Project 3

Christian Johnson Aidan Andersen

April 27, 2024

Contents

1 Given Information

Problem 1

- $c_1 = 257261 \pmod{303799}$
- $c_2 = 117466 \pmod{289279}$
- $c_3 = 260584 \pmod{410503}$

RSA Moduli:

- $n_1 = 303799$
- $n_2 = 289729$
- $n_3 = 410503$

Problem 2 $p = 1234567891$ $q = 987654323$ $e = 127$ $m = 14152019010605$

2 Python Code

```
def egcd(a, b):
    if a == 0:
        return b, 0, 1
    else:
        gcd, x, y = egcd(b % a, a)
        return gcd, y - (b // a) * x, x

def mod_inverse(a, m):
    gcd, x, y = egcd(a, m)
    if gcd != 1:
        raise Exception('Modular inverse does not exist')
    else:
        return x % m

def chinese_remainder_theorem(n, a):
    # Calculate N
    N = 1
    for ni in n:
        N *= ni
```

```python
    # Calculate x
    x = 0
    for ni, ai in zip(n, a):
        Ni = N // ni
        xi = ai * mod_inverse(Ni, ni) * Ni
        x += xi

    return x , N
```

3 Problem 1

3.1 Part A

Find an x with $0 \leq x \leq n_1n_2n_3$ and $x \equiv c_1 \pmod{n_1}$, $x \equiv c_2 \pmod{n_2}$, $x \equiv c_1 \pmod{n_2}$. $0 \leq x \leq 36132219741486913$

```python
n=[303799,289729,410503]
c=[257261,117466,260584]
[num, mod]=chinese_remainder_theorem(n,c)
f"x={num%mod}"
```

x=25990919649605545

3.2 Part B

Show that $0 \leq m^3 \leq n_1n_2n_3$ $0 \leq m^3 \leq 36132219741486913$

By definition, we know that if $a \equiv b \pmod{n}$ then $a^m \equiv b^m \pmod{n}$. Thus, if $m^3 \equiv c_i \pmod{n_i}$, then $m \equiv \sqrt[3]{c_i} \pmod{n_i}$. In RSA, m should be less than n_i, which means $m^3 < n_i^3$. This means that m must be less than 117466^3, at the most. It follows that m must also be less than $n_1 * n_2 * n_3$, since $36132219741486913 > 117466^3$

3.3 Part C

Show that $x = m^3$. In part A, we found x such that $x \equiv c_1 \pmod{n_1}$, $x \equiv c_2 \pmod{n_2}$, and $x \equiv c_3 \pmod{n_3}$. We also know from the given information that $m^3 \equiv c_1 \pmod{n_1}$, $m^3 \equiv c_2 \pmod{n_2}$, and $m^3 \equiv c_3 \pmod{n_3}$. We know, by definition, that solutions to the chinese remainder theorem are unique, therefore we can conclude that $x = m^3$.

3.4 Part D

Decode the message m.

$m^3 = 25990919649605545$
$m = \sqrt[3]{25990919649605545}$
$\therefore m = 296215.114998$

4 Problem 2

4.1 Part A

Find $m^e \pmod{p}$ and $m^e \pmod{q}$. Use the chinese remainder theorem to combine - $c \equiv m^e \pmod{pq}$

- $m^e \pmod{p} = 14152019010605^{127} \pmod{1234567891} = 1156569072$
- $m^e \pmod{q} = 14152019010605^{127} \pmod{987654323} = 812538893$

Chinese Remainder Theorem - Given $c_1 \equiv a \pmod{m_1}$ and $c_2 \equiv a \pmod{m_2}$, if m_1 and m_2 are relatively prime, then there is some $c \pmod{m_1 * m_2}$ such that $c \equiv c_1 \pmod{m_1}$ and $c \equiv c_2 \pmod{m_2}$.

In our case, we have $m^e \pmod{p}$ and $m^e \pmod{q}$. $gcd(p, q) = 1 \therefore$ Relatively Prime From this, we can see that there exists $c \pmod{p * q}$.

```
m=[14152019010605**127, 14152019010605**127]
p=[1234567891, 9876534323]
[num, mod]=chinese_remainder_theorem(p,m)
c=num%mod

# Result is c(mod m1*m2)
result=c%(p[0]*p[1])
f"result={result}"
```

```
result=9868895527985399785
```

4.2 Part B

```
m=[14152019010600**127, 14152019010605**127]
# adjusted me: 14152019010605 to 14152019010600 in me(mod p)
p=[1234567891, 9876534323]
[num, mod]=chinese_remainder_theorem(p,m)
c_new=num%mod

# Result is c(mod m1*m2)
f=c%(p[0]*p[1])
pq=p[0]*p[1]
f"f={f}\npq={pq}"
```

```
f=98688955279853997785
pq=121932521495352227793
```

```
[gcd,_,_] = egcd(result-f,p[0]*p[1])
f"GCD = {gcd}"
```

```
GCD = 121932521495352227793
```

```
# Check if GCD is a factor of PQ
gcd/(p[0]*p[1])
# THEY"RE EQUAL... WHY??
```

```
1.0
```