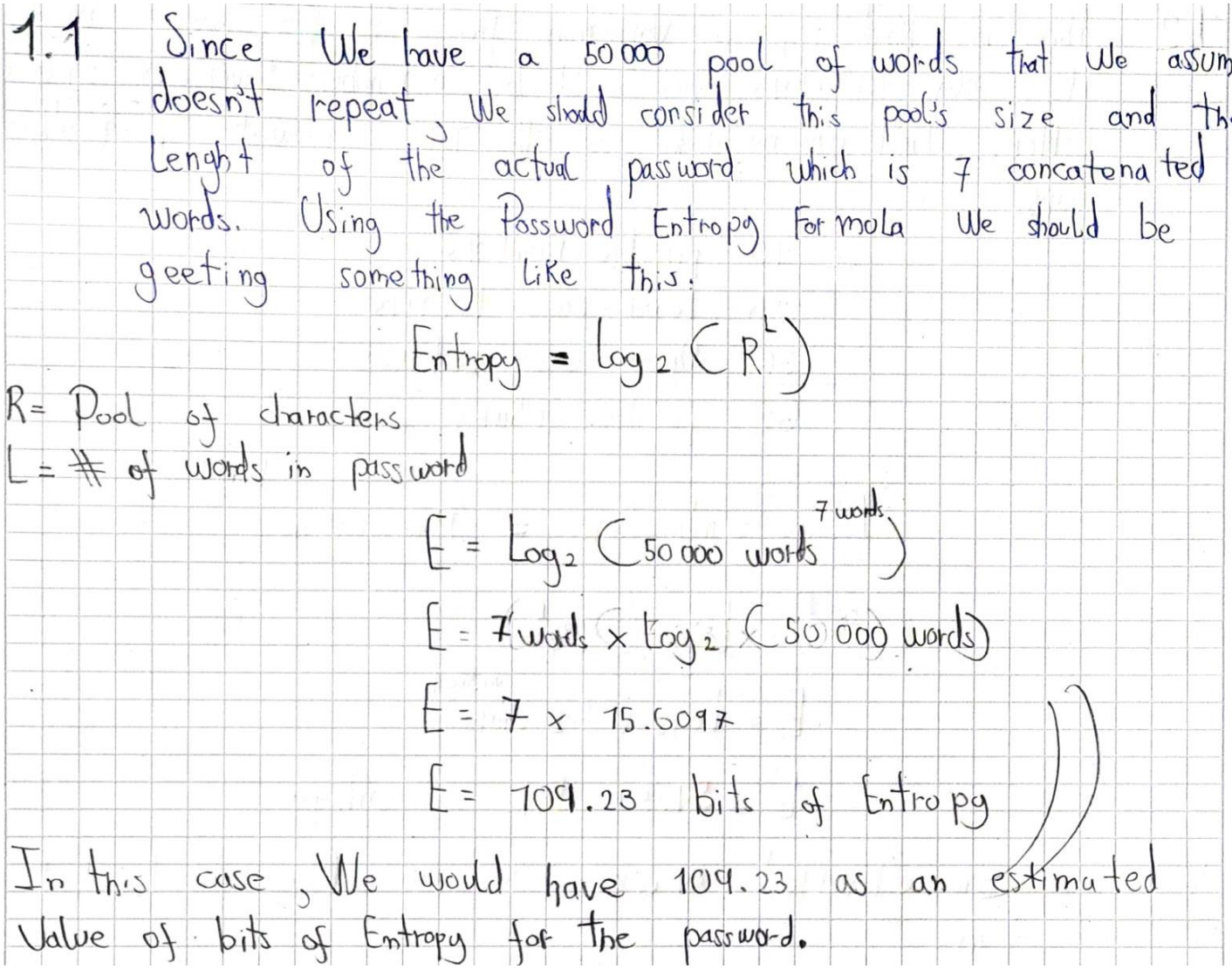
Homework 2

Guillermo Cisneros 2023-10-20

00206286

Exercise 1:

1. *Suppose a password is chosen as a concatenation of seven lower-case dictionary words. Each word is selected uniformly at random from a dictionary of size 50,000. An example of such a password is "mothercathousefivenextcrossroom". How many bits of entropy does this have?*



2. *Consider an alternative scheme where a password is chosen as a sequence of 10 random alphanumeric characters (including both lower-case and upper-case letters). An example is "dA3mG67Rrs". How many bits of entropy does this have?*

1.2 This is kind of a Variation from the first case since its only 10 chars but could be a number, lower or upper case letter. First we need to see the total amount of possible chars.

+ 10 possible numbers
26 Lower-Case Letters
26 Upper-Case Letters
62 possible chars

Possible combinations = Possible chars$^{\text{lenght}}$

$$E = \log_2(P_c^L)$$

$$E = \log_2(62 \text{ chars}^{\text{10 chars}})$$

$$E \approx 59.54 \text{ bits of entropy}$$

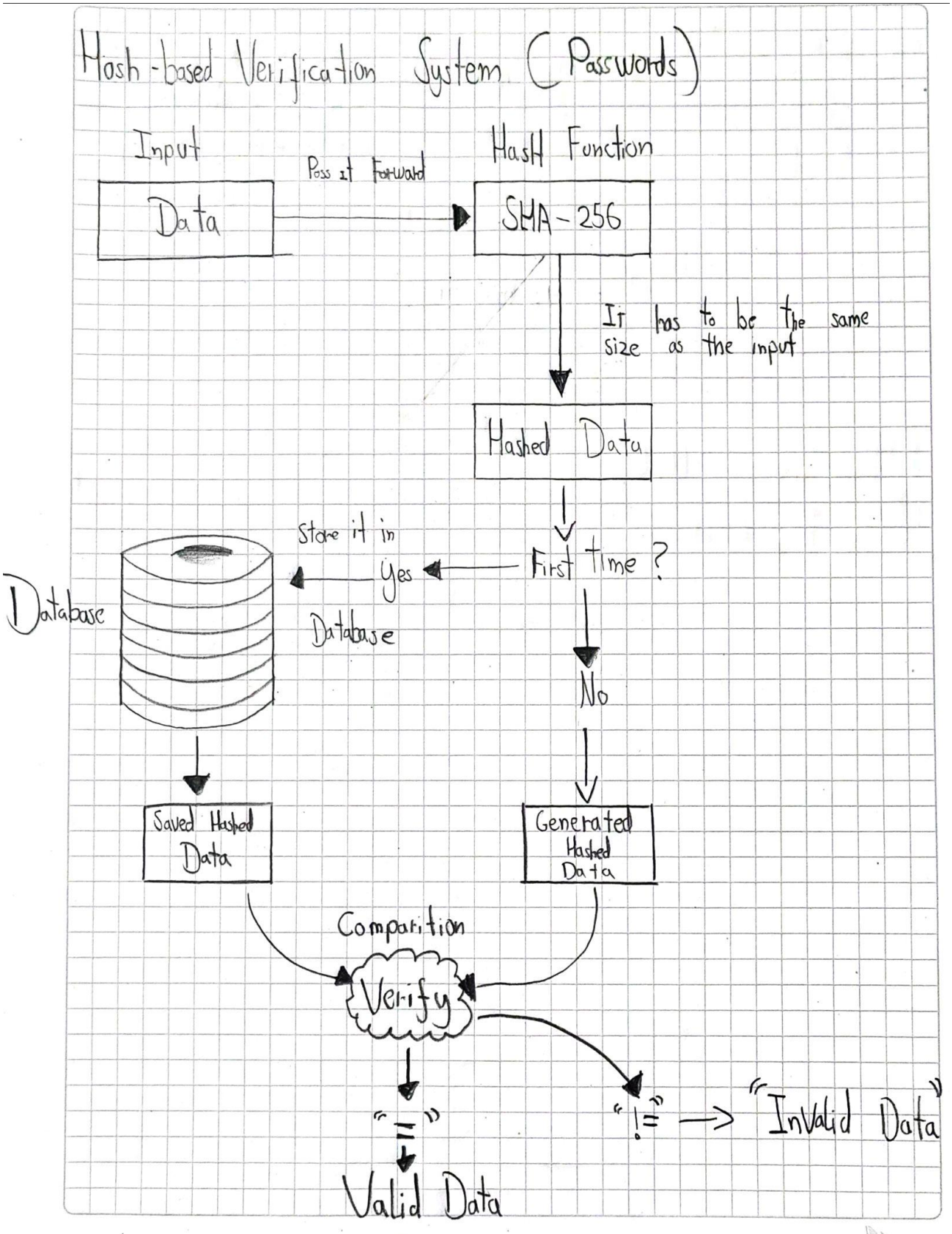
We have an estimated 59 or 60 bits of entropy for this kind of situation

3. *Which password is better, the one from 1. or 2.?*

1.3 On the level of Entropy, the first password would be the strongest since its entropy bits (109-110 bits) are Far Higher than those of the 10 char lenght.

Exercise 2:

1. ***Design a data verification system using hash functions. Explain the steps involved in the process.***



The Diagram in question describes a simple yet effective way to make a Hash-based verification system. This is based upon the use of passwords which have the intended effect of being stored in the system´s database for later verification. First off we as users must put our password as input data in the assumed UI of the service, the intention is to pass it through a Hash Function (SHA-256 just for example purposes).

The hashed data should have the same length as the one of the original forms of the data. The Next step is the most important in logistical matters since it will verify if the password has been submitted previously. If it's the first time, then there shouldn't be any more processes done but store it in the Database of the system. This in fact will give us later a saved hashed data. The next time we put the data as input in the interface, we will call the data stored in the Database and it will be compared with the one submitted.

Hash codes aren't supposed to be identical in normal cases and a show of this is that any little change can create always different hash codes. Since it's something that shouldn't happen, the stored and generated hash are compared and if they are the same then we can confirm de integrity and validity of the password submitted. Its not something fancy or anything but it's intended to be reliable in the long run.

2. ***Discuss the advantages and disadvantages of using hash functions for data verification.***

Hash function for this kind of things is actually very useful in the long run, they in contrast to other methods are quite fast since they don't really require a complex algorithm and can be easily implemented. Adding something to it, they are great at protecting data since no hash is unintentionally made the same, you can be sure that this system and the data it holds will have quite the protection. This also proves to be useful since no hash can be the same in the normal way, collisions as we know them are quite impossible unless you design a system that holds by comparing itself with a saved hash to verify someone. They are all around great since its an easy but logical way to approach this verification problem.

However, not everything is flowers and sunshine since in order to achieve some great things, we have to give various things in exchange. They are not susceptible to advanced attacks or techniques and since their only priority is authenticity, hackers or any malicious party can aim for their weakness. In their functionality, hash functions must emit codes in an orderly manner because if even the smallest of mistakes happen the whole system goes down. Another letdown to it is that even thou they provide integrity and authenticity for our communications; they were never stated to be secretive or deniable to the users.

3. ***Provide an example of a real-world application where a data verification system using hash functions is used.***

It's basically a given when you download files from the internet since they can be easily tampered of have omitted parts that in return gives us an incomplete work. Another way it used it's in a way the infrastructure we described since many login interfaces implement this method of hashing to verify passwords and user data. You can see this in various websites and mobile apps that require an authenticity segment to proceed to the app and many developers have decided to use. Its has even seen a resurgence when people started using digital signatures to confirm their authenticity and validation in official documents and files.

Exercise 3:

1. ***Define what a Message Authentication Code (MAC) is and how it is used in cryptography.***

MAC o Message Authentication Code it's a block of fixed data generated from the content of the original message we are sending. It basically serves to see if a message is safe in terms of not having its contents modified and to ensure that the message its authentic or is coming from the right sender. This is a way to know for the one receiving the message that this has not been altered or damaged in any way since its body is basically a part of the og message. Since cryptography is based on hiding or securing one´s data, MAC comes in handy for this task for allowing us to know the origin and safety of its content.

2. ***Explain the process of generating and verifying a MAC.***

For generating a MAC code, we first need to generate a secret or private key that only the two sides of the communication line can know, The sender and receiver. Once we have this, we need to choose a MAC generating code and apply a part of if not the entire message to it since any changes done to it can be easily verified. This plus our private key basically gives us our MAC code ready to use. Once we have this, its essential to send it along the original message since it's the only way we can verify the origins of it. All the previous work was for the generation of the code, but the verification also holds a bit of work to be done. The receiver now has the full content which is the message and the generated MAC from it and all he needs to do is to see if its valid. One thing that was essential in the previous process was the generation of the private key and since its known by both parties of the communication line. He needs to use the key with the message via the MAC function to generate a kind of makeshift MAC. This is good since we can use this actual MAC and compare it between the MAC sent by the sender. If they are the same, then its sure to assume it's the same message and it hasn't been tampered with along the communication process.

3. ***Discuss the importance of using MACs in secure communication systems.***

Just saying a MAC is important would be an understatement, its one of the only ways possible to see if a message is valid and if the communication line between two or more users is safe. How? Because aside from the key, the actual message is needed to remain the same since we are using parts of if not the entire message for the MAC. Any change could be essentially detected when you match both MACS by the receiving end and it surely is a simple yet effective way of seeing if a message its original. Any malicious party will be having a bad time trying to modify or damage the data since the MAC serves as the alarm for the communication line. You could even say in a way that this block of code is one of the stepping stones in the development of secure communication systems since its introduction has bring forth new ways of security like the Digital signatures that apply a similar logic like this one.

Exercise 4:

Given the values of p = 17 and q = 23, generate a pair of keys for RSA.

4. For the Key generation, We need to find n from the following

p = 17
q = 23

$n = pq$
$n = 17 \times 23$
$n = 391$

Now that We have n, Eulers totient is next...

$\phi(n) = (p-1)(q-1)$
$\phi(391) = (17-1)(23-1)$
$\phi(391) = 16 \times 22$
$\phi(391) = 352$

e is a value between 1 and 352 in this case, We will try n = 5 to Verify

e = 5 -> Public Key

$a = 5 \qquad b = 352$

1. $c = 352$
   $d = 5$
   $r = 2$

$352 \mid 5$
$02 \quad 70$
2

$2 = 352 - 70 \times 5$
$2 = b - 70a$

2. $c = 5$
   $d = 2$
   $r = 1$

$5 \mid 2$
$1 \quad 2$

$1 = 5 - 2$
$= a - (b - 70a)$
$= 71a - b$

3. $c = 2$
   $d = 1$
   $r = 0$

$2 \mid 1$
$0 \quad 2$

$0 = 2 - 2(1)$
$0 = 2 - 2(71a - b)$
$0 = a - 2(71a - b)$
$0 = 141a - 2b$

$e^{-1} = 141 \bmod 352$

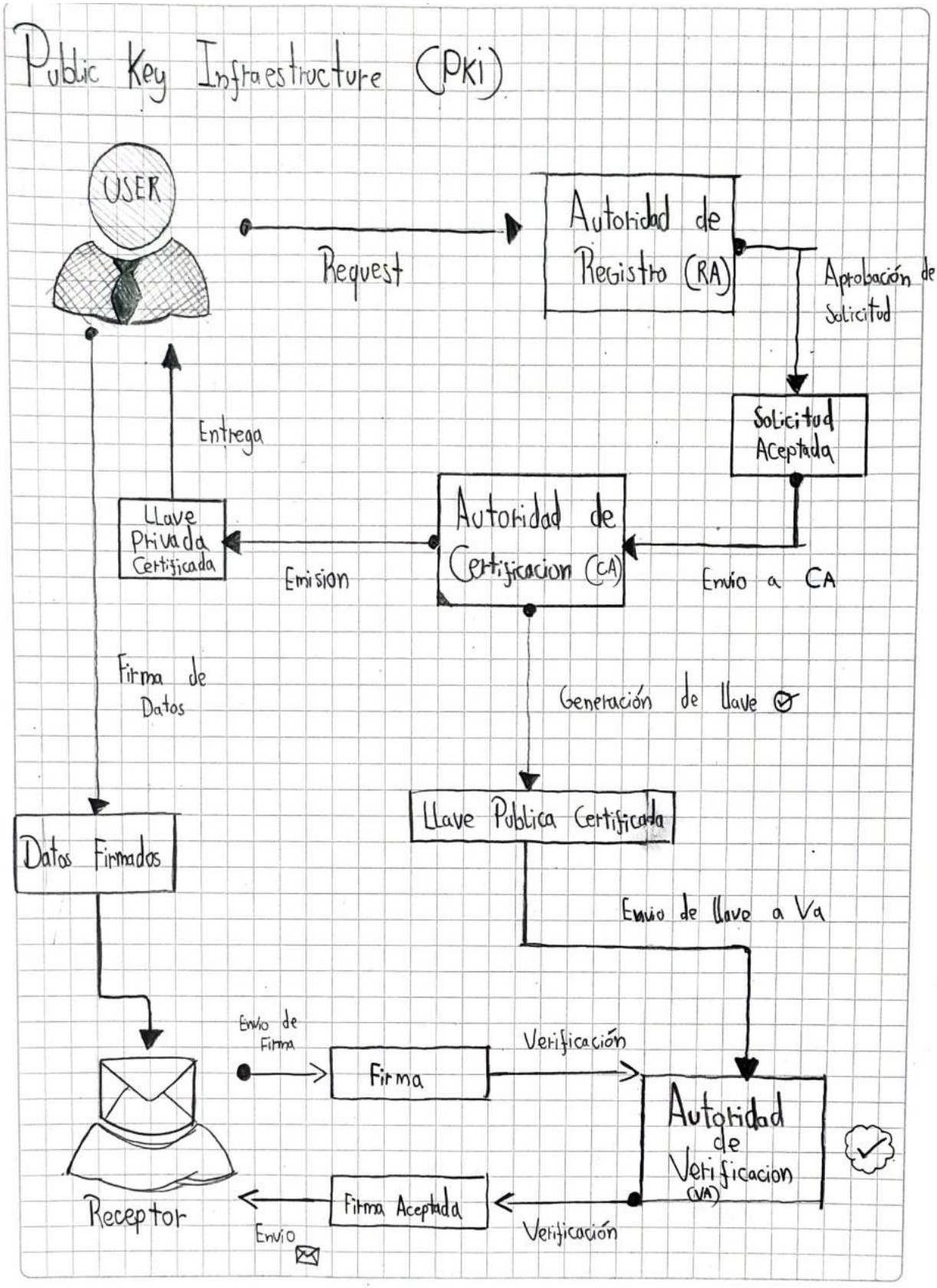
We now have our Keys

Public Key : (341, 5)

Private Key : (341, 141)

Exercise 5:

1. ***Design a public key infrastructure (PKI) system. Explain the components and their roles in the system.***



The previous diagram depicts how a PKI system is basically made. As all things, we first start with the user that will register to obtain a certificate. He does this by first sending a request to the RA or Registry Authority. The RA oversees receiving an analyzing the request made by the user to see if he meets the conditions of his request. Once this entity approves of this, it sends it forward to the CA or Certification Authority. CA play a similar role as the RA but they rather confirm the information given by them and permit the creation of the certified private and public key. Normally they send this private key again to the RA but in this case, it gives direct access to the user.

We must consider that the CA generate both the certified private and public keys an while the private key is sent to the user, the public one is sent to the Verification Authority for publication. It is one of the most important pieces of this system since it confirms whether or not that the data submitted by the user is legit. How is this done? Because since the user has already the private key, he will send the signed data with this to a receptor and this one will send the sign in question to the VA to see if its verified. Since the VA has access to the public key only, the sign made from the private key is the only thing that can match the files. The VA sees that it makes sense and proceeds to tell the receptor that the sign is valid from the validation standpoint. With this info, the receptor is able to accept any data sent with that sign from the original user.

2. Discuss the advantages and challenges of implementing a PKI system.

A PKI system its actually a very reliable way to validate the data sent over in a communication line. Why? Because you are backed up by three different entities that watch each step of the process. Nothing happens if any of these entities doesn't allow or permit further advance. It gives validation to the receiver that the sign made by the user has been certified and works as intended. In terms of security its possible to say we are in good hands because of the fact that the data is encrypted and only by the use of the private and public key can we hope to achieve a level of secrecy when we pass information. Looking more into this, the keys provide an authenticity layer to the whole process between user and receiver since only certified signatures or keys can permit any action.

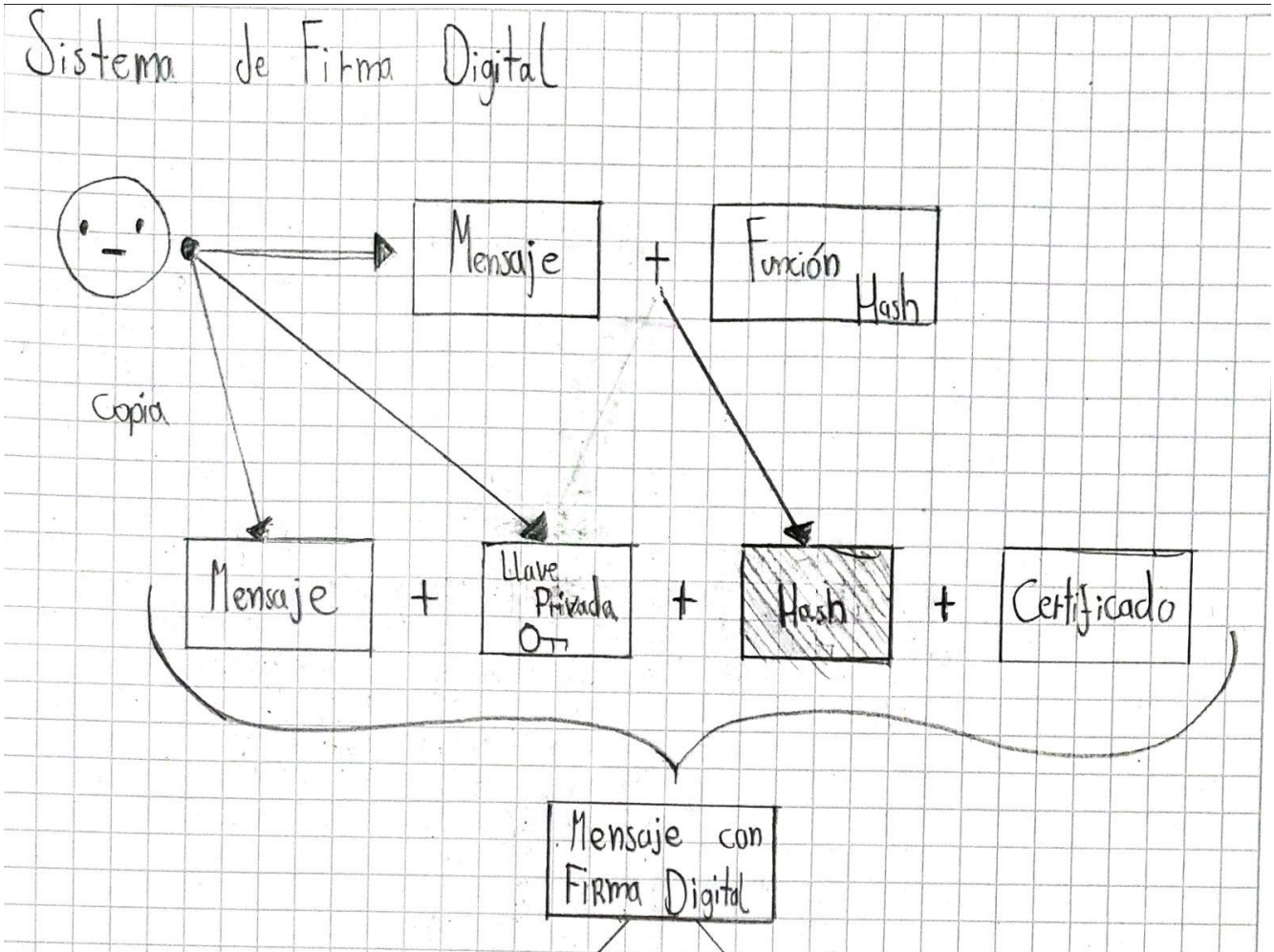
However, it also presents some challenges for an efficient use of it. The implemented system may seem overcomplicated by the fact that three authorities have to step in and that it may be too complicated to fully use. Aside from this, the cost of this type of systems does not come cheap since you have to consider what software to use, which hardware fulfills the tasks and with taking into account the human element. One of the most worrisome aspects is the keys since its susceptible to human errors. The way they are generated could mean a problem if one of the parties loses access to one of them since you would need to ask for another, erase or invalidate the previous one and its not so easy as you may think.

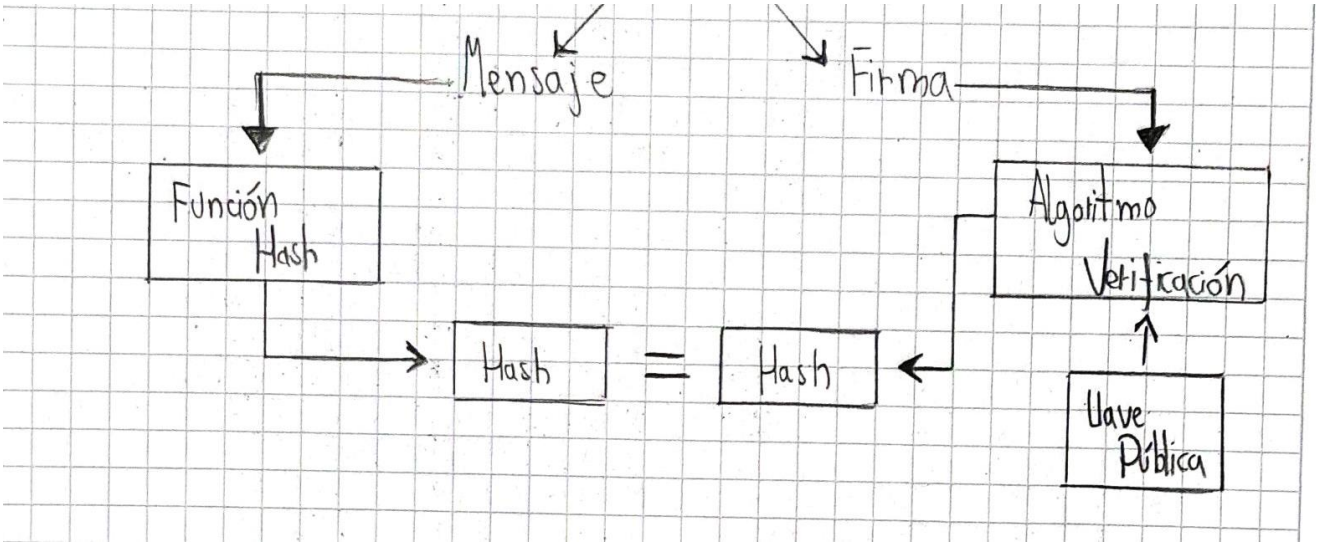
3. **Provide an example of a real-world application where a PKI system is used.**

One simple yet crucial example is any affair that involves money. This specially applies to any banking transaction made between two parties. The bank is the only one that can guarantee that it is you that acceded the system by your own private keys and this can also work by the user side since you can check if the banking site in question is authentic and reliable. They send in your private data for the transaction in a way that its really difficult to intercept it and overall means that any interaction between you and the banking site is secured and can't be revealed to anyone outside the communication. In the terms of transactions, it makes sure that both parties take responsibility for their actions during this process since it literally uses a signature made by the private key to entitle any action to their respective owners. It only made sense that a PKI system would be used by any banks since it's a secure way to interact with the clients and gives them a feeling of safety and security by the use of complex algorithms and tricks that allow all the involved parties to know each other´s authenticity.

Exercise 6:

Design a system for digital signatures based on public-key cryptography. Explain the steps involved in the process and the role of each component.





The previous diagram represents a digital signature system based upon the use of private and public keys. In a way its quite simple but hopes to encapsule the important elements needed to use a Digital Signature. The main responsible here is the User or client since he is the one that creates the message in the first place. He must create a copy that will pass through a hashing function that will provide him with a hash code. The process to create a digital signature is when you use your private key along with the hash code of the message and a certificate that holds who you are, where do you work and basically that the one sending the message is a valid person. This digital signature is annexed along with the original message to certified that this one has validity and the one that sends its who he says he is.

Once this process is done, the user send its message with the digital signature to the one person supposed to received it. The message is divided into the original message and the digital signature. The message is passed forward to a hash function in order to get a hash code for later verification. The Signature is passed forward to a verification algorithm which implements the use of a public key that can be used to generate another hash code. The last thing to do is to use both hash codes and start comparison. If both hashes match each other then it is safe to assume that the message or document is valid an that the signature belongs to the one that send the file.

References:

1. Autor DocuSign Contributor, D. C. (2020, October 30). *¿Qué es la infraestructura de clave pública o pki cuál es su relación con la firma electrónica?*. DocuSign. https://www.docusign.com/es-mx/blog/pki
2. Bello, E. (2022, October 20). *Conoce Las Herramientas de Ciberseguridad para proteger Tu Empresa*. Thinking for Innovation. https://www.iebschool.com/blog/herramientas-ciberseguridad-digital-business/
3. Cybersecurity. (2023, March 21). *What are the benefits and drawbacks of using hash functions for authentication?*. Hash Functions for Authentication: Benefits and Drawbacks. https://www.linkedin.com/advice/1/what-benefits-drawbacks-using-hash-functions-authentication
4. *Digital Signatures*. Digital Signatures - Practical Cryptography for Developers. (n.d.). https://cryptobook.nakov.com/digital-signatures
5. Miya, A. (2023, April 25). *Message authentication code (MAC) in cryptography - usemynotes*. Use My Notes. https://usemynotes.com/message-authentication-code-mac/
6. *Private Key Infrastructure Advantages and disadvantages*. Bartleby. (n.d.). https://www.bartleby.com/essay/Private-Key-Infrastructure-Advantages-And-Disadvantages-FJ63ZUM3G
7. *Public key infrastructure - secret double octopus*. Secret Double Octopus -. (2021, August 15). https://doubleoctopus.com/security-wiki/digital-certificates/public-key-infrastructure/
8. Sheldon, R. (2023, August 11). *What is message authentication code (MAC)?: Definition from TechTarget*. Security. https://www.techtarget.com/searchsecurity/definition/message-authentication-code-MAC