Information System Security HW#1 (fall 2020)

1-1

암호화 기능을 제공할 수 있어야 한다. 암호화 프로그램은 사용자로부터 평문 파일명을 입력 받고 암호에 사용할 비밀키 값을 입력 받을 수 있어야 한다. 반드시 PKCS5 패딩을 사용해야 한다. AES는 IV를 포함한 CBC모드로 구현되어야 한다. 암호화된 파일이 생성될 수 있어야 한다.

1-2

1-1에서 생성된 파일에 대해서 복호화 기능을 제공할 수 있어야 한다. 복호화 프로그램은 사용자로부터 암호문 파일명을 입력 받고 복호에 사용할 비밀키 값을 입력 받을 수 있어야 한다. 복호화된 파일이 생성될 수 있어야 한다.

Source Code

- AESAlgorithm.java

```
import javax.crypto.Cipher;
import javax.crypto.SecretKey;
import javax.crypto.spec.IvParameterSpec;
import javax.crypto.spec.SecretKeySpec;
import org.apache.commons.codec.binary.Base64;
import java.io.*;

public class AESAlgorithm {
    byte[] enc_iv = "A".repeat(16).getBytes(); // IV값

    // AES 암호화
    public String encode(String str, String enc_key) throws Exception {
        // key 길이가 32가 아닐경우 에러.
        if(enc_key.length() != 32){
            System.out.println("Key Length Error");
            return null;
        }
        // AES
        byte[] data = enc_key.getBytes();
        SecretKey secretKey = new SecretKeySpec(data, "AES"); // 키생성
        Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
        cipher.init(Cipher.ENCRYPT_MODE, secretKey, new
IvParameterSpec(enc_iv));
        byte[] encrypted = cipher.doFinal(str.getBytes("UTF-8"));

        // base64로 인코딩
        String enc_data = new String(Base64.encodeBase64(encrypted));
        return enc_data;
    }
```

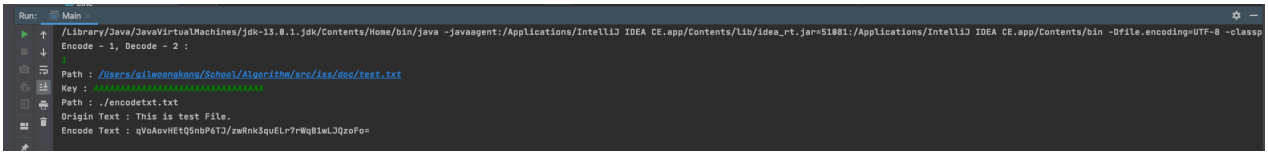
```
    // AES 복호화
    public String decode(String str,String enc_key) throws Exception {
        // key 길이 확인
        if(enc_key.length() != 32){
            System.out.println("Key Length Error");
            return null;
        }
        byte[] data = enc_key.getBytes();

        // AES
        SecretKey secretKey = new SecretKeySpec(data, "AES");
        Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
        cipher.init(Cipher.DECRYPT_MODE, secretKey, new
IvParameterSpec(enc_iv));

        // base64 인코딩 및 리턴
        byte[] decrypted = Base64.decodeBase64(str.getBytes());
        return new String(cipher.doFinal(decrypted), "UTF-8");
    }

    // 파일 읽기
    public String fileRead(String path) throws IOException {
        FileInputStream fileInputStream = new FileInputStream(path);
        byte[] readBuffer = new byte[fileInputStream.available()];
        while(fileInputStream.read(readBuffer) != -1){ }
        String r = new String(readBuffer);
        fileInputStream.close();
        return r;
    }

    // 파일 쓰기
    public boolean fileWrite(String path,String text) throws IOException {
        BufferedOutputStream bufferedOutputStream = new
BufferedOutputStream(new FileOutputStream(path));
        try {
            bufferedOutputStream.write(text.getBytes());
        }catch (Exception e){
            e.getStackTrace();
        }finally {
            bufferedOutputStream.close();
            return true;
        }
    }
}
```

- Main.java

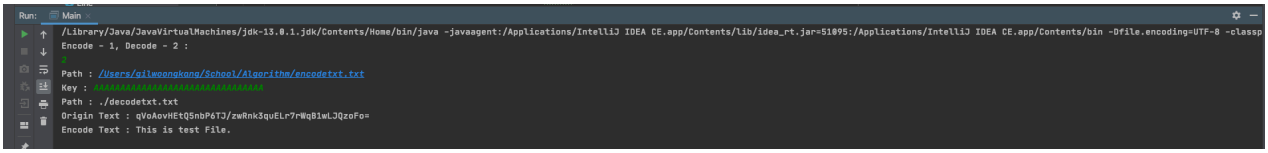
```java
import java.util.Scanner;

public class Main {
    public static void main(String[] args) throws Exception {
        Scanner scanner = new Scanner(System.in);
        AESAlgorithm aesAlgorithm = new AESAlgorithm();

        // 암호화/복화화 선택
        System.out.println("Encode - 1, Decode - 2 : ");
        int c = scanner.nextInt();
        scanner.nextLine();
        // 암호화 경우
        if(c == 1){
            // 경로 및 키 입력
            System.out.print("Path : ");
            String path = scanner.nextLine();
            System.out.print("Key : ");
            String key = scanner.nextLine();
            // 암호화 및 프로젝트 경로에 암호화 파일 생성
            String origin = aesAlgorithm.fileRead(path);    // 파일 원본
            String text = aesAlgorithm.encode(origin,key);  // 암호화한 파일 내용
            aesAlgorithm.fileWrite("./encodetxt.txt",text); // 파일 저장

            System.out.println("Path : ./encodetxt.txt");
            System.out.println("Origin Text : " + origin);
            System.out.println("Encode Text : " + text);
        }
        // 복호화 경우
        else if(c == 2){
            // 경로 및 키 입력
            System.out.print("Path : ");
            String path = scanner.nextLine();
            System.out.print("Key : ");
            String key = scanner.nextLine();
            // 복호화 및 프로젝트 경로에 복호화 파일 생성
            String origin  = aesAlgorithm.fileRead(path); // 복호화할 파일 내용
            String text = aesAlgorithm.decode(origin,key);  // 복호화 한 파일 내용
            aesAlgorithm.fileWrite("./decodetxt.txt",text);    // 파일 저장

            System.out.println("Path : ./decodetxt.txt");
            System.out.println("Origin Text : " + origin);
            System.out.println("Encode Text : " + text);
        }
    }
}
```

실행 결과

- 1-1 (암호화)


Run:
Main
/Library/Java/JavaVirtualMachines/jdk-13.0.1.jdk/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=51081:/Applications/IntelliJ IDEA CE.app/Contents/bin -Dfile.encoding=UTF-8 -classp
Encode - 1, Decode - 2 :
1
Path : /Users/gilwoongkang/School/Algorithm/src/iss/doc/test.txt
Key : ################################
Path : ./encodetxt.txt
Origin Text : This is test File.
Encode Text : qVoAovHEtQ5nbP6TJ/zwRnk3quELr7rWqB1wLJQzoFo=

- 1-2 (복호화)


Run:
Main
/Library/Java/JavaVirtualMachines/jdk-13.0.1.jdk/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=51095:/Applications/IntelliJ IDEA CE.app/Contents/bin -Dfile.encoding=UTF-8 -classp
Encode - 1, Decode - 2 :
2
Path : /Users/gilwoongkang/School/Algorithm/encodetxt.txt
Key : ################################
Path : ./decodetxt.txt
Origin Text : qVoAovHEtQ5nbP6TJ/zwRnk3quELr7rWqB1wLJQzoFo=
Encode Text : This is test File.