HACETTEPE UNIVERSITY

COMPUTER ENGINEERING DEPARTMENT

BBM465 2022 FALL

Assignment-3

December 6, 2022

Group Number: 37

Tuğrul ACAR
2210356144

Alper SOLMAZ
2200356039

Problem

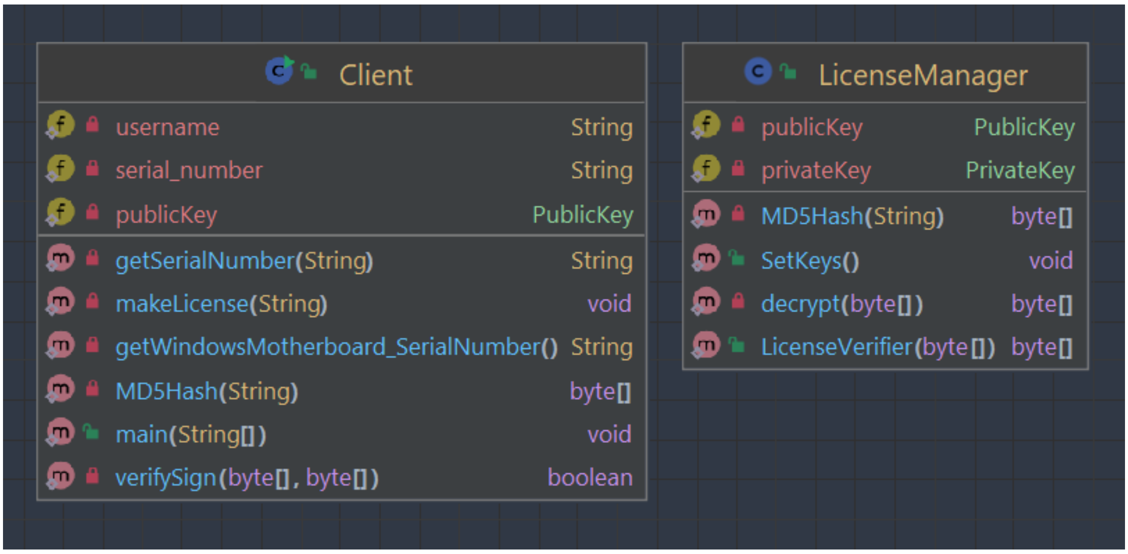
In this project, we are expected to develop a licensing framework by utilizing the methods of asymmetric cryptography, MD5, and digital signatures. Asymmetric cryptography offers to use a publicly available public key for encryption of the message which will be only decrypted by using the private key. The receiver is able to decrypt the encrypted message and it can create a digital signature of content for further authentication and verification purposes.

Important Notes About the Assignment

- We used JAVA 14 version.
- Detailed description of all functions given in the code as comments.
- Our program works on both Windows 10 and 11 operating system.
- Username and serial_number are hardcoded in the Client class.
- Encrypted messages, hash values and signature encoded as Base64 format before printed.

Solution for the Problem

We used 2 java classes in this project. Which is Client and LicenseManager.



Client	
username	String
serial_number	String
publicKey	PublicKey
getSerialNumber(String)	String
makeLicense(String)	void
getWindowsMotherboard_SerialNumber()	String
MD5Hash(String)	byte[]
main(String[])	void
verifySign(byte[], byte[])	boolean

LicenseManager	
publicKey	PublicKey
privateKey	PrivateKey
MD5Hash(String)	byte[]
SetKeys()	void
decrypt(byte[])	byte[]
LicenseVerifier(byte[])	byte[]

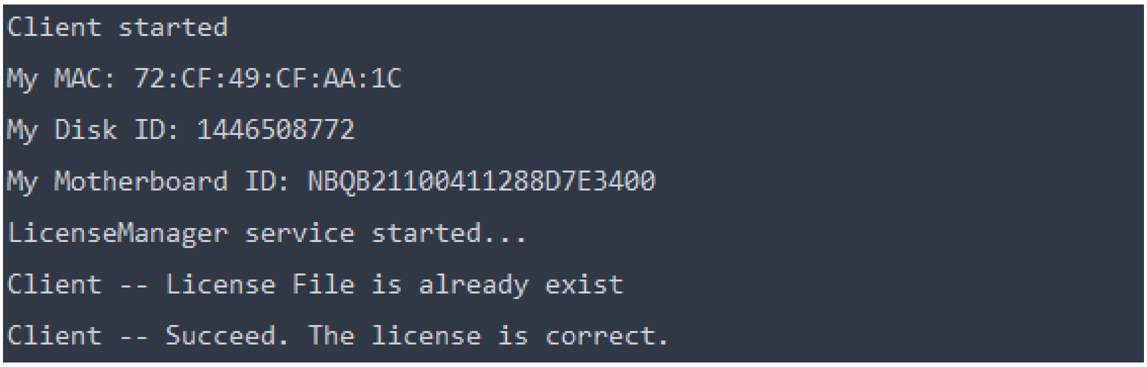
Figure 1 UML Diagram of The Program

Firstly, when program started key files which is exist in the application folder read. Then necessary user information MAC address etc. collected with different methods and combined as a string named UserTuple. After this setup process Client module checks the existence of the “license.txt” in the application folder which involves a digital signature signed by the license manager. If the license.txt exists, then client module calls verifySign method to verify the signature by using the public key. If server can not verify the signature or license.txt does not exist at all, then program calls makeLicense method with UserTuple to create a license.

username$serial_number$MAC_address$disk_serial_no$motherboard_serial_no

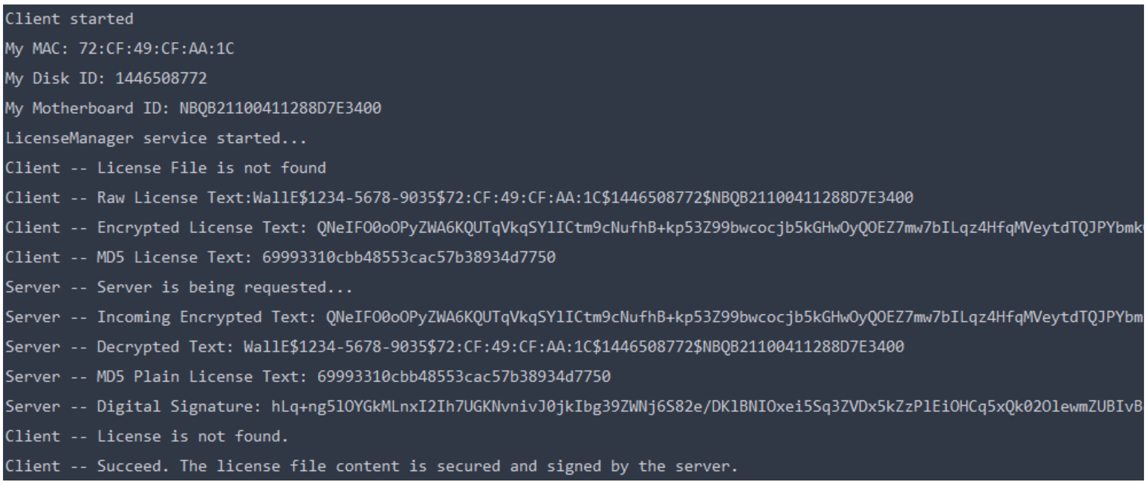
UserTuple Format

makeLicense method firstly encrypts the UserTuple with RSA algorithm via public key and send it to server by calling LicenseVerifier method. LicenseVerifier method decrypts the received data via private key. Then hashes the decrypted data (UserTuple) with MD5 hashing algorithm and signs the hash value via SHA256withRSA algorithm. Eventually returns the signature back to the makeLicense method. Client verifies the returned signature by calling verifySign method. If verification done successfully, license will be saved into license.txt file encoded as Base64 format.



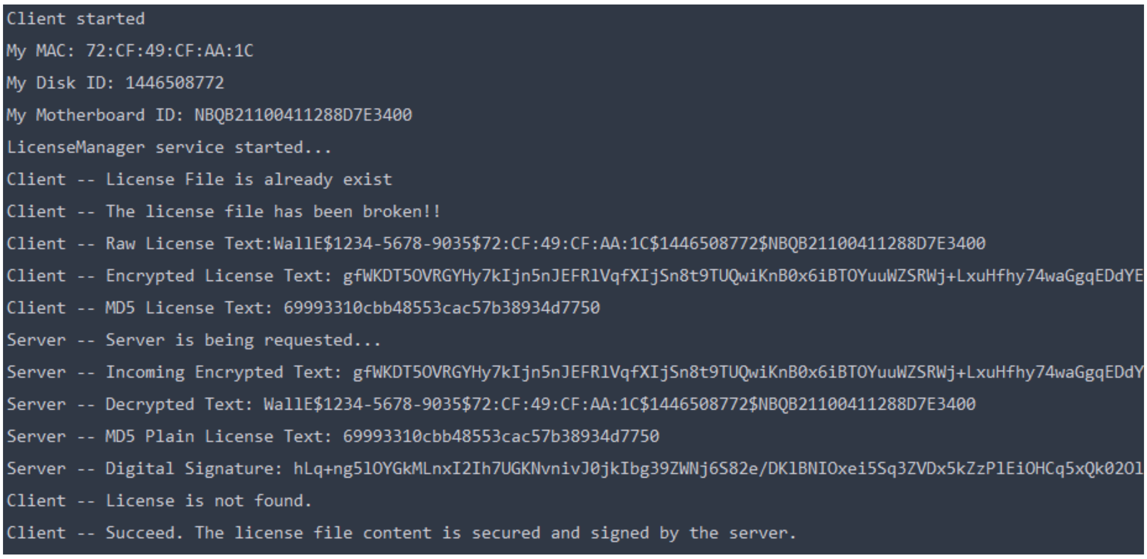
```
Client started
My MAC: 72:CF:49:CF:AA:1C
My Disk ID: 1446508772
My Motherboard ID: NBQB21100411288D7E3400
LicenseManager service started...
Client -- License File is already exist
Client -- Succeed. The license is correct.
```

Figure 2 License.txt exist and have valid license.



```
Client started
My MAC: 72:CF:49:CF:AA:1C
My Disk ID: 1446508772
My Motherboard ID: NBQB21100411288D7E3400
LicenseManager service started...
Client -- License File is not found
Client -- Raw License Text:WallE$1234-5678-9035$72:CF:49:CF:AA:1C$1446508772$NBQB21100411288D7E3400
Client -- Encrypted License Text: QNeIFO0oOPyZWA6KQUTqVkqSYlICtm9cNufhB+kp53Z99bwcocjb5kGHwOyQOEZ7mw7bILqz4HfqMVeytdTQJPYbmk
Client -- MD5 License Text: 69993310cbb48553cac57b38934d7750
Server -- Server is being requested...
Server -- Incoming Encrypted Text: QNeIFO0oOPyZWA6KQUTqVkqSYlICtm9cNufhB+kp53Z99bwcocjb5kGHwOyQOEZ7mw7bILqz4HfqMVeytdTQJPYbm
Server -- Decrypted Text: WallE$1234-5678-9035$72:CF:49:CF:AA:1C$1446508772$NBQB21100411288D7E3400
Server -- MD5 Plain License Text: 69993310cbb48553cac57b38934d7750
Server -- Digital Signature: hLq+ng51OYGkMLnxI2Ih7UGKNvnivJ0jkIbg39ZWNj6S82e/DKlBNIOxei5Sq3ZVDx5kZzPlEiOHCq5xQk02OlewmZUBIvB
Client -- License is not found.
Client -- Succeed. The license file content is secured and signed by the server.
```

Figure 3 License.txt does not exist.



```
Client started
My MAC: 72:CF:49:CF:AA:1C
My Disk ID: 1446508772
My Motherboard ID: NBQB21100411288D7E3400
LicenseManager service started...
Client -- License File is already exist
Client -- The license file has been broken!!
Client -- Raw License Text:WallE$1234-5678-9035$72:CF:49:CF:AA:1C$1446508772$NBQB21100411288D7E3400
Client -- Encrypted License Text: gfWKDT5OVRGYHy7kIjn5nJEFRlVqfXIjSn8t9TUQwiKnB0x6iBTOYuuWZSRWj+LxuHfhy74waGgqEDdYE
Client -- MD5 License Text: 69993310cbb48553cac57b38934d7750
Server -- Server is being requested...
Server -- Incoming Encrypted Text: gfWKDT5OVRGYHy7kIjn5nJEFRlVqfXIjSn8t9TUQwiKnB0x6iBTOYuuWZSRWj+LxuHfhy74waGgqEDdY
Server -- Decrypted Text: WallE$1234-5678-9035$72:CF:49:CF:AA:1C$1446508772$NBQB21100411288D7E3400
Server -- MD5 Plain License Text: 69993310cbb48553cac57b38934d7750
Server -- Digital Signature: hLq+ng5lOYGkMLnxI2Ih7UGKNvnivJ0jkIbg39ZWNj6S82e/DKlBNIOxei5Sq3ZVDx5kZzPlEiOHCq5xQk02Ol
Client -- License is not found.
Client -- Succeed. The license file content is secured and signed by the server.
```

Figure 4 License.txt exist but license broken.

Reference

- https://www.geeksforgeeks.org/java-program-to-get-system-motherboard-serial-number-for-windows-and-linux-machine/

- https://www.baeldung.com/java-mac-address

- https://stackoverflow.com/questions/69750026/create-sha256withrsa-in-twosteps

- https://www.baeldung.com/java-md5

- https://www.baeldung.com/java-rsa