INF140 Mandatory Assignment 1:

Question 1: Explain the meanings of the following security attributes in computer systems: confidentiality, integrity, authenticity, accountability, availability.

Confidentiality

Is the idea that users does not get control over any part of a system or a profile in which they can exploit information. Everything online should always be confidential and accessible to only those whom the persons in charge decides. In computer security we can divide this attribute in two categories, data confidentiality and privacy.

Data confidentiality: Makes sure that unauthorized users does not get access to confidential or private information in any way.

Privacy: Makes sure that the individual has full control over which information related to them should be available for others. Both when it comes to collection and storing and when it comes to who the information may be revealed and disclosed too.

Integrity

The meaning of integrity in terms of computer security is that the systems keeps it's structure. That it is whole and not adjusted or modified or destroyed in any malicious way such that the system does not work. This also applies to messages and other parts of computer security. The integrity of a system is supposed to keep it whole and correct. And just as confidentiality we also have two types of integrity, Data integrity and System integrity.

Data integrity: Makes sure that programs and information is only altered by authorized users, and only in a specified way.
System integrity: That a system works as it should and is not changed in any way by users that are unauthorized to the system.

Authenticity

Is an attribute that makes sure every user that has access to a system is able to be verified and trusted. And also that the validation process is secure and credible. A secure system is almost always reliable on the people that have access to the system. So in a way, authenticity may be one of the most important attributes in computer security.

Accountability

A computer system today will never be 100% secure, because there is always be a threat to the system. And because of that we need to have some sort of way to trace the breach to a system to the adversary or the people responsible for the breach in the system. There is always someone that can be held accountable for a breach in a system. Either it is from an inside man or is a breach from the outside we need some way to trace the breaches back to whoever got access, so that not only can they be held accountable, but also so that the system can be more secure in later stages.

Availability

Is an attribute that makes sure that each user that has access to the system and Is authorized has proper access to the services and that it works as it should.

Question 2: List and briefly define the kinds of threat consequences and the types of threat action that cause these consequences.

Unauthorized Disclosure

Defininition: Someone or something causes something to happen in where they get access to data which it is normally not authorized to access.

Threat action:

Exposure is when sensitive data is leaked, or "exposed" in a manor which can include both inside leaks, or when an error occurs so that information is leaked to an unauthorized source.

Interception is a type of attack in which a hacker can "intercept" or get access to data which he is unauthorized to access, mostly through the internet. Another great example is through Local area network in which any device can get access to information intended for other devices.

Inference is a type of attack in which data is being analyzed. In this case the attacker is searching for patterns it can use to gain information. A great example is analyzing password databases to figure out hashes or salt values etc.

Intrusion is simply an adversary overcomes the security systems and gains access to data he is normally not authorized to access.

Deception

Definition: Something happens in which case the system, user etc. is tricked into receiving proper information or data, but instead is given false information, in which case the user, system etc. believes the information to be true.

Threat action:

Masquerade is as the word suggest when a user is posing as somebody else to gain access to something he normally is not authorized to access. A great example of masquerade is the trojan horse which appears to work as it should, but in reality is gaining access to systems, databases etc.

Falsification is when you access a system to falsify and/or change data to something it should not be. A great example is to falsify your grade at school by getting unauthorized access to the school system.

Repudiation is a threat which is simply another way to say you are lying to the system about sending data, possessing data etc.

Disruption

Definition: a threat to systems and services in which case they won't work as they should. Examples could be that the system is significantly slower or does not even work as it should. Which could be

very harmful if done at bigger systems or services.
Threat action:

Incapacitation is an attack on the availability of a system. In which they target the system itself and try to damage or in some cases completely destroy the systems functions and services. This is mostly done through the use of viruses, worms, trojan horses. Done using methods which can easily get access to multiple part of a system.

Corruption is an attack on the integrity of a system. In which case they try to change the system itself to gain access to parts which the system does not intend to access.

Obstruction is a type of attack which targets the systems communication systems. This can be done by altering or even disabling the different communication networks which links them.

Usurpation

Definition: This may be the most dangerous of them all as in this circumstance, as an outside source, or an unauthorized user gets control over a system or service. Which may lead to leaked databases, changes in structures of systems. Company information leaks and etc. Almost all of the above can be done when getting control over a system.

Threat action:

Misappropriation is a type of attack intended to get full control of a system. This can be done in various different, but hard manors. And often needs someone to have installed malicious malware on multiple different platforms to be able to launch an attack at a host. In which case they try to get unauthorized access to process power or operating systems.

Misuse is a type of attack that includes getting full control of a system in some way. This can be through a hacker getting past security systems and getting unauthorized access to part of / or a system. In which they can do a lot of harm to the system or even disable it completely.

Question 3: . Refer to Figure 1.2 Security Concepts and Relationships in Section 1.1 in Chapter 1. Think of an example that involves all blocks in the figure and go through the relations of entities and relevant techniques in your example. Draw a similar figure that consists of concrete entities, techniques, countermeasures.

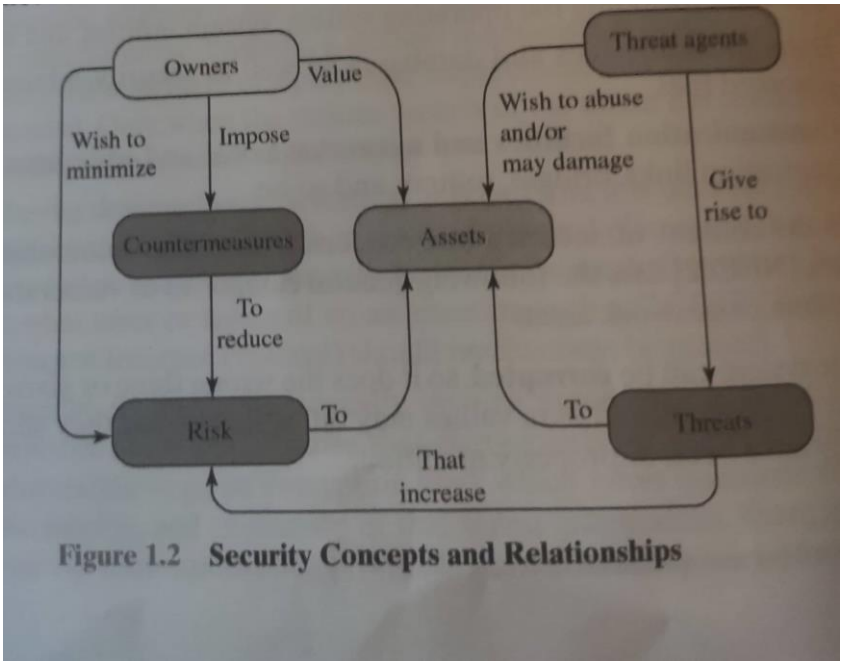


Figure 1.2 **Security Concepts and Relationships**

Scenario: Owner values a password which the threat agent, an adversary wishes to get hold of. So the adversary launches an attack which threatens and increase the risk to the password. In which case the owner would like to come with his own countermeasure to stop this.

The owner does not want any threats to his password. So he puts the password in a database along with other password. Then again that does not mean it is safe. So he hashes the passwords in the database. The threat agent is aware of these counter measures and has a mean to get hold of it anyway. Firstly he uses the threat intrusion so that he can get pass the systems protections and securities, and then performs an inference attack so that he can analyze the system for similarities. He abuses the weak passwords and finds out the hash so that he may figure out the correct password. Unlucky for the adverasary, the owner keeps check on his system and is always mentaining it making it harder for hackers to get the passwords. As it took too long time for the hacker the owner made sure to maintain his database. Upgrading it with new salt values to each password and of course new hash values. Which would make it even harder for the adverasary to get

the password the second time around.

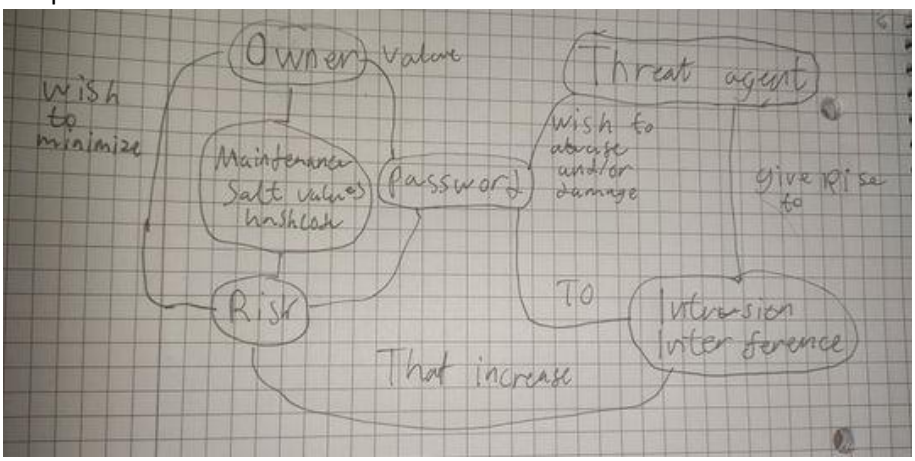


Figure 1 Question 3

Question 4:

Starting with the use of Vigenere cipher and then using Transposition and repeating it for round two, after that we do the opposite. We decrypt the message using our keys using the same ciphers.

Key1 = JISUAN

Key2 = 3415726

Message M = Cybersecurity is actually determined by the weakest link

When using a vigenere cipher the idea is that we exchange the letter we have with a letter written in an alphabetic coordinated system. Where the row is determined by the letter from our key, and the column is determined by the letter from our plaintext. Then we use it as a coordinate system, and find the letter which is in the same column as our key letter and same row as our plaintext letter, and then that is our letter for the ciphertext. We do this with all our letters from the plaintext. We do this by removing all spaces and etc. and by repeating our key for every letter there is in our plaintext. I did try to circle around the letters I changed when getting the encrypted message, and put numbers on the columns and rows for each letter I changed, but as there is multiple of the same letter It does get very crowded. So I stopped doing that after the 8th letter. However I continued changing the letters until i got the ciphermessages. Also using "*Practicaly Cryptography, Vigenere gronsfeld and autokey*" for doublechecking and going through it properly.

Doing decryption on this cipher is just as simple as the encryption. You simply do it backwards. Knowing the placement of the cipher letter and the key, you use the column as the key as normal and just find the cipher letter in the column then see which letter is in the row where you found the cipher letter. Very simple really, at least as long as you know the key.

Then we have the Columnar transposition cipher where you put your message into a table, with the length of your key and the as many rows as there are letters in your message. When we have put our message in the table we simply use the key and swap the order of the columns either numerically or alphabetical. Then we read it column for column and put the letters into our new ciphertext. Used

"*Practical Cryptography, Columnar transposition Cipher*" for help.

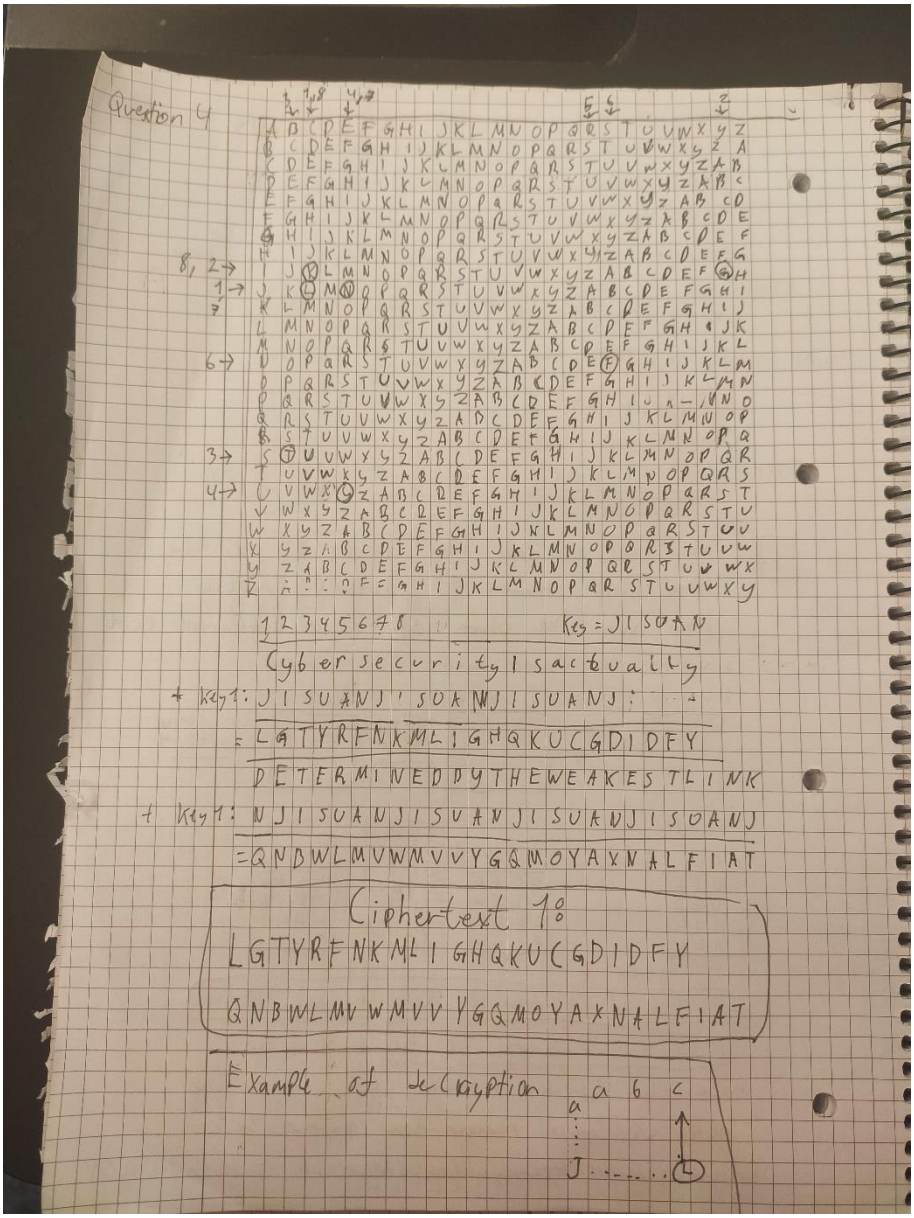


Figure 2 Question 4.1

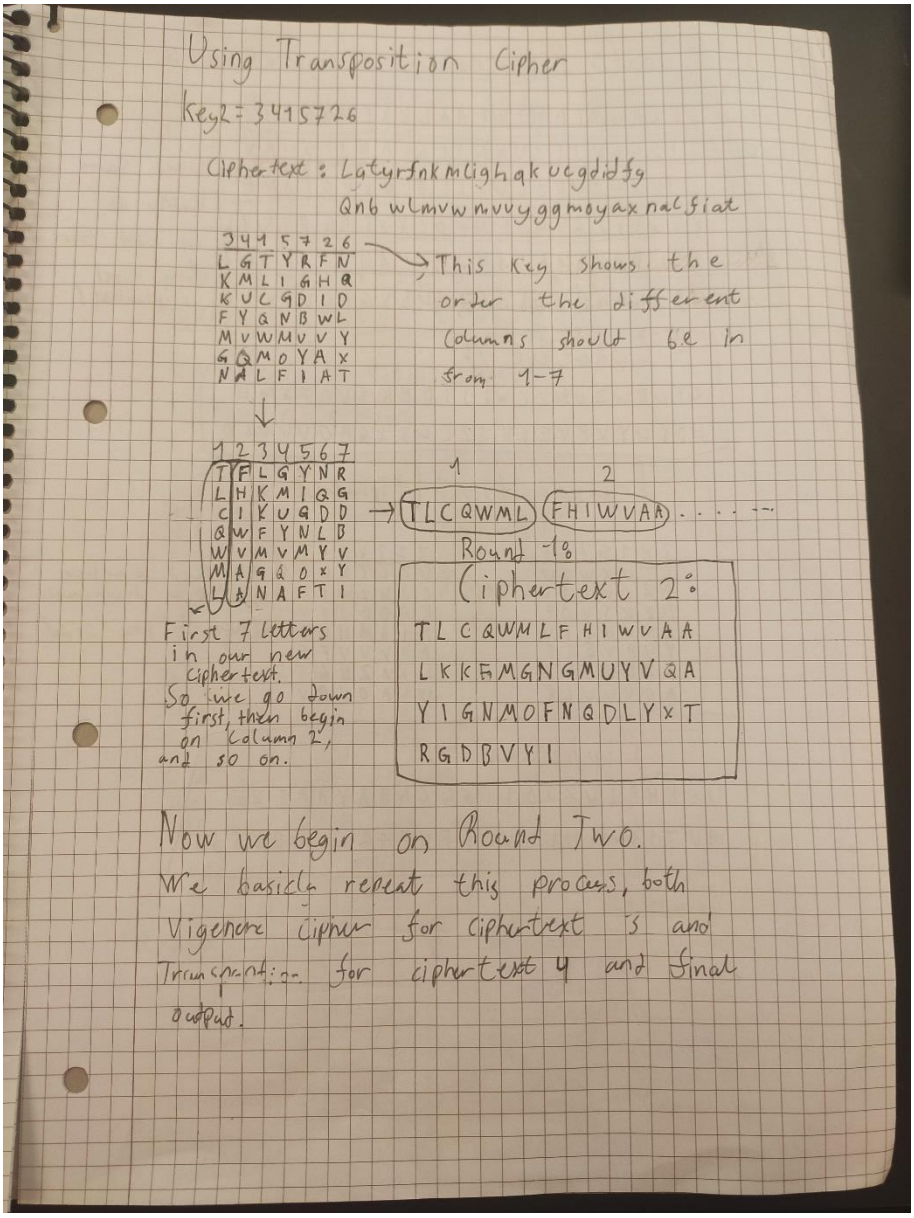


Figure 3 Question 4.1

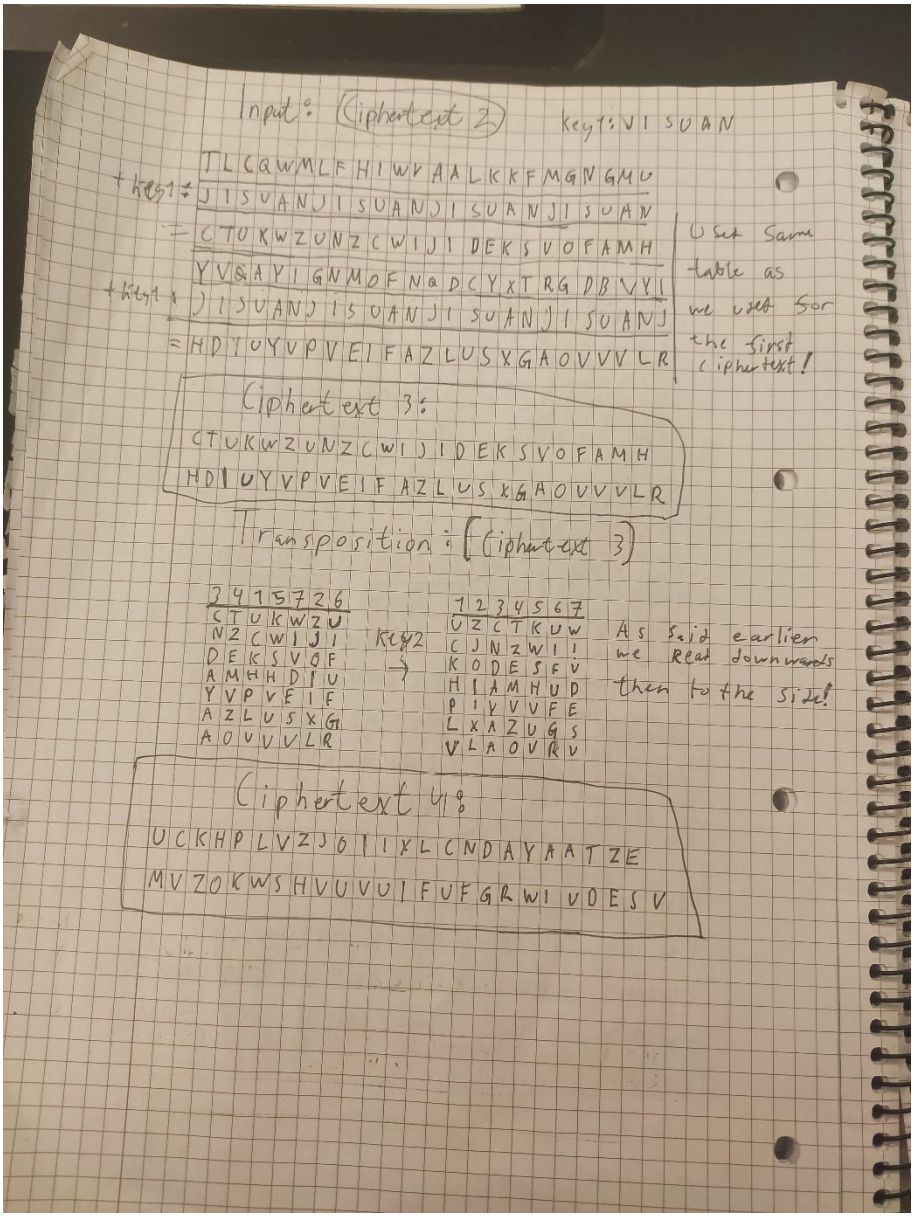


Figure 4 Question 4.2

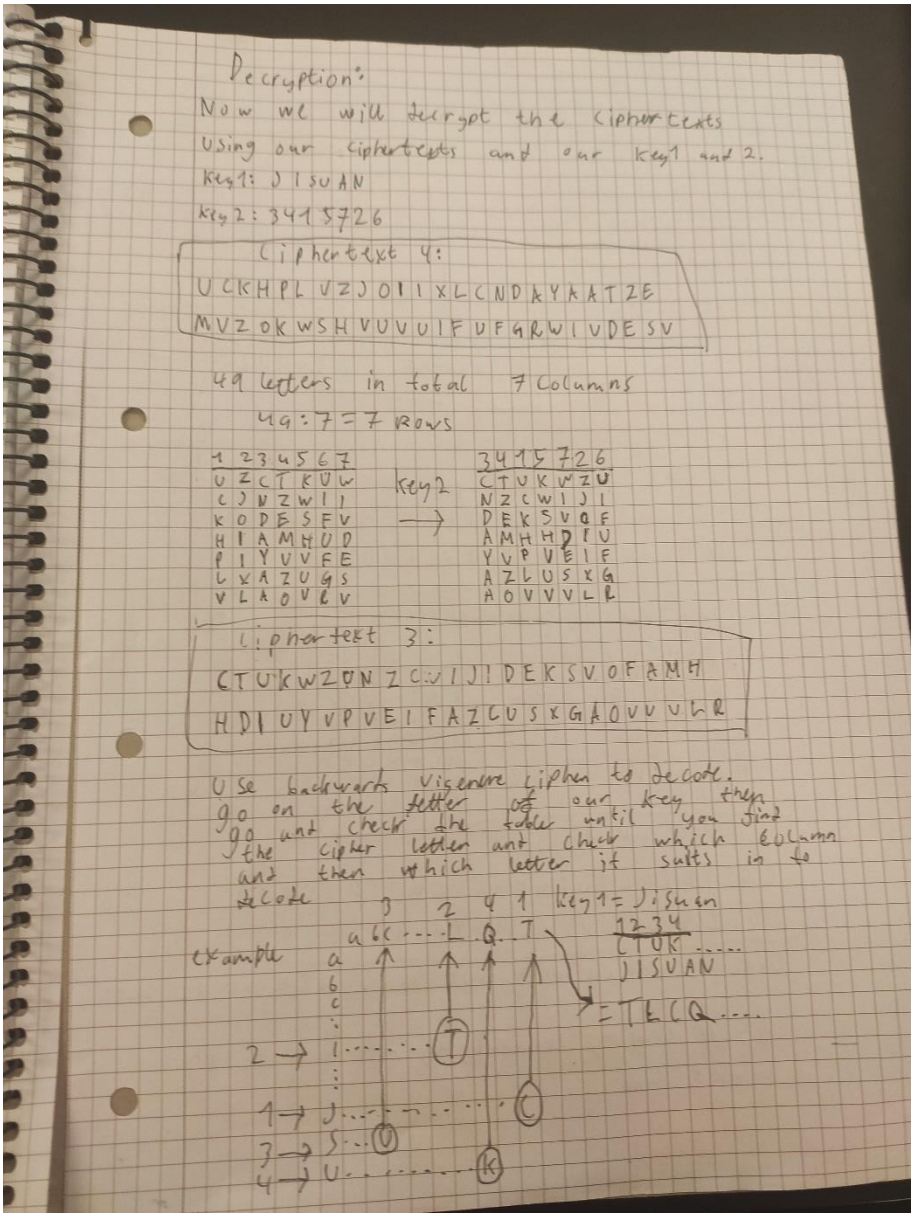
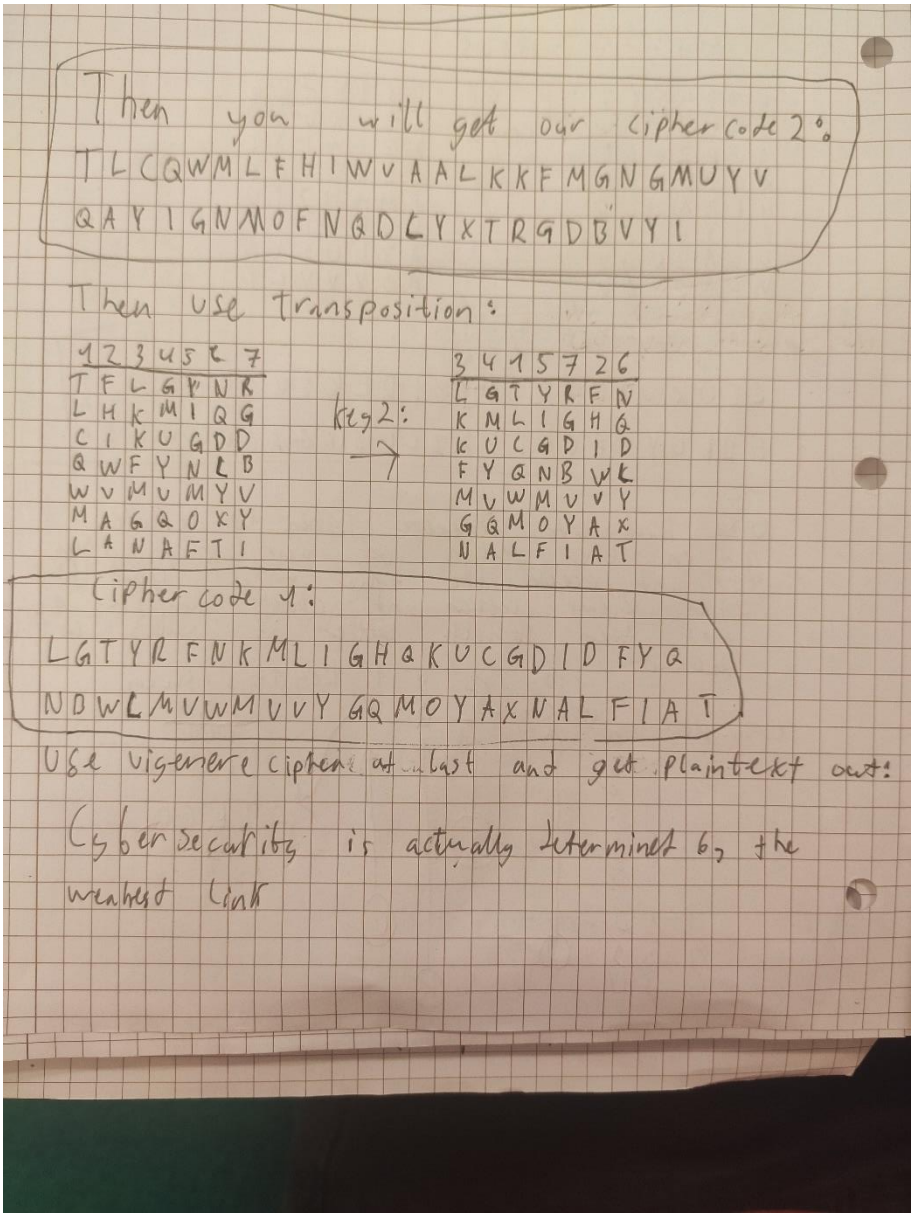


Figure 5 Question 4.3


Then you will get our cipher code 2:
TLCQWMLFHIWVAALKKFMGNGMUYV
QAYIGNMOFNQDLYXTRGDBVYI
Then use transposition:
1 2 3 4 5 6 7
T F L G Y N K
L H K M I Q G
C I K U G D D
Q W F Y N L B
W V M U M Y V
M A G Q O X Y
L A N A F T I
Key2:
3 4 1 5 7 2 6
L G T Y R F N
K M L I G H Q
K U C G D I D
F Y Q N B W L
M V W M U V Y
G Q M O Y A X
N A L F I A T
Cipher code 1:
LGTYRFNKMLIGHQKUCGDIDFYQ
NDWLMVWMUVYGQMOYAXNALFIAT
Use vigenere cipher at last and get plaintext out:
Cyber security is actually determined by the
weakest link

Question 5:

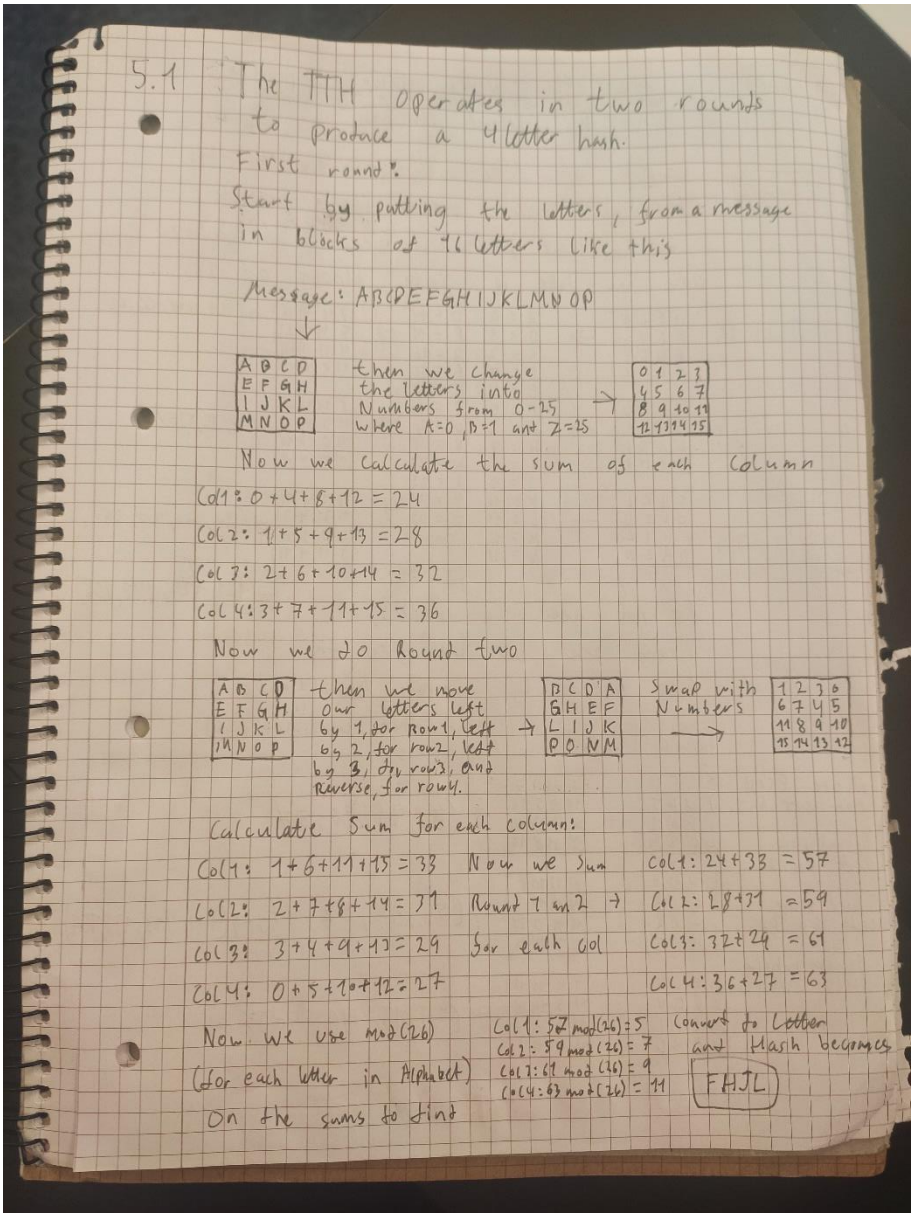


Figure 6 Question 5.1

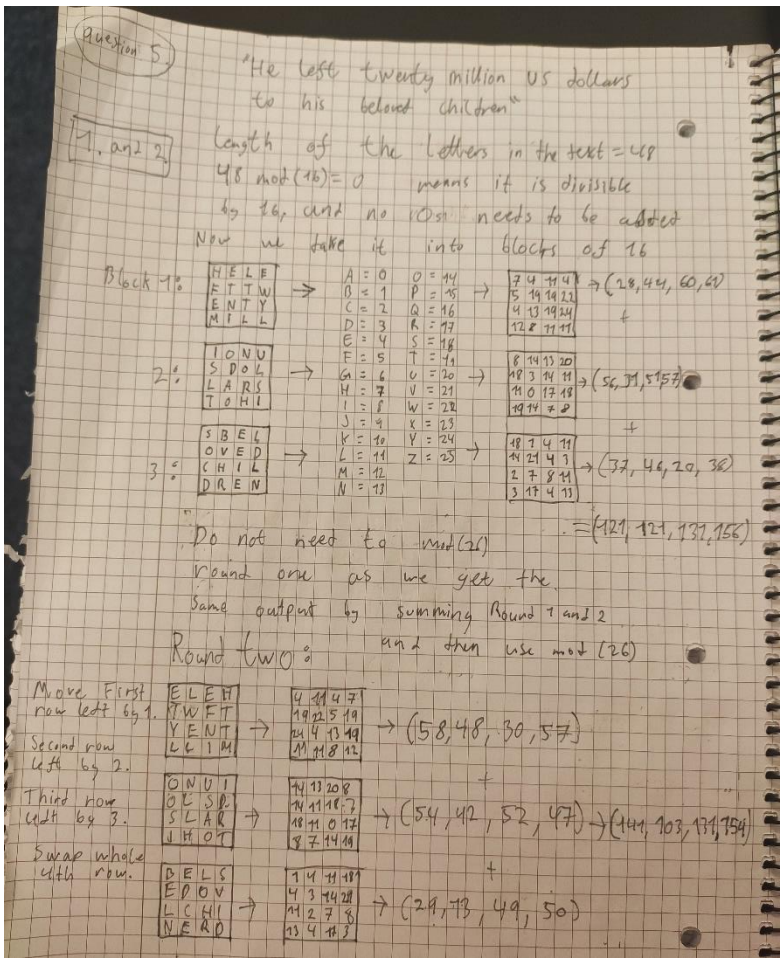


Figure 7 Question 5.2

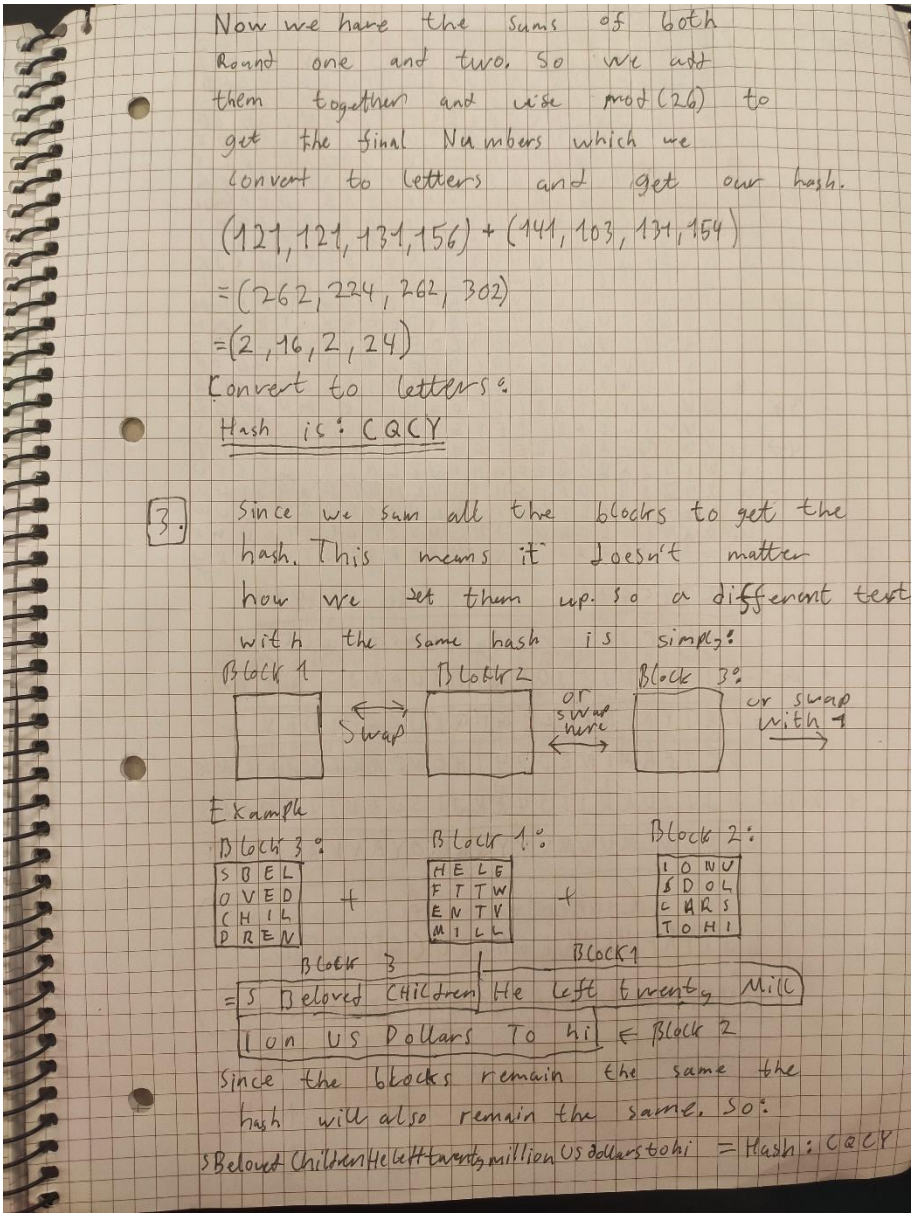


Figure 8 Question 5.3

Question 6:

Part one I did in python, while part two i did on paper as it got too hard to do it on python for me at this stage.

Figure 9 Question 6.1.1

Figure 10 Question 6.1.2

Figure 11 Question 6.1.3

Question 6.2

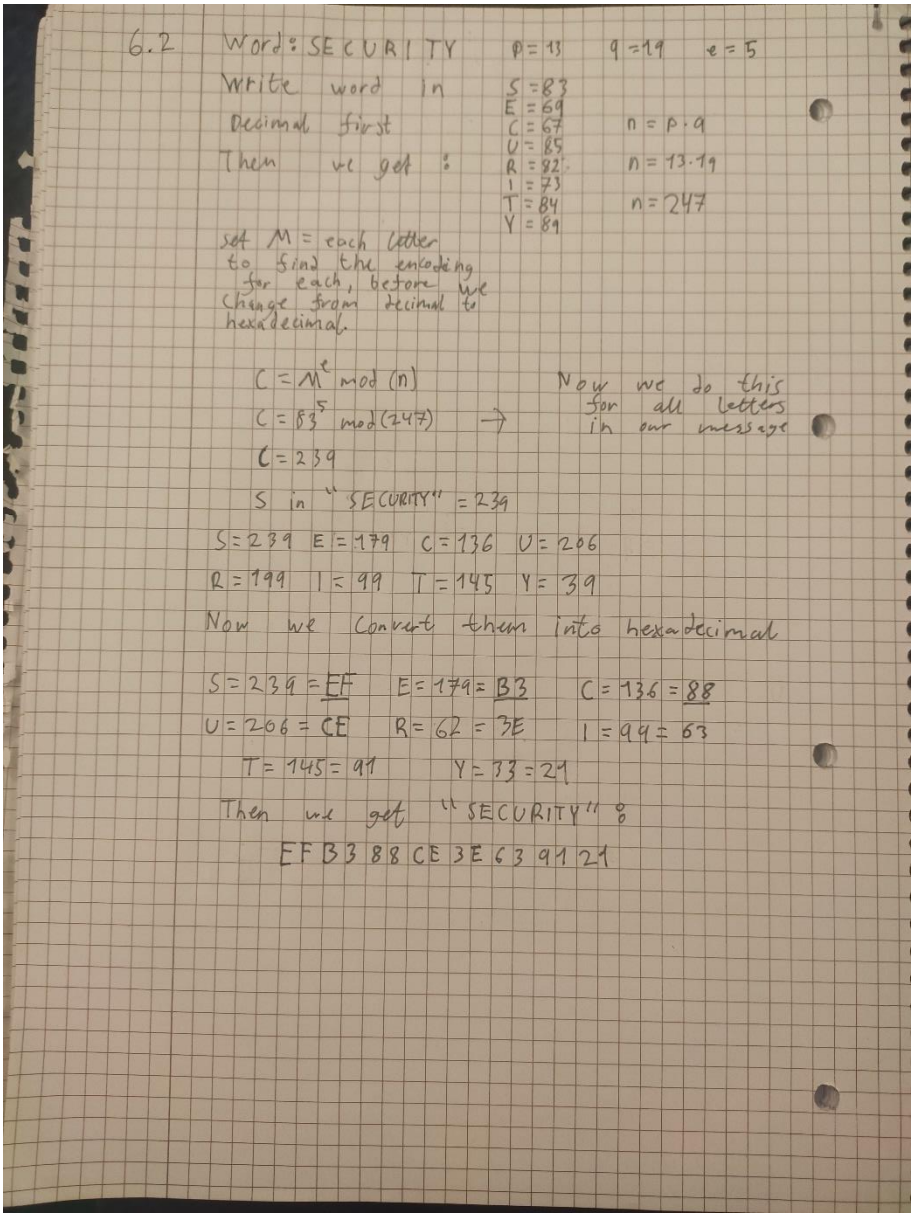


Figure 12 Question 6.2

Question 7:

Used python to find the key in numbers:

Figure 13 Question 10

And as we know from the text in the question 0 = A, 1 = B … So therefore we know that the key in numbers = 103 → is equal to the word BAD. So the vigenere key bob used is the word: BAD

The sentence, I found using *Practicaly Cryptography, Vigenere gronsfeld and autokey* to decrypt the vigenere cipher, using the keyword bad, which we got from our RSA decryption method. And the final message turned out to be:

> *The five most efficient cyber defenders are anticipation, education, detection, reaction and resilience do remember cybersecurity is much more than an it topic.*

Question 8:

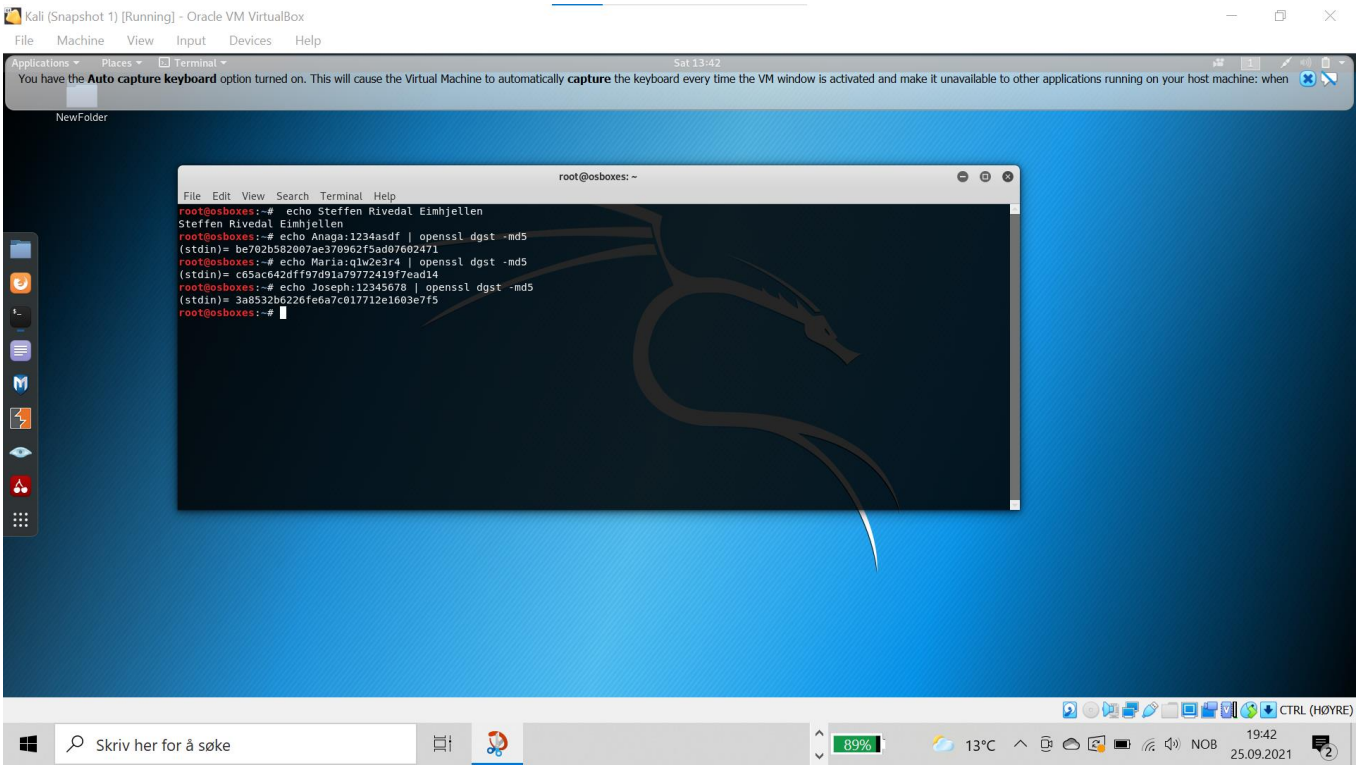


Figure 14 Question 8.1

Question 8.2

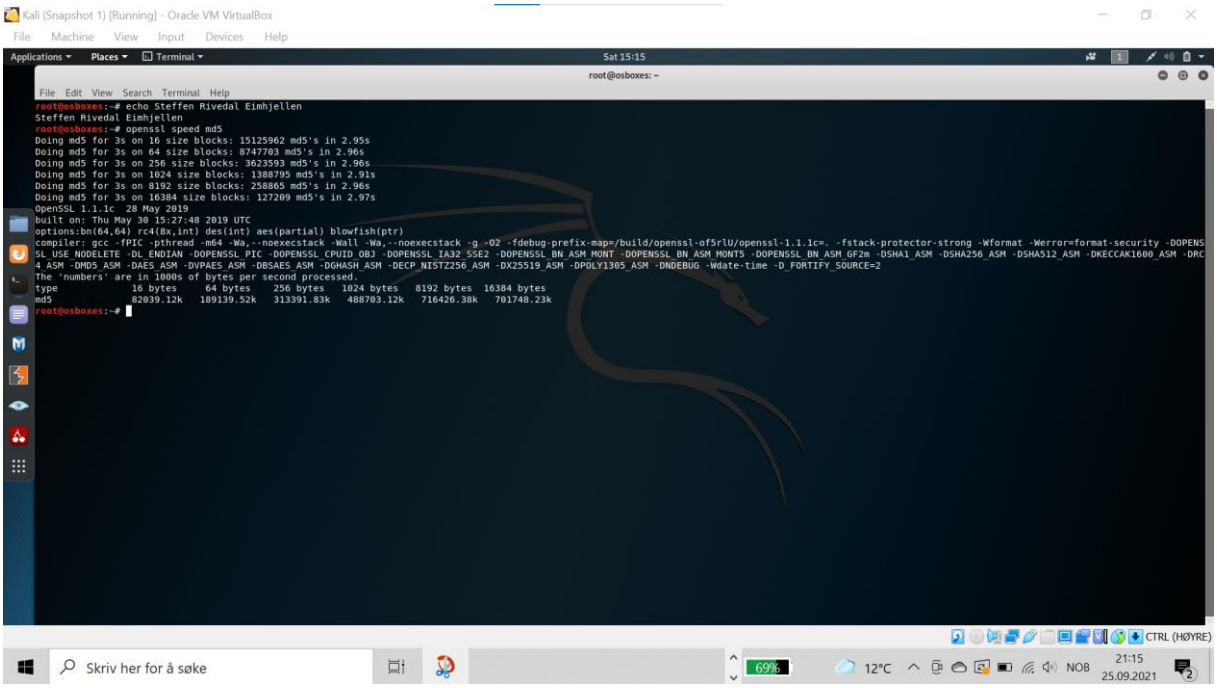


Figure 15 Question 8.2

As we can see here, we have the different speeds for the different size blocks using md5. To continue and figure out how to calculate the speed of Nikolay's password we have to use the 16 size blocks.

This gives us the speed of 15125962 md5's in 2.95 seconds. Now we use this to calculate the speed it takes for my pc to crack Nikolay's password. (I will calculate in estimates and not use every single decimal in the number etc. as it won't have much say on the final output anyway.)

First, we divide the initial speed on 2.95 seconds, so we get the speed pr second. Which is

$\frac{1.5*10^7 md5's}{2.95\ seconds} = 5 * 10\text{^}6$ md5's pr seconds

Then we use this number to find out how long it takes to crack Nikolay's password in the different scenarios.

• Nikolay chooses a password of length 4 with all digits:
Since we have 4 digits this means we need 10^4 amount of tries. This is because 10 is the number of digits we must guess (0-9), and 4 is the length of password.
Now we calculate the time it takes for us to crack the password. Then we use the amount of tries we at most need and divide it by the md5's pr second. To find the time it takes to crack a 4 long password of digits.

$$\frac{10^4}{5 * 10^6} = 1.9 * 10^{-3}\ or\ 0.0019\ seconds$$

So, it takes 0.0019 seconds to crack a password with my pc, when it is 8 characters long and comprised of digits.

• Nikolay chooses a password of length 8 with all digits:
Since we have 8 digits this means we need 10^8 amount of tries. Same as before because 10 is the number of digits we must guess (0-9), and this time 8 is the length so therefore It is ^8.
Now we calculate the time it takes for us to crack the password. Then we use the amount of tries we at most need and divide it by the md5's pr second. To find the time it takes to crack a 8 long password of digits.

$$\frac{10^8}{5 * 10^6} = 19\ seconds$$

So, it takes 19 seconds to crack a password with my pc, when it is 8 characters long and comprised of digits.

• Nikolay chooses a password of length 8, where each position either a digit or a lower-case letter:
In this case we use both digits and letters, and password of length 8 so this means we need 36^8 amount of tries. This is because we have 26 letters in the alphabet and 10 digits to use therefore, we get 36, and ^8 is as stated earlier because of password length.
Now we calculate the time it takes for us to crack the password. Then we use the amount of tries we at most need and divide it by the md5's pr second. To find the time it takes to crack a 8 long password of letters and digits.

$$\frac{36^8}{5 * 10^6} = 554\,818\ seconds\ or\ 154\ hours$$

So, it takes 154 hours to crack a password with my pc, when it is 8 characters long and comprised of digits and letters.

Question 9: Why is it considered that the additional salt in the UNIX password scheme increases security?

The addition of salt values to password security, along with hashed passwords, is a great way to hinder attackers. Although the salt value doesn't need to be guessed as a plaintext copy of the salt is stored alongside the hashed password, the use of salt values really makes a huge difference in password security. There are three main purposes the salt value serves; The first is to help prevent duplicated passwords from being found as it hinders user ids with the same password of being found. Because although the password might be similar, creating in that case a similar hash, the salt value will always be different so therefore getting access to a password will never hurt someone else's.

The second reason is because it makes a huge impact on brute force attacks. As the salt value will increase the difficulty of guessing a password by the factor of 4096. Which will greatly increase the password security. The last reason, similarly, to the first, it makes it almost impossible to know whether a person is using the same password for multiple systems. Since the hashed password in the different id files will not be the same, as mentioned previously. Hence why the use of salt values makes huge difference in the security of passwords.

Question 10: The following is an entry in the password file in a Linux system, which a root user can access : root:6Q8uKtWWm/dptau2a$E184j/HJuiuw2lsUT7yuBvTh3FioWj5KKUvPQT /1OJT4rtBACAm4NlEFV4n4x6ndTN3wD9A5uHOjEQQ/JJqN./:18142:0:99999:7:::

This entry is what we call /etc/shadow file, a format used on Linux. This file is used to store hashed passwords for Linux user's accounts. Hence why a root user can access this and not every user. For most people this might not look like it is very important, after all it is just a bunch of numbers, letters and special symbols. However, knowing about the etc/shadow file is quite important for people who needs to access it later. Such as system developers and admins, as they might need to know a thing or two to debug eventual user issues.

The most basic thing to understand in this file is that every field is separated by a (:) colon, and is frequently used in the shadow file. Now that we have explained the colons it is time to go through each field in the file. The first field is the username, which pretty much is self-explanatory, the name you use to login in with, in this case root. Field number two however, can seem very difficult to understand. Luckily, it's not, as it's just the encrypted password. Usually comprised of a id, salt value and the hashed password, only split by a $. First a $, then the user id then $, then a salt value then $ and lastly the hashed password. A noteworthy part is that the $id will differ for what type of algorithm is used. If it's MD5 it'll be 1, if it is 5 it'll be a sha256 algorithm and etc.

Now the next field is quite an interesting one, as it days since jan 1. 1970 that the password was last changed. The reasoning behind the use of 1970, is that it is recognized as the start of unix epoch time. The 4th field is the minimum days that has to go before a user can change password, which also changes everyday meaning it counts down until you are allowed to change, and is also connected with field 5. Which is the opposite, the maximum number of days before you must change password. The sixth field is the number of days before you are warned that the password is in need of change before it expiers. In this case the warning happens 7 days before password change. The 7th field is the number of days after a password has expired that the system disables the account. And lastly the 8th field stands for days since jan1. 1970 that the account is disabled and is no longer in use.

As you can see the last 2 does not have any number. As the root file is not supposed to be disabled, and for the most part other accounts will not have this either. This is mostly used on bigger connected systems, and only changeable by admins and roots. Also including field the 4 previous fields, which is changeable by admins. Now since this is only a root account it will of course not have been set timers on and therefore will not have any numbers written and that's why it only shows a new colon.

In addition, I gotta admit I took heavy inspiration in the writing from Cyberciti, Understanding etcshadow file, as it was quite hard to explain it in any other way, especially for the last 4 fields. However, the other fields I did use my own words and try to explain it as good as I could.

Sources:

Cyberciti, Understanding etcshadow file, information collected 26.09.21 URL:

https://www.cyberciti.biz/faq/understanding-etcshadow-file/

Practical Cryptography, Columnar transposition Cipher, information collected 19.09.21 URL:

http://practicalcryptography.com/ciphers/columnar-transposition-cipher/

Practicaly Cryptography, Vigenere gronsfeld and autokey, information collected 19.09.21 URL:

http://practicalcryptography.com/ciphers/classical-era/vigenere-gronsfeld-and-autokey/

Computer Security Principles and Practice 4th edition, by William Stalling and Lawrie Brown, Published by Pearson Education in 2018, information collected 17.09.21.