Information System Security HW#1 (fall 2020)

2

SHA256을 이용해서 생성된 결과값(256비트) 중, 처음 16비트가 모두 0인 결과값을 생성하는 임의의 입력값 스트링을 하나 이상 찾으시오.

Source Code

• SHAAlgorithm.java

```
import java.io.UnsupportedEncodingException;
import java.math.BigInteger;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.util.Random;
public class SHAAlgorithm {
   public byte[] sha256ToByte(String msg) throws NoSuchAlgorithmException,
UnsupportedEncodingException {
        MessageDigest md = MessageDigest.getInstance("SHA-256");
        md.reset();
        md.update(msg.getBytes("utf8"));
        return md.digest();
    }
    // 랜덤 문자열 생성
    public String randomStringGenerator(){
        StringBuffer temp = new StringBuffer();
        Random rnd = new Random();
        // 난수생성
        for (int i = 0; i < 20; i++) {
            int rIndex = rnd.nextInt(3);
            switch (rIndex) {
                case 0:
                    temp.append((char) ((int) (rnd.nextInt(26)) + 97));
                    break;
                case 1:
                    temp.append((char) ((int) (rnd.nextInt(26)) + 65));
                    break;
                case 2:
                    // 0-9
                    temp.append((rnd.nextInt(10)));
                    break;
```

```
}
return temp.toString();
}
```

Main.java

```
import org.apache.commons.codec.binary.Base64;
import java.io.UnsupportedEncodingException;
import java.security.NoSuchAlgorithmException;
public class Main {
   public static void main(String[] args) throws UnsupportedEncodingException,
NoSuchAlgorithmException {
       SHAAlgorithm shaAlgorithm = new SHAAlgorithm();
       // Gil3I9x4jj4yTJ5Dh1p8,mhsrM39uiGcbPPyyz930,sLrGGze569971g35dtZi
       for(int i=0;i<100000;i++){ // 10만번 수행
           String r = shaAlgorithm.randomStringGenerator();
                                                             // 랜덤 문자열 생
성
           byte[] s = shaAlgorithm.sha256ToByte(r); // SHA256 해시화
           if(s[0]==0 && s[1]==0){ // 2바이트 값이 모두 0 이라면, 즉 16비트가 0일
경우
               System.out.println(r); // 본래 문자열 출력
               System.out.println(byteArrayToHexString(s)); // 해시값의 hex값
출력
           }
       }
    }
    // hex to Byte[]
   public static byte[] hexStringToByteArray(String hex) {
       if (hex == null | hex.length() == 0) { return null; }
       byte[] ba = new byte[hex.length() / 2];
       for (int i = 0; i < ba.length; <math>i++) {
           ba[i] = (byte) Integer.parseInt(hex.substring(2 * i, 2 * i + 2),
16);
       return ba;
    }
    // byte[] to hex
    public static String byteArrayToHexString(byte[] bytes){
       StringBuilder sb = new StringBuilder();
       for(byte b : bytes){
           sb.append(String.format("%02X", b&0xff));
       }
```

```
return sb.toString();
}
```

실행 결과