All four assignments
Subject code:*DCS3100*
Subject name: *Introduction to Data and Cyber-Security*

Candidate no. 222

Monday 6 January, 2020

Contents

List of Figures

Listings

Part I

Assignment 1

0.1 Concepts illustration of CIA/AIC

What is CIA? CIA stands for Confidentiality Integrity and Authentication/ Availability and this model is a guide for policies information and these three elements are the most crucial components in Security. The CIA can be imagined in as a triangle. Confidentiality are a method and its designed to prevent the information from the reaching the wrong people and making sure that the right person can get it. The integrity is to have the ability to ensure that data is correct and it is not altered from the original sources. The information's such as concerned must be readily and accessible for the user all the times.

The following steps will show the illustration concepts of the CIA between Bob and Alice.

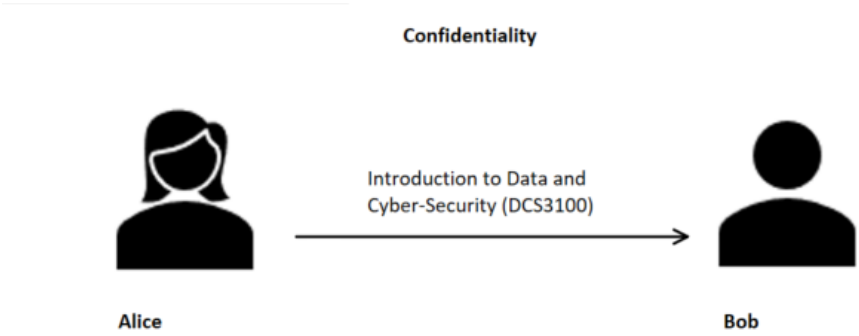


Figure 1: Confidentiality

Lets say Alice want to send a message to Bob, to her friend. The message should only be able to read by Alice and Bob. If a third person view their messages and they shouldn't exchange messages because the information they sharing it can leak and gives a serious consequence for both.

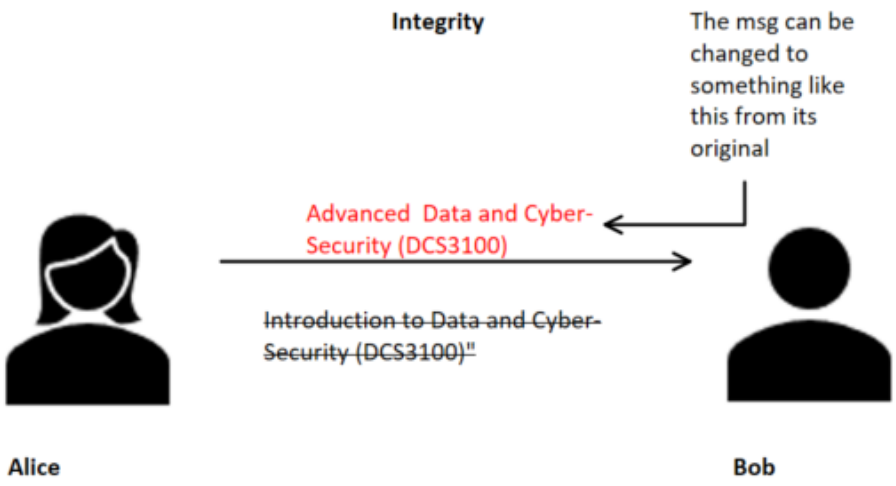


Figure 2: Integrity

Since Bob is receiver and he must be able to verify that the message content is accurate and unchanged. The message content can be modified accidentally or on purpose by a third person.

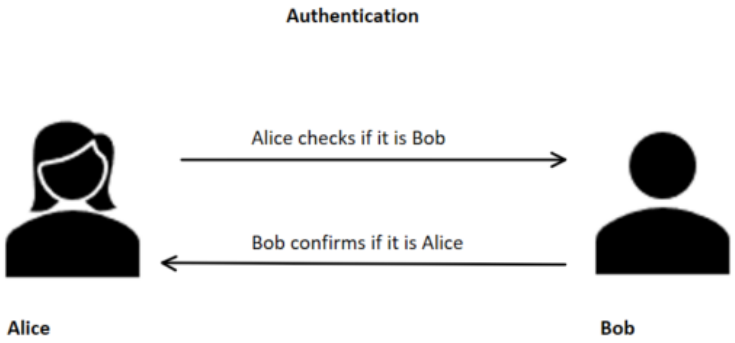


Figure 3: Authentication/ Availability

Alice and Bob should be able to confirm identity of the other party. Alice checks first the identity of the receiver then receiver confirms the sender's identity. If they have any suspect to confirm identity of the other party they have to avoid sending message to each other and figure out another solution.

0.2 Vigenere Cipher

Vigenere cipher is similar to Caesar crypto-system, but in Vigenere we are using several keys instead of just single key. the Vigenere cipher is a form of poly-alphabetic substitution method and this was constructed in the 16th century. This crypto method uses a given word as the private key and the letters in the key define how many character to shift the actual letter in the plain text.

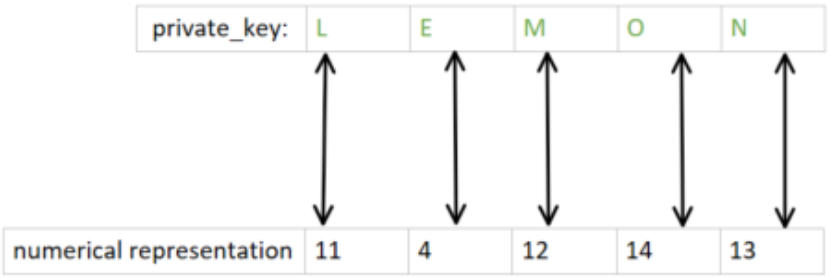


Figure 4: numerical representation

To encrypt Vigenere Cipher we need to use this mathematical formula and it's approximately the same formula as we using for Caesar.

$$C_i(m_i) = (m_i + K_i) mod 26$$

$C_i(m_i)$ is the encrypted character of the cipher text.
m_i is the character of the plain text.
In Vigenere we have to use the `i-th` character of the key for encrypting the `i-th` character.
mod 26 is the length of the English alphabet.
To Decrypt the cipher text to plain text we have to use this formula.

$$D_i(m_i) = (m_i - K_i) mod 26$$

$D_i(m_i)$ is the decrypted character in the cipher text.
To transfer the plain text into the cipher text we use the mathematical formula and using the character in private key in order to transform the letters

in plain text. The first character in plain text will be transformed with the help of the first character if the private key and second letter in the plain text will be transformed with help of the second character in the private key and so on. When all the character in private key are used we start over and do the same operations. After that we can just use Caesar encryption to encrypt the message.

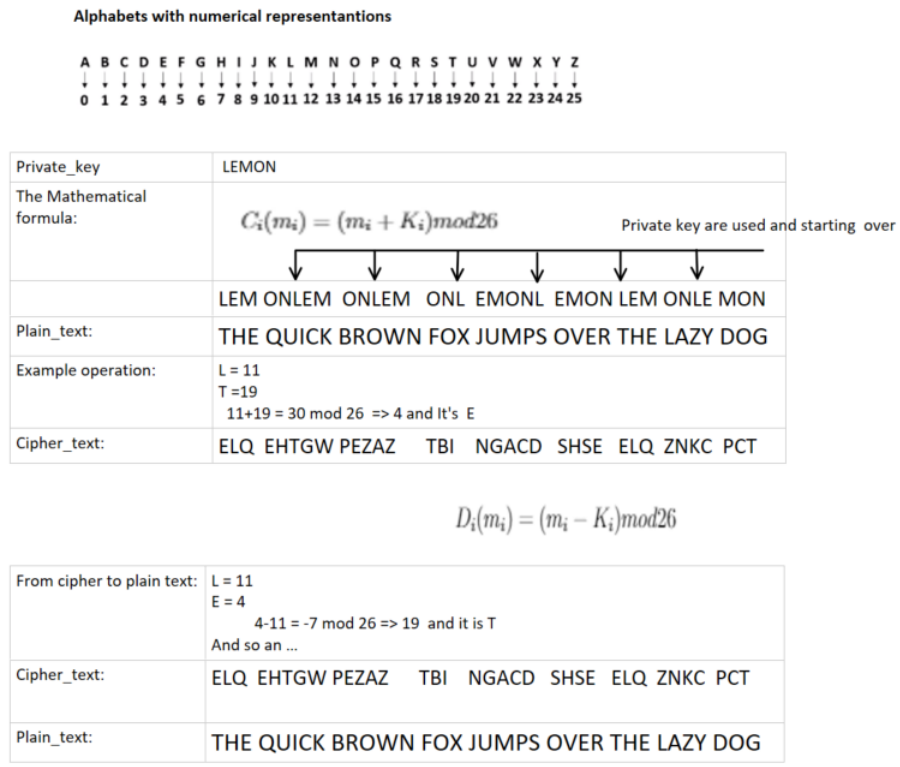


Figure 5: Single key operation

We basically shifting the letter T with as many characters to right as the numerical representation of L and shifting the second character of the plain text with as many letter to right as the numerical representation of second letter in private key etc.

0.3 Single Key Code

```
#alfa = ' abcdefghijklmnopqrstuvwxyz.'
alfa  = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ. '
# vigenere algorithm
#Mathematical formula is:  Ci (mi)  =(mi+ki) mod 28
# mod is  28 with space and .

def en_vigenere(plainText,key):
    #the text we want to encrypt
    plainText = plainText.upper()
    key = key.upper()
    cipherText = ''
    # repesenting the key index as far as key is concerned
    indexKey = 0
    # now we are going to concider all characters in
   plainText
    for char in plainText:
        #The number of shifts is equal to the index of the
   char in the alfabet and plus index of the char in the
   private key
        index = (alfa.find(char) + (alfa.find(key[indexKey]))
   ) % len (alfa) # this is the mathematical operation
        # adding the encrypted char to the cipherText
        cipherText = cipherText +alfa[index]
        # Now I'm concider the next letter and need to
   increment the key index
        indexKey = indexKey + 1

        # we need to start agin when we have concidered the
   last letter of key
        if indexKey == len(key):
            indexKey =0
    return cipherText

# Now I'm going to decrypt and using the following formula
# The number og shifts is equal to the index  of the char in
   the alfabet  and minus index of the char in the key
#Mathematical formula is:  Di (mi)  = (mi-ki) mod 28
def de_vigenere(cipherText, key):
    cipherText = cipherText.upper()
    key = key.upper()
    plainText = ''
    indexKey = 0

```

```
    for char in cipherText:
        index = (alfa.find(char) - (alfa.find(key[indexKey]))
    ) % len(alfa)
        plainText  = plainText + alfa[index]

        indexKey = indexKey +1
        if indexKey ==len(key):
            indexKey =0

    return plainText

if __name__ =="__main__":
    plainText = input("Enter some text to encrypt\n")
    encrypt = en_vigenere(plainText, 'LEMON')
    print("The encrypted message is: %s" % encrypt)
    decrypt = de_vigenere(encrypt, 'LEMON')
    print("The Decrypted message is: %s" % decrypt)
```

Listing 1: Vigenere Cipher with single key

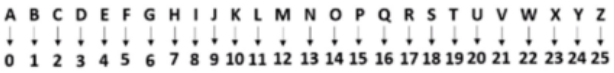
```
This is the output I got when I run the program.
PS C:\Users\m_rah\Desktop\crypto\en-decryption-algorithm\
   Vigenere> python .\vigenere.py
Enter some text to encrypt
The quick brown fox jumps over the lazy dog.
The encrypted message is: CLQNBDMOYMMV.I.KJ.
   JMUYYBDKSFSCKXTSMWEJKMOSSM
The Decrypted message is: THE QUICK BROWN FOX JUMPS OVER THE
   LAZY DOG.
```

Listing 2: Output of single key

0.4 Two Keys Code

This is basically the same method I use to encrypt the plain text with two keys. First I encrypting the plain text with help of the first key, when the plain text is encrypted with the first key, then I use the second key to encrypt the encrypted text again with help of the second key. The table is showing the encryption and decryption operation.

Alphabets with numerical representantions



Private_key_1	GREEN
The Mathematical formula:	$C_i(m_i) = (m_i + K_i) mod 26$ Private key are used and starting over
	GRE ENGRE ENGRE ENG REENG REEN GRE ENGR EEN
Plain_text:	THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG
Example operation:	G = 6 T =19 6+19 = 25 mod 26 => 25 and It's Z
Cipher_text_1:	ZYI UHOTO FEUNR JBD AYQCY FZIE ZYI PNFP HST
Private key_2	WATERMELON
Operation	Z = 25 W = 22 25+22=47 MOD 26 = 21 and its V in Alphabet, etc.
Cipher_text_2:	VYB YYAXZ TRQNK NSP EJEPU FSMV LCT DABP AWK

Decrypting from cipher text to plain text

$$D_i(m_i) = (m_i - K_i) mod 26$$

Private_key_2 = WATERMELON	W= 22 V = 21 21-22 = -1 mod 26 => 25 and it is Z And so an ...
Cipher_text_2:	VYB YYAXZ TRQNK NSP EJEPU FSMV LCT DABP AWK
Cipher_text_1:	ZYI UHOTO FEUNR JBD AYQCY FZIE ZYI PNFP HST
Private_key_1 = GREEN	G=6 Z=25 25-6 = 19 MOD 26 = 19 => T
Plain_text	THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG

Figure 6: Two Keys operation

Code

```
#alfa = ' abcdefghijklmnopqrstuvwxyz.'
alfa  = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ. '
# vigenere algorithm
#Mathematical formula is:  Ci (mi)  =(mi+ki) mod 28
# mod is  28 with space and .

def en_vigenere(plainText,key):
    #the text we want to encrypt
    plainText = plainText.upper()
    key = key.upper()
    cipherText = ''
    # repesenting the key index as far as key is concerned
    indexKey = 0
    # now we are going to concider all characters in plainText
    for char in plainText:
        #The number of shifts is equal to the index of the char in the alfabet and plus index of the char in the private key
        index = (alfa.find(char) + (alfa.find(key[indexKey])) ) % len (alfa) # this is the mathematical operation
        # adding the encrypted char to the cipherText
        cipherText = cipherText +alfa[index]
        # Now I'm concider the next letter and need to increment the key index
        indexKey = indexKey + 1

        # we need to start agin when we have concidered the last letter of key
        if indexKey == len(key):
            indexKey =0
    return cipherText

# Now I'm going to decrypt and using the following formula
# The number og shifts is equal to the index  of the char in the alfabet  and minus index of the char in the key
#Mathematical formula is:  Di (mi)  = (mi-ki) mod 28
def de_vigenere(cipherText, key):
    cipherText = cipherText.upper()
    key = key.upper()
    plainText = ''
    indexKey = 0
```

```
36
37      for char in cipherText:
38          index = (alfa.find(char) - (alfa.find(key[indexKey]))
    ) % len(alfa)
39          plainText  = plainText + alfa[index]
40
41          indexKey = indexKey +1
42          if indexKey ==len(key):
43              indexKey =0
44
45      return plainText
46
47 if __name__ =="__main__":
48      plainText = input("Enter some text to encrypt\n")
49      key_1  =    input("Enter the first key:\n")
50      encrypt1 = en_vigenere(plainText, key_1)          #
    Calling the Encrypting function to encrypt the message
    with the key 1
51      print("The encrypted message with key 1 is: %s" %
    encrypt1)
52      key_2 =     input("Enter the second key:\n")
53      encrypt2 = en_vigenere(encrypt1, key_2)           #
    Encrypting the message with the help of key 2. Calling the
     same function as I call when I encrypting the message
    with help of the key 1
54      print("The encrypted message wwith the key 2 is: %s" %
    encrypt2)
55
56      decrypt2 = de_vigenere(encrypt2, key_2)             #
    Decrypting the message to call the decrypting function,
    but first I decrypting the text with help of the second to
     get the encrypt1 text, then I decrypting the encrypt1 to
    get the plain text
57      print("Decrypted message with the key 2 is: %s" %
    decrypt2)
58      decrypt1 = de_vigenere(decrypt2, key_1)
59      print("The Decrypted message with the key 1 is: %s" %
    decrypt1)
```

Listing 3: Vigenere Cipher With two keys

```
1 PS C:\Users\m_rah\Desktop\crypto\en-decryption-algorithm\
    Vigenere> python .\vigenere.py
2 Enter some text to encrypt
3 The quick brown fox jumps over the lazy dog.
```

```
Enter the first key:
green
The encrypted message with key 1 is: ZYIDB.ZGOMHGS..FWS
    MPJQTDFDZICFILIMRRBAMJDKC
Enter the second key:
watermelon
The encrypted message wwith the key 2 is: TY HSKBRAZBGJCPR.
    BNZJJHXURHIWP ICMBBVMOZDDBG
Decrypted message with the key 2 is: ZYIDB.ZGOMHGS..FWS
    MPJQTDFDZICFILIMRRBAMJDKC
The Decrypted message with the key 1 is: THE QUICK BROWN FOX
    JUMPS OVER THE LAZY DOG.
```

Listing 4: Output of using two keys

0.5 Confusion and Diffusion

They are cryptography technique and purpose with the Confusion is that to make relationship between the statics of the cipher text and the value of the encryption key. On the contrary, diffusion attempts to hide the statistical structure of the plain text through expand out the influence respectively of each individual plain text numeral big piece. They both are properties of operation for secure cipher in cryptography and it was identified by Shannon in 1949. The Confusion is designed/ developed to boots the vagueness of cipher text and make certain that this technique gives no trace about the plain text and the correlation between the encryption key value and the statistics of the cipher text is maintained as complex as achievable. If someone gets control over the statistics of the cipher text and he/she couldn't be able to presume they key. On the other hand the diffusion is the increase the the redundancy of the plain text to cover the structure of the plain text to hinder to attack to calculate the key. The statistical structure of plain text can disappear into long range statistics of the cipher text and that no body can assume the key.

Part II

Assignment 2

0.6 AES Decryption Algorithm

AES is a symmetric block-cipher and is the latest and used to protect classified information and it is implemented in both hardware and software to encrypt the sensitive data. There are many ways that data can be exposed and it is extremely important that the companies or what ever organisation it is must protect the safeguarded information by using the right technology.

The block diagram will show the AES decryption algorithm. The block length is 128 bits and key length can be 128 or 192 or 256 bits and the number of round depends on the key length, if the key length is 128 bits then the number of round is 10 otherwise we increasing the number of round by 2 when the key length changes. We follow the following steps to decrypt for the particular the cipher text into plain text.

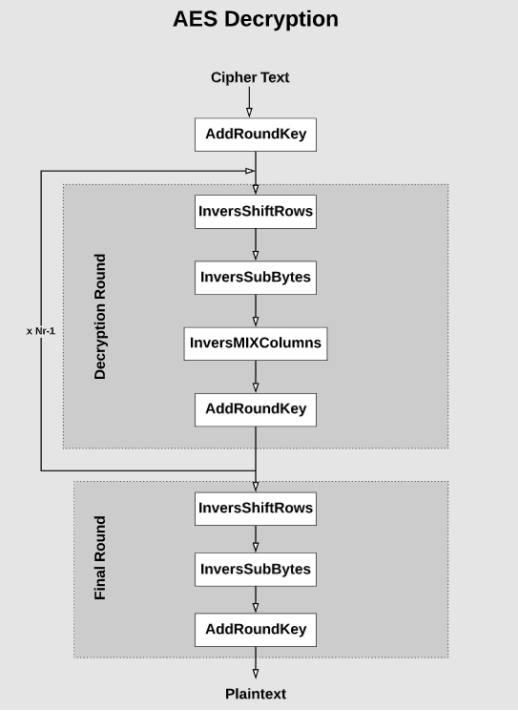


Figure 7: AES Decryption Algorithm [**book**]

First we take the inverse of input key of 128/192/256 bits and producing an expanded key, and this expanded key's size is related to the number of rounds to be performed. Second we AddRoundKey and expanded key from

previous step is used in this step. 3rd step is to SubBytes, applying the S-BOX or modifying the block by using an 8 bit substitution. The 4th step is shift rows and shifting the bytes of the bock by the ofsets. The 5th step is to MixColumns and it takes 4 bytes of each column and applies linaer transformation to the data.

0.7 AES Key Scheduling

Key scheduling is destined to expand the key into number of separate round key and producing the needed round key from initial key. I will show the steps in the following figures.

0.7.1 Round Key 1

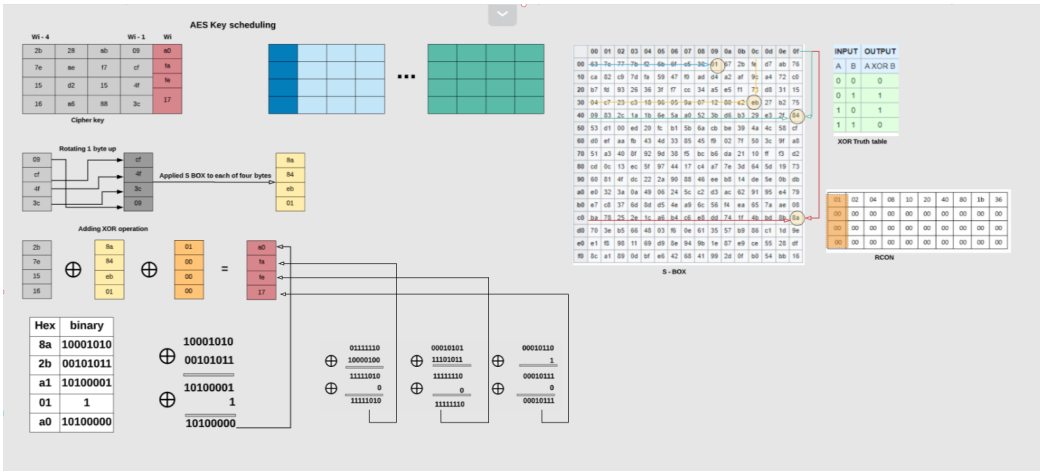


Figure 8: AES Key Scheduling Step: 1

- Calculating the words in positions that are a multiple of 4(W4,...,W40)
a: taking Wi-1 column and ratating 1 byte up
b: applying S-BOX to each of the 4 bytes
c: Adding XOR Wi-4 to the result and XOR RCON(i/4) to the result

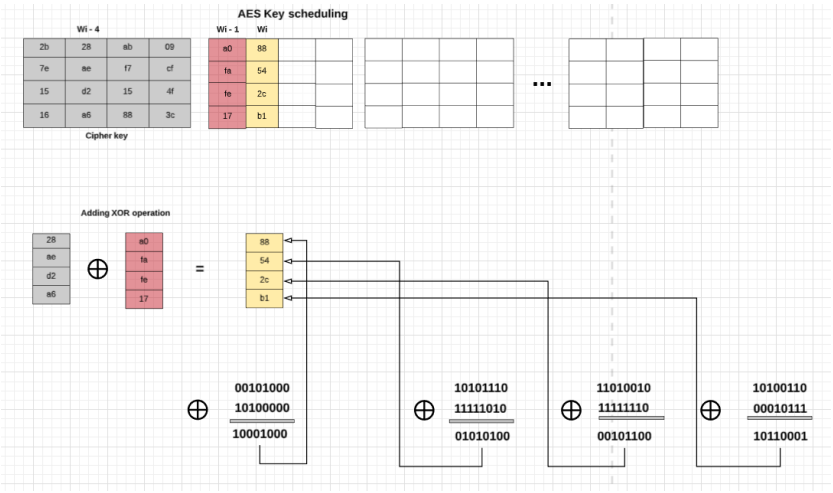


Figure 9: AES Key Scheduling Step: 2

- Calculating the remaining 32 bit words Wi

a: adding XOR to the previous word Wi-1, with the word 4 positions earlier Wi-4

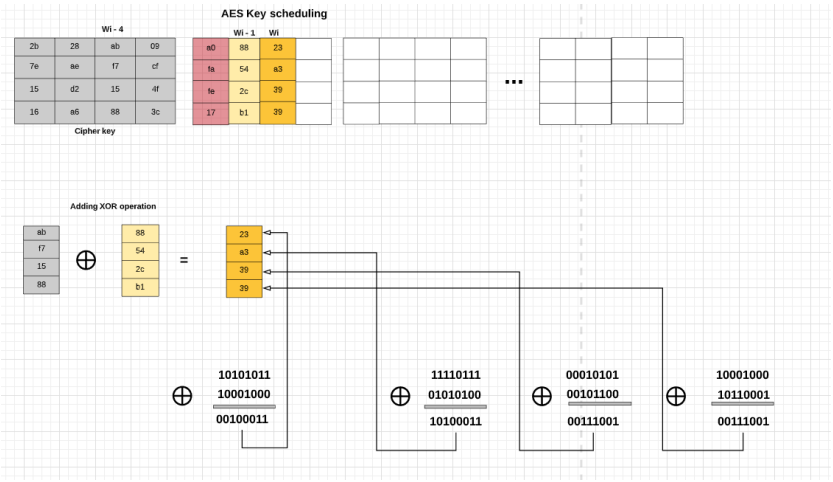


Figure 10: AES Key Scheduling Step: 3

- Calculating the remaining 32 bit words Wi

a: adding XOR to the previous word Wi-1, with the word 4 positions earlier Wi-4

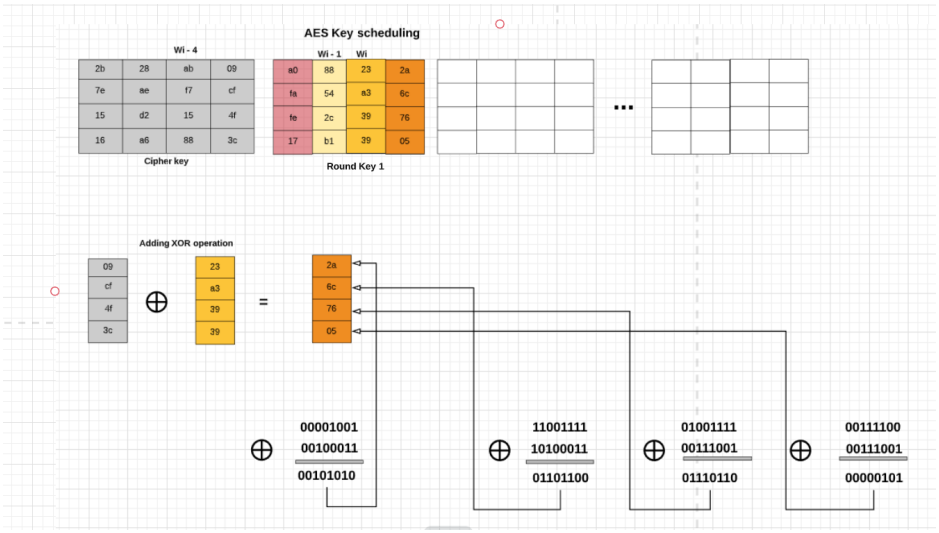


Figure 11: AES Key Scheduling Step: 4

- Calculating the remaining 32 bit words Wi

a: adding XOR to the previous word Wi-1, with the word 4 positions earlier Wi-4

0.7.2 Round Key 2

In Round key 2 we basically do the same operation as we did in the Round Key 1, but with a small changes. The changes is that we are not adding XOR operation on the Cipher Key but instead we applying XOR operation on the Round Key 1 using the RCON (02) not (01).

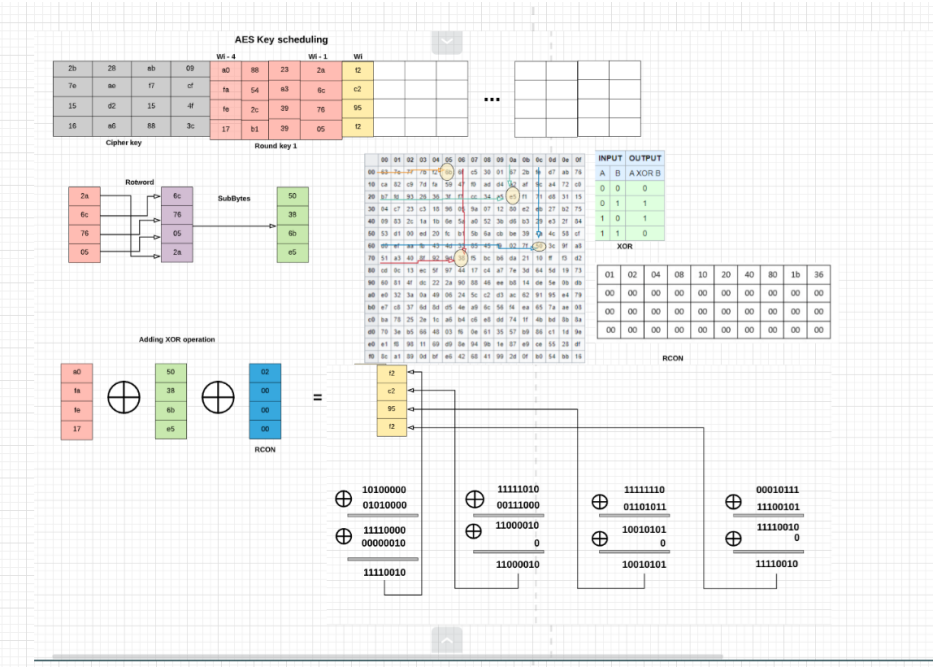


Figure 12: AES Key Scheduling Step: 5

- Calculating the words in positions that are a multiple of 4(W4,...,W40)
a: taking Wi-1 column and ratating 1 byte up
b: applying S-BOX to each of the 4 bytes
c: Adding XOR Wi-4 to the result and XOR RCON(02) to the result

0.7.3 Final Round key

This is the Final Round Key. We repeating the operation we did in the Round key 1 until the final round key. For Round key 1 we applied XOR operation on the Cipher Key and RCON(01), but for next round key we applying the XOR operation to the previous round key and taking the next RCON value that is not used yet.

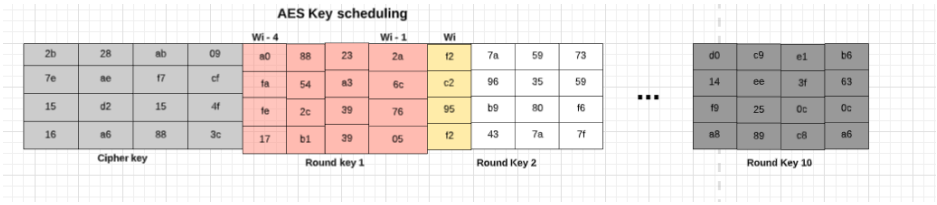


Figure 13: AES Key Scheduling round key 10

0.8 #Hashing

This is a process to converting an input of any length into a fixed sized string of text with the help of mathematical function. Trough an algorithm can the input text be converted into an array of letters and numbers. We have many formula that can be used to hash a message like, MD5, SHA1, SHA512 etc. A hash function must have some qualities that considered to be useful. A hash function must produce unique hash value and it should be impossible to produce the same hash output entering different inputs. Another important factor is the speed of the hash function it has to be fast to produce hash value. Hash function needs to be secure and needs to be non-invertible. The main properties of a hash function is: collision resistant, preimage, sencond preimage resistant.

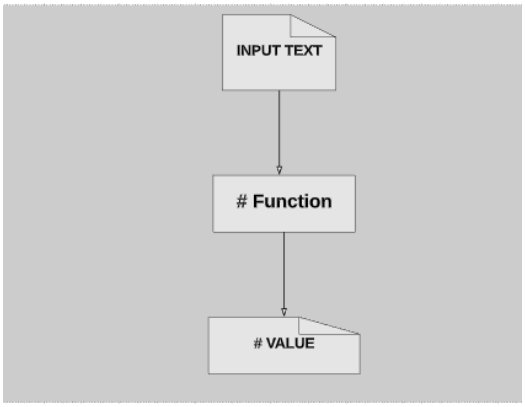


Figure 14: Hashing

0.9 Integrity

Alice and Bob as in a general measure they can set up by including asymmetric encryption method in their channel they are using to communicate. They have to ensure that the data is encrypted.This encryption method using two separate cryptography key one is public and the another is private key. Another method is to hash the message and it will protect the data integrity. Furthermore, they can analyze and match it with the original text.

0.10 Find the inverse

Finding the inverse of: $17y \equiv 1 \pmod{19}$
We want y such that:
$17y = 1 - 19x$
$17y \equiv 1 \pmod{19}$

Using Bezout's identity:
$1 = 17y + 19x$
Euclidean Algorithm:

$19 = 1 * 17 + 2$	$2 = 19 - 1 * 17$
$17 = 8 * 2 + 1$	$1 = 17 - 8 * 2$

Using backwards substitution:

$$\begin{aligned} 1 &= 17 - 8 * (19 - 1 * 17) \\ &= 17 - 8 * 19 + 8 * 17 \\ &= 9 * 17 - 8 * 19 \end{aligned}$$

$9 * 17 = 1 + 8 * 19 \equiv 1 \pmod{19}$
$y = 9, x = -8$
9 is y multiplicative inverse of $17(mod19)$

0.11 Symmetric & Asymmetric cryptography

They are two encryption method that we are using to hide the information from unauthorized people. Symmetric encryption require only one secret key to encrypt and decrypt the data or information and this secret key can be either a word, number or random string. Both parties who is involved in

the communication should know the secret key that is used to cipher and decipher the message and the AES is the most used symmetric algorithm. The downside of this type of cryptography must be to exchange the secret key.

Asymmetric encryption method uses two different keys to encrypt a message, its also know as public key cryptography. The public key can uses of everyone who wants to send us a message, but on the other hand the second private key is kept covert that no ones get know it. The secret key are exchanged over the internet and anyone who has the secret key can decipher the plain text. Assuming that the plain text is encrypted using a public key and it can only deciphered by using a secret key and if the plain text is encrypted using a secret key can be decipher using a public key. The most used asymmetric encrytion algortihms is RSA, ElGamal and DSA.

Part III

Assignment 3

0.12 The bullet points

The internet is insecure and the IoT devices we are connecting to it has high vulnerability and through this devices some random person can access to some sensitive information and he may use it against some one.

We are using cryptography method to encrypt and decrypt the information we are exchanging over the internet. This is method using public and secret key where the public key are available and the secret key is safe and the websites that is not using this method the history of visitor can intercept.

Hacking is to solve a creative problem in unexpected ways, but some of hackers want to learn how a system works while some others want to steal information and used to blackmail and someone is just kids and they just running the program without understanding it.

We have to be careful about what information we are dealing on the internet, because it can be stored in many places and if the information is public it can be stored and used by anyone who finds it and its very bad for our social life but in the worst case scenario it can destroy our life if are not careful enough of what are posting online.

0.13 The learning outcome

The coding challenge was to program a bot to navigate a maze. This challenge was very useful for me because I didn't know about the internet bot and I have learned how to program a bot in real and give it some instruction to take care of if someone attacks the system. A bot performs tasks faster than it would be possible for a human and the most web traffic is made of bots. Many companies running bots in their systems and if someone attacks the system the bot will block the attack or for example gives new direction to some place where there's no useful information and the attacker doesn't know that he attack the wrong system.

The password cracking games gives an idea of password battle and how an attacker might attempt to crack a password and it also gives an idea of how we can make stronger password. Before the attacker try to crack some-

one's password the attacker may need some information about the user then he/she can try to crack it. The attacker may start by guessing the user's password or run some simple script to crack it or in the worst case run advanced program to crack the password. The simple password is very easy to crack it can cracks by guessing it, but to have a strong password like: ${}_6@ = jKk9s - YmgC2dzpYnSF*?$ is very hard to crack for attackers. The stronger is the password the longer time it takes to be cracked and of course we need to use different password for different websites we are browsing just in case if one of the password is creaked then the other fine. The website who offer "Multi factor authentication" is strongly recommended to use.

Social Engineering is the move to manipulating someone to get their confidential information like, bank information, password and etc. In this part of the game I have learned about social engineering method and how people or companies can be manipulated to give them information. People getting very easy to be manipulate in this way since they don't have the knowledge to identify the phishing messages or email. The game was about to compare two apparently similar email and websites and find the differences between them, then analysed which of them was the phishing email or phishing websites and of course the phishing email/websites we don't trust and they are after to steal information.

The network attacks is an unauthorised action against the companies, government or private IT assets to destroy or modify them or steal the information. We can protect from a series of cyber attacks by buying cyber defences. To defend against cyber attacks we need to upgrade antivirus software and take back up company files, and educating the employers to recognise the phishing email.

To summarise this we as a developer need to upgrade the software of the company we are working in, educate the employers about social engineering (Phishing email), using strong password or multi factor authentication if they offers and different password for each website we are browsing and don't let an unauthorised person get access to the companies or private network.

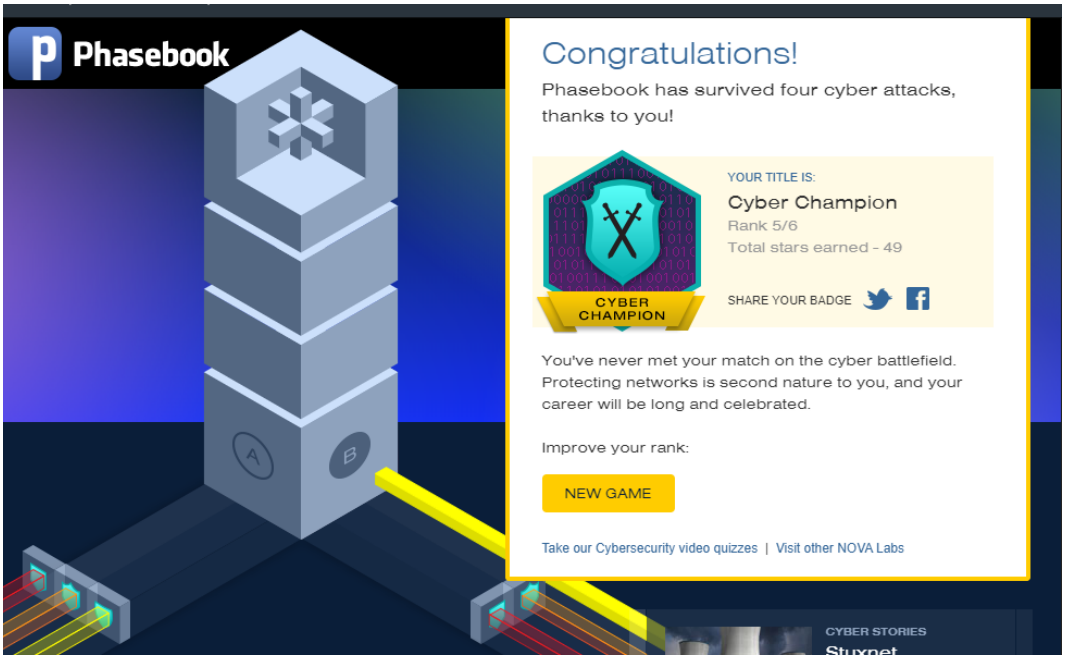


Figure 15: Cyber Champion [1]
This picture show the score of the game

Part IV

Assignment 4

0.14 Advantages of firewall

A firewall is just a wall that it keeps the intruders away from attacking networks and this firewall is placed in computers between the internet and the private network, in other word it is just a device. This device is designed to inspect all the networks traffic that goes throw the network and the internet and we can give some rules to this device to permit data to be shared. It can discard the packets that is not approved in protocols to accessing the network. The firewall system is to protect from unauthorised entry like for example when a private network is connected to the internet and it of course permit random people to access information from the external sources and it will prevent this access. We have two type of firewalls which one is software and the other one is hardware firewall. The firewall is to protect if something bad happens on the one side of the firewall and it won't let the other side be affected of that.

Out there, we have many sorts of attacks and some of them are IP spoofing (IPs), DoS and sniffer attack. The IPs is that the attacker is outside the network and pretending to be trusted device by using an external IP or an IP in range for the local network. The DoS attack is that it make the service disappear from the normal use by increasing the traffic until the system goes down. The last one or sniffer gives full overview of the information inside the packets and it can be a device or application that capturing network data exchanging and reading the packets.

The first generation of firewall is the packet filtering routers and it placed between the private network and the internet and this firewall work in IP layer of TCP protocol. The main benefit of this static filtering is that it gives low impact on the network performance and its cheap in many OS but on the other hand the downside of this type of filtering is that it operates in the network layer it analysing only the TCP and IP header.

The advantages of using firewall is that the cost/price is affordable and it is also easy to install. The implementation of packet filtering is easy since the it sue the current network routers and its high speed. On the contrary the drawback with packet filtering is that it doesn't understand the application layer and off course the packet filtering is not the secure one to use. It has some difficulty to setup some rules to the routers and it doesn't have any type user based authentication and information that comes from a specific user the packet filtering cannot authenticate it.

0.15 TLS

TLS is a widely used and it's the most important security protocols it gives end to end security over network. This protocol secure data via HTTPS and through encryption it gives confidentiality. This is true that client and server negotiate of choose which cryptography and algorithm key to use before the first byte of data is transferred between they. This process is very complicated and the shared secret is secure and even by an attacker who joins in middle of the connection and TLS makes difficult for an attacker to decrypt a secure https traffic. The diagram bellow show us the TLS handshake protocol where the connection is established in secure way. The server is authenticated and not the client and it starting with negotiation, where the client sends a message and it contains the newest version of TLS that client supports.

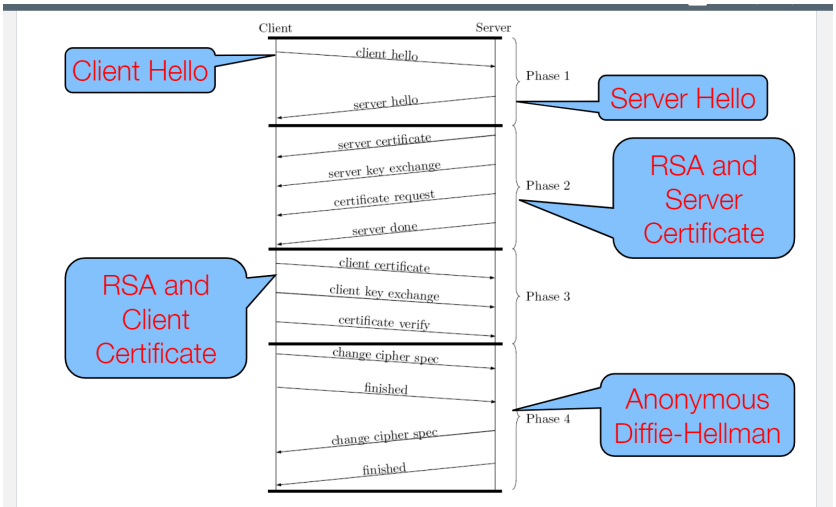


Figure 16: TLS handshake protocol [2]

0.16 IDS vs IPS

IDS and IPS are network infrastructure and IDS is a monitoring system and on the other hand the IPS is a control system. The IPS preventing the packet from delivery based on the content of the packets and IDS doesn't change the network packets. The table bellow show us the differences between the Intrusion Detection Systems and Intrusion Prevention Systems. The IDS examine and monitor network traffic for sign that specify attackers are using a known cyber threat to steal data. The IPS are placed in the same area of network as a firewall, between the internal and the external network.

IDS	BOTH	IPS
Detection & monitoring tools	Read network packets &compare the contents to a database of known threats	Is control system
Doesn't take action on their own		The ctr system rejects and accepts a packet based on the ruleset
Requires another system to look at the outcomes or a human		Requires that the DB gets regularly updated with new threat data

Figure 17: IDS vs IPS

0.16.1 Classification of IDS and IPS

2

The common classification of IDS are network IDS (NIDS) and host IDS (HIDS). They are a technique of security management for network and computer. The NIDS can be a hardware or software which are placed at various points along the network or installed on various computer. It examine the packets in both direction and offer real time detection. The HIDS is usually at the operating system level and collect data packets from sources internal to computer. The NIPS monitors the whole network for any threats analysing the protocol activities and WIPS monitors the wireless network for any sort of threats by analysing the WNP.

0.16.2 Factor for choosing a biometric modality

There are several key factor to consider when choosing a biometric modality. While these can be key factor general criteria for selection, but we have to realise that there is no single best modality for all conditions and implementations. These are the most important key factor to be taken in consideration before choosing a biometric modality.

Accuracy

This is the most important aspects to evaluate when choosing a biometric modality and this is based on different criteria including false acceptance rate, error rate, false reject rate and identification rate, etc.

Security

The security is an important factors and to choose which biometric modality is best for a project we have to consider what security level the project requires.

Acceptance

Another important factor is acceptance and the biometric modality must be user friendly. Culture also should be taken in consideration before choosing biometric modality and the user acceptance is the key to success.

Cost

The cost is also an important factor to consider when choosing biometric modality and not all this tools gives the same features, neither the same price. This tools also need to be maintained in the features and the cost of it also should be taken in consideration before choosing biometric modality for a project.

References

[1] *Cybersecurity Lab.*
Monday 6th January, 2020
. URL: https://www.pbs.org/wgbh/nova/labs/lab/cyber/.

[2] Kiran Raja Associate Professor.
Introductionto Data and Cyber-Security (DCS3100) Lecture18 -Introductionto Network Security.
2019
. URL: https://usn.instructure.com/courses/19603/files?preview=1105864.