COMP90043 Cryptography and Security: Assignment 1

Martin Raunkjær Andersen
mrandersen@student.unimelb.edu.au
1011164

August 24, 2018

1 Question 1

Euclid's algorithm and Divisibility properties

1A

Let a, b, c, d be integers and GCD(a, b) = 1. If c|a and d|b, prove that GCD(c, d) = 1.

This proof is based on two observations:

1. Let $a, b, c, x, y \in \mathbb{N}$ and c be a common divisor of a and b. c will also be a common divisor of ax and by for $x, y > 0$.
2. for $a, b, x \in \mathbb{N}$ if $b|a$ there exist x such that $bx = a$.

based on these observations, it is possible to state a proof by contradiction. if there exists a case of $GCD(c, d) = e$ where $e \in \mathbb{N}, e > 1$, it would be possible to find numbers $x, y \in \mathbb{N}$ and $x, y > 0$ where $cx = a$ and $dy = b$. Based on observation 1, e would then be a common divisor of a and b. As $e > 1$ this would make $gcd(a, b) > 1$. This is a contradiction. No such number e can exist if $gcd(a, b) = 1$, $c|a$ and $d|b$.

1B

You should perform the following implementation tasks in a language of your choice. You are at free to employ any underlying integer arithmetic library.

i *Implement the extended Euclid's algorithm and only submit the code here.*

ii *Implement a function in that takes two positive numbers a, n, a ¡ n and output the inverse of a mod n based on your extended Euclid's algorithm. Only submit the code for the function.*

The following code snippets contain the implementation of the extended euclidean algorithm and the inverse module in Python.

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

def modInverse(a, n):
    gcd, x, _ = extendedEuclideanAlgorithm(a, n)
    if gcd == 1:
        return x % n
    return None
```

1C

Now choose n as a product of two primes p1 and p2 (p1, p2 $> 10^{25}$), use your implementation in (b) and output the results of your inverse a mod n function for the following cases:

i *a is a random positive integer less than n*

ii *a is a multiple of* p_1

iii *a is n -1*

The following code snippet contains the answers to the questions in 1C. The commented values after a line is the result of the evaluation of that line.

```
p1 = 996622688165337716830009929239
p2 = 716029746093661370396795904119
n = p1 * p2
# 713611490358209018128116052497523311086662789934136218635441

i = modInverse(291357, n)
# 553456279592608982823045707330732906913618060715331682618277
ii = modInverse(p1 * 4, n)
# None
iii = modInverse(n-1, n)
# 713611490358209018128116052497523311086662789934136218635440
```

2 Question 2

General Security and Classical Ciphers

2A

Explain with an example how security risks and attacks are different. Name a security attack that has happened on computer systems in recent years. Describe how the attack took place in no more than half a page.

The Equifax data breach of 2017 is one of the most impactful cyber attacks in history, exposing private information such as names, social security numbers, birth dates and, in a few cases, credit card information of 145.5 million people[1]. It is a good example of a known *security risk* being taken advantage of to gain entry to a website. A security risk is a vulnerability in a computer system, while an *attack* is the act of breaching the system using such a vulnerability.

In the case of the Equifax hack, it was a known vulnerability called CVE-2017-5638 in the Apache Struts 2 framework, which various Equifax owned sites used, that the attackers used to gain entry. It allowed the attackers to perform a *Remote Code Execution* attack, by injecting code in the HTTP headers of requests to the sites, resulting in the complete compromise of the web servers and the applications running on them.[2]

The Apache Foundation were first notified of CVE-2017-5638 on february 14 of 2017. An update to address the vulnerability was released in the start of march, but Equifax did not apply the update, despite independent researchers notifying them of the vulnerability. From mid may til late august, attackers had full access to the servers, until the attacks were discovered and the update applied. In the start of september the breach was announced, followed closely by retirements in the top ranks of Equifax and numerous class action lawsuits.[1]

2B

Consider the following version of a classical cipher where plaintext and ciphertext elements are from integers 0 to 27. Note that this alphabet may be used when the plaintexts are 26 English characters and two punctuation symbols, viz: “, ”“.”. The encryption function, which takes any plaintext p to a ciphertext c, is given by

$$c = E(a, b)(p) = (ap + b) \ mod \ 28$$

where a and b are integers less than 28.

i *What is the decryption function for the scheme?*

The decryption function is:

$$D(c) = a^{-1}(x - b) \ mod \ 28$$

where c is the ciphertext and a^{-1} is the modular multiplicative inverse of a modulo 28

ii *How many different non-trivial keys are possible for the scheme?*

Excluding the trivial keys, there are a total of $a * b = 12 * 28 = 336$ different possible keys. a in this instance can only be 12 different values as it must be a coprime to the length of the alphabet, less than the length of the alphabet. b is the number of possible addition shifts to a, which is the length of the alphabet.

iii *What are the complexities of Ciphertext only Attack and Chosen Plaintext Attack on the scheme?*

- The complexity of a brute force Ciphertext only attack without prior knowledge is $O(a * b)$, as all possible keys must be checked. This number can be reduced if the attacker knows the plain text language or can perform traffic analysis.
- The complexity of a Chosen Plaintext Attack is $O(1)$, as the attacker can easily obtain the plaintext value for two ciphertexts, which can be used to obtain the key through a system of equations.

3 Question 3

Poly-alphabetic Cipher

For this question, we consider the Hill cipher given in the textbook on an alphabet A consisting of 26 English characters and the three punctuation symbols, viz: ",", "." and " " (space), which corresponds to integers 0 to 28. Here the plaintext (mod 29 characters) is processed successively in blocks of size m, $m > 1$ digits at a time. The encryption algorithm takes a block with m plaintext digits and transforms into a cipher block of size m using a key matrix of size $m \times m$ by the linear transformation is given by:

$$c_1 = (k_{1,1}\, p_1 + k_{1,2}\, p_2 + \ldots + k_{1,m}\, p_m) \ mod\ 29$$
$$c_2 = (k_{2,1}\, p_1 + k_{2,2}\, p_2 + \ldots + k_{2,m}\, p_m) \ mod\ 29$$
$$\ldots$$
$$c_m = (k_{m,1}\, p_1 + k_{m,2}\, p_2 + \ldots + k_{m,m}\, p_m) \ mod\ 29$$

3A

How many different keys are possible in the system?

The key permutation space is all the possible matrixes of $m \times m$ dimensions with each cell containing one of 28 values. In mathematical terms: 28^{m^2}.

3B

The "ciphertext only attack" on the system is hard to mount for the cipher. However, the cipher is easily broken with a known plaintext attack. Illustrate the steps for the attack.

As the Hill Cipher is completely linear, it is possible to determine the key of m dimensions given m pairs of known plaintext / ciphertext, in m linear equations with m unknown variables. One must guess m, but trial and error reveals the answer quickly. The following steps illustrate the known plaintext attack for a key matrix of 2 dimensions, given plaintexts $p1$, $p2$ and their corresponding ciphertexts $c1$, $c2$:

1. Set up the known pairs as such:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} p1_1 \\ p1_2 \end{bmatrix} = \begin{bmatrix} c1_1 \\ c1_2 \end{bmatrix}, \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} p2_1 \\ p2_2 \end{bmatrix} = \begin{bmatrix} c2_1 \\ c2_2 \end{bmatrix}$$

   Each of which represents two linear equations of the format:

$$a * p_1 + b * p_2 = c_1$$

$$c * p_1 + d * p_2 = c_2$$

2. Solve the linear congruences modulo the size of the alphabet. The values of the variables a, b, c, d denotes the key in the format:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

3. Use the discovered key to translate the rest of the message.

3C

An adversary discovers the following ciphertext encrypted using the Hill cipher with m = 4: (92 characters, no spaces)

BGB.D,LYIQJNBGSQLXWRIIBKESOWGEWSXCCAC.ZCPPW.YIAFPDNB UDOBPSFIKBSTRIQQFDOUHBZSRXVULMI,JVSGFUUG

If the following plaintext and ciphertext blocks are given, decrypt the cipher by giving the plaintext and the key used in the encipherment.

Plaintext	*Ciphertext*
CTRL	*JYZP*
CAPS	*QEPQ*
HOME	*CHZS*
PGUP	*GLXF*

References

[1] *Equifax data breach timeline — csrps.com - csrps*, https://csrps.com/meticulous-timeline-equifax-data-breach, (Accessed on 08/19/2018),

[2] *Equifax hack: How it all happened — netsparker*, https://www.netsparker.com/blog/web-security/how-equifax-data-breach-hack-happened/, (Accessed on 08/19/2018),