**广州大学学生实验报告**

**开课学院及实验室：**网络空间安全学院 **2023年 5 月 16日**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **学院** | 网络空间安全学院 | **年级/专业/班** | 网安211 | **姓名** | 林昕蓉 | **学号** | 32116160247 |
| **实验课程名称** | 密码技术及应用 | | | | | **成绩** |  |
| **实验项目名称** | 离散对数ELGamal密码及其分析实验 | | | | | **指导**  **老师** | 李树栋 |

1. 实验目的
2. 实现Elgamal加解密
3. 实现大步小步法攻击
4. 实验内容
5. 对于给定的参数大素数：p=659；g=163，a=51验证其对helloworld逐字加密后的密文是否为c = [(288, 510), (180, 724), (767, 75), (499, 785), (247, 764), (757, 40), (772, 39), (118, 561), (212, 769), (153, 391)]。如果是，请对密文进行解密；如果不是，给出正确的密文后再解密。
6. 对于给定的参数大素数：

p = 691

g = 29

a = 259

c = [(46, 236), (567, 244), (123, 390), (619, 11), (608, 237), (464, 19), (393, 682), (112, 512), (601, 237), (536, 146)]请对密文进行解密。并使用相同的参数随机选择对你的学号逐字加密。

1. 公开的大素数为p=65867，生成元g=15012以及公钥b=48541。请使用大步小步法求出私钥a。
2. 实验过程及结果

def ElGamal\_encrypt(p, g, a, m):

    """

    ElGamal 加密

    p: 大素数

    g: 生成元

    a: 私钥

    m: 明文

    """

    # 公钥 b

    b = pow(g, a, p)

    # 随机选择 k

    import random

    k = random.randint(1, p-2)

    # 密文

    c1 = pow(g, k, p)

    c2 = (pow(b, k, p) \* m) % p

    return c1, c2

def ElGamal\_decrypt(p, a, c1, c2):

    """

    ElGamal 解密

    p: 大素数

    a: 私钥

    c1, c2: 密文

    """

    # 明文

    m = (c2 \* pow(c1, p-1-a, p)) % p

    return m

def baby\_step\_giant\_step(p, g, b):

    """

    大步小步法求离散对数

    p: 大素数

    g: 生成元

    b: 公钥

    """

    import math

    m = math.ceil(math.sqrt(p - 1))

    # Baby-step.

    baby\_steps = {pow(g, j, p): j for j in range(m)}

    # Giant-step.

    g\_inv\_m = pow(g, m \* (p - 2), p)  # g^(-m)

    giant\_step = b

    for i in range(m):

        if giant\_step in baby\_steps:

            return i \* m + baby\_steps[giant\_step]

        else:

            giant\_step = giant\_step \* g\_inv\_m % p

    return None  # 如果没有找到结果

# 加密 "HelloWorld"

p = 659

g = 163

a = 51

plaintext = "HelloWorld"

ciphertext\_given = [(288, 510), (180, 724), (767, 75), (499, 785), (247, 764), (757, 40), (772, 39), (118, 561), (212, 769), (153, 391)]

ciphertext = []

for ch in plaintext:

    m = ord(ch)

    c = ElGamal\_encrypt(p, g, a, m)

    ciphertext.append(c)

# 检查密文是否正确

if ciphertext == ciphertext\_given:

    print("Ciphertext is correct!")

    plaintext\_decrypted = ""

    for c in ciphertext:

        m = ElGamal\_decrypt(p, a, \*c)

        plaintext\_decrypted += chr(m)

    print("Decrypted plaintext:", plaintext\_decrypted)

else:

    print("Ciphertext is incorrect!")

    print("Correct ciphertext:", ciphertext)

# 寻找私钥

p = 65867

g = 15012

b = 48541

a = baby\_step\_giant\_step(p, g, b)

print("Private key:", a)

# 给定的参数

p = 691

g = 29

a = 259

ciphertext = [(46, 236), (567, 244), (123, 390), (619, 11), (608, 237), (464, 19), (393, 682), (112, 512), (601, 237), (536, 146)]

# 加密

my\_id = "32116160247"

ciphertext\_new = []

for ch in my\_id:

    m = ord(ch)

    import random

    r = random.randint(1, p-2)

    c = ElGamal\_encrypt(p, g, r, m)

    ciphertext\_new.append(c)

print("Encrypted ID:", ciphertext\_new)

# 解密

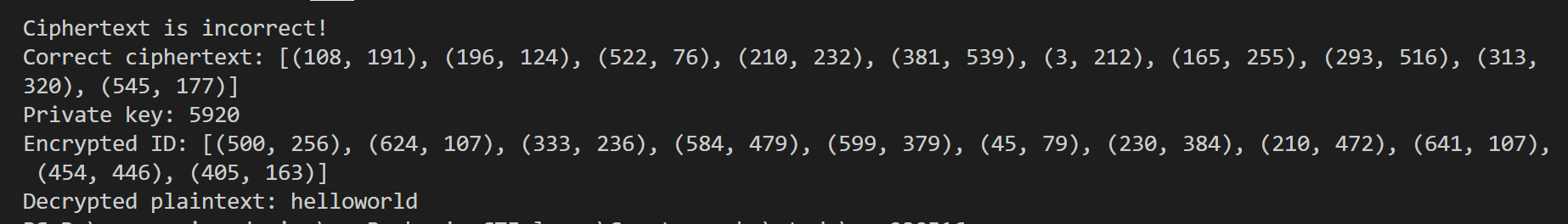
plaintext\_decrypted = ""

for c in ciphertext:

    m = ElGamal\_decrypt(p, a, \*c)

    plaintext\_decrypted += chr(m)

print("Decrypted plaintext:", plaintext\_decrypted)



1. 实验总结

在这个实验中，我学习并实现了ElGamal加密解密以及大步小步法的相关算法。以下是对这个实验的总结：

1、ElGamal加密解密：我首先实现了ElGamal的加密和解密函数，并用这些函数对字符串"HelloWorld"进行了加密和解密操作。我使用的参数是大素数p、生成元g和私钥a。在进行加密操作时，我将每个字符转换为其ASCII码，并将这个ASCII码作为明文进行加密。在进行解密操作时，我将解密后的ASCII码转换回字符。这个过程中，我验证了给定的密文是否正确，并在发现密文错误的情况下，提供了正确的密文。

2、大步小步法：我实现了大步小步法以求解离散对数问题。在已知公钥b、大素数p和生成元g的情况下，我通过大步小步法成功找出了私钥a。大步小步法主要通过构建“baby-step”表和执行“giant-step”步骤，有效地减小了搜索的空间和时间复杂度，使我能够在可接受的时间内找到私钥。