

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package identity_server;

import java.io.ByteArrayInputStream;
import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.io.ObjectInput;
import java.io.ObjectInputStream;
import java.io.ObjectOutput;
import java.io.ObjectOutputStream;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;

/**
 *
 * @author rapha
 */
public class Utils {
    public static byte[] cryptObject(Object o, Cipher cryptor)
        throws IOException, IllegalBlockSizeException, BadPaddingException
    {
        ByteArrayOutputStream bos = new ByteArrayOutputStream();
        ObjectOutput out = new ObjectOutputStream(bos);

        out.writeObject(o);
        byte[] bytes = bos.toByteArray();

        out.close();
        bos.close();

        return cryptor.doFinal(bytes);
    }

    public static Object decryptObject(byte[] bytes, Cipher decryptor)
        throws IOException, ClassNotFoundException,
        IllegalBlockSizeException, BadPaddingException
    {
        ByteArrayInputStream bis = new ByteArrayInputStream(
            decryptor.doFinal(bytes)
        );
        ObjectInput in = new ObjectInputStream(bis);

        Object o = in.readObject();

        bis.close();
        in.close();

        return o;
    }

    public static byte[] hashPassword(String password, int clientSalt,
        int serverSalt) throws NoSuchAlgorithmException
    {
        MessageDigest md = MessageDigest.getInstance("SHA-1");

        md.update(password.getBytes());
        md.update(String.valueOf(clientSalt).getBytes());
        md.update(String.valueOf(serverSalt).getBytes());

        return md.digest();
    }
}
```

