keys for AES in this project.

RSA

RSA, or in other words Rivest–Shamir–Adleman, is an asymmetric cryptographic algorithm. The most important differences from symmetric algorithms like DES or AES is that having two keys. A public key that we can share with anyone is used to encrypt data. And a private key that we keep only for ourselves and it's used for decrypting the data.

QUESTION-1

```
// ************************** QUESTION-1 ***************************************
// KEY PAIR GENERATOR FUNCTION
// RSA key pair generation with the key size of 2048 and the function returns the key pair.
public static KeyPair generateKeyPair() throws Exception {
    KeyPairGenerator generator = KeyPairGenerator.getInstance("RSA");
    generator.initialize( keysize: 2048, new SecureRandom());
    KeyPair pair = generator.generateKeyPair();

    return pair;
}
```

In Question 1, we generate RSA key pair with the function above. In that function we create an instance of key pair generator class and with the argument RSA. After that, we initialize the keys with the key size 2048. Lastly, the function returns the key pair. So, as a result, we obtain RSA key pair.

QUESTION-2

```
---------- 128 bit Symmetric Key ----------
Key SymmetricKey_128:
8197157DFB60FF9DB9BCDDA940615E56

CipherText SymmetricKey_128:
ncp2PEuRgu/zcCB+uPwERTVOIZL3pO55m9afA+4hyNAF19I3YtgdW1MyjTkMfiqbBHyzuCWsJr3+oOHdAqB0MPakxOZNL3oWSmmYFLhedNLBJT95b7
 +zW2SFFsXgb2adZmdPerH6TbHXRifApazwjDAJ9NccvpKmQmgLMz9AnEXCZaeWSpMZPLTeoEEKk+slANLHCU5CkG3BmQsBPxbR+cbmyMariQZ9JXjzu9hSLcxGegx1F4vybvHj9Y4d
 /e5SjaA8UQP8jq7sqig1a5FBvG4PlCXIyMWWJthhgjtjaeCr3xlU+kBSjV71dDUbbUggxkLIOMqLMrRoPFX/H8r+Qw==

Decrypted Message SymmetricKey_128:
8197157DFB60FF9DB9BCDDA940615E56

Is plaintext and decrypted message equal?
true
```

```
---------- 256 bit Symmetric Key ----------
Key SymmetricKey_256:
ED62DE883A70861C484DD38712824844587E63D8C26B79C2299A3C9CDFA65213

CipherText SymmetricKey_256:
LVWdWUQikXw03iipgjRJ1WAfPjLLoIo486OqLujh4mnsLOLubrlW0ciiLTAVtG1ieP02evgx/zUZ2hwnK1DXyubvrfrcQCdyiAyhgoe8cFI1nA66HAPOBDcdv0xsTFyzue0OId29w3ePLnn4Ne
 +9TQAky0KQBEKnSvgyLtIF757uNhoBROnK+h2rIRUu5sdpFifGbd9WjvZ6sWkihPdDsTOUnnwG9fVFNHPpWUjPLGy+krKXH2PzyUkubexOPjBDfw19j6H+KQcQIFb7ju/kmtUpg
 +9iVQ5VuUXXgNCmp22ft7B4L7NCI9ZBl3DOVgt8sBhyXha1xbgswMfDórLp9Q==

Decrypted Message SymmetricKey_256:
ED62DE883A70861C484DD38712824844587E63D8C26B79C2299A3C9CDFA65213

Is plaintext and decrypted message equal?
true
```

In Question 2, we create AES symmetric keys with the size of 128 and 256. After that, we print the keys for each size. Then, we encrypt them with the public key of RSA key pair obtained in Question 1. After the encryption, we print the cipher text to the console. After that we decrypt the message with the private key of RSA. Lastly, we check that decrypted message and the original text is equal.

QUESTION-3

```
------------------------ Question-3 ------------------------
Is decrypted message and the file content equal?
true

File content
You may do this homework in groups of two.
What to submit: Submit all your commented codes, output files and a report including your results, screenshots and comments via google classroom. In your codes, please clearly describe which
 code parts do which job. If you do not complete all the items asked above, please clearly indicate which items are completed.
You may do this homework in groups of two.
What to submit: Submit all your commented codes, output files and a report including your results, screenshots and comments via google classroom. In your codes, please clearly describe which
 code parts do which job. If you do not complete all the items asked above, please clearly indicate which items are completed.
You may do this homework in groups of two.
What to submit: Submit all your commented codes, output files and a report including your results, screenshots and comments via google classroom. In your codes, please clearly describe which
 code parts do which job. If you do not complete all the items asked above, please clearly indicate which items are completed.
You may do this homework in groups of two.
What to submit: Submit all your commented codes, output files and a report including your results, screenshots and comments via google classroom. In your codes, please clearly describe which
 code parts do which job. If you do not complete all the items asked above, please clearly indicate which items are completed.
You may do this homework in groups of two.
What to submit: Submit all your commented codes, output files and a report including your results, screenshots and comments via google classroom. In your codes, please clearly describe which
 code parts do which job. If you do not complete all the items asked above, please clearly indicate which items are completed.
You may do this homework in groups of two.
What to submit: Submit all your commented codes, output files and a report including your results, screenshots and comments via google classroom. In your codes, please clearly describe which
 code parts do which job. If you do not complete all the items asked above, please clearly indicate which items are completed.


H(m)
A6A8D75BE2B4D60F4BA85F16EDC17A2AA4D3E94685241F4748BEE6328B453272

Digital Signature
2C16F9E1D297BBB7ED5BD4DD0ABD77E8FFECCD2825419B540CBFB351BFFFFAC32CDA99D4E6F3F68452AFCCB41E0591A14ACDD9983B4C9DB2B4E923845B03A8A462D4C5CCE12A9329FD8E33FE2C34499AAED724A02AB8D87B0F5BFB1630EC7A002
 35DDBC95DD82AEEBF075E408FDAD08A11CE604D5407B7CB1446E073F7C51876BFEC1B3F620235B18FF8704F3D0F0C216720E8C049797BBA8970E7C68C9882E9001BD09BC420F0FB85C05547686695E5053AF47EF7EEC62A4FFC5432A88C98A7A
 3873865BFC970CE9B2B9BFAEC3AD8F4F0A16D13EB0BA75BF719B4462E7A06142543F6A5CED159F8563F929EA5A2A5408CBEECE0BA3CBB5A765CBFE4A3D57DBD
```

In Question 3, we create a message.txt file which will be encrypt. After that, we apply SHA256 Hash algorithm and print the H(m). After that, we encrypt the file with private key of RSA key pair. As a result of that, we obtain a digital signature. We extract the digital signature to a file and read it again to verify the digital signature. As a result of all the operations, we print the file content, H(m) and the digital signature in hex binary notation.

QUESTION-4

```
------------------------ Question-4 ------------------------

-- Key Size: 128 bit / Mode: CBC --
CBC finished.
Total time elapsed in nano seconds: 8587363700

-- Key Size: 256 bit / Mode: CBC --
CBC finished.
Total time elapsed in nano seconds: 9717098300

-- Key Size: 256 bit / Mode: CTR --
CTR finished.
Total time elapsed in nano seconds: 9056283300
```

In Question 4, we create another java file for the operations. In that file, we set the mode and encrypt an image file with that mode. Also, the key size parameter passes to the function for decision of the key size. In addition to that, we set a timer to obtain how much time passes for the operation. It returns the time in nanoseconds.