Faculty of Computers, Informatics and Microelectronics

Technical University of Moldova

PSI

Laboratory work # 2

Registration number verification

Author:
Petru Negrei

Supervisor:
A. Railean

November 2014

1 Introduction

1.1 Objective

- Design and develop a registration number verification mechanism
- That relies on public key cryptography

1.2 Overview

Registration numbers - nobody likes them, especially when they are split into several chunks that have to go to different edit widgets (so you must perform more than one copy/paste operation from the keygen. Nevertheless, they are the most common method of verifying the authenticity of the installed application, and most commercial software applies such checks.

2 Implementation

- Frequency analysis
- Each substitution number

2.1 Cypher module

Below there are external ruby libraries used in this program.

- *openssl* - OpenSSL provides general purpose cryptography which includes RSA.
- *base64* - The Base64 module provides for the encoding and decoding of binary data using a Base64 representation.
- *securerandom* - Generate a random number-string
- *SECRET* - some predefined secret string.
- *PUBLIC_KEY_LOC* - public key generate by RSA.
- *KEY_LOC* - encrypted string with the private key.

```
require 'openssl'
require 'base64'
require 'securerandom'

SECRET = "s3cr3t"
PUBLIC_KEY_LOC = "public_key.pub"
KEY_LOC = "key.txt"
```

Generator steps:

- First we generate a random string with a secret string.
- Then we encrypt that key using the way of public key cryptography.
- Then encode the key in base64.
- Save the encoded key in a file, later to be read.

```ruby
class Generator

 def initialize
  @rsa = OpenSSL::PKey::RSA.new 2048
  save_public_key
  save_serial
 end

 private

 #'O4KLIVc-5u8C11E-s3cr3t-S9NSu1A-s4qWzcI-lUI9k_0-RV9ETlY
 # "P0X2lPU3OQAsQMIRUaocU3bMn8VAv6cK7xcs3cr3t7rPd4_Y"
 def generate_key
  tmp = (0..5).map { || generate_numbers } << SECRET
  tmp.shuffle.join #"-"
 end

 def generate_numbers
  SecureRandom.urlsafe_base64(5)
 end

 def save_public_key
  File.open(PUBLIC_KEY_LOC, 'w+') { |file| file.write @rsa.public_key.to_pem}
 end

  #encrypt with the private RSA key
  def encrypted_key
   @rsa.private_encrypt generate_key
  end

  def encoded_key
   Base64.encode64 encrypted_key
  end

  def save_serial
   File.open(KEY_LOC, 'w+') { |file| file.write encoded_key }
  end
end
```

```ruby
 module Verifier

  class Checker

   def initialize key
    @key = key
    @rsa = OpenSSL::PKey::RSA.new File.read(PUBLIC_KEY_LOC)
   end

   def result
    !(decrypted_key =~ /"#{SECRET}"/)
   end

   private

   def decoded_key
    Base64.decode64 @key
   end

   def decrypted_key
    @rsa.public_decrypt decoded_key
   end

  end

   def self.verify key
    Checker.new(key).result
   end
 end
```

Verifier steps:

- Reverse the base64 encoding.
- Decrypt the key using public key.
- Check the number and return a boolean value.

```
Generator.new
key = File.open(KEY_LOC, "rb").read
p Verifier.verify key
```

```
# public_key.pub
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxdoL7Xt2IrekMgQ8m2IF
IM/DnfsNvURqRL/Meds+acd5Uw9qyTZJ0AdDwNPJRP7+iRxeX9O/QQ376II/F342
/gJVZGNUaVfTAEvkfAI8uX6RT1UyQOF6FCH5DwO1h6iQqPKlspQtqDqoZedKzSS5
kJJ+yAh17xgM53YysgQAXoNO90/HsMQ1+jr7oIKnD84yBhYxfCGMTXNDYGMM3nXa
vyx5nmrlHk+W0TzSBawpFB1SMkOcHNoZvTJQ+TWcHj+pyzO9noFgrUq/KVuVDS9d
K6uekbs9t7gIRBaRcBL5wWnf/c5L2jlye9l3OZKs0mbVYRqnyt0dr2zi+JuJgkB5
YwIDAQAB
-----END PUBLIC KEY-----
```

```
# key.txt
xM7zDVYX2s/O7PTIMnJXbw62WAxLG7IAf60rB5dsNdJnwWyjk0UayIiOjZVJ
Ruc7hjO7g1koC+nDYFD72ArmFjOa1l1/AUQ9acLSl3/SJa7Wa5MhHRqi7ySZ
PcYbgNtgv/A7g/m5pPgtwFmXZy4io2IfbGbFZYssxhUZevuXRDO0JV3GaJPU
FvTI5u3JtFg6mrSZF1lZZdBhMnOKdpJFqBtkeGveXJaW7GCxHuVPYrOwqCb+
zq+aISwkuzA3p+K7XsoKGFMjPaZh1Pjwm4BkddNi1Jw4vvRZxWl/ZaJnMxYe
s7awg4qWayft8OYgdNx16fSpX2FVrqcQ5/jcUcVK3g==
```

```
# result
true
```

3 Conclussion

After making this laboratory work I learn more about public key cryptography and how it can be used in the case of hiding information about registration numbers or serial number. Also implemented a checker for verification of validity of the serial number, the method used here is far from reality of generating and verificiation of serial numbers but it serve the purpose to show the mechanism of publick key cryptography.