Cryptography

Lab 1

Important: The goal of this exercise is to become familiar with openssl signing and verification. Parameters (both key length and the hash algorithm) are intentionally set so one can easily forge the signatures. Never use md5. Never use RSA with key length less than 2048 bits.

An automated grading system Gras uses digital signatures to ensure authenticity of files.

A public/private RSA key pair was generated together with a self-signed certificate.

GenCertificate
```
openssl genrsa -out cakey.pem 360
openssl req -new -md5 -key cakey.pem -out cacsr.csr
openssl req -x509 -md5 -days 365 -key cakey.pem -in cacsr.csr -out cacertificate.pem
```

Then, for each student and for each problem a file with grades is generated:

grade.txt
```
Student number: 123456
Problem number: 5/1
Grade: 0
```

In the next step, each file is signed:

SignService
```
openssl dgst -md5 -sign cakey.pem -out grade.sign grade.txt
```

Later, when before grades are sent out by email the signatures are verified.

Your goal is to generate forged grade.sign files.

You can download essential files from here:
`https://zagorski.im.pwr.wroc.pl/courses/crypto2018/lab5.tar.gz`

Problem 1 (5 points) The size (368-bits) of the RSA key is quite low. Try to find a private key and generate the signature under a modified grade.txt file by yourself.

You can use *e.g.,* CADO-NFS `http://cado-nfs.gforge.inria.fr/` or Msieve `https://sourceforge.net/projects/msieve/` to factor the key.

Problem 2 (5 points) People at Gras realized their mistake and they changed the way RSA key is generated.

Fortunately for you they did not change the way the sign SignService works (except the name of the file) – they still use md5 for signing.

(May be helpful: only the first three lines of the files are read).

VerifyService

```
openssl dgst -md5 -verify <
        (openssl x509 -in cacertificate.pem -pubkey -noout)
         -signature grade.sign grade.txt
```

GenCertificate

```
openssl genrsa -out cakeySec.pem 2048
openssl req -new -sha256 -key cakeySec.pem -out cacsrSec.csr
openssl req -x509 -sha256 -days 365 -key cakeySec.pem -in cacsrSec.csr -out cacertSec.pem
```

VerifyService

```
openssl dgst -md5 -sign cakeySec.pem -out grade.2048.sign grade.txt
```