CS 458 A3
Patrick Wong
pn2wong
20317267

[Question 1b]

Reading in both ciphertexts, we XOR the two ciphertexts byte by byte and store this as another character string, XOR12. Then we do a dictionary attack on this XOR12. For example, we guess a common string found in most sentences, " the " (spaces included), and XOR this string to every possible five character block in XOR12. We find that " the " maps to "March" – this means that " the " exists in one plaintext, and "March" exists in the other plaintext. We can build on this newly discovered word, "March", by turning it into " March ", and XOR that to the XOR12. Eventually, enough text (guesstext1 or guesstext2) is discovered to make a search that produces a small amount of results in Wikipedia.

By estimating the start and end of the string guesstext1 (set to be 300 characters in length) from the Wikipedia article, the exact start and end of the string can be confirmed with an XOR between XOR12 and guesstext1 - if unintelligible output is produced, then the guessed start or end is incorrect. The output between XOR12 and guesstext1 will be a 300 character length string from another Wikipedia article, guesstext2. The source of this article can be easily confirmed by another search on Wikipedia.

Below is the python code used to guess the plaintexts.

```
f1 = open('ciphertext1','r')
f2 = open('ciphertext2','r')
f1 = f1.read()
f2 = f2.read()

def btoc(c1,c2):
    return chr(ord(c1) ^ ord(c2))

s = ""
lf1 = len(f1)
for i in range(lf1):
    s += btoc(f1[i], f2[i])

s_new = ""

#https://en.wikipedia.org/wiki/Srm_%28Unix%29
#https://en.wikipedia.org/wiki/OBD_Memorial

the_string = "Attempting to secure delete a file with multiple hard links results in a warning from srm stating that the
current access path has been unlinked, but the data itself was not overwritten or truncated. This is an undocumented
feature of srm 1.2.8 on Mac OS X 10.9, and is erroneously documented in 1.2."
#the_string = "The project was launched in November 2006 and the online database was opened for the public access on
March 31, 2007. The main sources of information are funds of the Central Archive of the Russian Ministry of Defence
(TsAMO) and funds of Military-Memorial Center of the Armed Forces of Russia, inclu"
ts2 = ""
ts_len = len(the_string)


i = 0
while i < lf1:
    y = 0
    s_new = ""
    for j in range(i,i+ts_len):
        s_new += btoc(the_string[y],s[j])
        y += 1
#    print str(i) + ": " + s_new                #to get the plaintext
    i += 1
    if i+ts_len >= lf1:
        i = lf1
```

[Question 2]

(a)

Since we only want one record, we follow this format to get Lucille's salary: q(C) = q(C or T) + q(C or !T) – q(S)

Thus, our tracker is T: WHERE Occupation='Staff'

We are given that have the employees are either Staff or Specialists. We are also given K = N/8.

Checking if N/2 is in [k, N-k]

- ⇨ N/8 <= N/2 <= N – N/8
- ⇨ N/4 <= N <= (2 – ¼)N
- ⇨ N/4 <= N <= (7/4)N

Since N is an integer, then this inequality holds true.

The three queries:

SELECT SUM(Salary) FROM Employee WHERE Occupation='Staff' or Name='Lucille' (C or T)

SELECT SUM(Salary) FROM Employee WHERE Occupation<>'Staff' or Name='Lucille' (C or !T)

SELECT SUM(Salary) FROM Employee (S)

(b)

We can get Rachel's salary by using a modified binary search. Assuming that salaries are evenly distributed in the range of [0, 200000] to simplify the math, specifically that the number of employees with salary >= 100k is in (k, N-k), and similarly for employees with salary < 100k. Then, we initiate the binary search like so:

1. SELECT COUNT(*) FROM Employee WHERE Salary >= 100k OR Name='Rachel'
2. SELECT COUNT(*) FROM Employee WHERE Salary >= 100k

If both queries fail, this means that there is no one with Salary >= 100k, including Rachel.

Otherwise, both queries are correct. We then have two possibilities: (i) Rachel's salary >= 100k, or (ii) Rachel's salary < 100k. If it is scenario (i), we repeat (1) and (2), except changing salary to 150k. Otherwise, we have scenario (ii): Rachel's salary < 100k. This results in query 1 having a count greater than query 2 by one. Then we have a new set of queries:

3. SELECT COUNT(*) FROM Employee WHERE Salary >= 100k OR (Name='Rachel' AND Salary < 50k)
4. SELECT COUNT(*) FROM Employee WHERE Salary >= 100k OR (Name='Rachel' AND Salary > 50k)

Since we know the count of employees where Salary >= 100k is determined from a number of records in between [k, N – k], then the second part of the query (Name='Rachel' AND Salary <= 150k) or (Name='Rachel' AND Salary > 150k) will add 0 or 1 to the COUNT of the query. We can continuously apply binary search on the bounds, until we have a bound where the lower bound differs from the upper bound by 1, e.g. (Name='Rachel' AND Salary <= 4907 AND SALARY > 4906), then it is obvious that Rachel's salary is 4907.

The assumption that that the number of people with Salary >= 100k was used for the sake of brevity; in reality, we need to find a lower and upper bound on Salary such that Rachel's salary is not contained within these bounds. Also, that the COUNT of this range is within (k, N – k) – thus if Rachel's salary is found, adding 1 to the count will not make the query fail.

(c)
This table is not 3-anonymous. Given birthdate of 83* and occupation of Specialist, it does not match at least 3 other records in the table. We can produce a 3-anonymous table by removing the second digit in birthdate:

Name	Birthdate	Occupation	Allegiance
*	7**	Specialist	Quendor
*	8**	Specialist	Quendor
*	7**	Staff	Quendor
*	7**	Specialist	Antharia
*	7**	Staff	Antharia
*	7**	Specialist	Quendor
*	8**	Specialist	Antharia
*	8**	Specialist	Antharia
*	7**	Staff	Kovalli
*	7**	Staff	Kovalli

Thus, there are at least three records with birthdate 7** and occupation of Specialist, 7** and Staff, and 8** and Specialist.
This table is 1-diverse.

[Question 3]
(a)
Under fair use, a university student can bypass a copy protection mechanism if, for a course project or assignment, screen capture software is not sufficient enough for their purposes.

(b)
TracFone was opposed to cellphone unlocking if it was not clear that phone trafficking was forbidden; the purchase of prepaid phones that are unlocked and resold, usually abroad, for profit.
The Alliance of Automobile Manufacturers and General Motors were opposed because they wanted their mobile connectivity software to be exempt from legal unlocking.

(c)
Devices or machines that can be purchased by consumers (including voting machines), motorized land vehicles, and implanted medical devices (i.e. a pacemaker) and their corresponding monitoring systems are given research exemptions.

(d)
It is legal to bypass copy protections for educational purposes, i.e. a course project or lectures, but it is not legal to distribute such software publically. This allows the software to be available to non-academia, and thus be used for unlawful acts.

[Question 4]
The average case is linear on the number of blocks in the message, with a worst case O(NbW). N is the number of blocks in the message, b the number of bytes per block, and W is the number of possible words the block can be.

[Question 5]
We can encrypt the original plaintext and then apply the CBC encryption scheme on the resulting message. This provides the property of confidentiality – even if the CBC scheme is broken, the attacker receives an encrypted payload, rather than a plaintext message. The problem with CBC is that after a padding attack, the resulting message was in plaintext.

[Question 6]
We can completely omit step 5 in section 3.1 Since the number of padded bytes is listed in the final byte of the message, we do not need to check the alternating padding byte sequence as given in the article.