Facharbeit am
Städt. Gymnasium Bergkamen
von Malte Bublitz (Q1)

Thema:
Asymmetrische Kryptosysteme am Beispiel des ElGamal-Algorithmus

Fach:
Mathematik

Betreuende Lehrkraft:
Frau Meermeier

Inhaltsverzeichnis

4. Primitive Wurzel

7 A ! g . $\mathbb{Z}_p^*$ > <)
"

$$g^q \bmod p = -1 \bmod p$$

- 2 &* $) % 2 3
4 3$! ,

5. Das ElGamal-Verfahren

- # 4 35 % /BC 6 # ' 6 3
- (0 ! , - # 4 3
+ 7 f^{-1} # % f
! , - *) 5 # 4 3
D=) %+ !
) ! % ,

1. Schlüsselerzeugung[4]

&
+ "

/, # % ! + ! + $! p %+)
! ! $p = 2 * q + 1$ %+ %) 2
4 $! ,

, < + ! % $ % !

4 $x=8$ % 6 I

$C=(40353607,4)$,

. $z=a^{p-1-x} \bmod p=40353607^{11-1-8} \bmod 11=9$ + . "

$$M=z*b \bmod p$$
$$M=9*4 \bmod 11$$
$$M=3$$

6. Fazit

% !) % 5

@ + ! 4 0 !

, - > *

" - $a=g^k$ D

$! * <

) % $ I !

) !

@ % ,

7. Anhänge

A. elgamal.py

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
#
# ElGamal Beispielimplementierung
#
from random import randint

class ElGamalKey(object):
      def __init__(self):
            self.p = 11
            self.q = 5
            self.g = 0
            while not self.g**self.q % self.p == -1 % self.p:
                  self.g = randint(2, self.p-1)
            self.x = randint(1, self.p-2)
            self.A = self.g**self.x % self.p

      def getPrivateKey(self):
            return (self.x)

      def getPublicKey(self):
            return (self.p, self.g, self.A)

      def __repr__(self):
            return "<ElGamal Key Object :: priv(x=%s), pub(p=%s, g=%s, 
A=%s)>" % (self.x, self.p, self.g, self.A)
```

```
def Crypt(M, key):
    pubkey = key.getPublicKey()
    p = pubkey[0]
    g = pubkey[1]
    A = pubkey[2]
    if M < p:
        k = randint(1, p-2)
        a = g**k
        b = M * A**k % p
        C = (a, b)
        return C

def DeCrypt(C, key):
    privkey = key.getPrivateKey()
    x = privkey
    pubkey = key.getPublicKey()
    p = pubkey[0]
    a = C[0]
    b = C[1]
    z = a**(p-1-x) % p
    M = z * b % p
    return M

def main():
    myKey = ElGamalKey()
    print(myKey.__repr__())
    C = Crypt(3, myKey)
    print('C =',C)
    M = DeCrypt(C, myKey)
    print('M =', M)


if __name__=='__main__':
    main()
```

B. Literaturverzeichnis

Cameron 2003: Peter J. Cameron, Notes on Cryptography, 2003
http://www.maths.qmul.ac.uk/~pjc/notes/crypt.pdf

Dölle 2002: Lukas Dölle, Einwegfunktionen - Varianten und Beispiele, 2002
http://www2.informatik.hu-berlin.de/Forschung_Lehre/algorithmenII/Lehre/WS2002-2003/Perlen/Variationen.pdf

Meier 2005: Andreas M. Meier, The ElGamal Cryptosystem, 2005
http://wwwmayr.in.tum.de/konferenzen/Jass05/courses/1/papers/meier_paper.pdf

Plutarch/Eyth 1868: Πλούταρχος: Plutarchi vitae parallelae - Lysander, ca. 96 n. Chr., Übersetzung aus dem Altgriechischen von Eduard Eyth, Hoffmansche Verlags-Buchhandlung 1868
http://books.google.de/books?id=vcwCAAAAMAAJ&hl=de

Wätjen 2008: Dietmar Wätjen, Kryptographie. Grundlagen, Algorithmen, Protokolle, Spektrum, ²2008, ISBN 978-8-8274-1916-3

C. Abbildungsverzeichnis