الجمهورية الجزائرية الديموقراطية الشعبية
وزارة التعليم العالي و البحث العلمي
جامعة ابن خلدون –تيارت
كلية الرياضيات و الإعلام الألي
قسم :علوم الحاسب

مذكرة مقدمة لاستكمال متطلبات شهادة الماستر

لكلية الرياضيات والاعلام الالي

قسم: علوم الحاسب

تخصص:

هندسة البرمجيات

من اعداد :

روتال منير

صاف أحمد

تنفيذ إجراءات التشفير فيNodeJS

Implémentation de routines cryptographiques en NodeJS

تناقش يوم 18سبتمبر 2022 في تيارت امام أعضاء اللجنة المؤلفة من :

شيخاوي أحمد	الرتبة أستاذ جامعي	
دحماني يوسف	الرتبة أستاذ جامعي	المشرف
حمداني لعابدية	الرتبة مساعد أستاذ جامعي	

2021/2022

شكر و عرفان

قال رسول الله ﷺ ﴿لا يَشْكُرُ اللَّهَ مَنْ لا يَشْكُرُ النَّاسَ ﴾

رواه أحمد وأبو داود والبخاري في الأدب المفرد وابن حبان والطيالسي، وهو حديث صحيح صححه العلامة الألباني

وقال ﴿ مَنْ صَنَعَ إِلَيْكُمْ مَعْرُوفًا فَكَافِئُوهُ ، فَإِنْ لَمْ تَجِدُوا مَا تُكَافِئُونَهُ فَادْعُوا لَهُ حَتَّى تَرَوْا أَنَّكُمْ قَدْ كَافَأْتُمُوهُ ﴾ رواه أبو داود والنسائي بسند صحيح.

الحمد الله على إحسانه و الشكر له على توفيقه و امتنانه و نشهد أن لا إله إلا الله وحده لا شريك له و نشهد أن سيدنا و نبينا مُحَمَّد ﷺ عبده و رسوله الداعي إلى رضوانه و على آله.

اما بعد

كل الشكر والثناء إلى الله عزو جل ، على وتوفيقنا لإتمام هذا البحث المتواضع ،أتقدم بجزيل الشكر إلى الوالدين العزيزين الذين أعانوني و شجعوني على الاستمرار في مسيرة العلم ،و إكمال الدراسة الجامعية و البحث؛

كما أتوجه بالشكر الجزيل إلى المشرف البروفسور " دحماني يوسف " على توجيهاته العلمية و التي ساهمت في إتمام نقائص هذا العمل و إلى كل أساتذة الإعلام الالي وخاصة الدكتور "بوداعة بوجمعة" ،كما أتجه بخالص شكري و تقديري إلى كل من ساعدني من قريب أو من بعيد على إنجاز و إتمام هذا العمل.

وشكر خاص للسيد شيخاوي أحمد والسيدة حمداني عابدية ، إنه لمن دواعي سروري أن تكونوا لجنة تحكيم لهذا العمل

وفي الاخير

لا أقول إلاكما قال سيدنا سلمان عليه السلام

﴿رَبِّ أَوْزِعْنِي أَنْ أَشْكُرَ نِعْمَتَكَ الَّتِي أَنْعَمْتَ عَلَيَّ وَعَلَىٰ وَالِدَيَّ وَأَنْ أَعْمَلَ صَالِحًا تَرْضَاهُ وَأَدْخِلْنِي بِرَحْمَتِكَ فِي عِبَادِكَ الصَّالِحِينَ ﴾

ملخص

يساعد نظامنا على إجراء التشفير و البصمة الالكترونية للملفات و توليد كلمات السر بNodeJS .

في هذا البحث نقدم حلاً يوفر الخصوصية و السرية، النزاهة، المصادقة و عدم التنصل.

الهدف الرئيسي لبحثنا هو توفير نظام آمن للتشفير و فك التشفير بطرق حديثة وأخرى تقليدية والبصمة الإلكترونية للملفات.

Abstract

Our system helps to perform encryption and electronic fingerprinting of files and generate passwords with NodeJS.

In this paper, we present a solution that provides privacy, confidentiality, integrity, authentication and non-repudiation.

The main objective of our research is to provide a secure system for encryption and decryption by modern and traditional methods and electronic fingerprinting of files.

الكلمات المفتاحية :

، تشفير ، بصمة الكرتونية، نظام أمن ،توليد مفاتيح ، التوقيع الرقمي، تشفير حديث ، تشفير تقليدي ، توزيع مفاتيح التشفير

key words

Encryption, electronic fingerprint, security system, key generation, digital signature, modern encryption, traditional encryption, distribution of encryption keys.

1 الفهرس

قائمة الجداول

قائمة الأشكال

قائمة الصور

قائمة الاختصارات

- **AES: (Advanced Encryption Standard)معيار التشفير المتقدم**
- **CBC : (Cipher Block Chaining)**
- **CTR : (Counter) وضع العداد**
- **DES : (Data Encryption Standard)معيار تشفير البيانات**
- **ECB : (Electronic Code Book)**
- **IP : (Initial permutation) تبديل الترتيب الابتدائي**
- **IP^{-1} : (Final permutation) التبديل النهائي**
- **MD5: (Message Digest) ملخص الرسالة**
- **NIST: (National Institute of Standards and Technology)المعهد الوطني للمعايير والتكنولوجيا**
- **P-BOX : (permutation box) التبديل و التمديد من اجل حساب**
- **PC-1 : (Permuted choice 1))تبديل الترتيب الاول**
- **PC-2: (Permuted choice 2) تبديل الترتيب الثاني**
- **PKI: (Public Key Infrastructure) البنية التحتية للمفاتيح العمومية**
- **S-BOX: (Substitution boxes)صناديق الاستبدال**
- **SHA : (Secure Hash Algorithm) خوارزمية التجزئة الآمنة**
- **SSL : (Secure Sockets Layer) طبقة مآخذ التوصيل الآمنة**
- **URL : Uniform Resource Locator() مُحدد الموارد الموحد**
- **VPN : (Virtual Private Network) شبكة خاصة افتراضية**

مقدمة عامة :

إن التزايد الهائل في وسائل الاتصال واستخدامها على نطاق واسع، جعل المعلومات أكثر عرضة للخطر للقرصنة او السرقة او التجسس من قبل المتسللين الذين يسعون للحصول على اتصالات ومعلومات سرية ، لذلك يجب حماية الاتصالات والمعلومات من أي طرف غير مصرح له بذلك، هذا من بين الأسباب التي جعلت التشفير منذ السبعينيات مجالًا نشطًا جدًا للبحث العلمي وصناعة برامج التشفير و من بينها برنامجنا.

2 الفصل الاول : التشفير

2-1مقدمة :

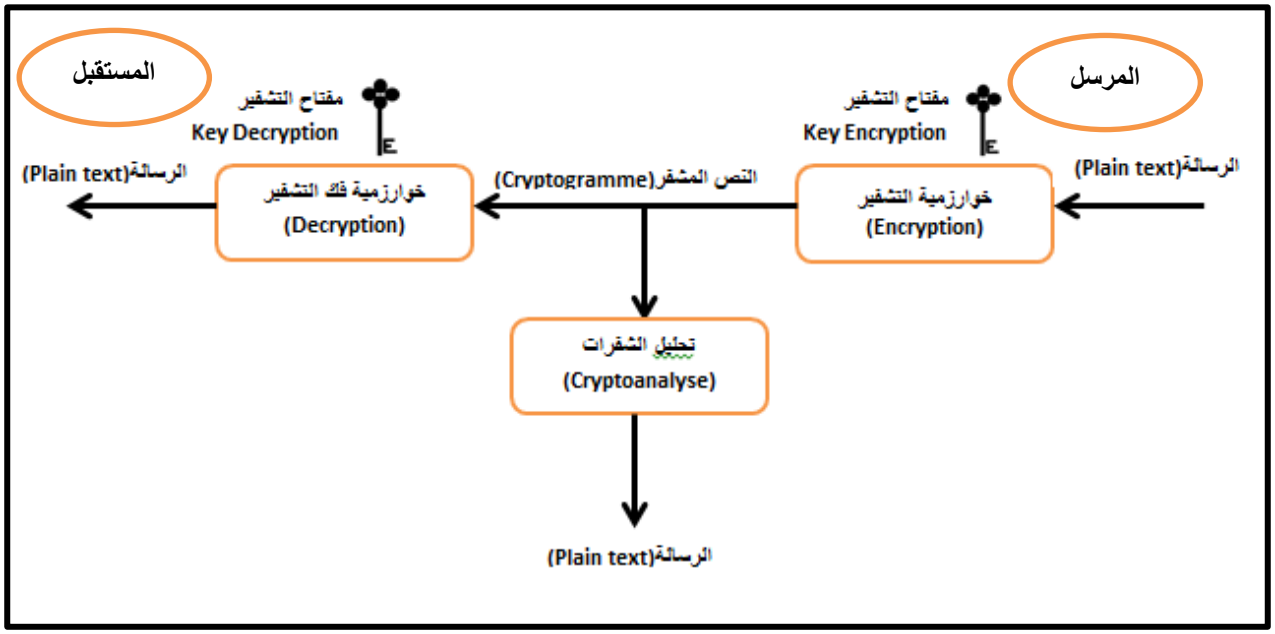
كان في الماضي اكثر مستخدمي التشفير هم المنظمات العسكرية والاستخباراتية. أما اليوم ، فإن التشفير في كل مكان تقريبا! تعد آليات الأمان التي تعتمد على التشفير جزءًا لا يتجزأ من أي نظام كمبيوتر تقريبًا.

يعتمد المستخدمون (غالبًا عن غير قصد) على التشفير في كل مرة يتصلون فيها بموقع ويب آمن أو عندما يرسلون رسائل او أي ملف

في هذا الفصل سوف نتعرف على تعريف التشفير، اهم مصطلحاته ،مزاياه و خوارزميات التشفير التقليدية و الحديثة

.

2-2المصطلحات الرئيسية :



الشكل 1 : المبدأ العام لخوارزمية التشفير

- التشفير(Cryptography): التشفير هو طريقة لحماية المعلومات والاتصالات من خلال استخدام الرموز ، بحيث لا يتمكن من قراءتها ومعالجتها سوى من تستهدفهم المعلومات.

في علوم الكمبيوتر ، يشير التشفير إلى المعلومات الآمنة وتقنيات الاتصال المستمدة من المفاهيم الرياضية ومجموعة من الحسابات القائمة على القواعد تسمى الخوارزميات ، لتحويل الرسائل بطرق يصعب فك شفرتها. تُستخدم هذه الخوارزميات لإنشاء مفتاح التشفير والتوقيع الرقمي والتحقق لحماية خصوصية البيانات وتصفح الويب على الإنترنت والاتصالات السرية مثل معاملات بطاقات الائتمان والبريد الإلكتروني.[1]

- **علم التشفير:** الرسالة او النص العادي (Plaintext): المعلومات في شكلها العادي.

علم التشفير = التشفير + تحليل الشفرات

[1] Kathleen Richards ، (27 ،09 ، 2021)، cryptography ، (30 ، 6 ، 2022) ، https://www.techtarget.com/searchsecurity/definition/cryptography

مفتاح التشفير / فك التشفير (Key Encryption): في التشفير ، المفتاح keyهو سلسلة من الأحرف المستخدمة داخل خوارزمية تشفير لتغيير البيانات بحيث تظهر بشكل عشوائي. مثل المفتاح المادي ، يقوم بتأمين (تشفير) البيانات بحيث لا يتمكن سوى شخص لديه المفتاح الصحيح من فتحها (فك تشفيرها) [2]

التشفير(encryption): التشفير هو طريقة لخلط البيانات بحيث يمكن للأطراف المصرح لها فقط فهم المعلومات. من الناحية الفنية ، إنها عملية تحويل النص العادي plaintext المقروء على الإنسان إلى نص غير مفهوم ، يُعرف أيضًا باسم النص المشفرciphertext . بعبارات أبسط ، يأخذ التشفير بيانات قابلة للقراءة ويغيرها بحيث تظهر عشوائية. يتطلب التشفير استخدام مفتاح تشفير: مجموعة من القيم الرياضية التي يتفق عليها كل من مرسل ومتلقي الرسالة المشفرة.

نص مشفر (Cryptogram): هو نتيجة التشفير النص عادي plaintext باستخدام إحدى خوارزميات التشفير. يُعرف النص المشفرplaintext أيضًا بالبيانات المشفرة لأنه يحتوي على شكل من أشكال النص العادي plaintext الذي لا يمكن قراءته بواسطة الإنسان أو الكمبيوتر بدون خوارزمية التشفير المناسب لفك تشفيره.

فك التشفير(Decryption): هو عملية تحويل البيانات المشفرة ciphertext إلى معلومات يمكن التعرف عليها. إنه عكس ال تشفير encryption ، الذي يأخذ بيانات قابلة للقراءة ويجعلها غير قابلة للتمييز.

قد يتم تشفير الملفات ونقل البيانات لمنع الوصول غير المصرح به. إذا حاول شخص ما عرض مستند مشفر ، فسيظهر كسلسلة عشوائية من الأحرف. إذا حاول شخص ما "التطفل" على اتصال شبكة مشفر ، فلن يكون للبيانات أي معنى.[3]

تحليل الشفرات(Cryptoanalyse): عملية استعادة النص الواضح أو العثور على المفتاح.

---

[2] ما هو مفتاح التشفير؟ (2022، 6، 15)، https://www.cloudflare.com/learning/ssl/what-is-a-cryptographic-key/

[3] Decryption ، (2022، 6، 15)، https://techterms.com/definition/decryption

وأول من كان له الفضل في تطوير تحليل الشفرات هو العالم العربي الكندي [4]

نظام التشفير(Cryptosystem): هو مجموعة من خوارزميات التشفير اللازمة لتنفيذ خدمة التشفير.

2-3أهداف التشفير واهم استخداماته :

2-3-1 أهدافه:

الخصوصية أو السرية (**Confidentiality**) :سيتمكن الأشخاص المصرح لهم فقط من الحصول على المعلومات .

النزاهة (Integrity) :يجب أن يكون من الممكن التحقق من أن الرسالة لم يتم تعديلها أثناء رحلتها.

المصادقة (Authentication) :عندما يكون الاتصال آمنًا بين طرفين.

عدم التنصل(Non-repudiation) :لا يجب على أي طرف أن ينكر لاحقًا أنه قد اتخذ إجراءً معينًا أو أنه نقل معلومات معينة او ينكر انه توصل بمعلومات معينة.

استخداماته :

يوجد العديد من الاستخدامات المهمة للتشفير يقابل معظمنا التشفير كل يوم.

الاستخدامات الشائعة:

1. في كل مرة تستخدم فيها ماكينة صراف آلي أو تشتري شيئًا عبر الإنترنت باستخدام هاتف ذكي، يتم استخدام التشفير لحماية المعلومات التي يتم نقلها.

2. تأمين الأجهزة، مثل التشفير لأجهزة الكمبيوتر المحمولة.

[4] الكندي (2022، 5 ، 16) ، من ويكيبيديا، https://ar.wikipedia.org/wiki/%D8%A7%D9%84%D9%83%D9%86%D8%AF%D9%8A#%D8%A7%D9%84%D8%AA%D8%B4%D9%81%D9%8A%D8%B1

3. تستخدم معظم مواقع الويب السليمة" طبقة المقابس الآمنة(SSL) "، وهي شكل من أشكال تشفير البيانات عند إرسالها من موقع ويب وإليه. وهذا يمنع المهاجمين من الوصول إلى تلك البيانات أثناء نقلها. ابحث عن رمز القفل في شريط URL وحرف "s" في "https://" للتأكد من أنك تجري معاملات آمنة ومشفرة عبر الإنترنت.

4. يتم أيضًا تشفير رسائل WhatsApp الخاصة بك، وقد يكون لديك أيضًا مجلد مشفر على هاتفك.

5. يمكن أيضًا أن يتم تشفير بريدك الإلكتروني باستخدام بروتوكولات مثل.OpenPGP

6. تستخدم الشبكات الافتراضية الخاصة(VPN) التشفير، ويجب تشفير كل ما تخزنه في السحابة. يمكنك تشفير محرك الأقراص الثابتة بالكامل، بل إجراء مكالمات صوتية مشفرة.

7. يستخدم التشفير لإثبات سلامة وصحة المعلومات، وهذا باستخدام ما يعرف بالتوقيعات الرقمية. التشفير جزء لا يتجزأ من إدارة الحقوق الرقمية وحماية المؤلفات.

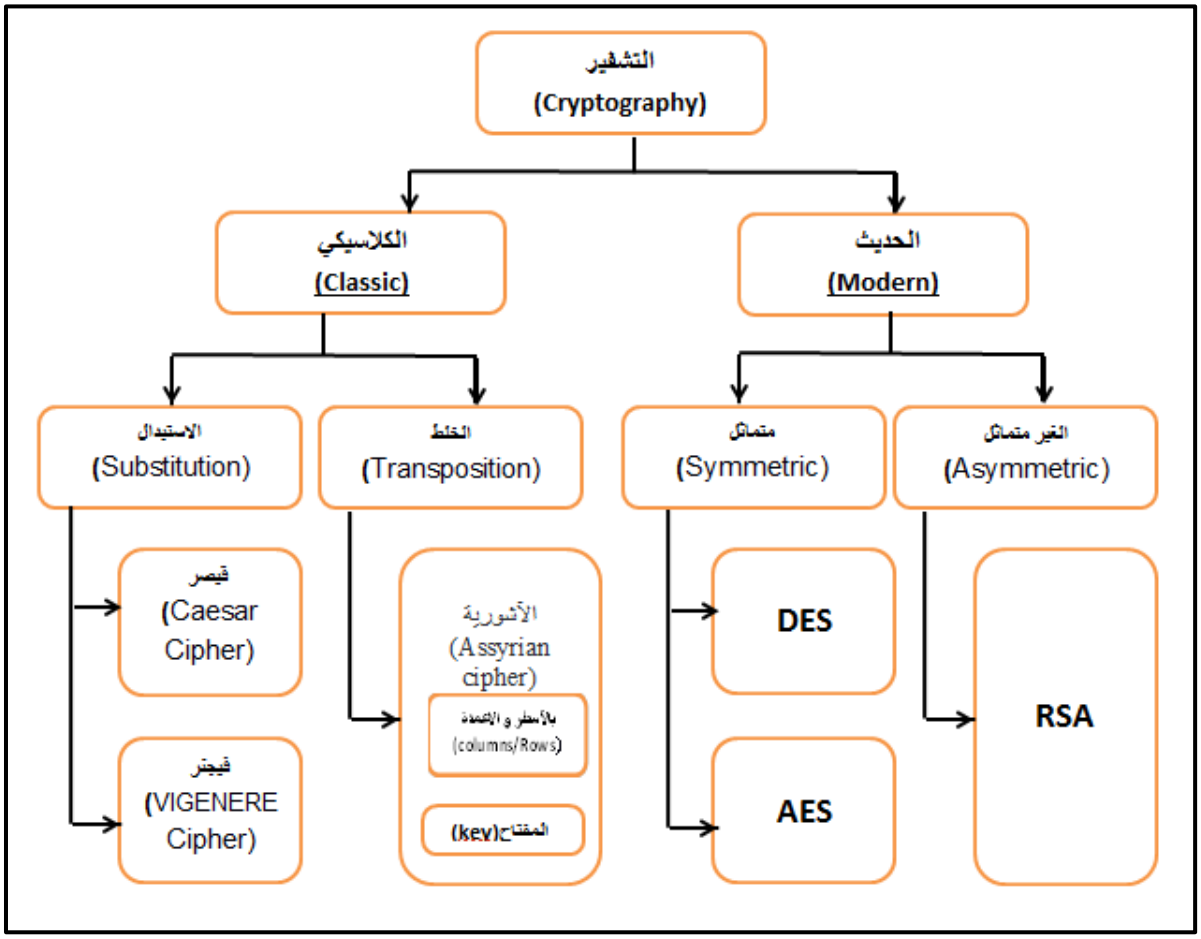
8. يمكن استخدام التشفير لمحو البيانات. نظرًا لأنه يمكن أحيانًا إعادة المعلومات المحذوفة باستخدام أدوات استعادة البيانات، فإنك إذا قمت بتشفير البيانات أولاً وتخلصت من المفتاح، فلن يمكن لأي شخص أن يسترد إلا النص المشفر وليس البيانات الأصلية.

9. إن التشفير وسيلة لحماية المعلومات الخاصة من السرقة أو الاختراق في مجال الأمن الإلكتروني[5] .

[5] kaspersky ، الاستخدامات المهمة للتشفير ،(2022 ، 7 ، 16)، https://me.kaspersky.com/resource-center/definitions/encryption

2-4 طرق التشفير:

في مجال التشفير(cryptography) هناك قسمين اساسيين من التشفير: التشفير الكلاسيكي(Classic) و التشفير الحديث (Modern)وكل منهما فيه عدة اقسام الشكل 2 يوضح ذلك.



الشكل 2 : أنواع التشفير

2-4-1 التشفير الكلاسيكي (Classic Cryptography):

من اقدم أنواع التشفير ظهر قبل النصف الثاني من القرن العشرين (الحرب العالمية الأولى و الثانية) ومن مميزاته سهل وغير معقد و لا يحتاج الى الآلة يمكن عمله بليد فقط ومن عيوبه سهل الفك ولا يدعم اللغة العربية و الفراغات و الرموز والأرقام ويعتمد على مبدأين :

2-4-1-1 التشفير بالاستبدال (Substitution) :

توجد به خوارزميتين: Caesar Cipher و VIGENERE Cipher

1) الخوارزمية الاولى (**Caesar Cipher**):

طورها جول سيزار يتم فيها استبدال كل حرف بالحرف التالي مع عدد الخطوات

الحروف	ABCDEFGHIJKLMNOPQRSTUVWXYZ
حروف سيزار (3+)	DEFGHIJKLMNOPQRSTUVWXYZABC

- مثال : شفير كلمة **TIARET** مع الإزاحة ب 3

الكلمة المشفرة هي **WLDUHW**

- مثال : فك تشفير كلمة **PRKDPHG** مع الازاحة 3

فك تشفير كلمة **PRKDPHG** هي **MOHAMED**

او بطريقة الجبرية :

$$Ci = (Pi + k) \% 26$$

$$Pi = (Ci - k) \% 26$$

Ci:رقم الحرف المشفر في مصفوفة الحروف

Pi:رقم الحرف الأصلي في مصفوفة الحروف

k:المفتاح

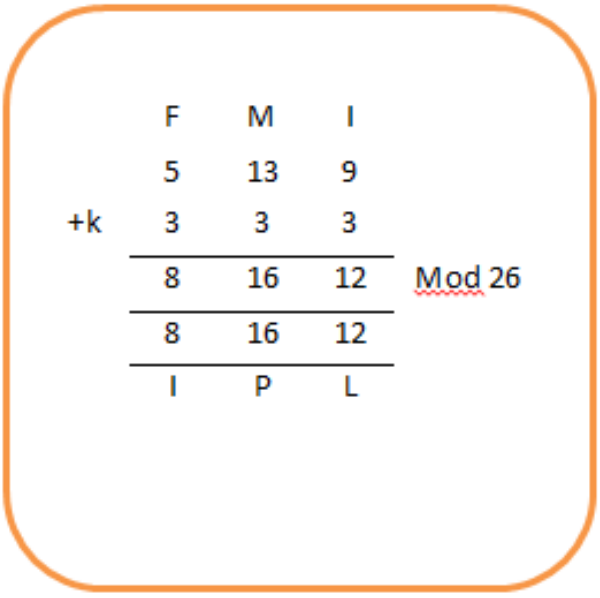
- مثال الطريقة الجبرية :

تشفير كلمة FMI بإزاحة 3 :

1- تحويل الحرف إلى الرقم الذي يتطابق مع ترتيبه في الأبجدية بدءًا من 0 .

(A=0, B=1, C=2, …, Y=24, Z=25)

2- حساب و تحويل الرقم الناتج الى الحرف الذي بطابقه في الترتيب الأبجدي النتيجة هي النص المشفر (**Cryptogram**)



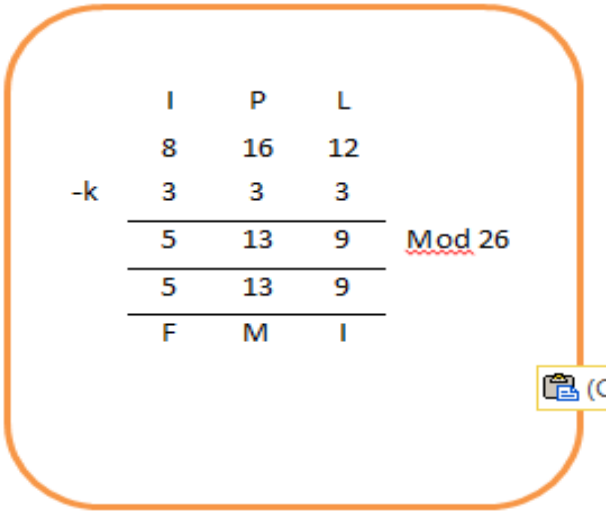
الشكل 3 : مثال تشفير جبريا

فك تشفير " **IPL**" بإزاحة 3

1- تحويل الحرف إلى الرقم الذي يتطابق مع ترتيبه في الأبجدية بدءًا من 0 .

(A=0, B=1, C=2, …, Y=24, Z=25)

2- حساب و تحويل الرقم الناتج الى الحرف الذي بطابقه في الترتيب الأبجدي النتيجة هي النص الأصلي(Plaintext)



الشكل 4 : مثال فك تشفير جيريا

اخترعت هذه الشفرة من قبل العالم الفرنسي بليز دي فجنير أفضل طرق اليدوية المعروفة للشفرات متعددة الأحرف تَستخدم شفرة فيجنر مربع فيجنر لإجراء عملية التشفير او بطريقة جبرية

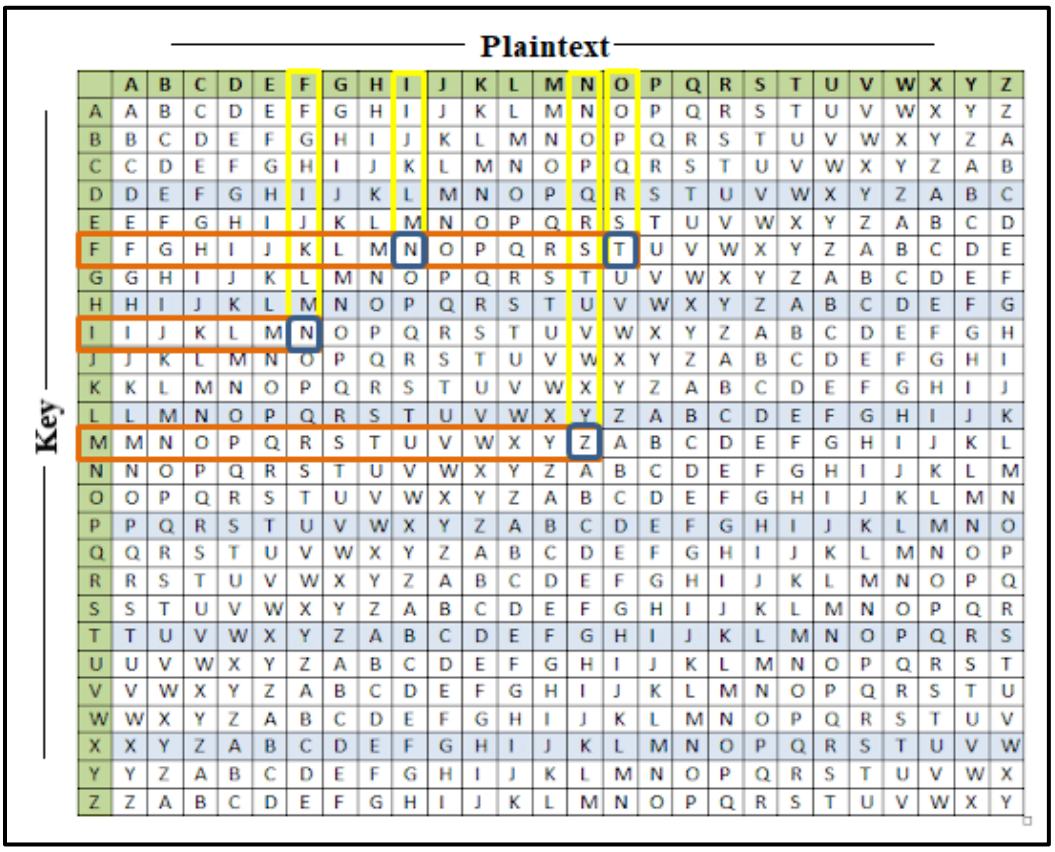
Plaintext

Key	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

جدول 1:مربع فيجنر

هذه الطريقة تحتاج الى مفتاح key ويجب ان يكون النص الواضح plaintext بنفس طول واذا كانت اقل منها طولا نكرر احرف المفتاح Key على التوالي . مثال : شفّر النص الواضح INFO اذا علمت ان المفتاح هو FMI باستخدام خوارزمية مربع فجنيير.

Plaintext	I	N	F	O
Key	F	M	I	F
Cryptogram	N	Z	N	T



جدول 2:كيفية الاسقاطات في مربع فيجنر

طريقة فك التشفير: Decryption

1- نعتبر الصف الاول هو احرف المفتاح.

2- نختار اول حرف بالمفتاح وننزل عمودياً وصولاً للحرف المشفر. نرى ما يقابله في العمود.

مثال :النص المشفر Cryptogram هو **NZNT** و المفتاح هو **FMI**

بتطبيق 1و2 في الشكل هذا الجدول نجد

النص المشفر Cryptogram	N	Z	N	T
المفتاح Key	F	M	I	F
النص الواضح Plaintext	I	N	F	O

او بطريقة الجبرية :

1- يكون النص الواضح plaintext بنفس طول واذا كانت اقل منها طولا نكرر احرف المفتاح Keyعلى التوالي حتى يصبح طول key مساوي لطول النص الواضح plain text

2- تحويل الحرف إلى الرقم الذي يتطابق مع ترتيبه في الأبجدية بدءًا من 0 .

(A=0, B=1, C=2, …, Y=24, Z=25)

التشفير $C[i]=(P[i] + key[i]) \bmod 26$

فك التشفير $P[i] = (C[i] - key[i]) \bmod 26$

	I	N	F	O	
	8	13	5	14	
+k[i]	5	12	8	5	
	13	25	13	19	
	13	25	13	19	Mod 26
	N	Z	N	T	

الشكل 5 : التشفير فيجنر جبريا

النص الواضح (plaintext): INFO

2-1-4-2 تشفير الخلط (Transposition):

1) خوارزمية التشفير الآشورية **Assyrian cipher** :

ربما تكون تقنية التشفير الآشوريةAssyrian cipher أول دليل على استخدام أجهزة التشفير في اليونان منذ 600 قبل الميلاد ، لإخفاء الرسائل المكتوبة على شرائح من ورق البردي (الشكل 8). كلا الطريقتين تشتركان في ان الرسالة توضع في مصفوفة ثنائية البعد[6]

[6] Cryptage par transposition، (10 ، 7 ، 2022)، https://web.maths.unsw.edu.au/~lafaye/CCM/crypto/transpo.htm

صورة 1: المقياس (الاسطوانة الخشبية) المستعمل في التشفير

تتكون تقنية التشفير هذه عن طريق **التشفير الخلط (Transposition)** من لف شريط من ورق البردي على أسطوانة خشبية تسمى مقياس ، ثم كتابة النص بشكل طولي على الشريط. بمجرد فتح البردى ، لم تعد الرسالة مفهومة بشكل مباشر.

لفك تشفير الرسالة ، يجب أن يكون لدى المستلم أسطوانة من نفس القطر: لذلك من المفهوم أن مفتاح التشفير هنا هو القطر المذكور ، وبالتالي هشاشة التشفير الآشوري. في الواقع ، كان كافياً (للمتسللين في ذلك الوقت) اختبار الأسطوانات خطوة بخطوة بأقطار مختلفة: يقال في هذه الحالة أنه يمكن كسر الطريقة بشكل منهجي.[7]

مثال :

لتكن المصفوفة m(6,5) و النص الواضح " plaintext MESSAGE SECRET ATRANSPOSER" كما هو موضح في الشكل التالي :

M	**E**	**S**	**S**	**A**	**G**
E		**S**	**E**	**C**	**R**
E	**T**			**A**	**T**
R	**A**	**N**	**S**	**P**	**O**
S	**E**	**R**			

صورة 2: مصفوفة الخلط

[7] omnilogie ، (14/09/2009) ، Le chiffrement assyrien ، (2022 ، 7 ، 10)، https://omnilogie.fr/O/Le_chiffrement_assyrien

إذن النص المشفر هو " MEERSE TAESS NRSEAS AC P GRTO "

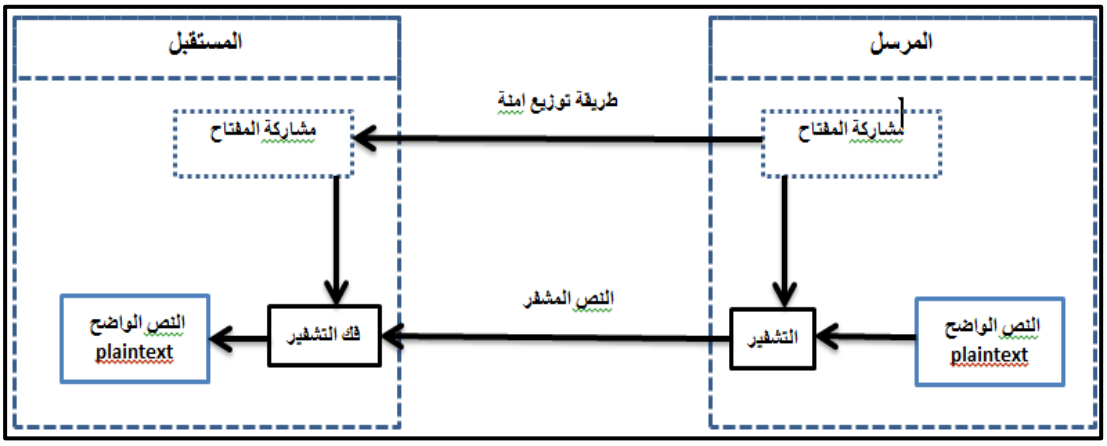
2-4-2 التشفير الحديث (Modern Cryptography):

توجد خوارزميتين للتشفير هما الأكثر شيوعًا: التشفير **المتماثل** و**غير المتماثل** .يشير الاسمان إلى ما إذا كان يتم استخدام المفتاح نفسه للتشفير ثم لفك التشفير أم لا:

2-4-2-1 مفاتيح التشفير المتماثلة :

1) التشفير بالمفاتيح المتماثلة

يقصد بالمفاتيح المتماثلة ، اي انه يوجد لدى المرسل و المستقبل **مفتاح واحد** معروف لديهما لتشفير و فك التشفير ، وهي **كلمة السر**. **key**



الشكل6 : مخطط مفاتيح التشفير المتماثلة

2) خصائص التشفير بالمفتاح المتماثل**Symmetric Key Encryption :**

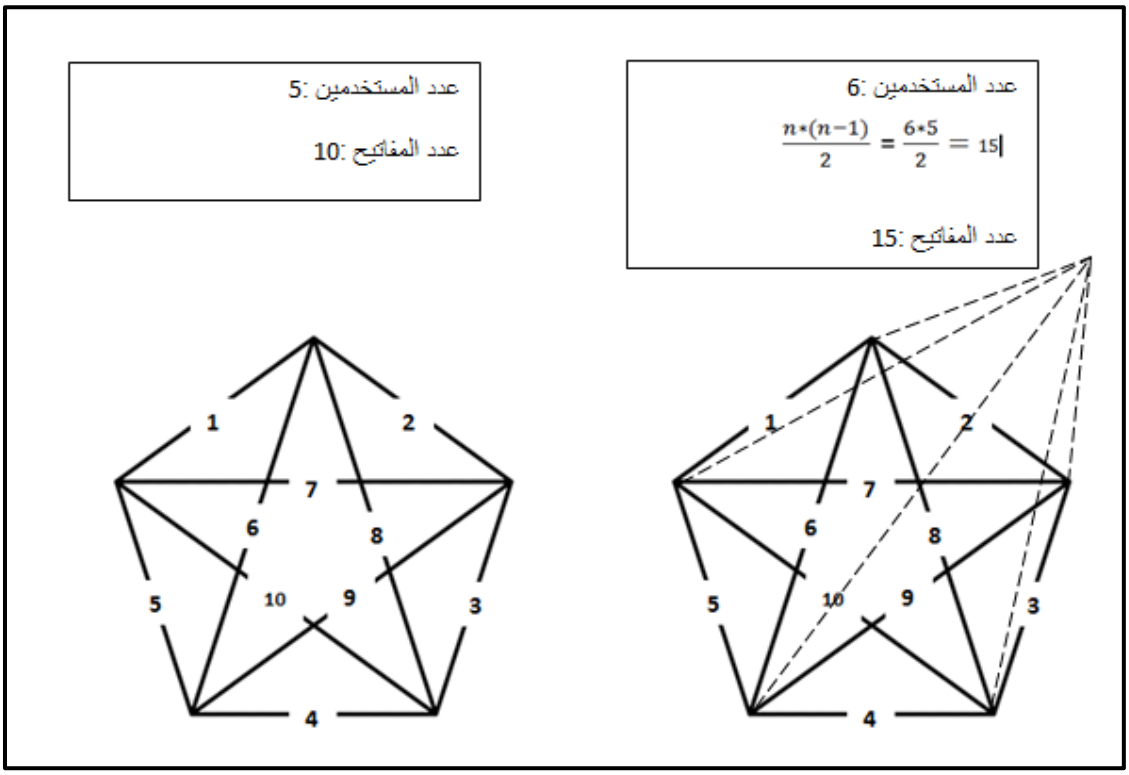
1- يجب على جميع الأشخاص مشاركة المفتاح السري قبل إرسال المعلومات.

2- يوصى بتغيير المفاتيح بانتظام لمنع أي هجوم على النظام.

3- هناك حاجة إلى وجود آلية قوية لتبادل المفتاح بين الأطراف المتواصلة، بما أن المفاتيح مطلوبة لتغييرها بانتظام، تصبح هذه الآلية باهظة الثمن ومكلفة.

4- لتمكين الاتصال بين مجموعة من الاشخاص نحتاج إلى وجود مفتاح بين كل شخصين، حيث يكون عدد المفاتيح المطلوبة لمجموعة بها n من الأشخاص هي $n \times (n - 1) / 2$.[8]

[8] Taha al Mohamed ، (16 10 2020)، التشفير المتماثل Symmetric Encryption والتشفير الغير متماثل Asymmetric Encryption،(30 7 2022)،
https://www.linkedin.com/pulse/%D8%A7%D9%84%D8%AA%D8%B4%D9%81%D9%8A%D8%B1-%D8%A7%D9%84%D9%85%D8%AA%D9%85%D8%A7%D8%AB%D9%84-symmetric-encryption-%D9%88%D8%A7%D9%84%D8%AA%D8%B4%D9%81%D9%8A%D8%B1-



الشكل 7 : مثال توضيحي يبين الية توزيع المفاتيح

1) يكون طول المفتاح عدد الـ (Bits) في هذا التشفير أصغر، وبالتالي فإن عملية فك التشفير تكون أسرع من تشفير المفتاح غير المتماثل، لذا فإن قوة معالجة نظام الكمبيوتر اللازمة لتشغيل خوارزمية متماثلة أقل[9].

3) عوائق تشفير المفتاح المتماثل :

1- **تكوين المفتاح** :قبل أي اتصال يحتاج الطرفان إلى الاتفاق على مفتاح سري، هذا يتطلب وجود آلية إنشاء مفتاح سري بشكل آمن.

2- **مشكلة الثقة** :بما أن المرسل والمستقبل يستخدمان نفس المفتاح فهناك مطلب ضمني بأن يثق المرسل والمستقبل في بعضهما البعض، فمثلا قد يفقد المستقبل المفتاح للمهاجمين ولم يتم إبلاغ المرسل!

%D8%A7%D9%84%D8%BA%D9%8A%D8%B1-%D9%85%D8%AA%D9%85%D8%A7%D8%AB%D9%84-taha?trk=public_profile_article_view

[9] نفس المرجع

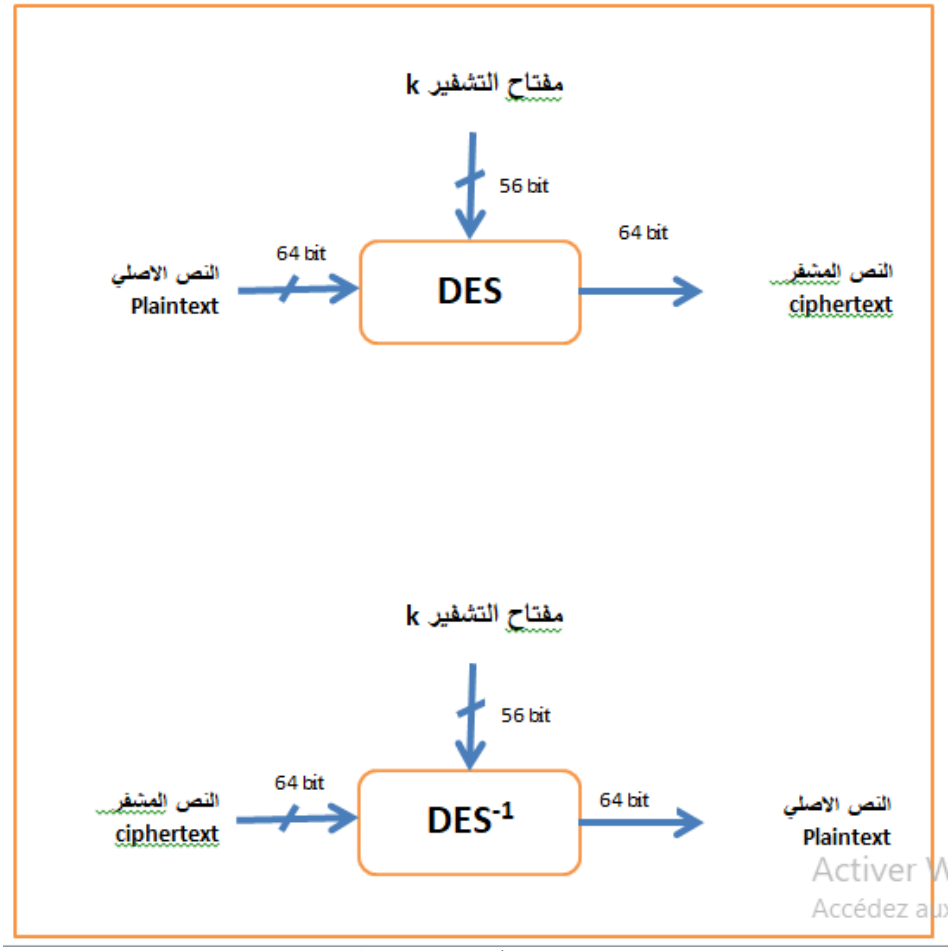
3- **التواصل العام** :يحتاج الناس إلى تبادل المعلومات مع أطراف غير معروفة وغير موثوق بها، كالتواصل بين البائع عبر الإنترنت والعميل.

أدت جميع هذه القيود إلى ظهور أنظمة تشفير المفتاح الغير متماثل[10].

4) خوارزمية التشفير بالمفتاح المتماثل **DES**

كانت DES أهم خوارزمية تشفير على مدار الثلاثين عامًا الماضية.

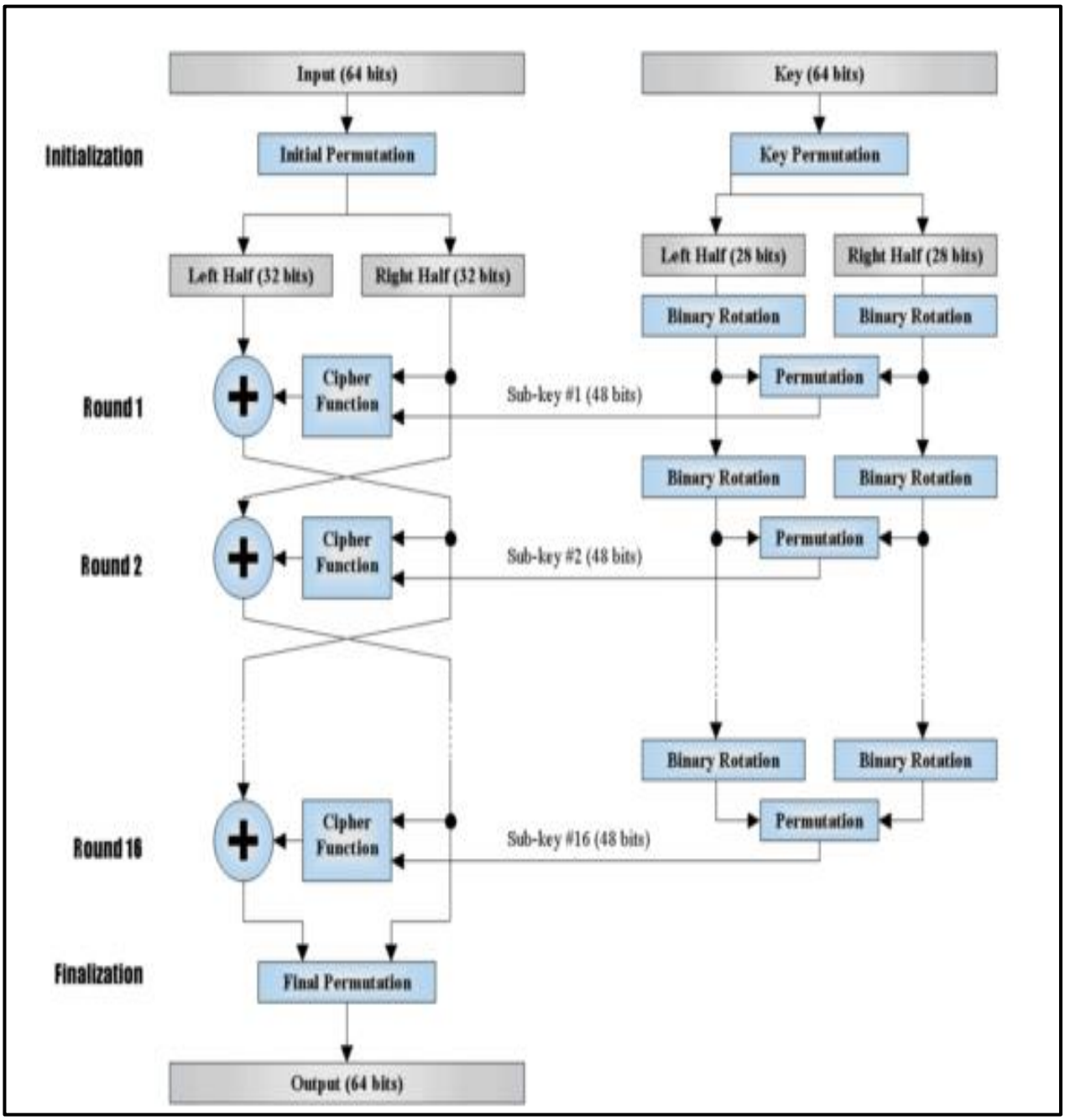
- حجم المفتاح الفعال هو 56 بت (إجمالي 64 بت مع 8 بتات يتم تجاهلها).
- تتكون الخوارزمية من 16 خطوة مع 16 مفتاحًا فرعيًا مكونًا من 48 بت (مفتاح واحد لكل خطوة).
- يعمل DES في عدة أوضاع: (ECB و CBC وCTR ...)
- مخطط عملية DES كما يلي:



الشكل 8 : مخطط عملية التشفير وفك تشفيرDES

[10] نفس المرجع

مخطط عملية التشفير DES بالتفصيل



الشكل 9 : مخطط عملية التشفير DES

الخطوة 1: إنشاء 16 مفتاح فرعي ، طول كل منها 48 بت

1- تحويل المفتاح بالجدول PC-1 الشكل 15

PC-1

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

جدول 3: جدول التحويل PC-1

فسيؤدي ذلك إلى تغيير البت 57 من المفتاح الأصلي إلى البت الأول للمفتاح الجديد و البت 49 الى البت الثاني في المفتاح الجديد وهذا ينطبق على كل الجداول التالية

2- قسّم هذا المفتاح إلى الأجزاء اليمنى واليسرى Left Half(28bits)0 و Right Half(28bit)0 نمر عليهم في 16 لينتج لدينا 16 مفتاحا فرعيا

3- تحويل الى اليسار Left Shifts بستعمال الجدول التالي :

Round Number i:	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Nombre of left Shift :	1 1 2 2 2 2 2 2 1 2 2 2 2 2 2 1

جدول 4: الازاحة الى اليسار

الان عندنا 16 مفتاح مختلف

4- نجمع كل زوج من مفاتيح $Left\ Half(28bits)_i$ و $Right\ Half(28bit)_i$ ثم نقوم بإجراء **التحويل** وفقًا لجدول:PC-2

PC-2

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

جدول 5: التحويل ب PC-2

يحتوي كل زوج من المفاتيح الفرعية على 56 بت ، لكن PC−2 يستخدم 48 منها فقط

الان انتهينا من تهيئة المفاتيح(48bit) sub_key #i [1..16]=i

الخطوة 2: تشفير كل قطعة (bloc) 64 بت من البياناتM

1- تحويل البيانات M في النظام الثنائي Binary

2- تحويل البيانات بالجدول IP (*Initial permutation*) الشكل 17

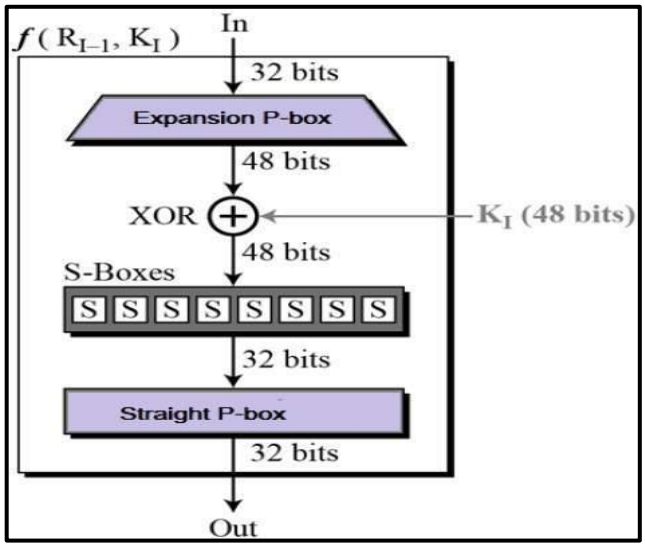
IP							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

جدول 6: التحويل IP

3- الناتج من تحويل IP يتقسم الى نصفين Left Helf (32bit) ,Right Helf(32bit)

1. Left Helf $_n$ = Right Helf $_{n-1}$
2. Right Helf $_n$ = Left Helf $_{n-1}$ + ***f***(Right Helf $_{n-1}$,K_n)

الدالة ***f()*** cipher function:



الشكل 11 : رسم تخطيطي يمثا الدالة Cipher Function

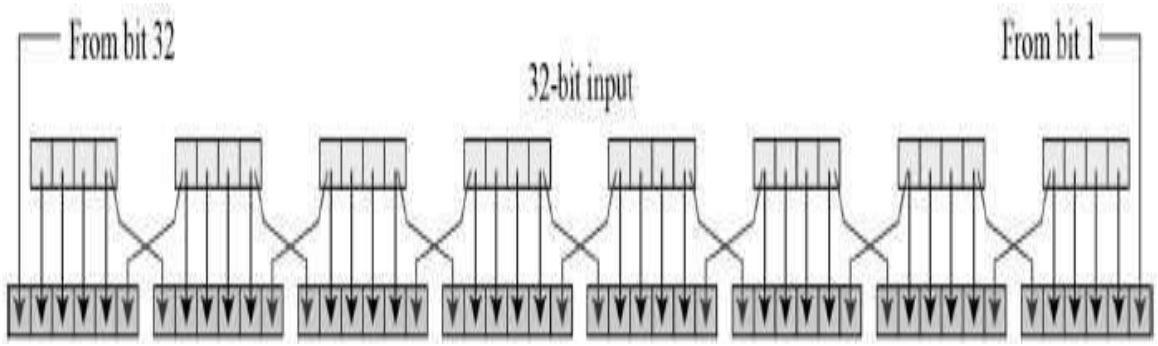
4- الجدول P-BOX يمدد Right Helf(32bit) من 32 bit الى 48bit

sub_key = input_SBoxes XOR **ناتج تمديد P-BOX**

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

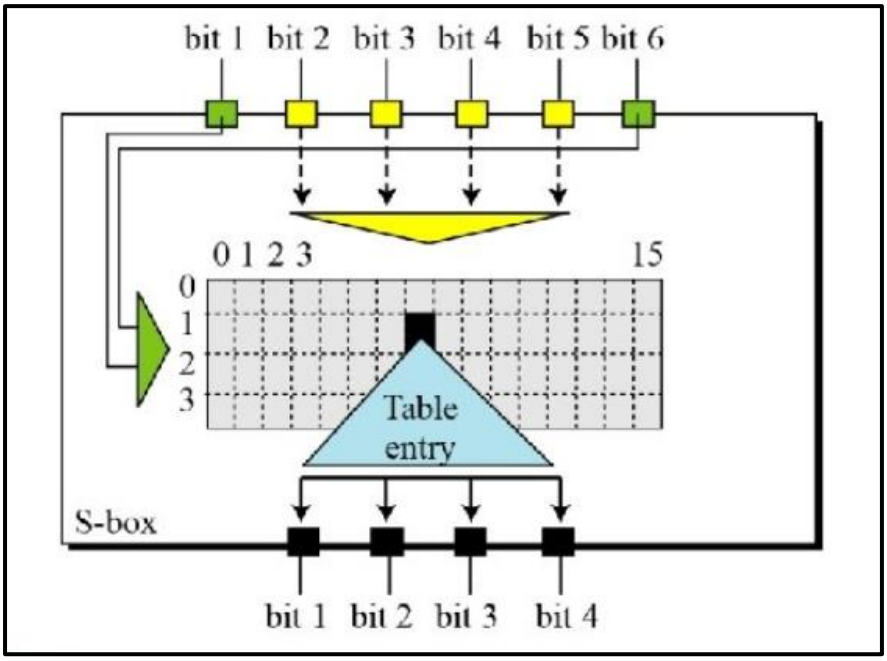
جدول7 : جدول التمديد P-BOX

أو بشكل اوضح الشكل 12 :



الشكل 12 : رسم تخطيطي يمثل عملية التمديد

5- الناتج 8 أجزاء (bloc) كل جزء طوله 6bit تمر على SBox الخاص بها كما في الشكل 13



الشكل 13 : ألية عملSBox

S_1	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
S_2	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S_3	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S_4	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14
S_5	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
S_6	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S_7	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
S_8	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

جدول 8 :SBox

لفهم الية عمل cipher function بتفصيل تجدونها في الموقع www.tutorialspoint.com

6- تتكرر $f()$ مع كل $sub-key$ 16 مرة في الأخير ينتج لنا $Left\ Helf(32bit)_{16}$ و $Right\ Helf(32bit)_{16}$ نقلب بينهما للحصول على:

$$Bloc\ (64bit) = Right\ Helf(32bit)_{16} + Left\ Helf(32bit)_{16}$$

7- ثم اجراء التحويل النهائي Final Permutation حسب الجدول IP^{-1}

IP-1

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

جدول 9 : جدول التحويل النهائي IP^{-1}

بعد اجراء التحويلات ينتج لنا النص المشفر output (64bit)

فك التشفير هو عملية التشفير العكسي ، يتم تنفيذ الخطوات المذكورة أعلاه ، ولكن في التكرارات الـ 16 ، يتم عكس وضعي المفاتيح الفرعية اليمنى واليسرى.

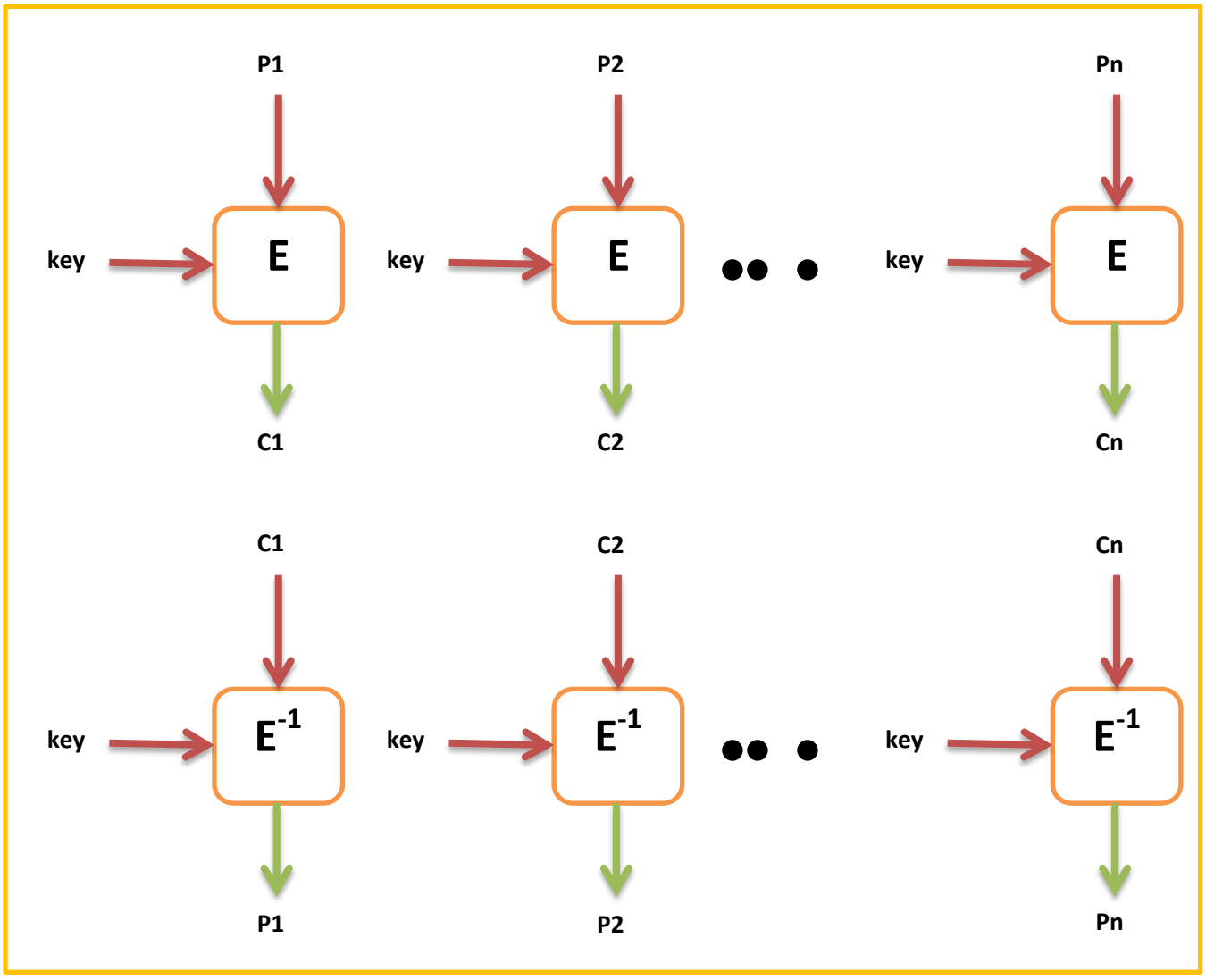
5) أوضاع تشغيل DES

1) *ECB (Electronic Code Book)*:

إنه أبسط طريقة لتشفير النص ويتم دمج المفتاح هناك لتشكيل تشفير النص ، أي إدخال متطابق ينتج دائمًا إخراجًا متطابقًا.

دالة التشفير **E** ، دالة فك التشفير $\mathbf{E^{-1}}$ ، النص الواضح **P** ، النص المشفير **C** ، المفتاح **key**

2) *رسم تخطيطي لألية تشفير و فك التشفير* ***ECB***

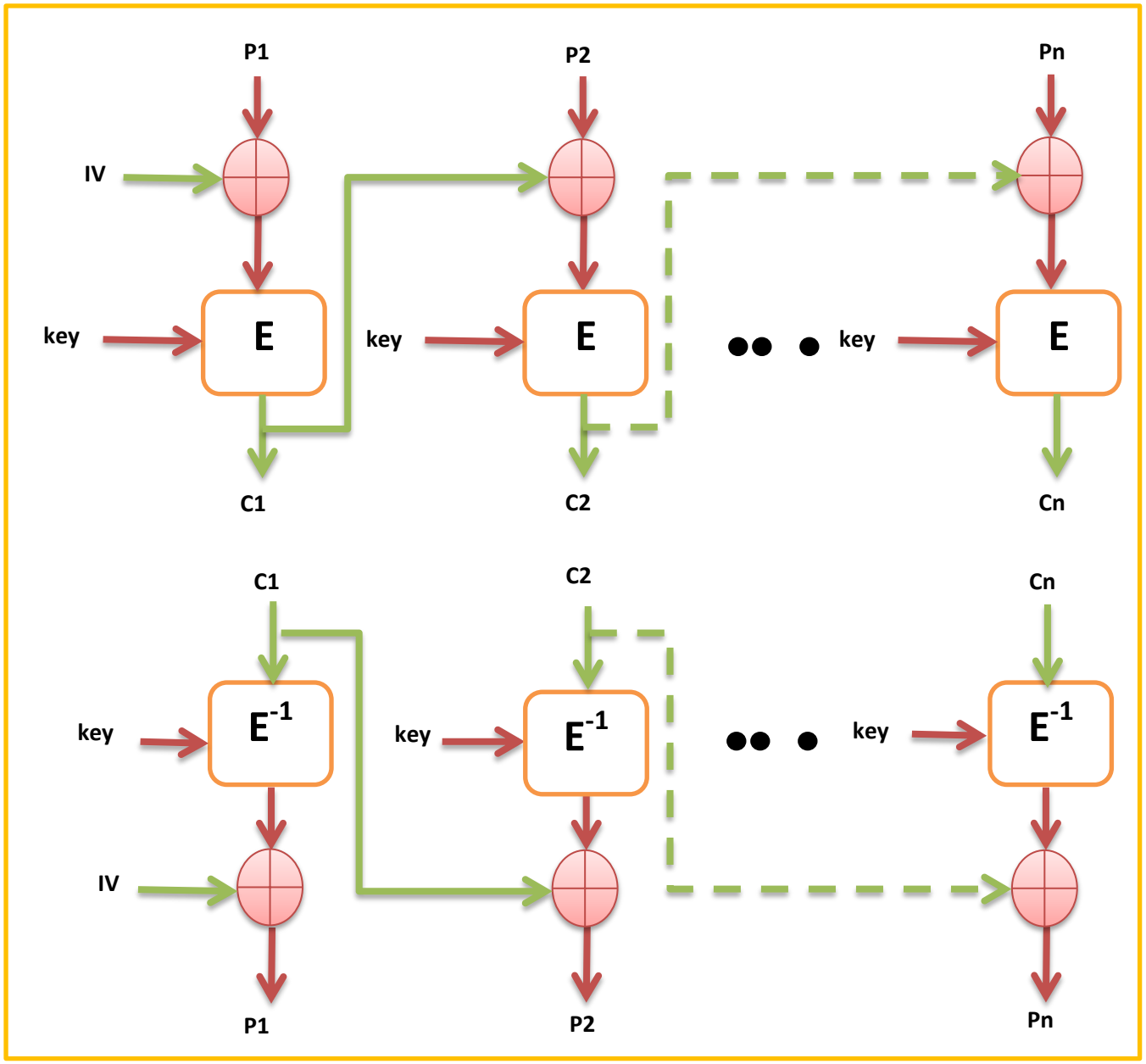


الشكل 14 : ألية التشفير و فك تشفير ECB

3) ***CBC (Cipher Block Chaining)*** :

في هذا الوضع ، يتم تشفير كل كتلة كـ ECB ، ولكن يتم إضافة عامل ثالث مأخوذ من الإدخال السابق.
في هذه الحالة ، لا ينتج الإدخال المتطابق (نص عادي) إخراجًا متطابقًا

4) ***رسم تخطيطي لألية تشفير و فك التشفير CBC***

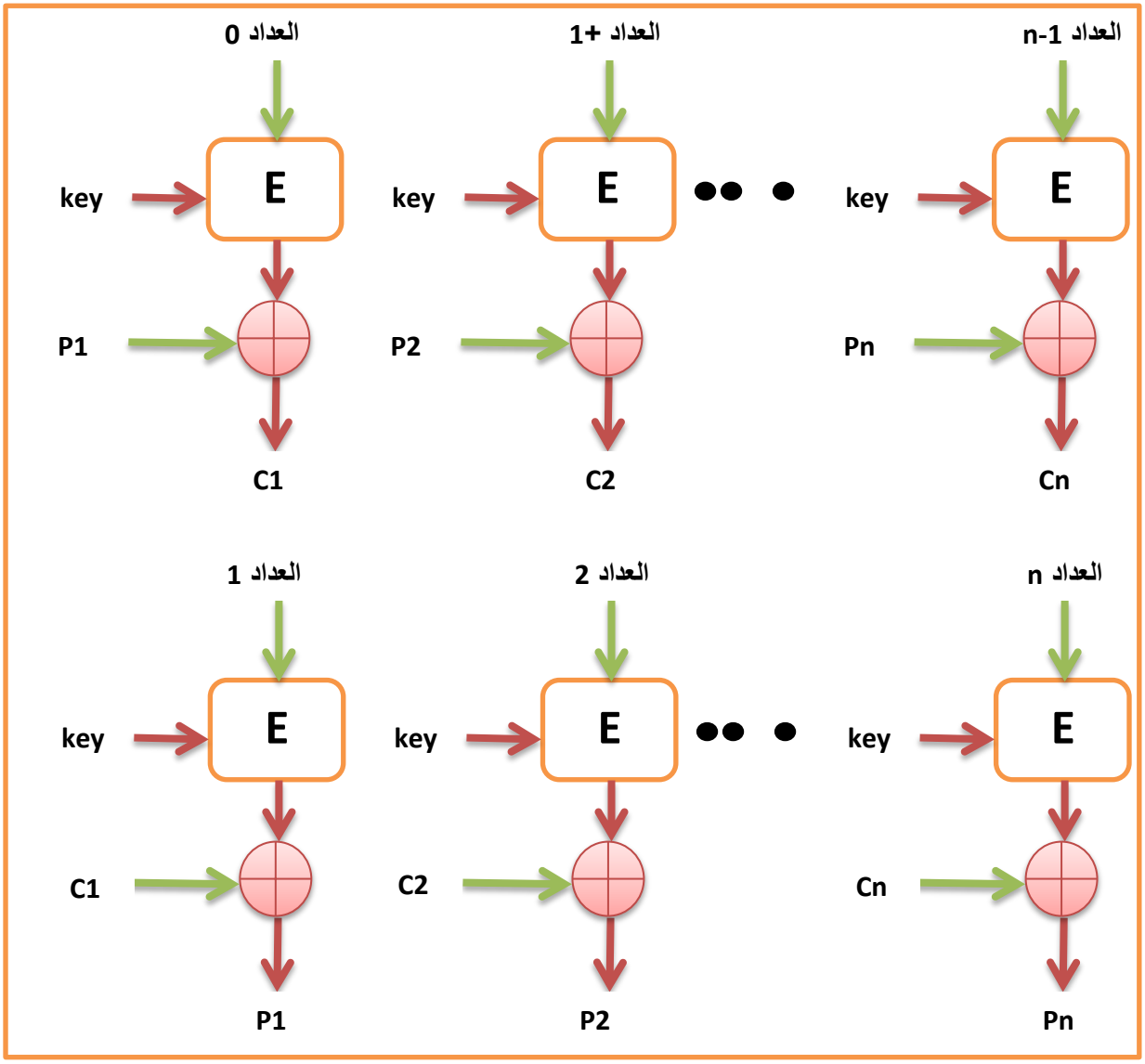


الشكل 15 : رسم تخطيطي لألية تشفير و فك التشفيرCBC

5) ***Counter (CTR)***

يتم ترقيم الكتل بالتسلسل ، ثم يتم دمج رقم الكتلة هذا مع المفتاح keyويتم تشفيره باستخدام تشفير كتلة E ، نتيجة هذا التشفير مع النص العادي لإنتاج النص المشفر عن طريق XOR.

6) *رسم تخطيطي لألية تشفير و فك التشفير* ***CTR***



الشكل 16 : رسم تخطيطي لألية تشفير و فك التشفيرCTR

تم االإستعانة في الشرح

1- بكتاب The DES Algorithm Illustrated by J. Orlin Grabbe

2- مدونة **المبرمج العربي**

3- موقع https://www.tutorialspoint.com

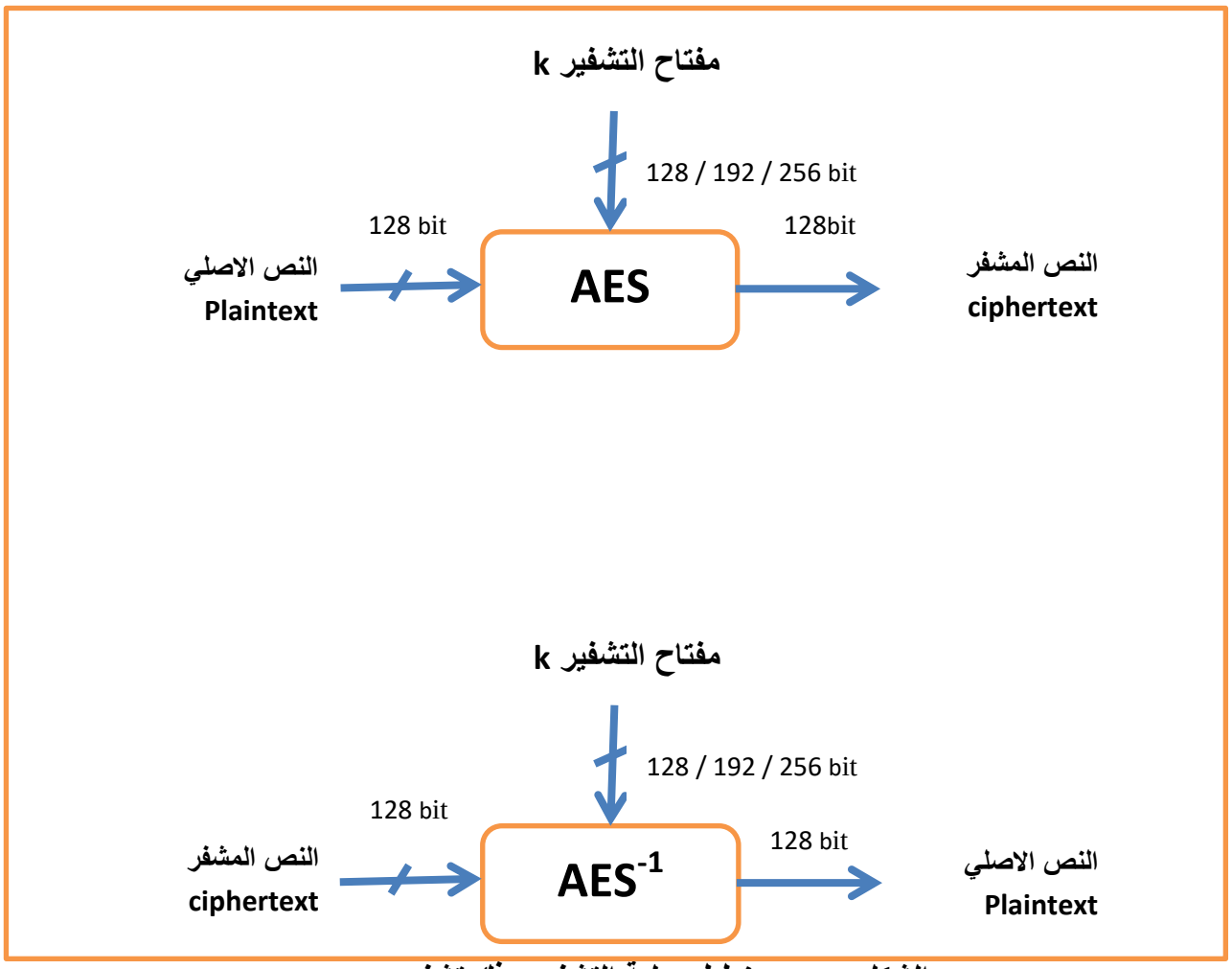
4- DATA ENCRYPTION STANDARD DES Rajdeep Shaktawat Information Security

6) معيار التشفير المتقدم AES(Advanced Encryption Standard)

معيار التشفير المتقدم (AES) هو مواصفة لتشفير البيانات الإلكترونية أنشأها المعهد الوطني الأمريكي للمعايير والتكنولوجيا (NIST) في عام 2001. يستخدم AES على نطاق واسع اليوم لأنه أقوى بكثير من DES على الرغم من كونه صعب تنفيذ[11]

- حجم المفتاح هو 128bit أو 192bit أو 256bit
- يقوم بتشفير البيانات في كتل من 128 bit .

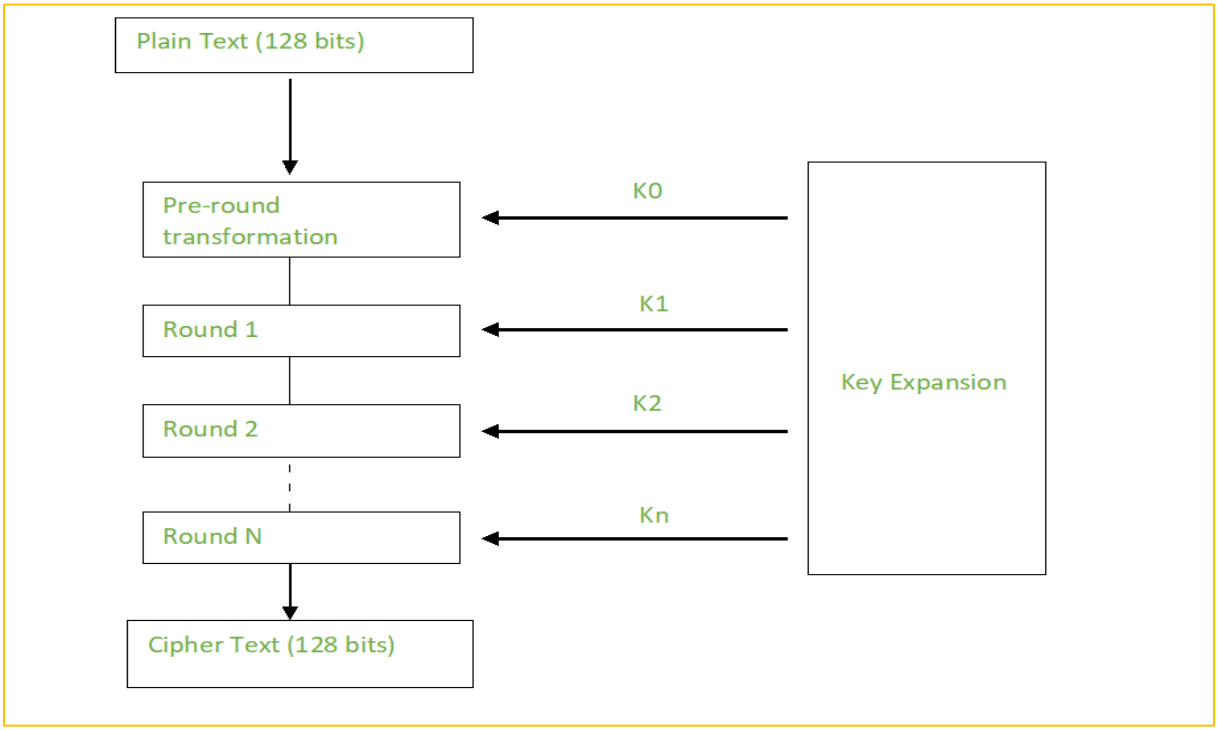
الشكل 18 يبسط آلية العمل AES، حيث Plaintext هنا هي البيانات قبل التشفير و ciphertext هي البيانات المشفرة.



الشكل 17 : مخطط عملية التشفير وفك تشفير AES

[11] https://www.geeksforgeeks.org/advanced-encryption-standard-aes/

مخطط عام الشكل 18 **AES** :



الشكل 18 : مخطط AES

ينفذ AES العمليات على bytes البيانات. نظرًا لأن حجم الكتلة هو 128bit ، فإن التشفير يعالج 128bit (16bytes) من بيانات الإدخال في المرة الواحدة.

1-يعتمد عدد جولات (RoundKey) على طول المفتاح

Key= 128bit –> 10 rounds

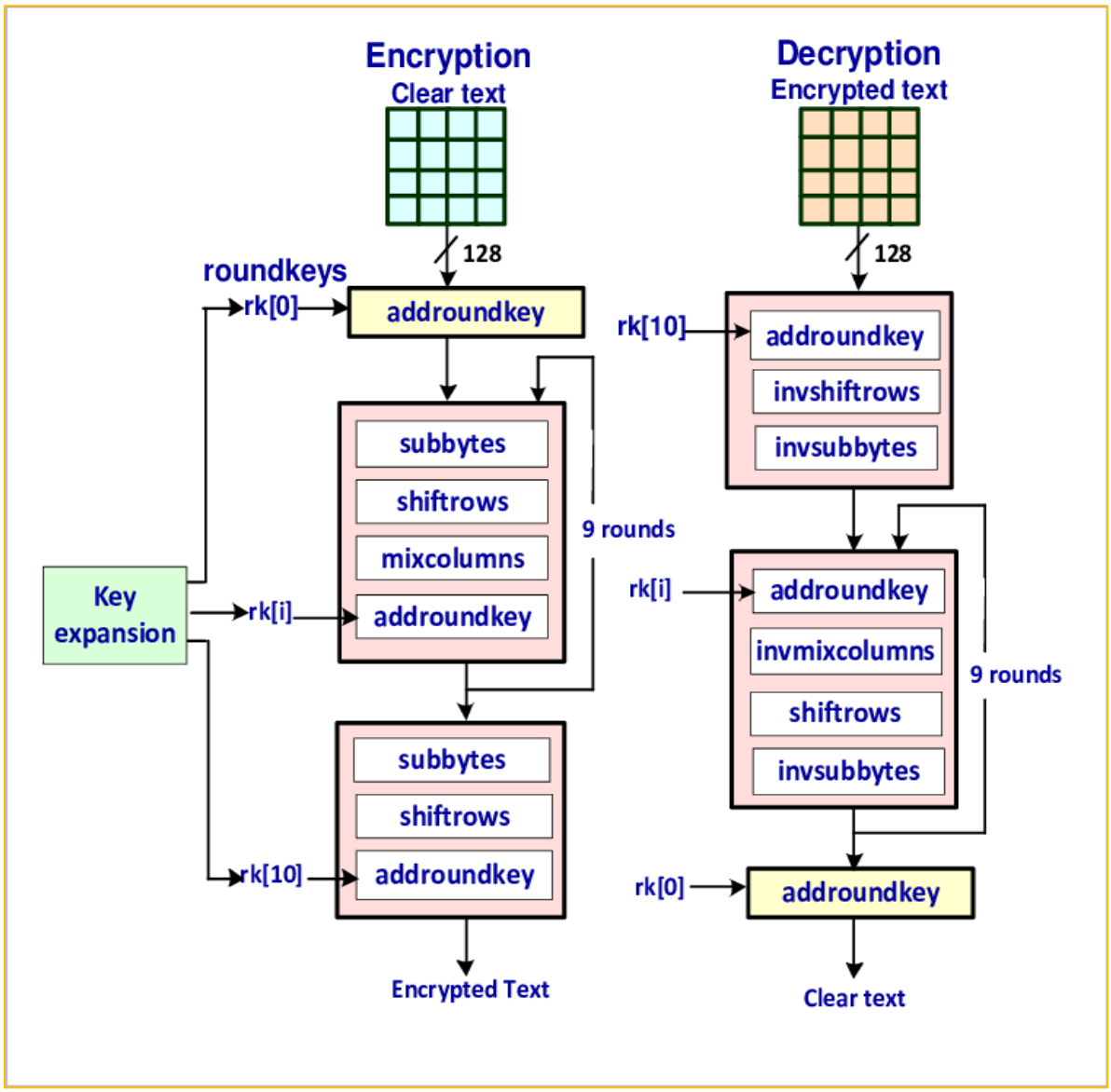
Key= 192bit –> 12 rounds

Key= 256 bit –>14 rounds

مخطط عملية التشفير وفك التشفير (Key = 128bit)AES:

تبدأ عملية التشفير AES (الشكل 20) بإضافة roundkey(0) تليها تسعة دورات من 4 عمليات و الدورة 10 من 3 عمليات . أما فك التشفير هي **معكوس** نظيرتها في خوارزمية التشفير. العمليات الأربع هي كما يلي:

- ✓ ADD ROUND KEY
- ✓ SUB BYTE
- ✓ SHIFT ROW
- ✓ MIX COLUMN



الشكل 19 : مخطط عملية التشفير AES

قبل بدء الخوارزمية يجب تحويل كل من النص الواضح plaintext الى مصفوفة 4*4

P1	**P2**	**P3**	**P4**
P5	**P6**	**P7**	**P8**
P9	**P10**	**P11**	**P12**
P13	**P14**	**P15**	**P16**

جدول 10 : State

1) خوارزمية توسيع المفتاح Key Expansion :

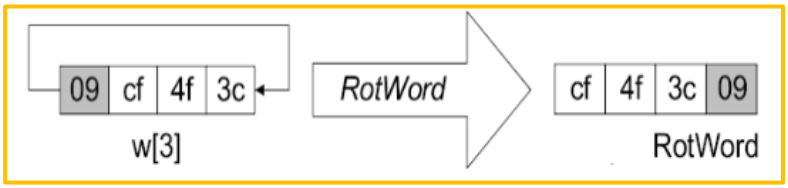
تحويل المفتاح الى مصفوفة 4*4 وينتج من هذا الاخير (مصفوفة) 11 مفتاح(k) أي 44 كلمة (W)

في هذه الخوارزمية لإيجاد المفتاح الموسع W[i] نتبع الخطوات التالية:

أ. إذا كان i مضاعفات 4 ($i=4n, n \in N^*$)

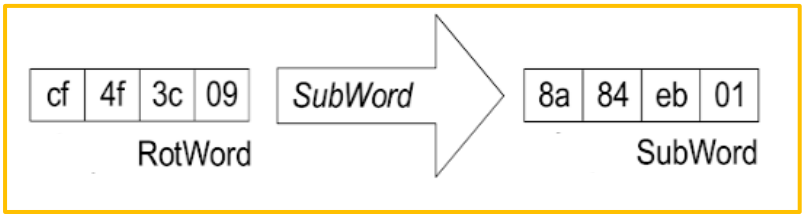
K[n] : W[0] = K[n−1] : W[0] **XOR** SubWord(RotWord(K[n−1] : W[3])) **XOR** Rcon[i]

- RotWord: إزاحة الكلمة (K[n−1] :W[3]) الى اليسار ب 1byte



الشكل 20 : تنفيذ RotWord

- SubByte: نستبدل كل byte من الكلمة السابقة (RotWord) بما يقابلها في جدول S-Box



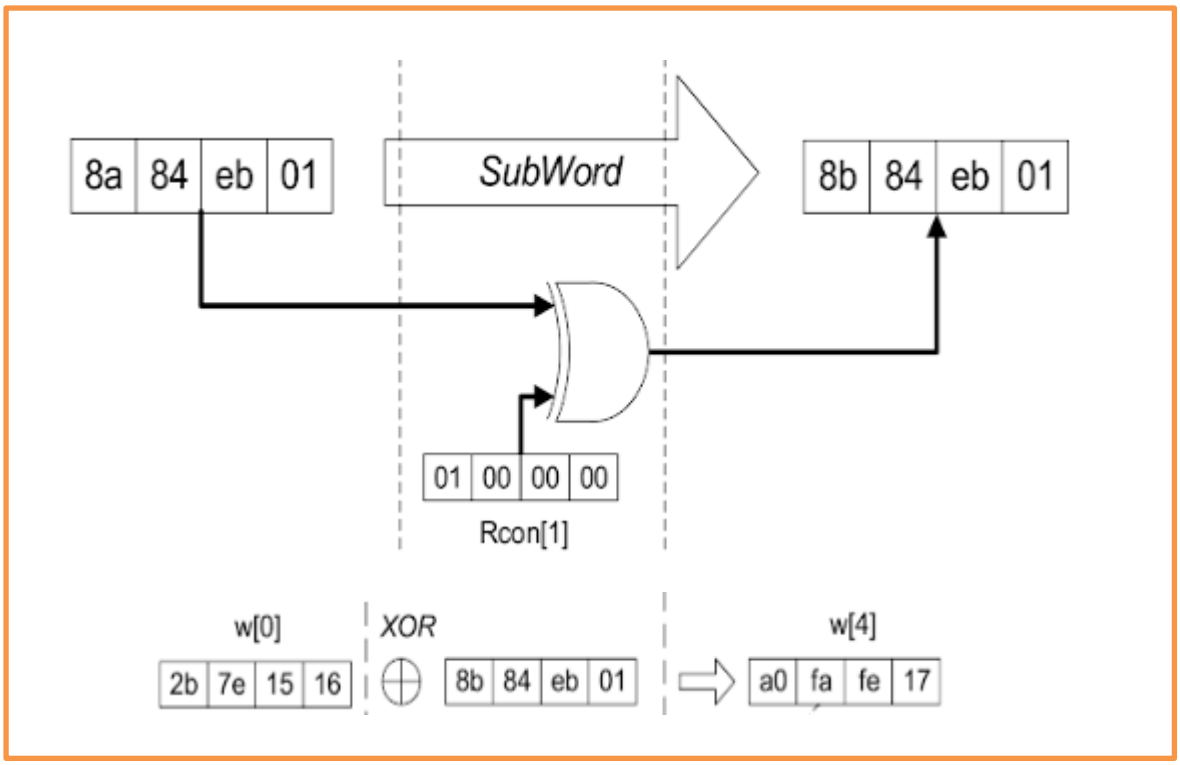
الشكل 21 : تنفيذ SubWord

- نجمع SubWord (XOR) مع Rcon[i] مع الكلمة K[n−1] : W[0] وفق العملية الثنائية XOR . و Rcon يختلف من اجل كل دورة وفق الجدول 11 :

i	1	2	3	4	5	6	7	8	9	10
RC[i]	01	02	04	08	10	20	40	80	1b	36

جدول 11: Rcon

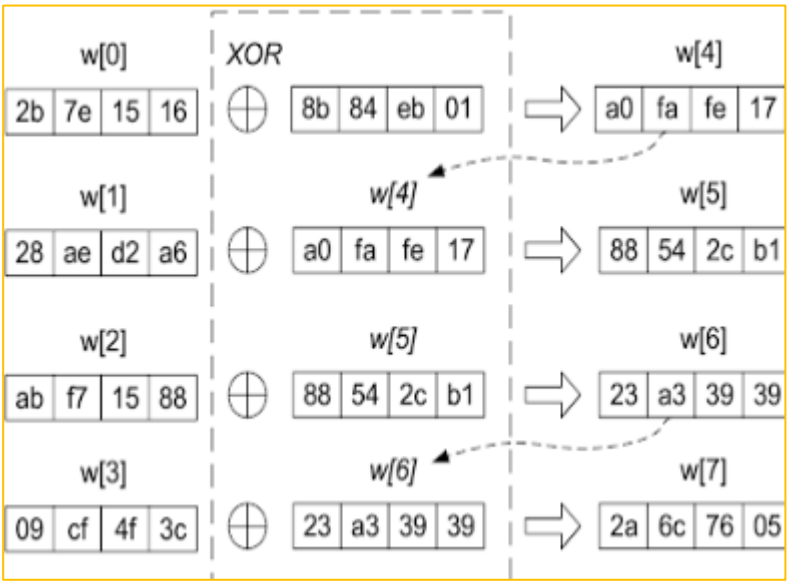
و الشكل التالي يوضح العملية الثنائية XOR :



الشكل 22 : حساب الكلمة الأولى للمفتاح الأول

ب. إذا كان i ليس من مضاعفات 4 ($i=4n+m$, $m \in \{1,2,3\}$, $n \in N^*$)

K[n] : W[i] = K[n -1]: W[i] XOR K[n]: W[i - 1]

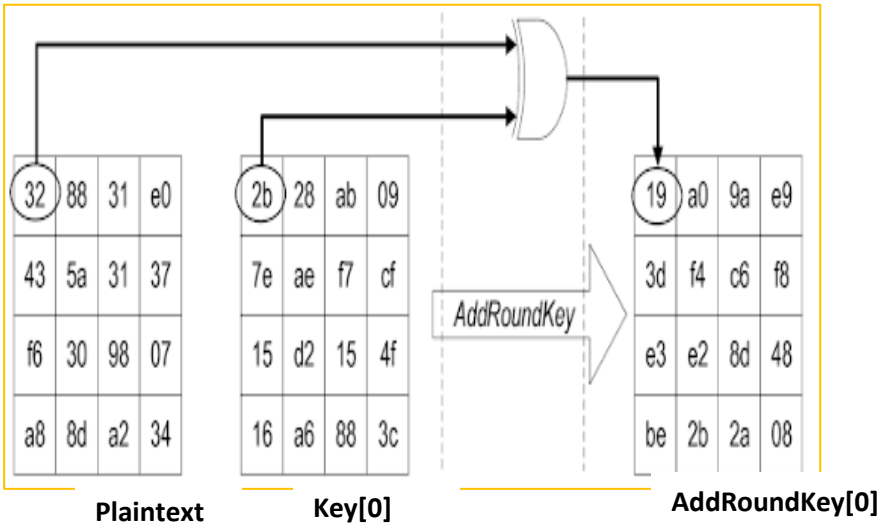


الشكل 23 : حساب مفتاح الحلقة الاولى

2) **خورزمية AES** : تتالف من قسمين رئسيين هما التشفير **Encryption** و فك التشفير **Decryption**

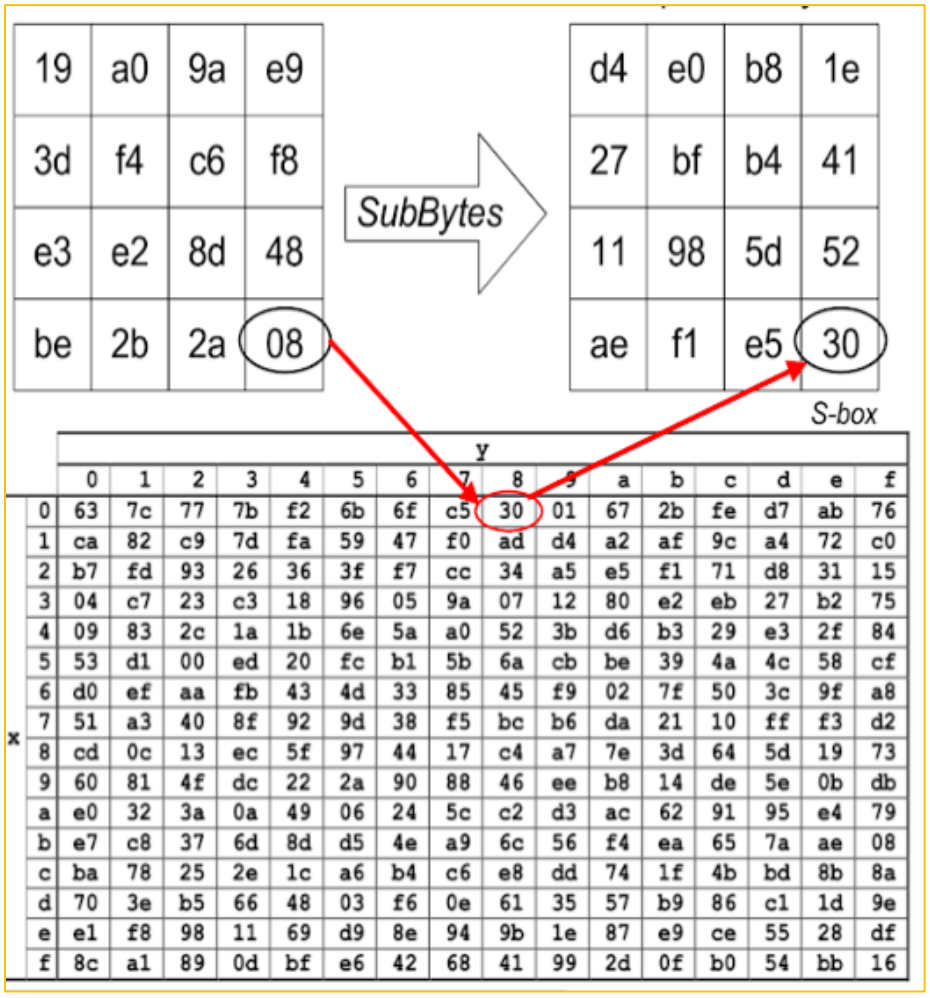
أ. **التشفير Encryption:**

- AddRoundKey : إضافة مفتاح التشفير الى النص بواسطة العملية الثنائية XOR الشكل 24



الشكل 24 : AddRoundKey[0]

- SubByte : نأخذ كل بايت من AddRoundKey ونبحث عن البديل في جدول S–box الشكل 25



x \ y	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

الشكل 25 : تنفيذ SubByte

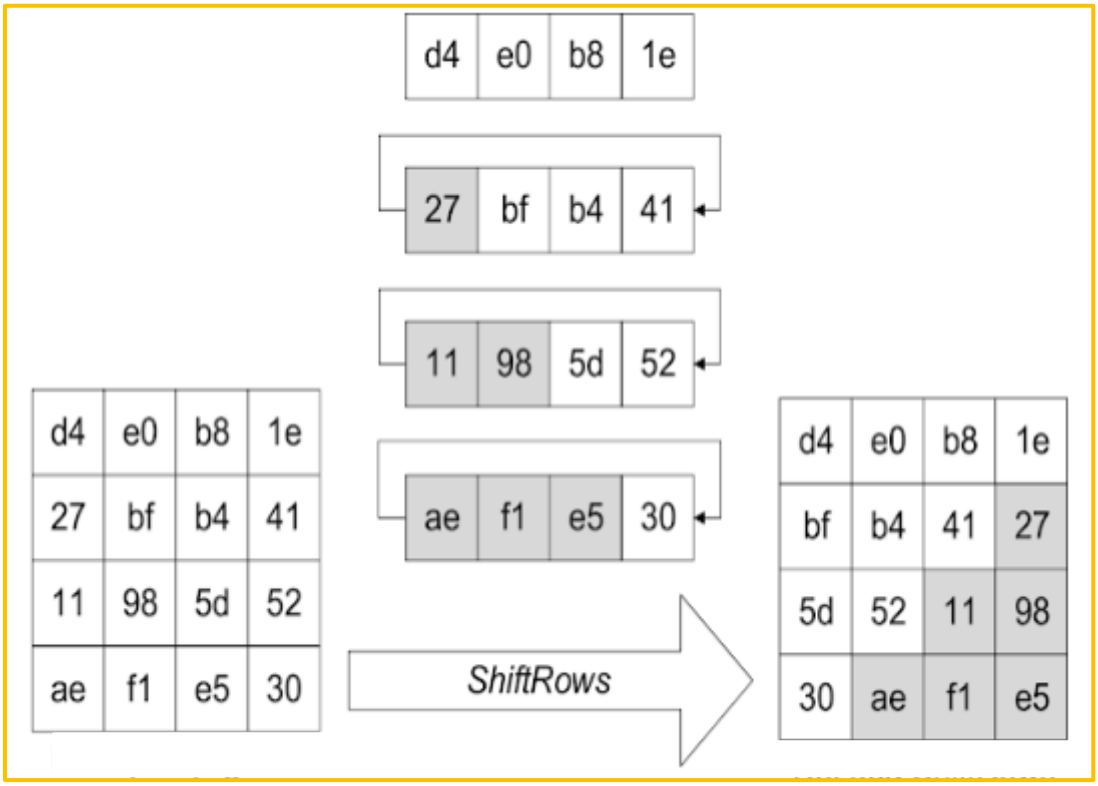
- Shift Row: هذه الخطوة هي تبديل بسيط (إزاحة دائرية) الشكل 26. عملها هو:

– لم يتم تعديل صف الحالة الأول.

– يتم إزاحة السطر الثاني بمقدار 1 بايت إلى اليسار بطريقة دائرية.

– تم إزاحة الصف الثالث إلى اليسار بمقدار 2 بايت بطريقة دائرية

– تم إزاحة الصف الرابع إلى اليسار بمقدار 3 بايت بطريقة دائرية



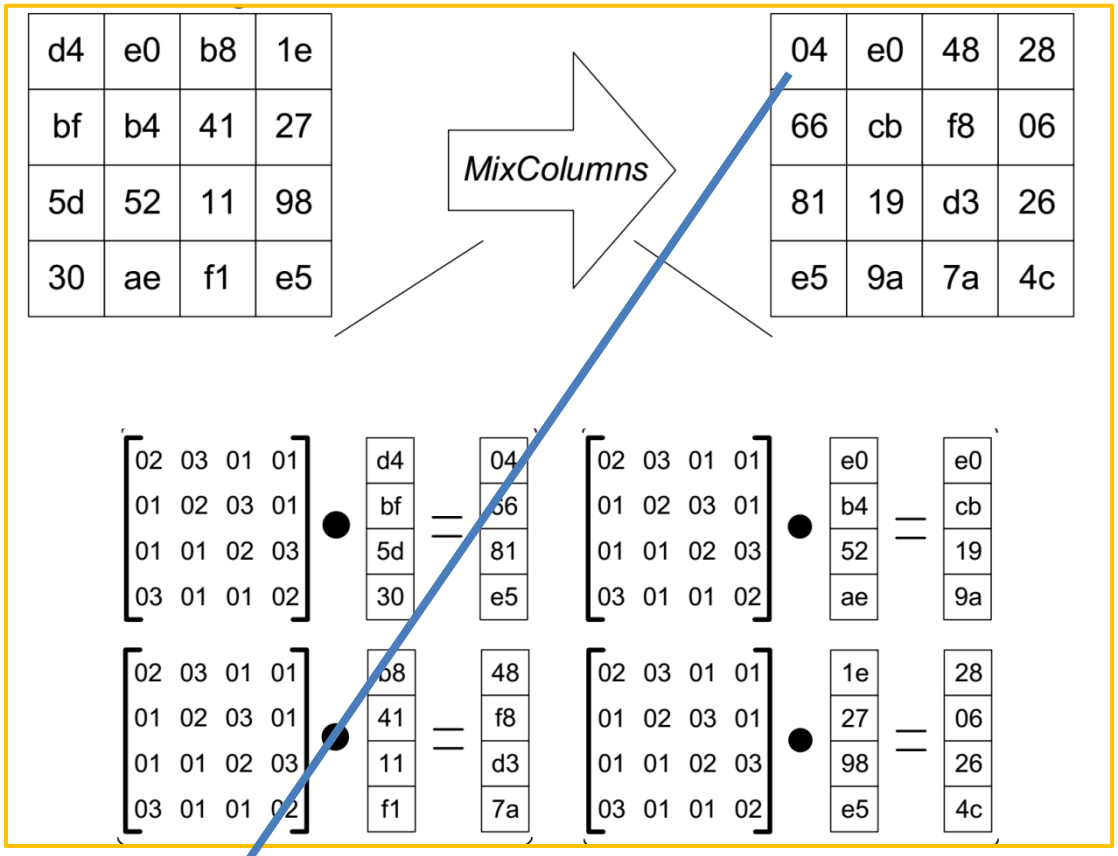
الشكل 26 : تنفيذ shiftRows

- **MixColumns:** تقوم الدالة بضرب كل عمود من مصفوفة ShiftRows مع مصفوفة تحويل محددة بواسطة معيار AES الجدول 12، باستخدام $GF(2^8)$ كثير الحدود غير القابل للاختزال إذا كان الضرب ينتج عنه كثير حدود من الدرجة أكبر من 7 ، فإن كثير الحدود يتم تقليله أي نضيف m(x) ونحتفظ بالباقي. بالنسبة إلى كثير الحدود. الشكل 27

$$m(x) = x8 + x4 + x3 + x + 1 \ (11B)$$

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

جدول 13: مصفوفة التحويل المستخدمة في دالة MixColumns



الشكل 27 : تنفيذMixColumns

$$04 = d4.02 + bf.03 + 5d.01 + 30.01 = d4.02 + (bf.02 + bf.01) + 5d.01 + 30.01$$

Bf.03

ب. فك التشفير Decryption :

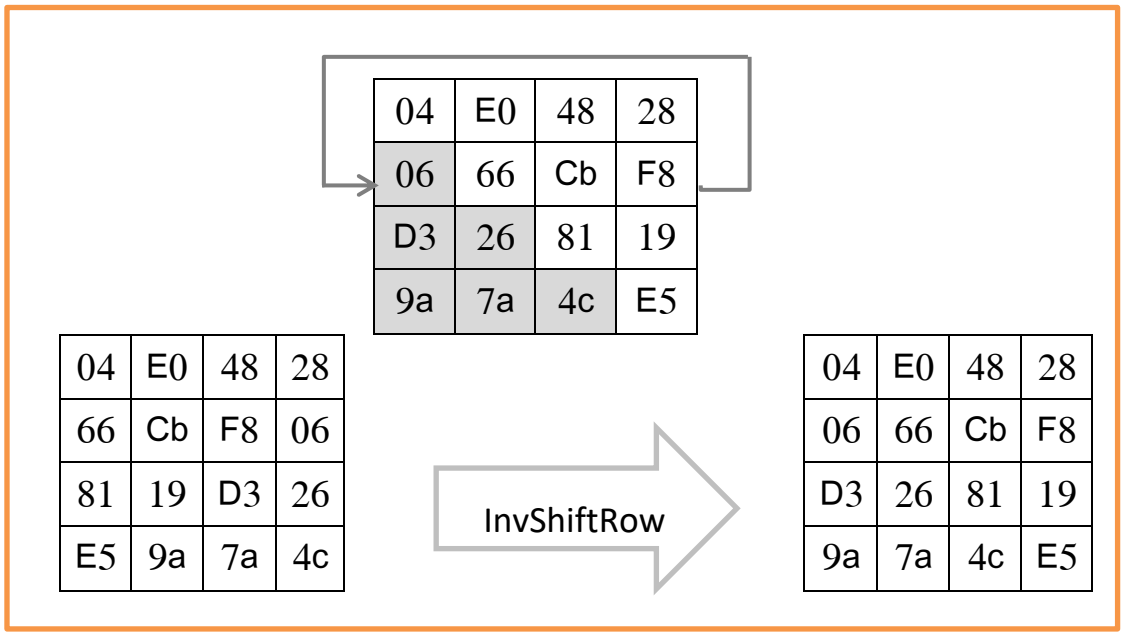
- AddRoundKey : إضافة مفتاح التشفير (key[10]) الى النص المشفر بواسطة العملية الثنائية XOR الشكل 24
- InvShiftRow : هذه الخطوة هي تبديل بسيط (إزاحة دائرية) الشكل 28 عملها هو:

– لم يتم تعديل صف الحالة الأول.

– يتم إزاحة السطر الثاني بمقدار 1 بايت إلى اليمين بطريقة دائرية.

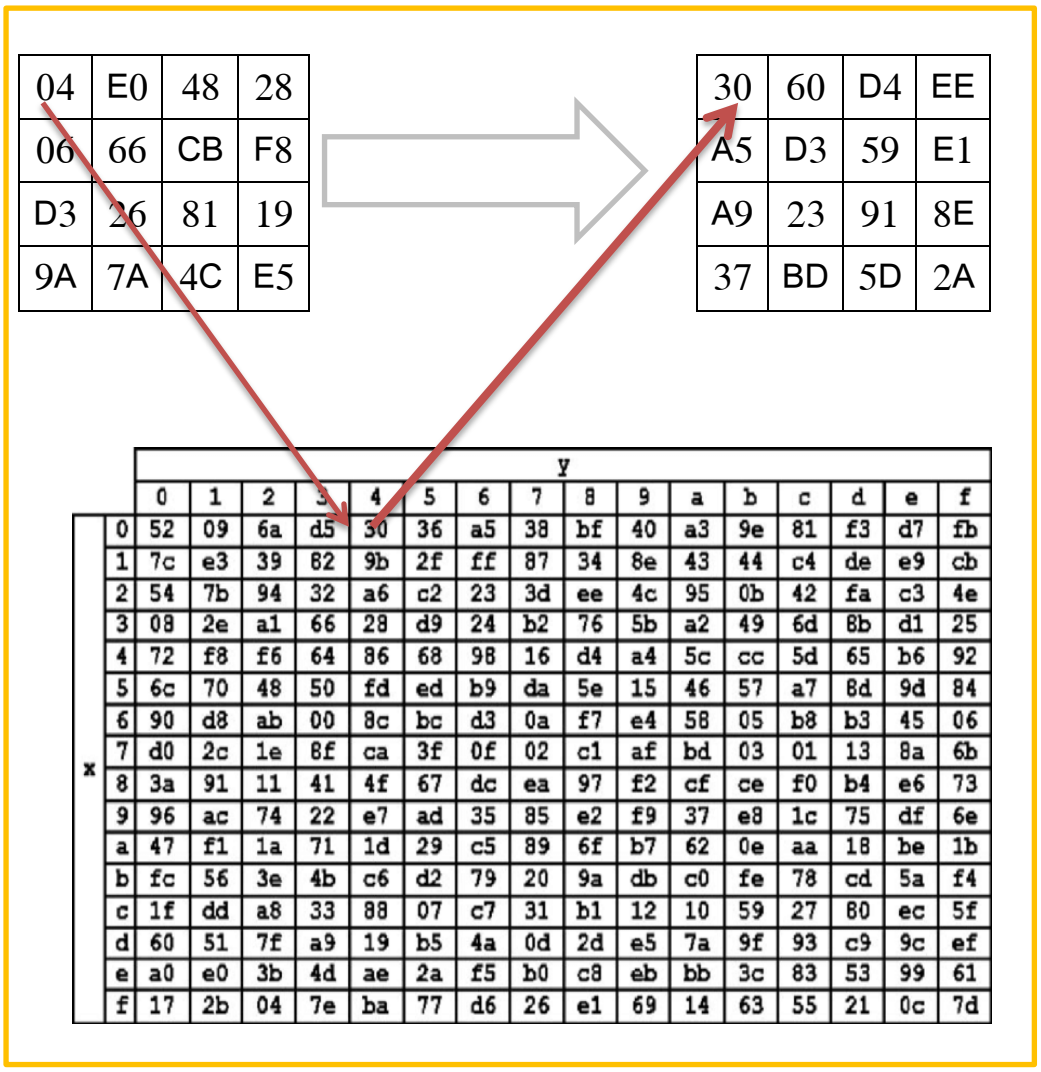
– تم إزاحة الصف الثالث إلى اليمين بمقدار 2 بايت بطريقة دائرية

– تم إزاحة الصف الرابع إلى اليمين بمقدار 3 بايت بطريقة دائرية



الشكل 28 : تنفيذ InvShiftRows

- InvSubBytes: نأخذ كل بايت من InvShiftRow ونبحث عن البديل في جدول InvS-box الشكل 29 :



x \ y	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	52	09	6a	d5	30	36	a5	38	bf	40	a3	9e	81	f3	d7	fb
1	7c	e3	39	82	9b	2f	ff	87	34	8e	43	44	c4	de	e9	cb
2	54	7b	94	32	a6	c2	23	3d	ee	4c	95	0b	42	fa	c3	4e
3	08	2e	a1	66	28	d9	24	b2	76	5b	a2	49	6d	8b	d1	25
4	72	f8	f6	64	86	68	98	16	d4	a4	5c	cc	5d	65	b6	92
5	6c	70	48	50	fd	ed	b9	da	5e	15	46	57	a7	8d	9d	84
6	90	d8	ab	00	8c	bc	d3	0a	f7	e4	58	05	b8	b3	45	06
7	d0	2c	1e	8f	ca	3f	0f	02	c1	af	bd	03	01	13	8a	6b
8	3a	91	11	41	4f	67	dc	ea	97	f2	cf	ce	f0	b4	e6	73
9	96	ac	74	22	e7	ad	35	85	e2	f9	37	e8	1c	75	df	6e
a	47	f1	1a	71	1d	29	c5	89	6f	b7	62	0e	aa	18	be	1b
b	fc	56	3e	4b	c6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
c	1f	dd	a8	33	88	07	c7	31	b1	12	10	59	27	80	ec	5f
d	60	51	7f	a9	19	b5	4a	0d	2d	e5	7a	9f	93	c9	9c	ef
e	a0	e0	3b	4d	ae	2a	f5	b0	c8	eb	bb	3c	83	53	99	61
f	17	2b	04	7e	ba	77	d6	26	e1	69	14	63	55	21	0c	7d

الشكل 29 : تنفيذ InvSubByte

- InvMixColumns : تستخدم مصفوفة ثابتة هي مقلوب مصفوفة التحويل المستخدمة في دالة MixColumns الجدول 13 من اجل ضرب العمود كما هو موضح في الشكل 30 مع إستخدام $GF(2^8)$:

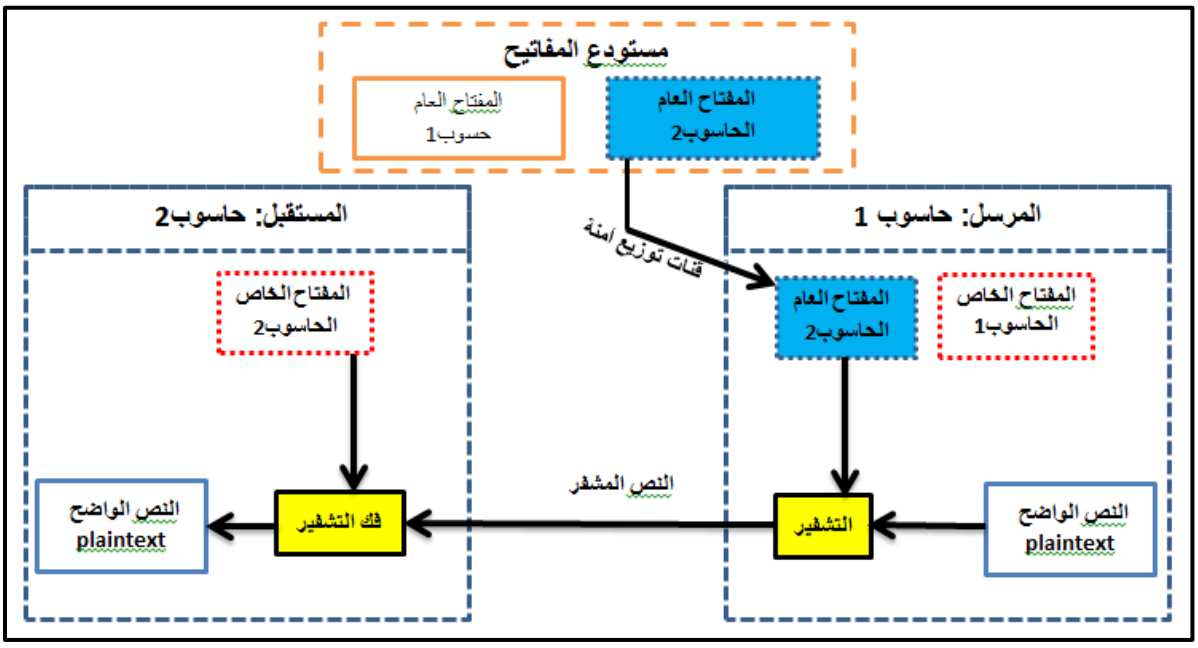
$$\begin{bmatrix} s_{3,c} \\ s_{2,c} \\ s_{1,c} \\ s_{0,c} \end{bmatrix}' = \begin{bmatrix} \{0e\} & \{09\} & \{0d\} & \{0b\} \\ \{0b\} & \{0e\} & \{09\} & \{0d\} \\ \{0d\} & \{0b\} & \{0e\} & \{09\} \\ \{09\} & \{0d\} & \{0b\} & \{0e\} \end{bmatrix} \begin{bmatrix} s_{3,c} \\ s_{2,c} \\ s_{1,c} \\ s_{0,c} \end{bmatrix}$$

الشكل 30 : تنفيذ InvMixColumns

2-2-4-2 مفاتيح التشفير غير المتماثل:Asymmetric Key Encryption

2-2-4-2-1 آلية عمل هذه التقنية :

بعد القيام بتكوين المفتاحين من جهة موثوقة (مستودع المفاتيح)، تقوم بإرسال المفتاح العام لمن تريد المستقبل، المفتاح العام هو من يقوم بتشفير النص الواضح (plaintext) وليس فك التشفير، الطرف الثاني يقوم بتشفير النص الواضح عن طريق استخدام مفتاح العام للمستقبل الذي تم إرساله اليه، بعد ذلك يقوم الطرف المستقبل بإرسال الرسالة المشفرة الى المرسل الأصلي الذي قام بإرسال المفتاح العام له، عند استلام المرسل الرسالة المشفرة ، فإنه يقوم بفك التشفير عن طريق المفتاح الخاص به ، هو فقط من يستطيع فك التشفير



الشكل 31 : مخطط مفاتيح التشفير غير المتماثل

2-2-2-4-2 لتشفير بالمفتاح الغير المتماثل :

1- يحتاج كل شخص إلى الحصول على زوج من المفاتيح المختلفة (مفتاح خاص Private Key ومفتاح عامPublic Key)، حيث ترتبط هذه المفاتيح ببعضها رياضيا، فعند استخدام أحد المفاتيح للتشفير يمكن للآخر فك تشفير النص المشفر مرة أخرى إلى النص الأصلي.

2- يتطلب وضع المفتاح العام Public Key في مستودع عام Public Repository وبالتالي يسمى نظام التشفير هذا أيضًا تشفير المفتاح العام.Public Key Encryption

3- على الرغم من أن المفاتيح العامة والخاصة مرتبطة رياضيا، فإنه ليس من الممكن الحصول على واحد بمعلومية الآخر.

4- عندما يحتاج **الحاسوب 1** إلى إرسال البيانات إلى **الحاسوب 2**، فإنه يحصل على المفتاح العام **للحاسوب 2** من المستودع، ثم يقوم بتشفير البيانات ونقلها، ومنها يستخدم **الحاسوب 2** المفتاح الخاص به للحصول النص العادي.

5- طول المفاتيح (عدد Bits) في هذا التشفير كبير، وبالتالي فإن عملية فك التشفير هي أبطأ من تشفير المفتاح المتماثل، لذا فإن الطاقة المطلوبة لتشغيل الخوارزمية الغير المتماثلة تكون أعلى.

6- أنظمة التشفير المتماثلة هي مفهوم طبيعي يمكن فهمه، ولكن في المقابل يصعب فهم أنظمة التشفير الغير متماثلة لاحتوائها على مفاهيم رياضية معقدة[12].

2-4-2-3 عوائق التشفير بالمفتاح الغير متماثل:

1- يحتاج المستخدم إلى الوثوق بأن المفتاح العام الذي يستخدمه في التواصل مع شخص ما، هو حقًا المفتاح العام لذلك الشخص ولم يتم خداعه من قبل طرف ثالث ضار.

2- يتم تحقيق ذلك عادةً من خلال Public Key Infrastructure PKI تتألف من طرف ثالث موثوق به، حيث يقوم الطرف الثالث بإدارة وتحقيق صحة المفاتيح العامة بشكل آمن، عندما يُطلب من الطرف الثالث توفير المفتاح العام لأي شخص متصلX ، يكون موثوقًا به لتوفير المفتاح العام الصحيح.

3- عادة ما يتم تضمين المفاتيح العامة التي تم التحقق منها شهادة موقعة رقميا Digital Signature Certificateمن قبل جهة خارجية موثوق مفادها أنها تشهد أن مفتاح عام معين ينتمي إلى شخص أو كيان معين فقط[13].

2-4-2-4 خوارزمية تشفير المفتاح العام **RSA**

RSA هي الخوارزمية الأكثر شهرة ، من بين خوارزميات تشفير المفتاح العام ، وهي تأخذ اسمها من أسماء مخترعيها الثلاثة: R. Rivest و A. Shamir و L. Adleman. تم اختراعه في عام 1976 ، وهو أول بروتوكول تشفير للمفتاح العام. يوفر أمانًا عاليًا ويستند إلى سلسلة من العمليات الحسابية في شكل أسي معياري (modulo N) على أعداد أولية كبيرة. RSA هو اليوم نظام عالمي يخدم في العديد من التطبيقات.

يتم استخدامه في المعاملات الآمنة على الإنترنت ، ويتم تنفيذه أيضًا في العديد من معايير تكنولوجيا المعلومات مثل معايير المجتمع الدولي للاتصالات المالية بين البنوك أو مسودة معيار X9.44 للصناعة

[12] Taha ALMAHMUDI ، (2020، 10 ،16) ، Symmetric VS Asymmetric Encryption ،(2022، 7 ، 10) ، https://www.linkedin.com/pulse/%D8%A7%D9%84%D8%AA%D8%B4%D9%81%D9%8A%D8%B1-%D8%A7%D9%84%D9%85%D8%AA%D9%85%D8%A7%D8%AB%D9%84-symmetric-encryption-%D9%88%D8%A7%D9%84%D8%AA%D8%B4%D9%81%D9%8A%D8%B1-%D8%A7%D9%84%D8%BA%D9%8A%D8%B1-%D9%85%D8%AA%D9%85%D8%A7%D8%AB%D9%84-taha?trk=public_profile_article_view

[13] نفس المرجع

المصرفية الأمريكية. تستمد أمانها من مشكلة العوامل ومبدأها كما يلي: لدينا زوج من المفاتيح ، أحدهما عام (N، e) والآخر خاص (N، d). لتوليد هذه المفاتيح من الضروري تنفيذ الخطوات التالية [14]

1- اختيار عددين أوَليين عشوائيّن كبيرين مختلفين **p** و **q** و حساب **N=p * q**
2- حساب **Φ(n) = (p-1) * (q-1)**
3- اختيار **e** حيث **1=PGCD(e,Φ(N)) ,1<e<Φ(N)** .
4- حساب **d** بحيث :**(1<d<Φ(N))** **e * d mod Φ(N) = 1** .

شكل المفتاح العام **(N,e)**

شكل المفتاح الخاص **(N,d)**

2.4.2.2.4.1 تشفير الرسائل RSA :

لنفرض أن A و Bيريدان أن يتواصلا فيما بينهما. لنفرض أَنَّ مفتاح A العمومي هو (N_A,e_A) أما المفتاح الخصوصي هو (n_A,d_A) ومفتاح B العمومي هو (N_B,e_B) والمفتاح الخصوصي. (N_B,d_B)

لنفرض أنَّ A يريد أن يرسل رسالة إلى , B لذا عليه فعل التالي:

1 يحصل على المفتاح العام للمستقبل B والذي هو (N_B,e_B)).
2 تحويل الرسالة M الى شكل يتوافق مع الحساب نظام أسكي ASCII
3 ناتج التشفير C لهذا الرقم هو$C=M^{(e_B)} \pmod{N_B}$
4 يُرسِلِ C إلىB.

2.4.2.2.4.2 فك تشفير الرسائل RSA:

ليحصل B على الرسالة في شكلها العادي plaintext يستخدم مفتاحه الخاص (N_B,d_B) ويحسب $M=C^{(d_B)} \pmod{N_B}$

M الناتج هو الرسالة التي ارسلها A الى B

[14] Kernouf Y, Zerrouki C ، Simulation de quelques attaques sur le cryptosystème RSA ،(10، 2022،7)
https://www.ummto.dz/dspace/bitstream/handle/ummto/13144/Kernouf%20Y%2C%20Zerrouki%20C..pdf?sequence=1&isAllowed=y

2.4.2.2.4.3 مثال تشفير RSA:

توليد المفاتيح :

1- اختيار عددين أوَليين عشوائيين كبيرين مختلفين **p =11** و **q=13**

وحساب N: **N=p * q=11*13=143**

2- حساب Φ: **Φ(n) = (p-1) * (q-1)=10 * 12 =120**

3- اختيار **e** حيث

. 1<e=7<Φ(N)=120 ,PGCD(e,Φ(N))= PGCD(7,120) =1 ↔e=7

4- حساب **d** بحيث: **(1<d<Φ(N))** . **e * d mod Φ(N) = 1**

e * d mod Φ(N) = 1 ↔ 7 * d = 1 mod 120 ↔ d=103
7*103 = 1 mod 120 ↔721=120*6+1

اذا المفتاح العام : (N,e)=(143,7)

المفتاح الخاص :(N,d)=(143,103)

- تشفير الرسائل $C=M^e \bmod N = M^{103} \bmod 143$
- فك تشفير الرسائل : $M=C^d \bmod N=C^7 \bmod 143$

بحيث الرسالة الاصلية M و الرسالة المشفرة C

2-4-3 التجزئة(Hash) :

التجزئة (Hash)هي من المواضيع التي تستخدم بكثرة في هذا الوقت في المجال امن المعلومات

2-4-3-1 تعريف وظائف التجزئة (Hash)

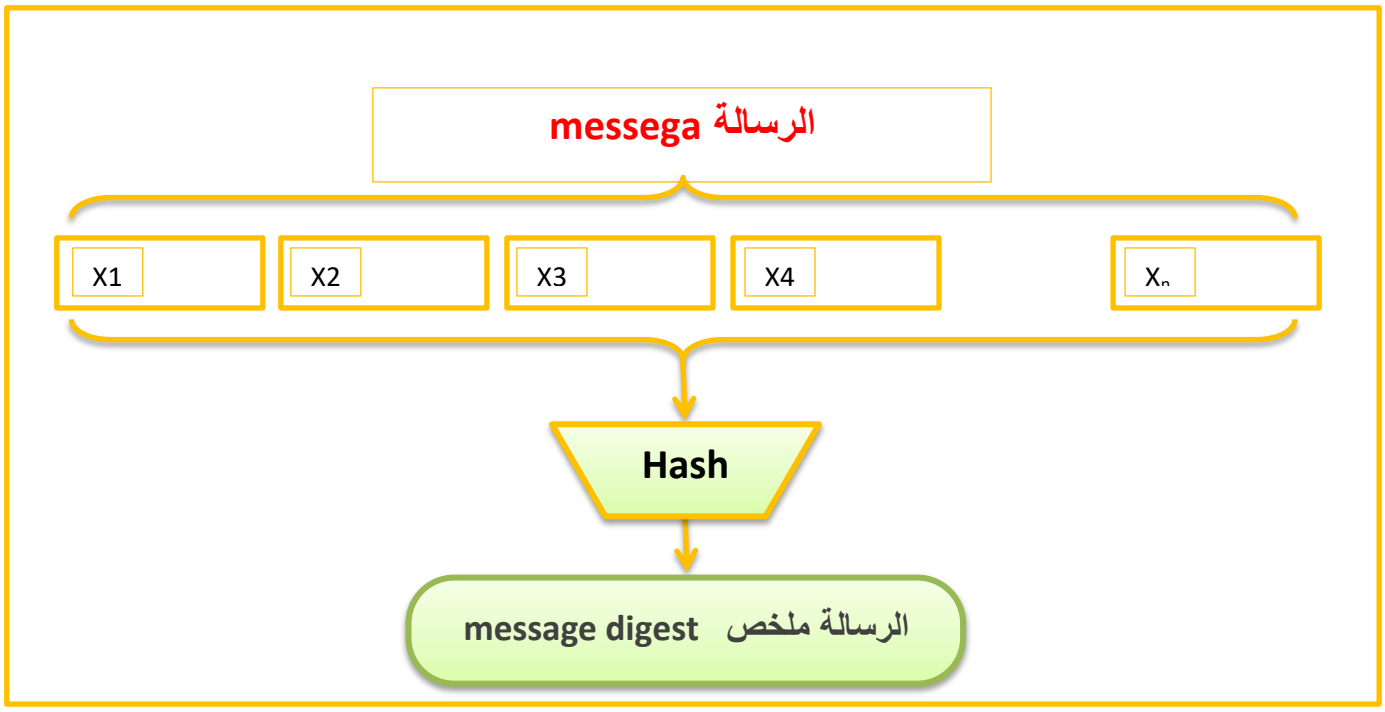
1- وظائف التجزئة (Hash):هي محاولة "لتسمية" البيانات بمعرف فريد وطول ثابت[15]

2- وظائف التجزئة (Hash) هي إحدى بدهيات التشفير المهمة وتستخدم على نطاق واسع في البروتوكولات. يحسبون ملخصًا لرسالة (message digest)عبارة عن سلسلة بتات قصيرة

[15] NCCIC ، File Haching ، (2022، 07، 10)،
https://www.cisa.gov/uscert/sites/default/files/FactSheets/NCCIC%20ICS_Factsheet_File_Hashing_S508C.pdf

ذات طول ثابت. بالنسبة لرسالة معينة، يمكن اعتبار ملخص الرسالة (message digest) أو قيمة التجزئة (Hash)بمثابة بصمة للرسالة (Digital fingerprint)، أي تمثيل فريد للرسالة ،على عكس جميع خوارزميات التشفير الأخرى[16]

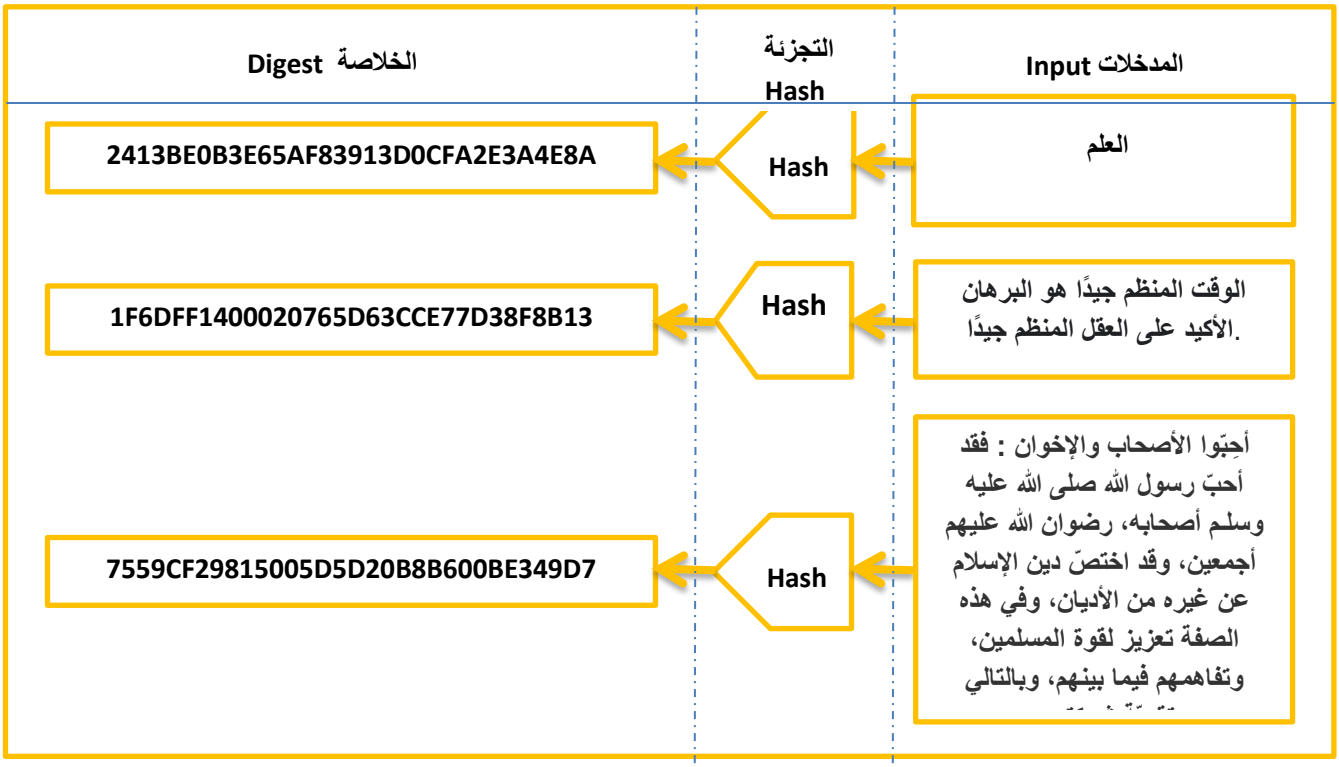
المخطط التالي يوضح كيف تعمل وضيفة التجزئة Hash



الشكل 32 : سلوك المدخلات والمخرجات الرئيسي لوظيفة التجزئة

[16] jan peelzl /christof paar ، **understanding cryptography**، **382**، **springer**، **2010** ، **293**

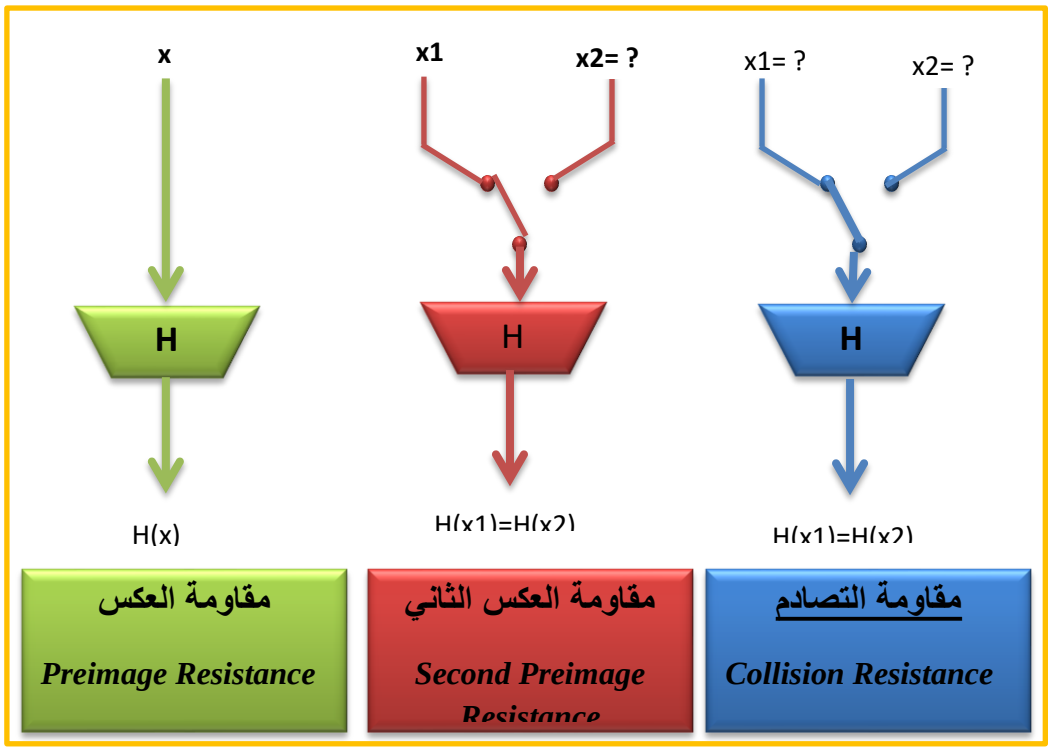
2-4-3-1-1 مثال حول تطبيق الدالة التجزئة **(Hash MD5)** على عدة رسائل مختلفة الأطوال :



الشكل 33 : المدخلات والمخرجات الرئيسية سلوك وظائف التجزئة

2-4-3-2 متطلبات الأمن لوظائف التجزئة(Hash Functions)

ناك ثلاث خصائص مركزية تحتاج وظائف التجزئة إلى امتلاكها حتى تكون **آمنة**:



الشكل 34 : الخصائص الأمنية الثلاثة لوظائف التجزئة

أ. مقاومة العكس او **Preimage resistance** :

بفرض قيمة التجزئة h يجب ان يكون من الصعب ايجاد رسالة m بحيث يكون h = Hash(m) .يرتبط هذا المبدأ ب الدالات وحيدة الاتجاه او غير العكوسة.

الدالات التي تفتقر لهذه الصفة تكون معرضة لهجوم العكس.

ب. مقاومة العكس الثاني او**Second-preimage resistance** :

بفرض وجود دخل x1 يجب ان يكون من الصعب ان نجد دخلا اخر x2 حيث

x1 ≠ x2 بحيث يكون Hash(x1)=Hash(x2) .

يصطلح احيانا إلى هذه الخاصية بـ*مقاومة العكس الضعيف* والدالات التي تفتقر لهذه الصفة تكون معرضة لهجوم العكس الثاني.Second-preimage

ت. مقاومة التصادم او**Collision resistance** :

يجب ان يكون من الصعب ايجاد رسالتين مختلفتين x1 و x2 بحيث يكون : Hash(x1)=Hash(x2)

يطلق على زوج كهذا تصادم التجزئة Collision Hash . يشار احيانا إلى هذه الخاصية بـ*ِ مقاومة العكس القوي* .

يجب ان يكون طول قيمة التجزئة على الاقل ضعف الطول المطلوب لمقاومة العكس, والا فستوجد التصادمات ب.Birthday attack [17]

2-4-3-3 خصائص وظائف التجزئة :

1. يمكن تطبيق hash(x) على الرسائل x من أي حجم.

2. طول الإخراج الثابت(x)hash ينتج قيمة تجزئة z بطول ثابت.

3. الكفاءة(x)hash سهلة الحساب نسبيًا.

4. مقاومة Preimage بالنسبة لمخرج معين z ، من المستحيل العثور على أي مدخلات x بحيث تكون z =(x)hash ، أي hash(x) أحادية الاتجاه.

[17] marefa ، دالة هاش تشفيرية ، (2022، 7 ، 16)،
https://www.marefa.org/%D8%AF%D8%A7%D9%84%D8%A9_%D9%87%D8%A7%D8%B4_%D8%AA%D8%B4%D9%81%D9%8A%D8%B1%D9%8A%D8%A9/simplified#%D8%A7%D9%84%D8%AE%D8%B5%D8%A7%D8%A6%D8%B5

5. مقاومة ما قبل الصورة الثانية إذا كانت x1 ، وبالتالي hash(x1) ، فمن غير المجدي حسابيًا إيجاد أي x2 بحيث hash(x1)=hash(x2)

6. مقاومة الاصطدام من غير المجدي حسابيًا العثور على أي أزواج x2 = x1 بحيث أن hash(x1)=hash(x2) .

2-4-3-4 أمثلة دوال التجزئة المشهورة :

ظهر عدد كبير من هذه الدوال خلال العقدين الماضيين لا يسعنا ذكرها في هذا المقام

نشرح اهمها وهي (MD5,SHA1,SHA256,SHA512 ...).

2-4-3-4-1 **MD5** :

*2.4.3.4.1.1 تعريف **MD5**:*

هي خوارزمية تجزئة تشفير يمكن استخدامها لإنشاء قيمة سلسلة 128 بت من سلسلة طول عشوائية. على الرغم من وجود حالات عدم أمان تم تحديدها مع MD5 ، إلا أنها لا تزال مستخدمة على نطاق واسع. يتم استخدام MD5 بشكل شائع للتحقق من سلامة الملفات.[18]

تم تطويرها من قبل مصممها الأصلي الدكتور *رونالد ريفست* (Ronald Rivest) في 1991 يتم فيها تقسيم المدخلات الى عدة حزم كل حزمة 512 bit

2.4.3.4.1.2 خصائص MD5:

1- سهلة التنفيذ وقليلة التكلفة.

2- تُوفّر مخرجًا مختلفًا لكلّ مدخل مهما صغر الفرق بينهم وهو ما يُسمّى بالبصمة (Fingerprint).استحالة الرّجوع من قيمة الاختزال hash إلى الرسالة الأصليّة.

2.4.3.4.1.3 مخطط عملية MD5 كما يلي:

[18] Dale Liu ، Next Generation SSH2 Implementation ،Syngress ،st Edition1 ،2008 ،64-41
https://www.sciencedirect.com/topics/computer-science/message-digest-algorithm-5

2.4.3.4.1.4 تطبيقات خوارزمية التشفير MD5

1- التأكيد على صحّة الملفّات:(Data Integrity)

2- علم التوقيع الرقمي:(Digital Signature) ، إثبات هويّة المُرسل

3- كلمة المرور (Authentication)

2-4-3-4-2 **SHA(Secure Hash Algorithm)** :

SHA تعني خوارزمية التجزئة الآمنة. نشر المعهد الوطني للمعايير والتكنولوجيا إصدارات مختلفة من SHA. نذكر البعض منها :

2.4.3.4.2.1 SHA1

تعريف SHA1 :

ينتج SHA-1 ملخصًا للرسالة استنادًا إلى مبادئ مشابهة لتلك المستخدمة من قبل رونالد لريفست(Ronald Rivest) من معهد ماساتشوستس للتكنولوجيا في تصميم خوارزميات هضم الرسائل MD2 و MD4 و MD5 ، ولكنه يولد قيمة تجزئة أكبر (160 بت مقابل 128 بت).[19]

مخطط عملية SHA1 :

2.4.3.4.2.2 مخطط عملية MD5 كما يلي:

2.4.3.4.2.3 :SHA256

SHA 256 هو جزء من عائلة خوارزميات SHA 2 ، حيث يرمز SHA إلى خوارزمية Secure Hash. نُشر في عام 2001 ، وكان جهدًا مشتركًا بين NSA و NIST لتقديم خليفة لعائلة SHA 1 ، التي كانت تفقد قوتها ببطء ضد هجمات القوة الغاشمة (attacks-force).

[19] sha-1 ، (2022، 7 ، 16)، من ويكيبيديا، https://en.wikipedia.org/wiki/SHA-1#Development

تمثل أهمية 256 في الاسم قيمة ملخص التجزئة النهائي ، أي بغض النظر عن حجم النص العادي ، ستكون قيمة التجزئة دائمًا 256 بت[20]

2.4.3.4.2.4 مخطط عملية SHA256 كما يلي:

2.4.3.4.2.5 SHA512:

إنه جزء من مجموعة من خوارزميات التجزئة تسمى SHA-2 والتي تتضمن SHA-256 أيضًا والتي تُستخدم في blockchain البيتكوين للتجزئة.[21]

2.4.3.4.2.6 مخطط عملية SHA512 كما يلي:

[20] Karin Kelley ، simplilearn ، (2022، 7 ، 16)،
Introduction To Cyber Security | Cyber Security Training For Beginners | CyberSecurity | Simplilearn [ملف فيديو]،
https://www.youtube.com/watch?v=z5nc9MDbvkw&t=65s

[21] Zaid Khaishagi ، (21 ، 06 ، 2019) ، Cryptography: Explaining SHA-512 ، (2022، 7 ، 16)،
https://medium.com/@zaid960928/cryptography-explaining-sha-512-ad896365a0c1

3 الفصل الثاني: التقنيات المستخدمة :

3-1مقدمة :

خلال هذه الرسالة ، استخدمنا العديد من لغات البرمجة و الادوات لتحقيق مشروعنا .

هناك الكثير من الخيارات ، من الصعب اختيار الأفضل فيما يتعلق بالجوانب المختلفة مثل ،السرعة الأداء والأمن؛ في هذا الفصل سنشرح كل لغات البرمجة و االتقنيات و نركز في هذا الفصل على VS Code و إطار العمل NodeJs

3-2تقنيات التطوير الأمامية :

3-2-1 لغة التنسيق HTML 5 :

- هو اختصار لجملة (*Hypertext Markup Language 5*) وهذا يعني أنها لاتعتبر لغة برمجة.
- تعتبر لغة ترميز لصفحات الويب web .
- سهلة التعلم .
- تستخدم لتحديد صفحات **الويب** (**Pages Structure**).
- ملفات HTML لها امتداد "html. "

صورة 3:شعار HTML5

3-2-2 تعريف و مميزات CSS :

- **تعني** Cascading **S**tyle **S**heets .
- من أجل تحسين تصميم العناصر الموجودة في الصفحة.
- يصف طريقة عرض صفحة ال HTML .
- ملفات CSS لها امتداد "css. "
- تهتم بالخطوط والألوان والهوامش وتحديد العرض والإرتفاع .

3-2-3 مميزات لغة الCSS :

- التحكم بتصميم الموقع في ملف واحد.
- يمكنك إعادة إستخدام ملف CSS في عدة صفحات .
- يمكنك إنشاء تصميمات لكل أنواع الشاشات (الحاسوب ,الهاتف ,.....).
- سهلة التعلم .

- سهلة الصيانة .

3-2-4 لغة البرمجة JavaScript :

- تم إصدار أول نسخة سنة 1995 ,طورتها شركة نتسكيب Netscape.
- تعتبر من أكثر اللغات شيوعا في العالم.
- سهلة التعلم .
- تستخذم في متصفحات الويب web .
- متعددة المنصات .
- تدعم البرمجة كائنية التوجه .
- ملفات JavaScript لها امتداد ".JS ".

3-3تقنيات التطوير الخلفية:

3-3-1 ماهو ال Node.js؟

صورة 4 : شعارNodeJs

_ هي تقنية لإنشاء تطبيقات الويب وبشكل أكثر دقة جانب الخادم في JavaScript

-من المعروف أنه سريع جدًا لسببين أساسيين أنه مبني على محرك Chrome V8 المتطور و آلية إستقبال وإرسال المدخلات والمخرجات 0/1 التي يطلق عليها بالإنجليزية «Non-blocking » في مقابل آلية ال « Blocking » التي تنتهجها اللغات الأخرى وعلى رأسها PHP .

_ هي بيئة خادم مفتوحة المصدر.

_ مجاني .

_ لايعتبر لغة برمجة .

_ تم تطويره في عام 2009 من قبل مبرمج أمريكي يعيش في ألمانيا يدعى «ريان دال ».

– يدعم البرمجة غير متزامنة.

_ مكتوب بلغة C,JavaScript, C++

– يتميز بوقت معالجة أقل والقدرة على التعامل مع طلبات متعددة في وقت واحد.

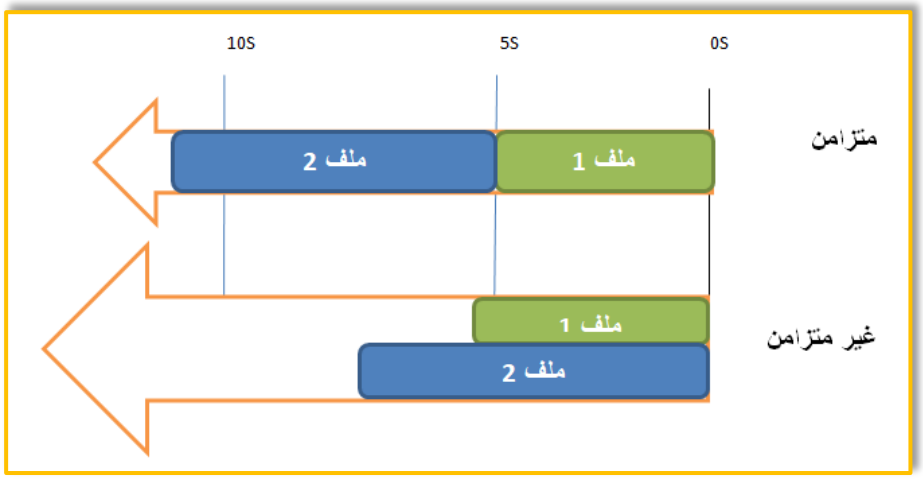
– ويشتغل على العديد من الأنظمة الأساسية (Unix / Linux و Windows و macOS).

ما هو محرك V8؟

يستعمل Nodejs محرك الجافا سكريبت V8 الذي تم تطويره من طرف مبرمجي Google Chrome وتم تصميمه لتحسين أداء JavaScript بشكل أسرع وأخف وذلك بفضل اعتماده على تقنية متطورة في عملية ال compilation تسمى JIT Compilation أو (Just In Time Compilation) التي تقوم بتحويل البرنامج إلى أكواد تفهمها الآلة أثناء تنفيذ (Exécution) البرنامج وليس قبل بدء التنفيذ عكس ما كان عليه الحال قبل ظهور V8 عندما كانت جل المحركات تعتمد آلية ال Interprétation التقليدية.

آلية Non – Blocking :

_ تنفيذ الطرق بشكل غير متزامن, هنا مثال لتوضيح هذا النموذج:



الشكل 35 : نموذج غير محظور ، يتم تنزيل كلا الملفين في نفس الوقت وينتهي الأمر بشكل أسرع

الشكل 37 يمثل تنزيل ملفين, باستخدام نموذج المتزامن والغير المتزامن في النموذج المتزامن يتم تنزيل الملفات واحدا تلو الآخر وفي النموذج الغير المتزامن يتم تنزيل الملفات بالتوازي .وبالتالي فإن إجمالي الوقت التنفيذ لتنزيل الملفين يكون أقصر في هذا الأخير وهذا السبب هو الذي جعله سريع وميزه عن باقي اللغات .

3-3-2 ما الذي يمكن أن يفعله Node.js؟

_ يمكن لـ Node.js إنشاء محتوى صفحة ديناميكي.

_ يمكن لـ Node.js إنشاء الملفات على الخادم وفتحها وقراءتها وكتابتها وحذفها وإغلاقها.

_ يمكن لـ Node.js جمع بيانات النموذج.

_ يمكن لـ Node.js إضافة أو حذف أو تعديل البيانات في قاعدة البيانات الخاصة بك[22].

3-3-3 ما هو ملف Node.js؟

_ تحتوي ملفات Node.js على مهام سيتم تنفيذها على أحداث معينة.

_ حدث نموذجي هو شخص يحاول الوصول إلى منفذ على الخادم.

_ يجب بدء ملفات Node.js على الخادم قبل أن يكون لها أي تأثير.

_ ملفات Node.js لها امتداد ".js"[23].

3-3-4 من يستخدم NodeJs:

حسب موقع W3Techs يتم استخدام Node.js بواسطة 1.9٪ من جميع مواقع الويب التي نعرف خادم الويب الخاص بها.[24]

3-3-5 الشركات الرئيسية التي تستخدم NodeJS:

1. LinkedIn

[22] w3schools ، What is Node.js، (10 ، 07 ، 2022) https://www.w3schools.com/nodejs/nodejs_intro.asp
[23] نفسه المرجع السابق
[24] w3techs ، Usage statistics of Node.js ، (10 ، 07 ، 2022)، https://w3techs.com/technologies/details/ws-nodejs

2. Netflix
3. Uber
4. Trello
5. PayPal
6. NASA
7. eBay
8. Medium
9. Groupon
10. Walmart
11. Mozilla

3-3-6 فوائد استخدام Node.js:

- سهل التعلم.
- يستخدم في الواجهة الأمامية والخلفية في نفس الوقت ,عكس اللغات الأخرى يجب عليك تعلم بعض اللغات الأخرى احتياجات التطوير بما في ذلك قواعد البيانات.
- سريع وسهل في تنفيذ المشاريع .

3-3-7 إعداد بيئة Node.js :

من أجل إعداد بيئة Node.js أنت بحاجة إلى برنامجين وهما :

- تثبيت Node.js و npm.
- محرر النصوص .

3-3-7-1 كيفية تثبيت Node.js و npm؟

لكل نظام تشغيل طريقة مميزة لتثبيت Node.js. يختلف ملف التثبيت الأساسي من نظام التشغيل إلى نظام التشغيل. ومع ذلك ، فقد حرص منشئو Node.js على تزويدك بالملفات الضرورية لكل نظام ,نأخذ مثال عن تثبيت Windows أما باقي الأنظمة لمعرفة تثبيتها يوجد الشرح الكامل في الموقع :

- تثبيت في نظام : macOS
- تثبيت في نظام : Linux
- تثبيت في نظام : Ubuntu

كيفية تثبيته على نظام Windows :

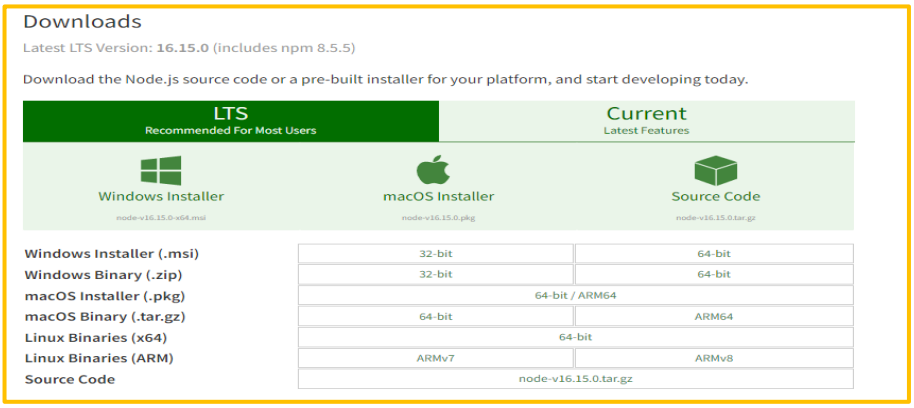
اتبع الخطوات :

➢ **تنزيل Windows Installer :**

تحتاج إلى تنزيل ملف (Windows Installer.msi) من الموقع الرسمي:

Windows Installer (.msi) ويحتوي على مجموعة من الملفات المثبتة الأساسية لتثبيت الإصدار الحالي والأخير, ويحتوي أيضا على حزم npm أنت لست بحاجة لي تثبيته.

عند التنزيل ، حدد الإصدار الصحيح بناءً على نظام التشغيل الخاص بك. على سبيل المثال ، إذا كنت تستخدم نظام تشغيل 64 بت ، فقم بتنزيل الإصدار 64 بت ، وإذا كنت تستخدم الإصدار 32 بت ، فقم بتنزيل الإصدار 32 بت كما هو موضح في الشكل :



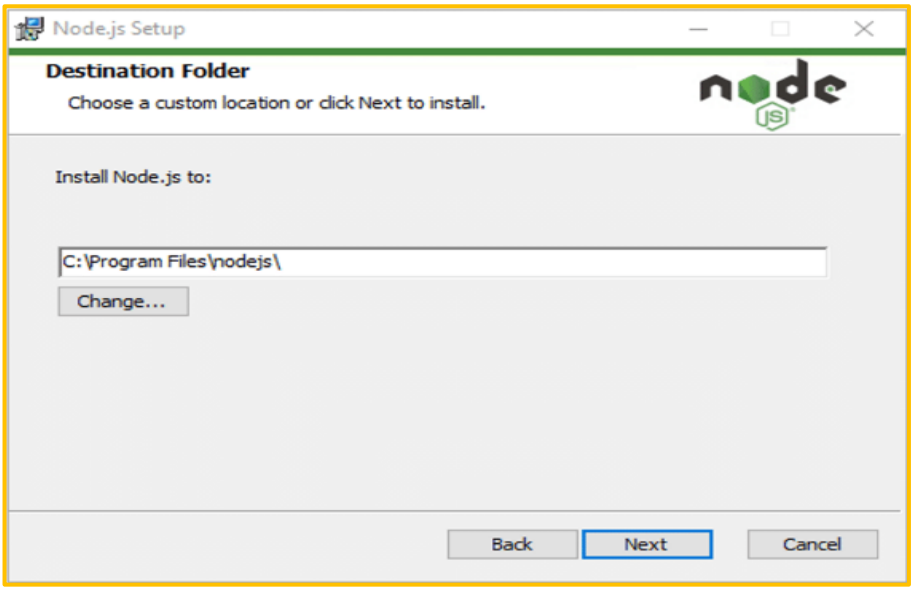
صورة 5: واجهة التنزيل NodeJs

بدأ عملية التثبيت : بمجرد فتح وتشغيل ملف msi تبدأ عملية التثبيت ,لكن يجب الموافقة على بعض التعليمات قبل البدأ .

أنقر نقرتين فوق الملف msi,سيطلب منك الموافقة على إتفاقية الترخيص ,ثم حدد مربع أوافق "I accept the terms in License Agreement" ثم أنقر على التالي :

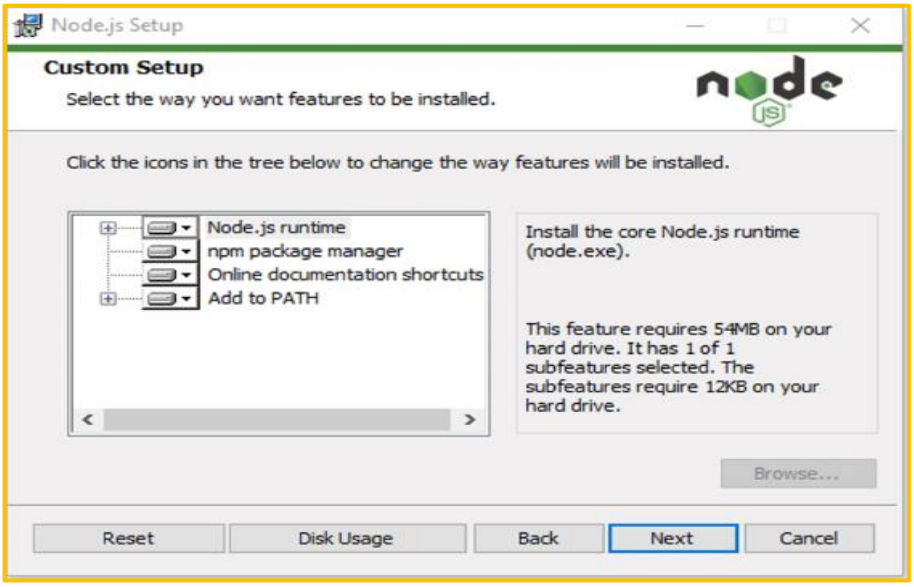
صورة 6: أقبل الإتفاقية

بعد ذلك حدد المسار الذي تريد تثبيت فيه NodeJS,إذا كنت لاتريد تغيير المسار أنقر مرة أخرى على التالي :



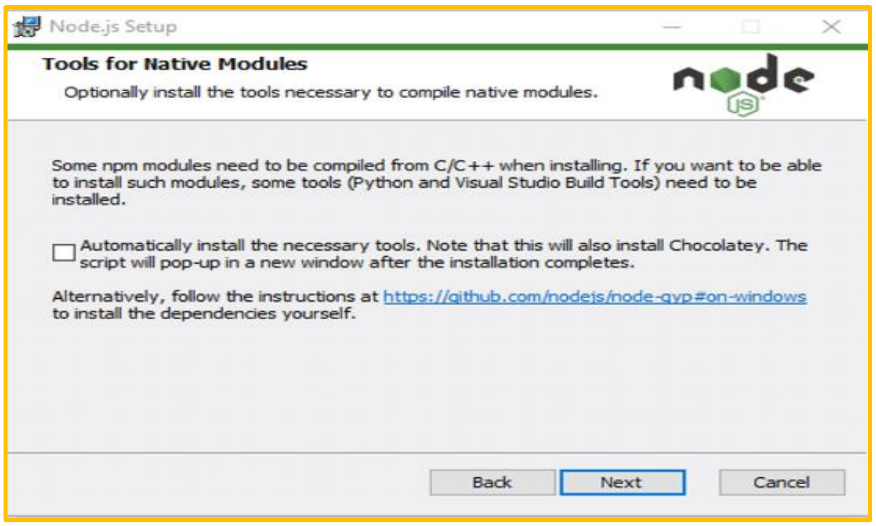
صورة 7 : تحديد مسار التثبيت

ستظهر لك الشاشة التالية خيارات التكوين المخصصة. إذا كنت تريد تثبيتًا قياسيًا بميزات Node.js الافتراضية ، فانقر فوق الزر "التالي". بدلاً من ذلك ، يمكنك تحديد العناصر الخاصة بك من أيقونات الشجرة قبل النقر فوق التالي:



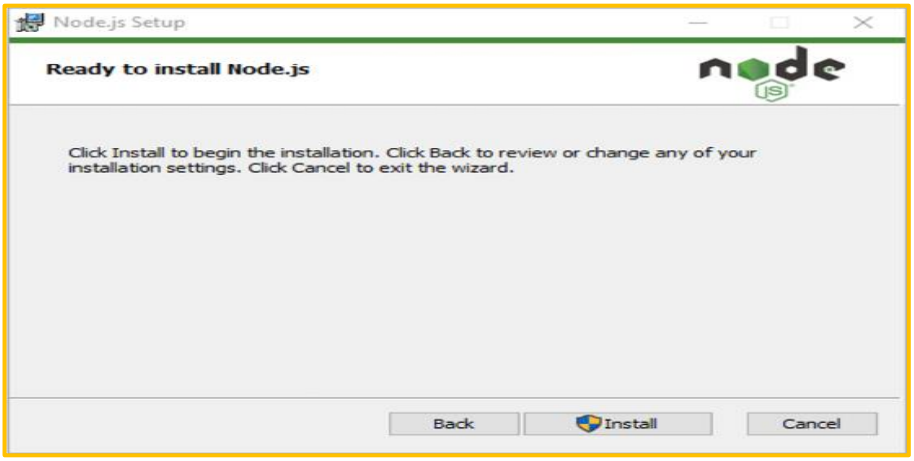
صورة 8: خيارات التكوين المخصصة

يمنحك Node.js خيارات لتثبيت أدوات للوحدات النمطية الأصلية .إذا كنت مهتمًا بها ، فانقر فوق مربع الاختيار لتحديد تفضيلاتك ، أو انقر فوق "التالي" للمتابعة مع الإعداد الافتراضي :



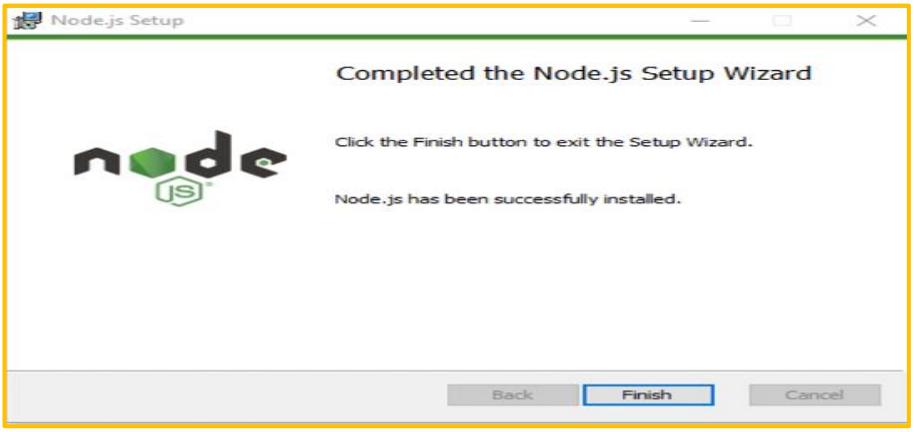
صورة 9: أدوات الوحدات المنطقية

- **تشغيل تثبيت Node.js على Windows** : أنقر فو زر التثبيت لبدء عملية التثبيت :



صورة 10: البداء في تثبيت NodeJs

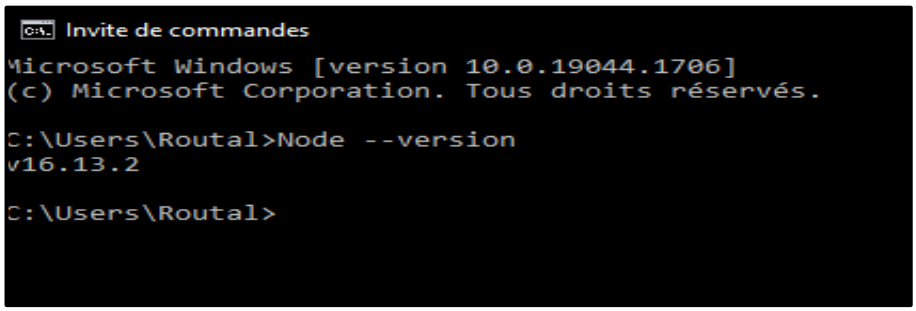
سوف ينتهي التثبيت في ثوانٍ أو دقائق ويعرض لك رسالة نجاح. انقر فوق الزر "إنهاء" لإغلاق أداة التثبيت



صورة 11: نهاية تثبيت NodeJs

- **تحقق من تثبيت :** عملية تثبيت NodeJS إنتهت ,الآن أنت بحاجة إلى التأكد من أنه تم التثبيت بنجاح أم لا ,وذلك عن طريق فتح موجه الأوامر على الجهاز "cmd" وكتابة الأمر التالي :

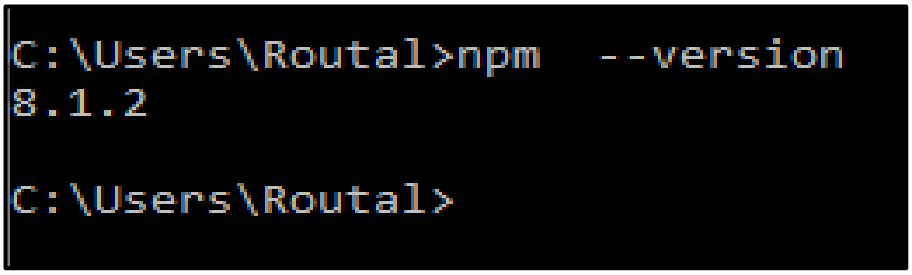
```
Node --version
```



صورة 12: التحقق من عملية التثبيت

وللتحقق من إصدار npm ، قم بتشغيل هذا الأمر :

```
npm --version
```

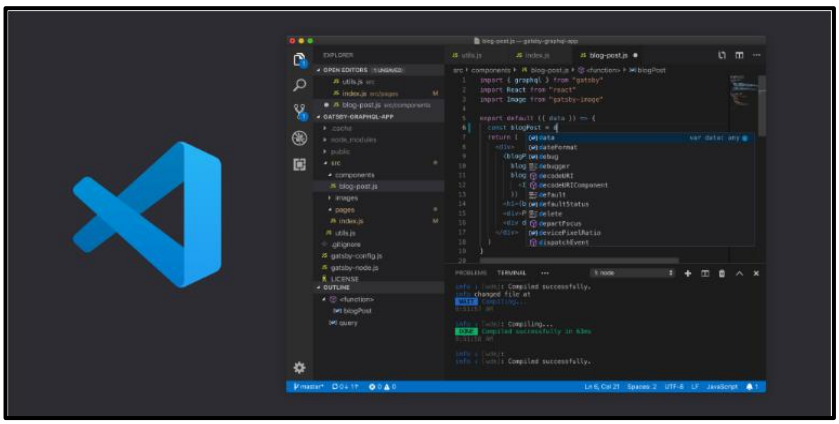


صورة 13: التحقق من إصدار npm

3-3-8 محرر النصوص :

نحتاج لكتابة الكود الخاص بنا محرر نصوص, ونأخذ على سبيل المثال **Visual Studio Code**:

- **Visual Studio Code** :



صورة 14:واجهة محرر VS Code

- يدعم عدد كبير من لغات البرمجة
- تم تطويره بواسطة Microsoft لنظام التشغيل Windows و Linux و MacOS.
- مجاني ومفتوح المصدر.
- سهولة عمل Debugging للكود .
- يأتى مدمج معه GIT لسهولة التحكم فى المشروع.
- الإتمام التلقائي للأوامر .
- يدعم الإضافات والمكتبات الخارجية بشكل يتيح لك بناء محرر النصوص بالشكل الذي يناسبك .
- متاح للتحميل من هنا.

3-3-9 الوحدات النمطية modules :

3-3-9-1 ماهي الوحدة النمطية :

هي مجموعة من مكتبات JavaScript المستخدمة بواسطة تطبيق Node.js, ويوجد وحدات مدمجة دون تثبيتها (يمكنك الإطلاع عليها هنا) .

3-3-9-2 تضمين الوحدات(Include Modules) :

يمكنك استخدام الدالة « require» مع إسم الوحدة :

```
var a = require('اسم الوحدة');
```

مثال:_استخدم التطبيق للوصول إلى وحدة HTTP وإنشاء خادم في الشكل التالي:

```
app.js > ...
1  var http =require('http');
2  var dt = require('./module');
3
4  http.createServer(function (req, res) {
5    res.writeHead(200, {'Content-Type': 'text/html; charset=utf-8'});
6    res.write('مرحبا بكم! ');
7    res.end();
8  }).listen(8080);
```

createServer: طريقة إنشاء خادم HTTP .

res.writeHead('...'): 200 تعني رمز الحالة أن كل شيئ على ما يرام و 404 لم يتم العثور على الصفحة , والوسيطة الثانية هي كائن يحتوي على رؤوس الاستجابة.

Req : استلام طلب http .

Res : استجابة http للإرسال .

Res.write('...') : ارسال جزء من النص استجابة http .

Res.end() : نهاية استجابة http .

Listen(8080) :لبدء قبول الاتصال على رقم المنفذ "8080".

3-3-9-3 إنشاء وحدة :

يمكننا إنشاء وحدة وإدراجها في التطبيق.

مثال:

إنشاء وحدة تقوم بإرجاع التاريخ والوقت الحالي, وحفظها في ملف module.js :

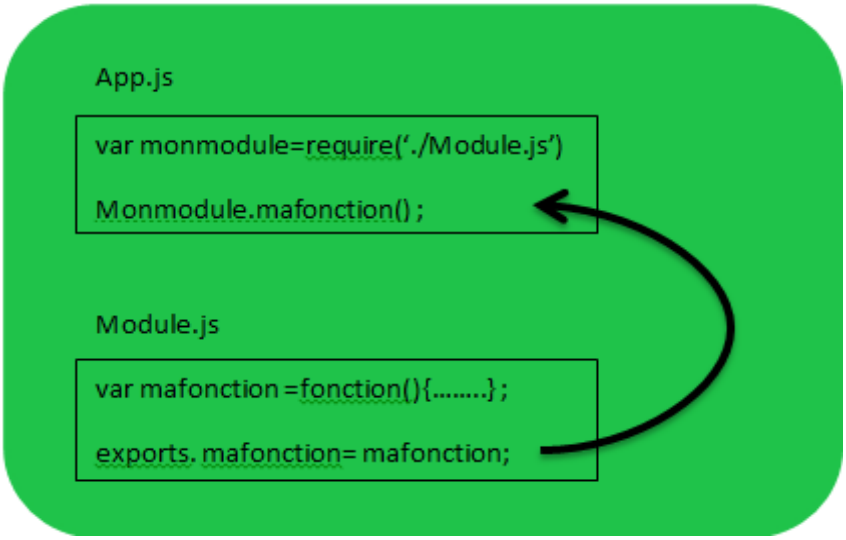
```
JS module.js > DateTime
1   exports.DateTime = function () {
2       return Date();
3     };
```

استخدم كلمة «exports»في اتاحة للوصول الى الخصائص الموجودة خارج ملف الوحدة .

تضمين الوحدة للوصول الى الوقت والتاريخ الحالي :

```
JS app.js > ...
1   var http = require('http');
2   var dt = require('./module');
3
4   http.createServer(function (req, res) {
5     res.writeHead(200, {'Content-Type': 'text/html; charset=utf-8'});
6     res.write ("التاريخ والوقت الحالي:" + dt.DateTime ()) ;
7     res.end();
8   }).listen(8080);
```

تعني "./" أن الوحدة module.js موجودة في نفس ملف app.js .



الشكل 36 : تصدير وظيفة

جميع الوحدات تعتمد على هذا المبدأ

أ. احفظ الكود في ملف app.js ,ثم قم بتشغيل الملف

```
E:\PFE MASTER 2\document word\ex_code_node>node app.js
```

4-9-3-3 خادم ملفات :

تسمح لك وحدة نظام الملفات Node.js بالعمل مع نظام الملفات على جهاز الكمبيوتر.
لتضمين وحدة نظام الملفات استخدم :

```
var a = require('fs');
```

الاستخدامات الشائعة لنظام الملفات :

قرائة الملفات Read files

إنشاء ملفات Create files

تحديث الملفات Update files

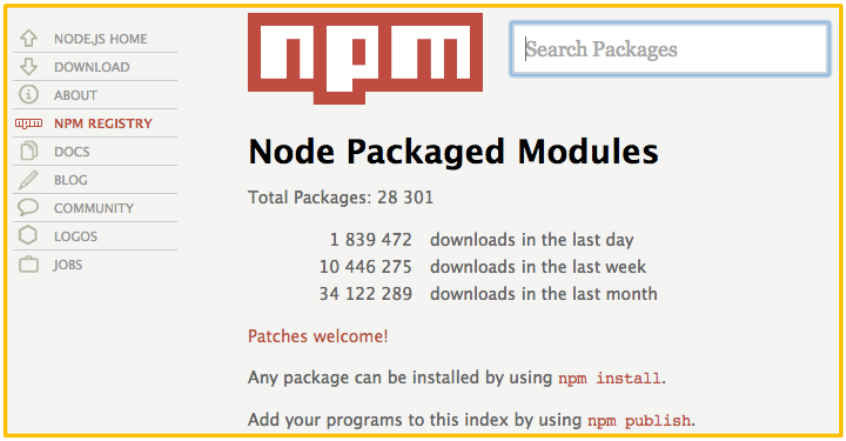
حذف الملفات Delete files

إعادة تسمية الملفات Rename files

3-3-10 نظام ادارة الحزم NPM (Node Package Manager)

3-3-10-1 ما هو NPM :

هو مدير الخاص بالحزم او البرمجيات الموجودة في بيئة العمل Nodejs ,ويوجد آلاف الحزم المجانية للتنزيل والاستخدام متاحة غلى الموقع https://www.npmjs.com/.



صورة 15: موقع npm

3-3-10-2 ما هي حزمة package :

تحتوي الحزمة في Node.js على جميع الملفات التي تحتاجها الوحدة النمطية.
الوحدات النمطية هي مكتبات JavaScript يمكنك تضمينها في مشروعك.

3-3-10-3 تنزيل حزمة Package :

لتنزيل الحزمة سهل للغاية ,افتح واجهة سطر الأوامر واطلب من NPMتزيل الحزمة التي تريدها, ويتم تثبيته في الدليل الحالي (node /nom_du_package) :

```
npm install nom_package
```

3-3-10-4 إلغاء تثبيت حزمة package :

لإلغاء تنزيل الحزمة سهل ,افتح واجهة سطر الأوامر واطلب من NPMإزالة الحزمة التي تريدها ,ويتم حذها من الدليل الحالي (node /nom_du_package) :

```
npm uninstall  nom_package
```

3-3-10-5 مكتبة JavaScript لمعايير التشفير :

عبارة عن مجموعة متزايدة من خوارزميات التشفير القياسية والآمنة المطبقة في JavaScript باستخدام أفضل الممارسات والأنماط. إنها سريعة ولديها واجهة متسقة وبسيطة للمزيد أكثر من المعلومات حولها هذا الموقع الرسمي.

- تثبيتها :

```
npm install crypto-js
```

- تضمينها :

```
var cryptoJS= require('crypto-js');
```

3-3-11 إطار عمل Express.js :

3-3-11-1 ماهو Express :

هو إطار عمل لتطوير تطبيقات الويب يعتمد على منصةnode.js ,وهو مفتوح المصدر تم اطلاقه سنة 2010 ,يسمح لنا بإدارة خدمة عالية المستوى في طلب والاستجابة http , للمزيد أكثر من المعلومات حول Express .

3-3-11-2 تثبيت Express :

```
npm init-y
```

أمر لإنشاء ملف package.json لتطبيقك .

```
npm install express --save
```

_قم بتثبيت Express وحفظه في قائمة.

للمزيد أكثر من المعلومات تثبيت Express .

3-11-3-3 تضمين وحدة Express :

```
var express = require('express');
```

4-11-3-3 إنشاء كائن Express :

```
var app= express();
```

5-11-3-3 التوجيه Routing :

يشير التوجيه إلى تحديد كيفية استجابة التطبيق لطلب العميل لنقطة نهاية معينة ، وهي URI (أو مسار) وطريقة طلب HTTP محددة (GET و POST وما إلى ذلك), يمكن أن يحتوي كل مسار على وظيفة معالج واحدة أو أكثر ، والتي يتم تنفيذها عند مطابقة المسار, يأخذ تعريف المسار الهيكل التالي:[25]

```
app.METHODE(uri, fonction)
```

- app : هو مثيل ل express .
- METHODE :هي طريقة طلب http بأحرف صغيرة ,ويوجد أنواع كثيرة للطلب نذكر منها طلبين (GET ,POST) :

	GET	POST
إرسال البيانات	URL	request body
الرؤية	مرئي في عنوان URL	غير مرئي في عنوانURL
الأمان	ليست البيانات السرية	البيانات السرية
حجم البيانات	بيانات صغيرة	البيانات الكبيرة

[25] http://expressjs.com/en/starter/basic-routing.html(15 ، 07 ، 2022) ، Basic routing ، Express

- Uri : على سبيل المثال : ('about', '/' , '/cours/td/')
- Fonction تستقبل وظيفتين req ,res
  Req :طلب http .
  Res : الرد على الإرسال .

1-5-11-3-3 الخصائص المفيدة لطلب **HTTP** :

_____تحليل الطلبات الواردة في برمجة وسيطية قبل معالجتها المضمنة بداخل body .

تثبيتها :

```
.npm install body-parser
```

تضمينها :

```
.var bodyParser = require('body-parser');
```

مثال :

```
//POST user[name]=mounir & user[email]= routalmounir1@gmail.com
```

req.body.user.name ➔mounir

req.body.user.email ➔ routalmounir1@gmail.com

2-5-11-3-3 إرسال استجابة **HTTP** :

_____ **res.send(data):** يرسل استجابة HTTP مع محتوى البيانات(سلسلة أو كائنًا أو مصفوفة).

مثال عن إرسال محتوى HTML صغير :

```
Res.send('< !DOCTYPE html><html><body><p>
```

مرحبا بالجميع

</p></body></html>') ;

. HTTP إنهاء استجابة : res.end()

3-3-12 قوالب جافا سكريبت المضمنة EJS :

3-3-12-1 ماهو EJS ؟

هي لغة نماذج بسيطة تتيح لك إنشاء ترميز HTML باستخدام JavaScript عادي. لا يوجد إعادة اختراع للتكرار والتحكم في التدفق. إنه مجرد JavaScript عادي.

للمزيد أكثر من المعلومات الموقع الرسمي EJS

- تثبيتها وتصمينها :

```
npm install ejs –save
let ejs=require('ejs') ;
```

- مراحل عمل نموذج :

1. تحديد نوع Template .

```
app.set('view  engine', 'ejs') ;
```

2. إنشاء ملف views وهذا الاسم مهم حتى يجب express أن ينظر فيه .
3. إنشاء ملف بداخل views "index.ejs" هذا الملف هو الذي يوجد في محتوى عرض Template.
4. نمرر الملف index .

```
res.render('index',{title :'.....'}) ;
```

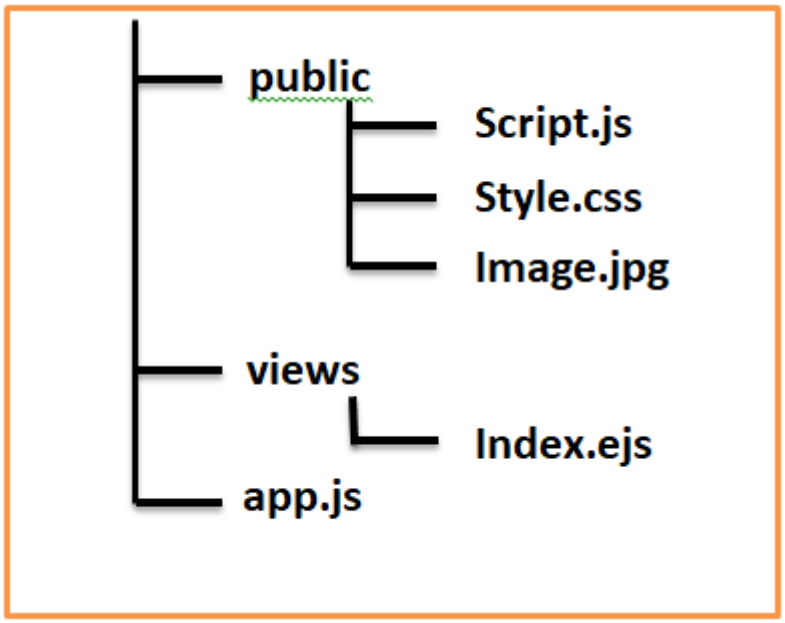
وهذه الميزة جديدة في Template .

- تضمين قوالب أخرى في القالب الحالي :

```
<%- include chemin/fichier.ejs %>
```

- إرفاق ملفات ثابتة بقالب: يمكن إرفاق قالب (ejs) بالملفات الثابتة (الصور ، css، js من جانب العميل) .

يجب أن تكون هذه الملفات في دليل يمكن الوصول إليه بواسطة الخادم السريع ومن الأحسن أن يكون اسم هذا الملف (public) .



طريقة الوصول الى الدليل بواسطة الخادم السريع express .

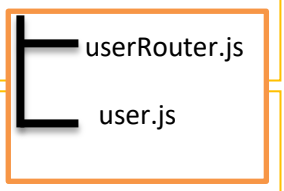
```
app.use(express.static('public')) ;
```

- express.route :يسمح لنا بإنشاء route في ملفات منفصلة والتي بدورها تكون مسؤولة عن ملفات أي مجرد تصدير وظائف أو كائنات:

مثال :ننشئ ملف userRouter.js يكون مسؤول عن ملف user

• userRouter.js

```
var express = require('express');
var router = express.Router();
router.get('/', fonction1);
router.post('/check', fonction2);
module.exports = router;
```




```
var express = require('express');
var serv= express();
var rout = require ('./userRouter');
serv.use('/register', rout);
serv.listen(8080);
```

المسارت هي : GET /register, POST /register/check

4 الفصل الثالث: التحليل والتصميم :

4-1مقدمة :

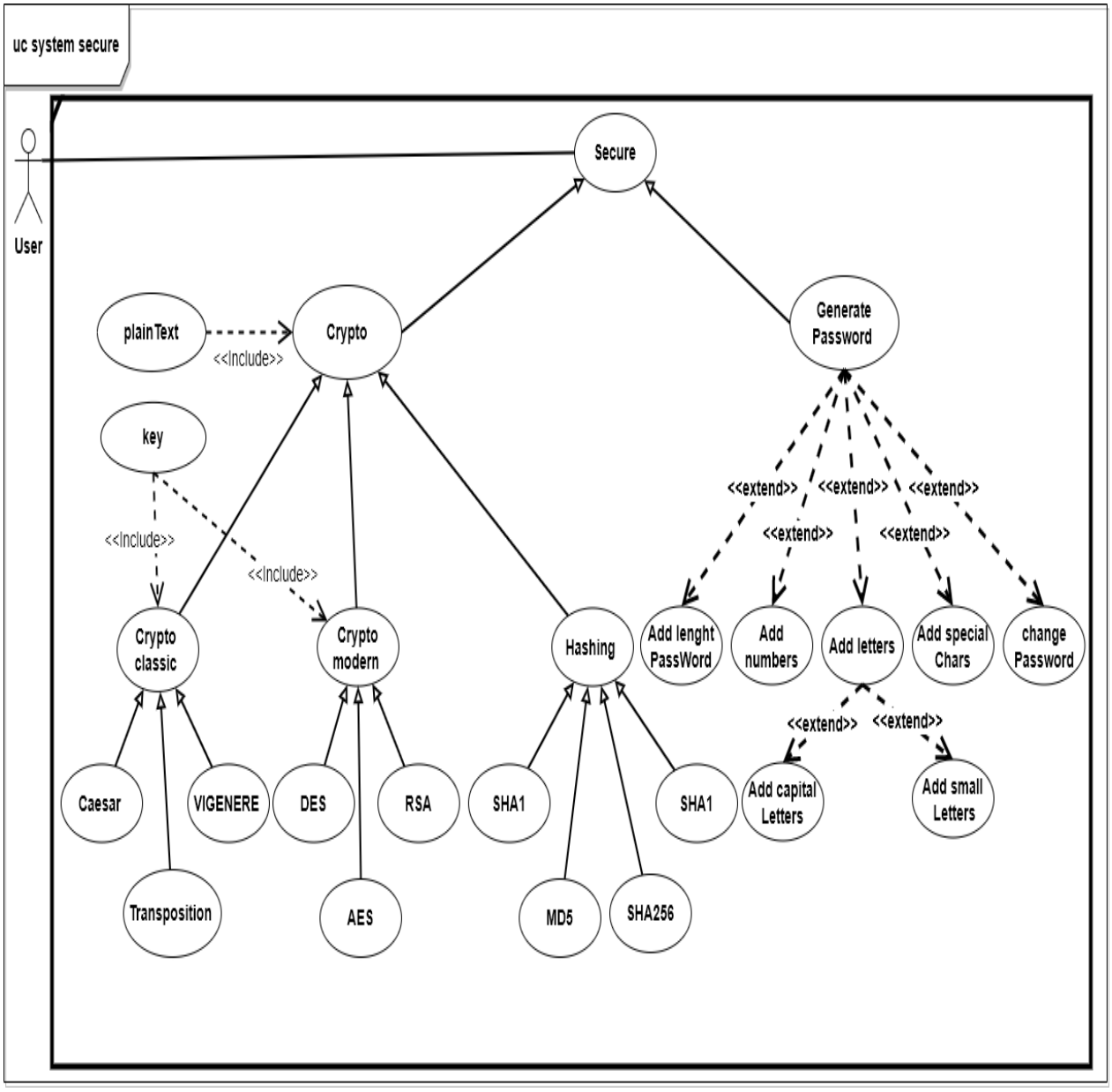
مراحل التحليل والتصميم هي أساس النظام الذي سننشئه ؛ تعطي هذه المراحل فكرة عن نظامنا .

في هذا الفصل ، نعرض المخططات التي تصف عدة جوانب

- مخطط حالات الاستخدام **use case .**
- مخطط التسلسل **sequences .**

4-2مخطط حالة الاستخدام Use Case Diagram:

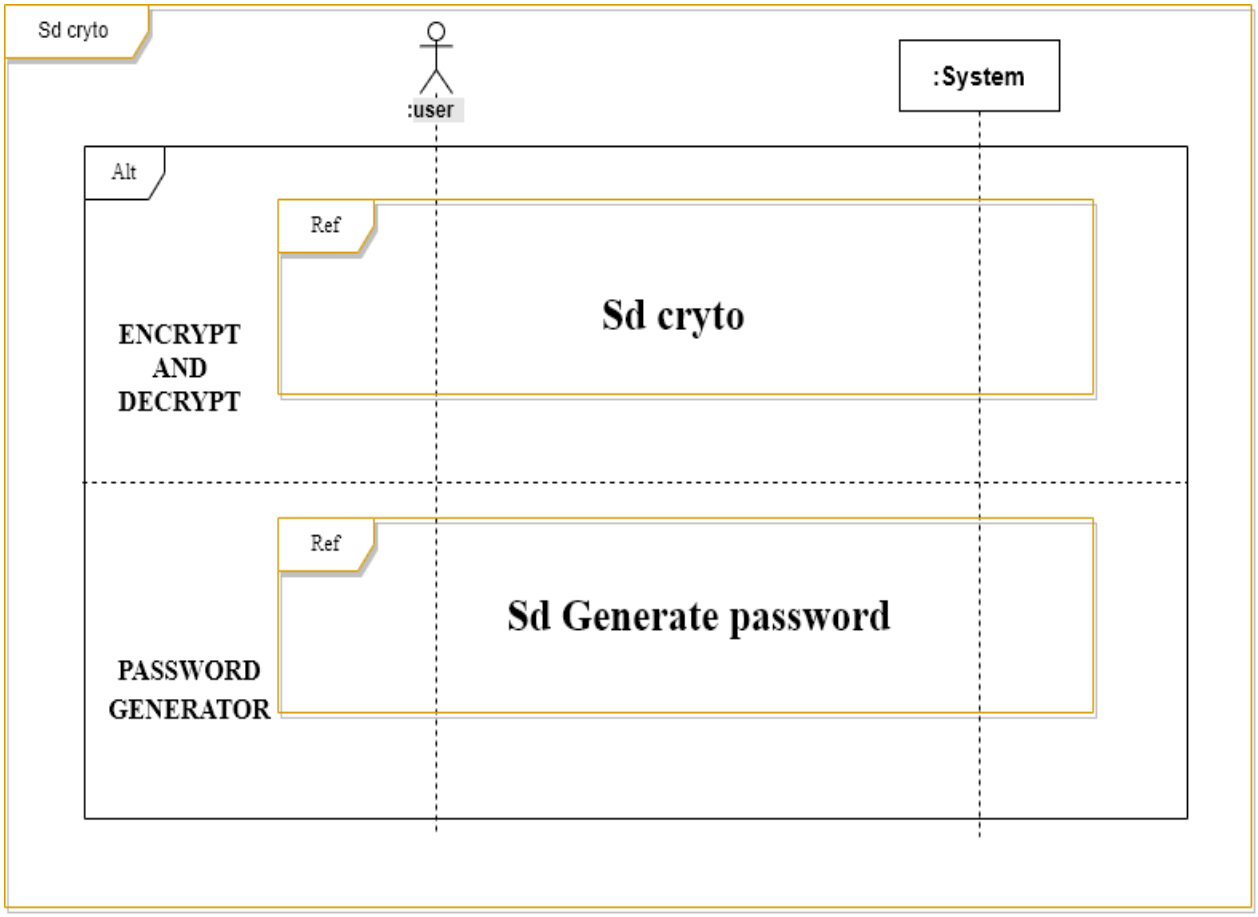
هي وصف لسلوك النظام. هذا الوصف مكتوب من وجهة نظر المستخدم الذي أخبر النظام للتو أن يفعل شيئًا معينًا. تلتقط حالة الاستخدام التسلسل المرئي للأحداث التي يمر بها النظام استجابةً لتحفيز مستخدم واحد.[26]



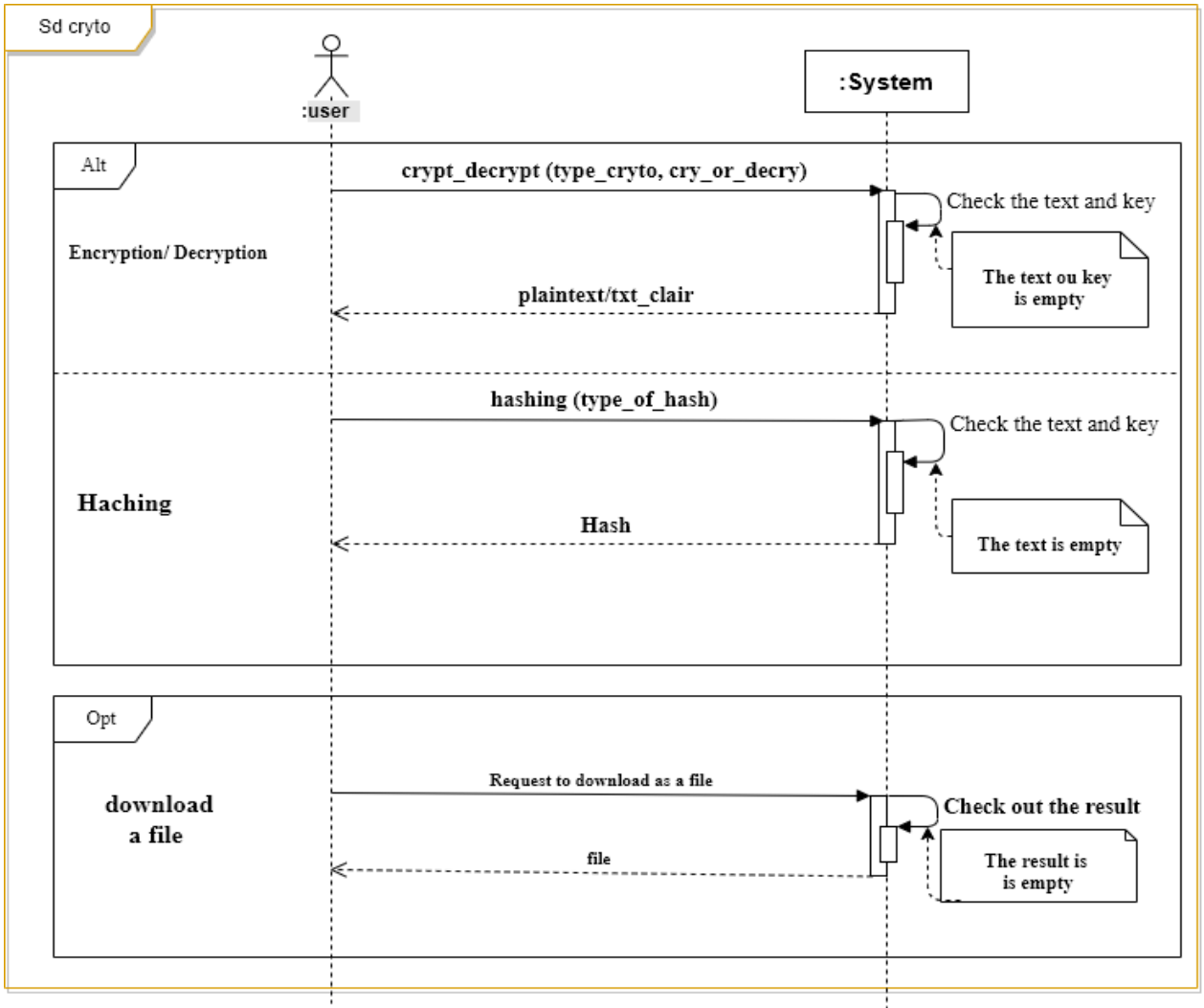
[26] **R. C. Martin, "Uml for java (tm) programmers," 2003.**

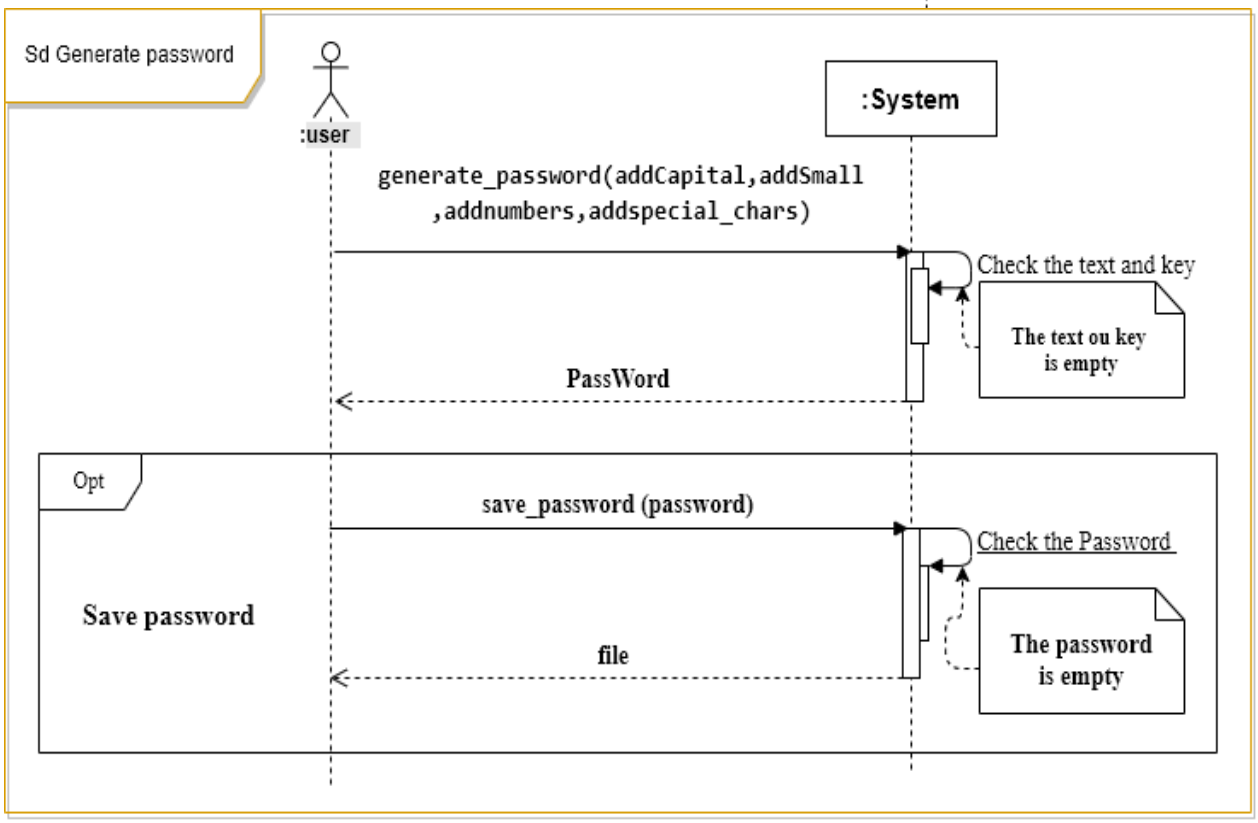
4-3المخططات التسلسلية Sequence diagram :

يوضح مخطط التسلسل أو مخطط تسلسل النظام (Sysytem Sequence diagram) تفاعلات العملية مرتبة في تسلسل زمني في مجال هندسة البرمجيات. يصور العمليات المتضمنة وتسلسل الرسائل المتبادلة بين العمليات اللازمة لتنفيذ الوظيفة. عادةً ما ترتبط مخططات التسلسل بحالة الاستخدام .Use Case تسمى مخططات التسلسل أحيانًا مخططات الأحداث أو سيناريوهات الأحداث. .27



[27] Sequence diagram (2022،6، 15)، من ويكيبيديا، https://en.wikipedia.org/wiki/Sequence_diagram


Sd cryto
:user
:System
Alt
crypt_decrypt (type_cryto, cry_or_decry)
Check the text and key
Encryption/ Decryption
The text ou key is empty
plaintext/txt_clair
hashing (type_of_hash)
Check the text and key
Haching
Hash
The text is empty
Opt
Request to download as a file
Check out the result
download a file
file
The result is is empty


Sd Generate password
:user
:System
generate_password(addCapital,addSmall ,addnumbers,addspecial_chars)
Check the text and key
The text ou key is empty
PassWord
Opt
save_password (password)
Check the Password
Save password
file
The password is empty

5 الفصل الرابع : الواجهات المختلفة لموقعنا :

5-1مقدمة :

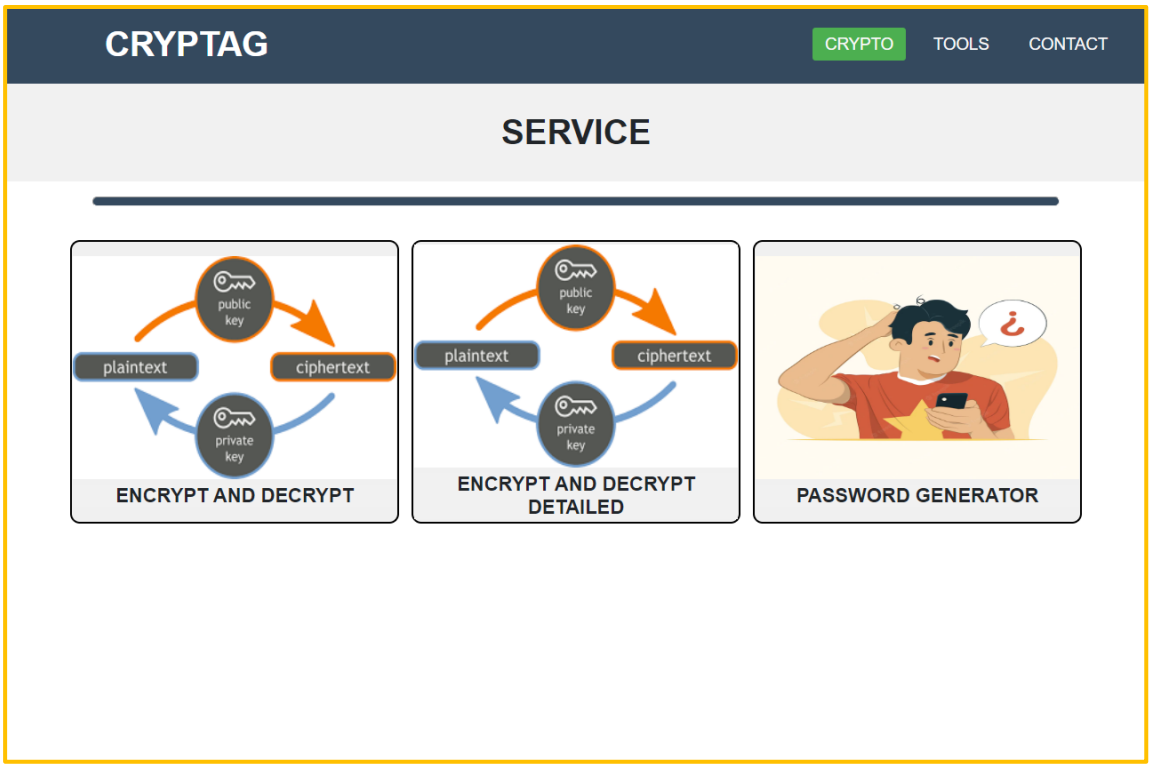
بعد أن تحدثنا أساسيات التشفير وتقنيات التطوير المستخدمة، سننتقل إلى المرحلة الأخيرة من مشروع التخرج الخاص بنا .

يتطلب تنفيذ مشروعنا عددًا من أدوات التكنولوجيا والتطوير. لقد اخترناها على أساس خصائصها التي نراها مناسبة للتنفيذ .

وأخيرًا ، سنختتم هذا الفصل من خلال تقديم الواجهات الرئيسية لموقعنا .

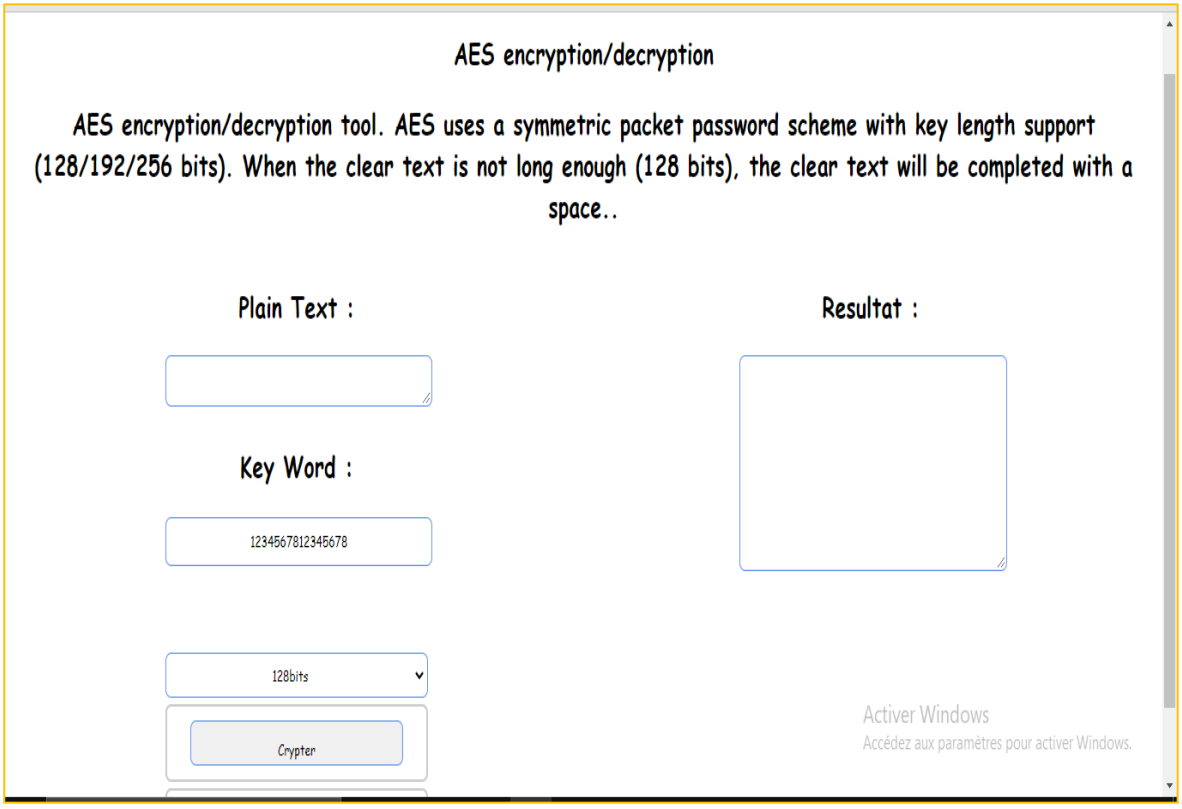
5-2الواجهة الرئيسية :

تمثل نافذة القائمة الرئيسية للموقع, تقدم قائمة الصفحة الرئيسية وتوليد كلمة السر والتشفيرو فك التشفير **بمكتبة CryptoJS و التشفيرو فك التشفير بكل مراحله صورة 16.**



صورة 16:الواجهة الرئيسية

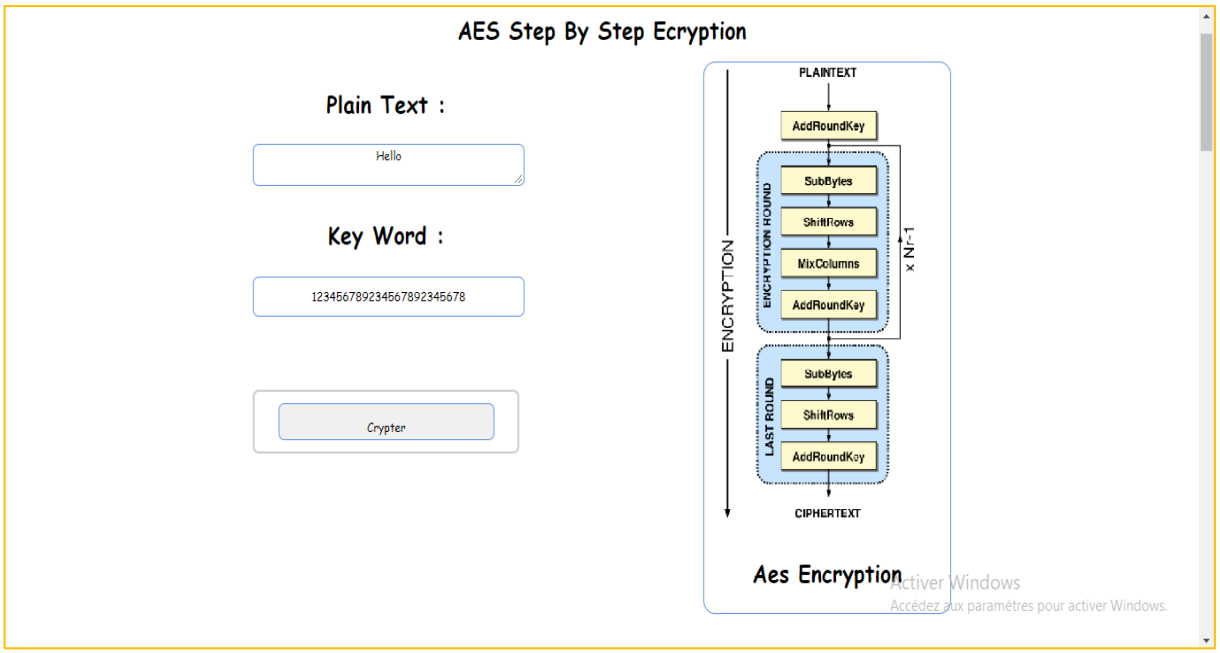
5-3واجهة تشفير و فك التشفير AES :

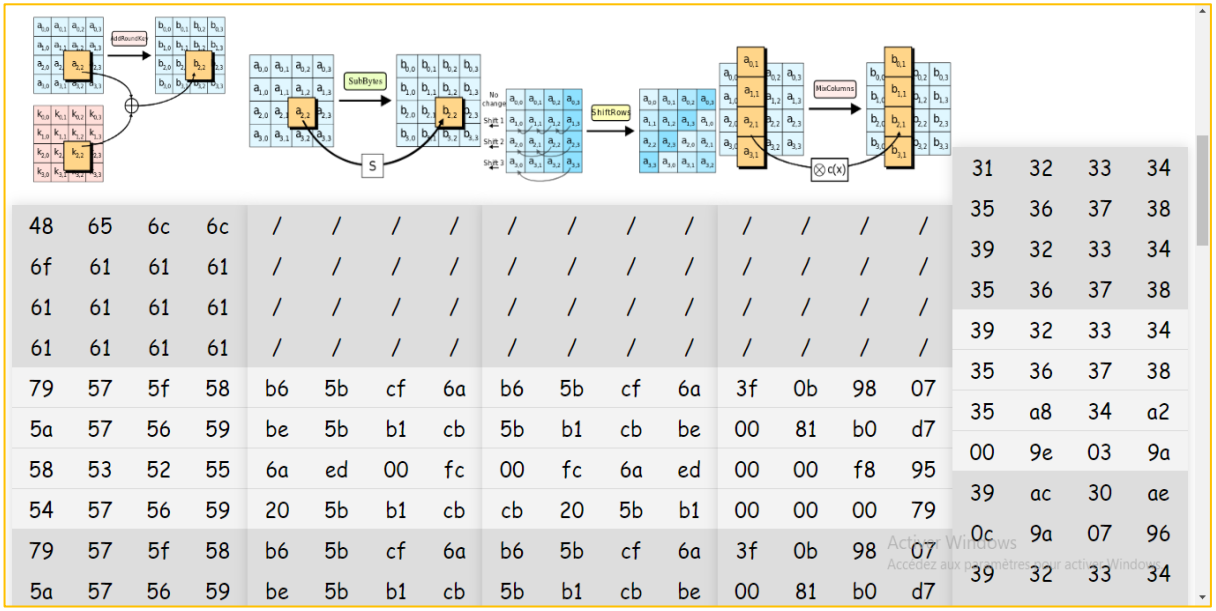


صورة17: واجهة تشفير و فك التشفير AES

5-4واجهة التشفير AES بتفصيل خطوة بخطوة :

هده الواجهة لتشفير و فك التشفير مع اظهار المراحل المتمثلة في كل من Key Addition، SubBytes ، ShiftRows، MixCols، Round Key Value صورة 19



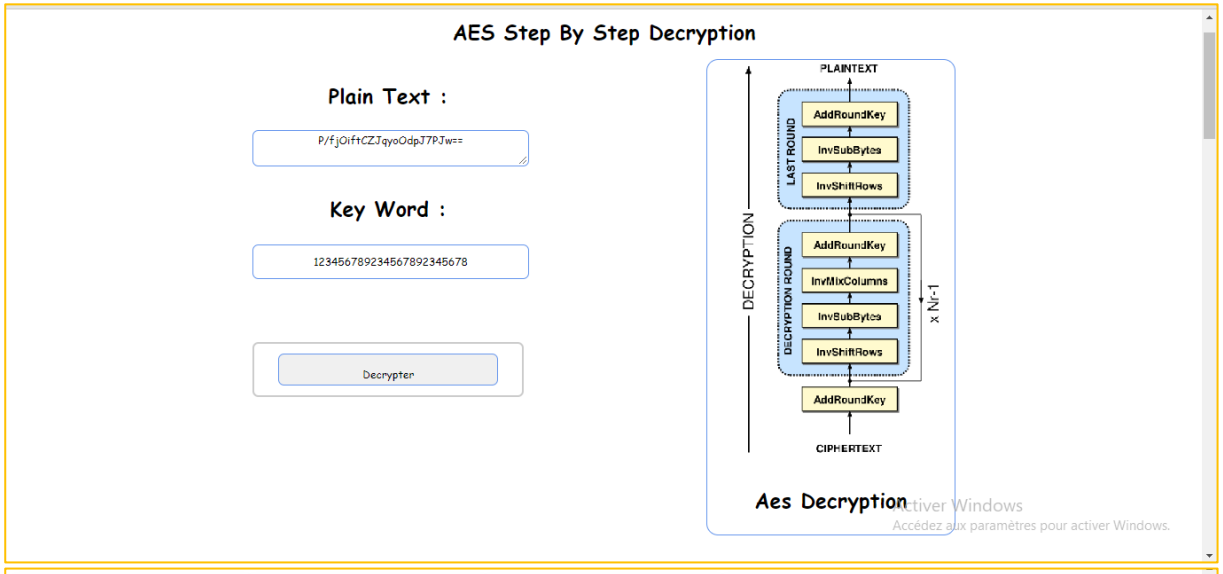


48	65	6c	6c	/	/	/	/	/	/	/	/	/	/	/	/
6f	61	61	61	/	/	/	/	/	/	/	/	/	/	/	/
61	61	61	61	/	/	/	/	/	/	/	/	/	/	/	/
61	61	61	61	/	/	/	/	/	/	/	/	/	/	/	/
79	57	5f	58	b6	5b	cf	6a	b6	5b	cf	6a	3f	0b	98	07
5a	57	56	59	be	5b	b1	cb	5b	b1	cb	be	00	81	b0	d7
58	53	52	55	6a	ed	00	fc	00	fc	6a	ed	00	00	f8	95
54	57	56	59	20	5b	b1	cb	cb	20	5b	b1	00	00	00	79
79	57	5f	58	b6	5b	cf	6a	b6	5b	cf	6a	3f	0b	98	07
5a	57	56	59	be	5b	b1	cb	5b	b1	cb	be	00	81	b0	d7

31	32	33	34
35	36	37	38
39	32	33	34
35	36	37	38
39	32	33	34
35	36	37	38
35	a8	34	a2
00	9e	03	9a
39	ac	30	ae
0c	9a	07	96
39	32	33	34

صورة 18: واجهة التشفير بتفصيل

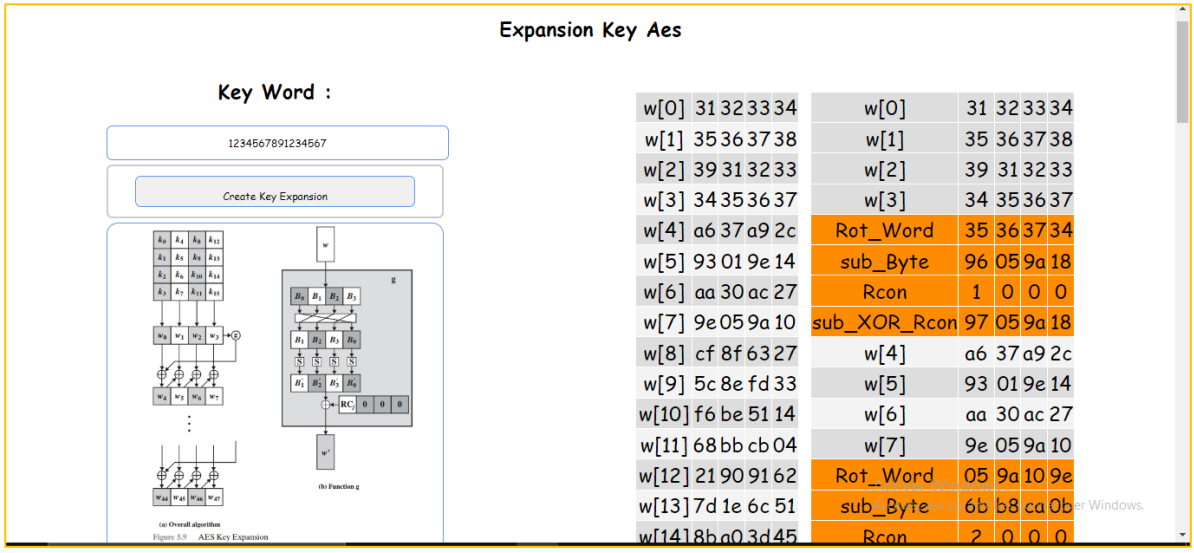
5-5 واجهة فك التشفير بتفصيل :



3f	f7	e3	3a	/	/	/	/	/	/	/	/	/	/	/	/	6c	a4	b0	69
27	ed	9	92	/	/	/	/	/	/	/	/	/	/	/	/	74	be	5a	c1
6a	ca	83	9d	/	/	/	/	/	/	/	/	/	/	/	/	ca	99	6e	2e
a4	9e	cf	27	/	/	/	/	/	/	/	/	/	/	/	/	09	c5	97	6b
53	53	53	53	d7	1b	f7	b8	d7	1b	f7	b8	0d	44	26	9a	c5	7d	f9	c1
53	53	53	53	1e	3b	72	b2	b2	1e	3b	72	3e	e9	49	1e	18	1a	ea	a8
a0	53	ed	b3	3e	0b	38	bd	38	bd	3e	0b	76	cd	d1	9e	be	27	34	ef
ad	5b	58	4c	fa	23	08	48	23	08	48	fa	32	bf	d4	14	c3	5c	f9	45
c8	39	df	5b	3a	56	ac	cd	3a	56	ac	cd	a2	b9	aa	80	b8	06	34	6b
26	f3	a3	b6	a5	00	42	6d	6d	a5	00	42	b3	29	52	f6	dd	67	13	69
c8	ea	e5	71	8f	18	e5	66	e5	66	8f	18	2a	d3	73	34	a6	3d	de	47
f1	e3	2d	51	c7	8d	bf	0b	8d	bf	0b	c7	b4	f4	9e	31	7d	7b	cd	aa
1a	bf	9e	eb	3e	65	c5	45	3e	65	c5	45	d1	bc	07	68	63	40	27	86
6e	4e	41	9f	b9	40	f8	5c	5c	b9	40	f8	a7	db	72	e1	65	61	27	02

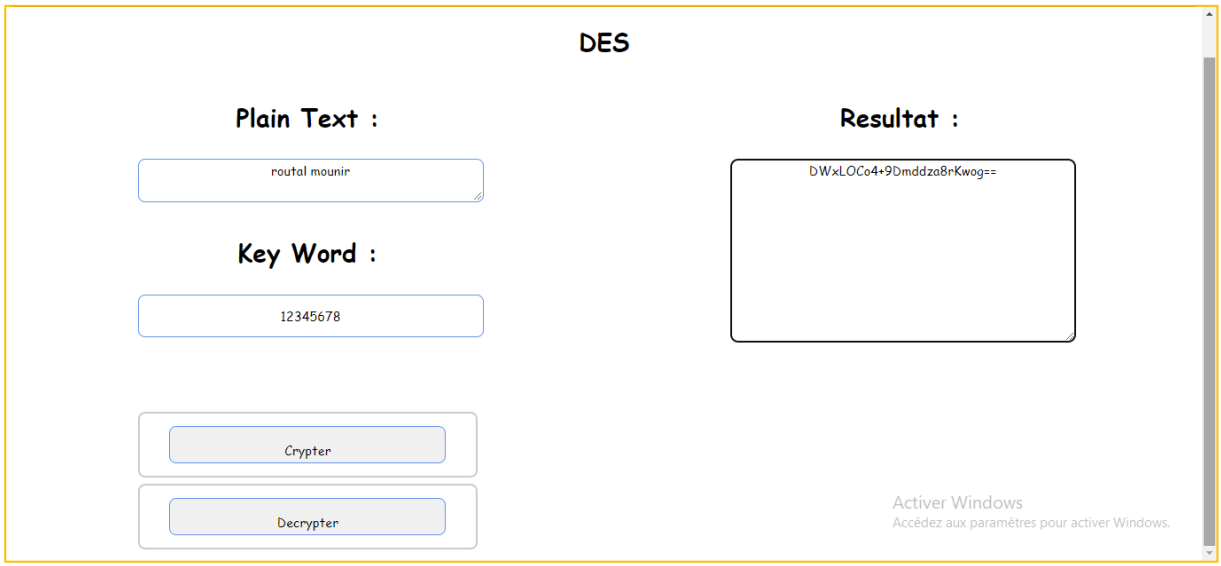
صورة 19 : واجهة فك التشفير

5-6 واجهة تمديد مفتاح التشفير Expansion key :



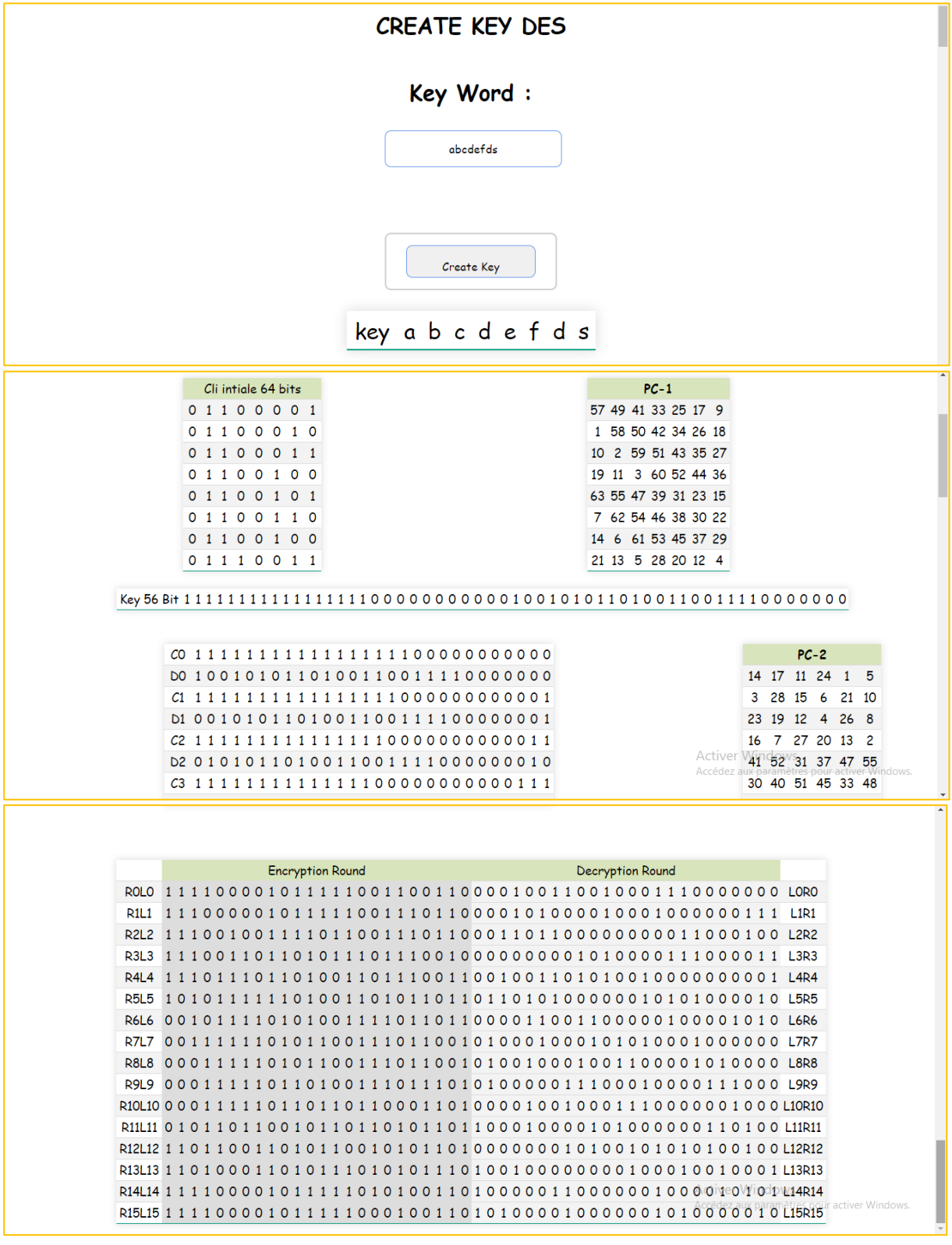
صورة20 : واجهة Expansion key

5-7واجهة التشفير و فك التشفير DES :



صورة21 : واجهة التشفير و فك التشفير DES

5-8واجهة إنشاء المفتاح DES :

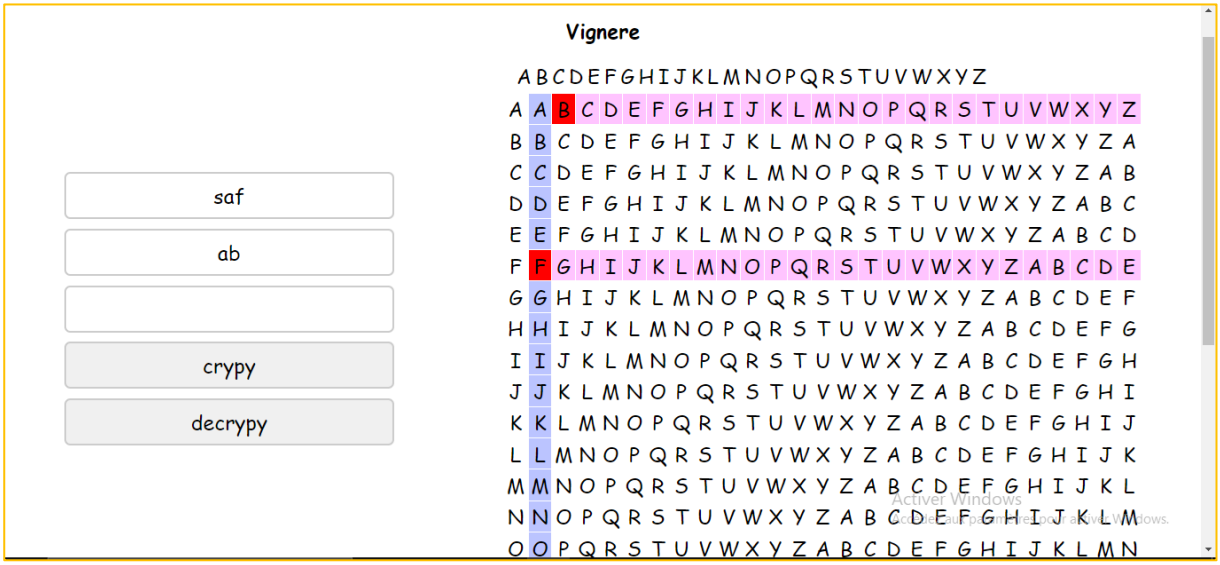


صورة22 : واجهة إنشاء المفتاح DES

5-9واجهة تشفير و فك التشفير قيصر Cesar :

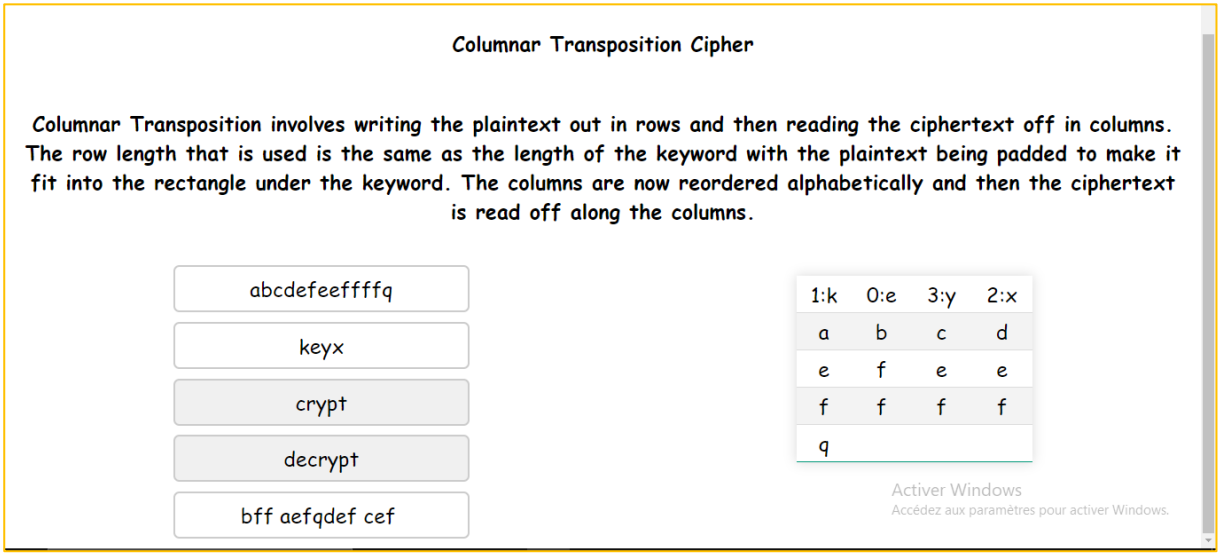
صورة23 : واجهة التشفير وفك التشفير Cesar

5-10 واجهة التشفير وفك التشفير Vignere :



صورة24 : واجهة التشفير وفك التشفير Vignere

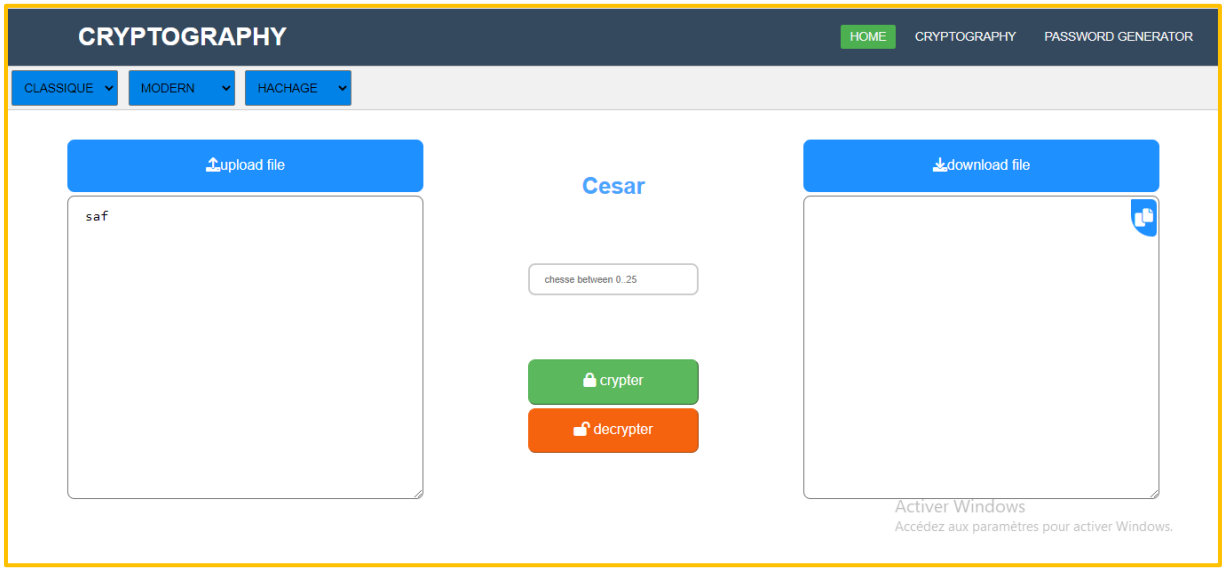
11-5 واجهة التشفير وفك التشفير Trasposition :



صورة25 : واجهة التشفير وفك التشفير Trasposition

12-5 واجهة التشفير وفك التشفير بمكتبة CryptoJS :

هذه الواجهة لتشفير النصوص وفك التشفير ويوجد ثلاث أنواع من التشفير كلاسيكي والحديث والهشاج .

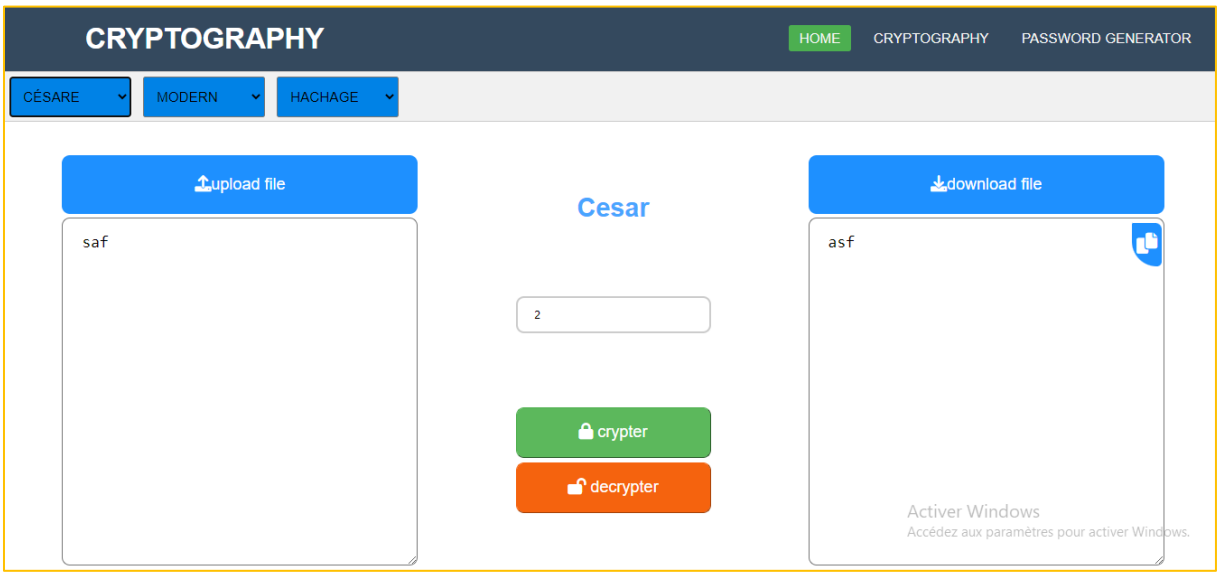


صورة 26:واجهة التشفير وفك التشفير

التشفير الكلاسيكي :

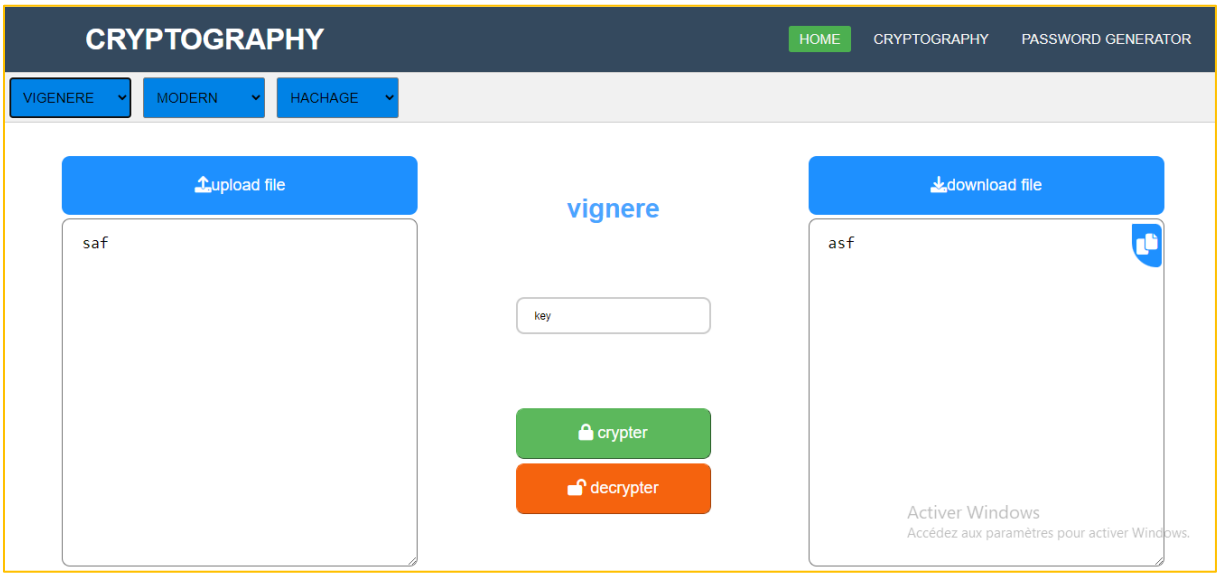
يحتوي على إلى ثلاث خوارزميات :

13-5 خوارزمية قيصر César:



صورة 27: التشفير الكلاسيكي Cesar

14-5 : Vignere

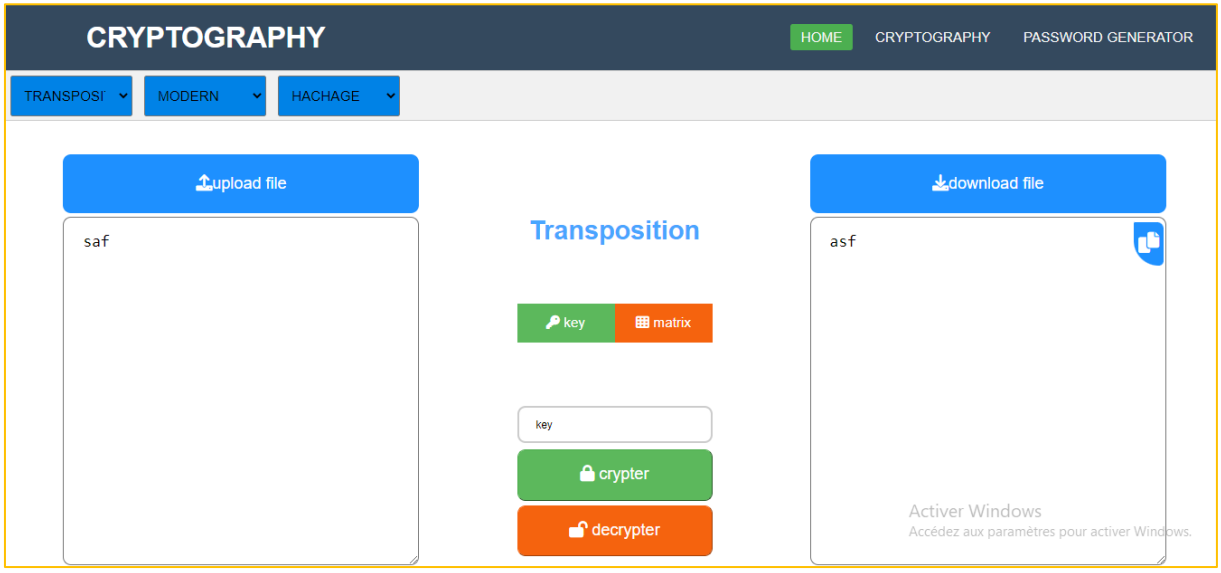


صورة 28:التشفير الكلاسيكي Cesar

15-5 Transposition :

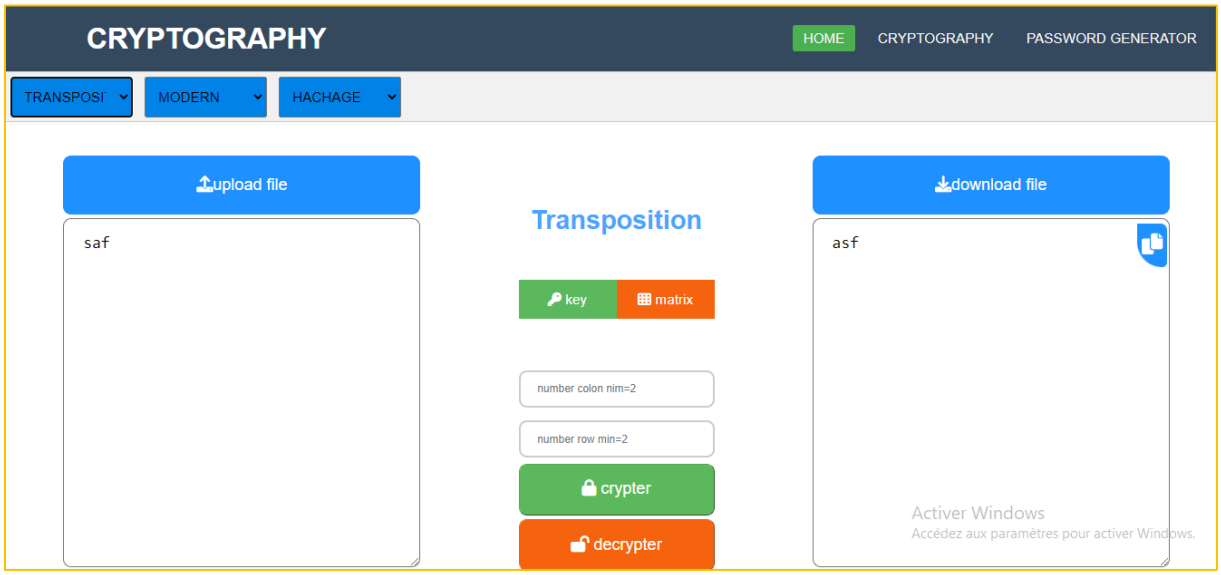
يوجد نوعين التشفير عن طريق المفتاح أو الصفوف و الأعمدة .

1. عن طريق المفتاح



صورة 29:التحويل عن طريق المفتاح

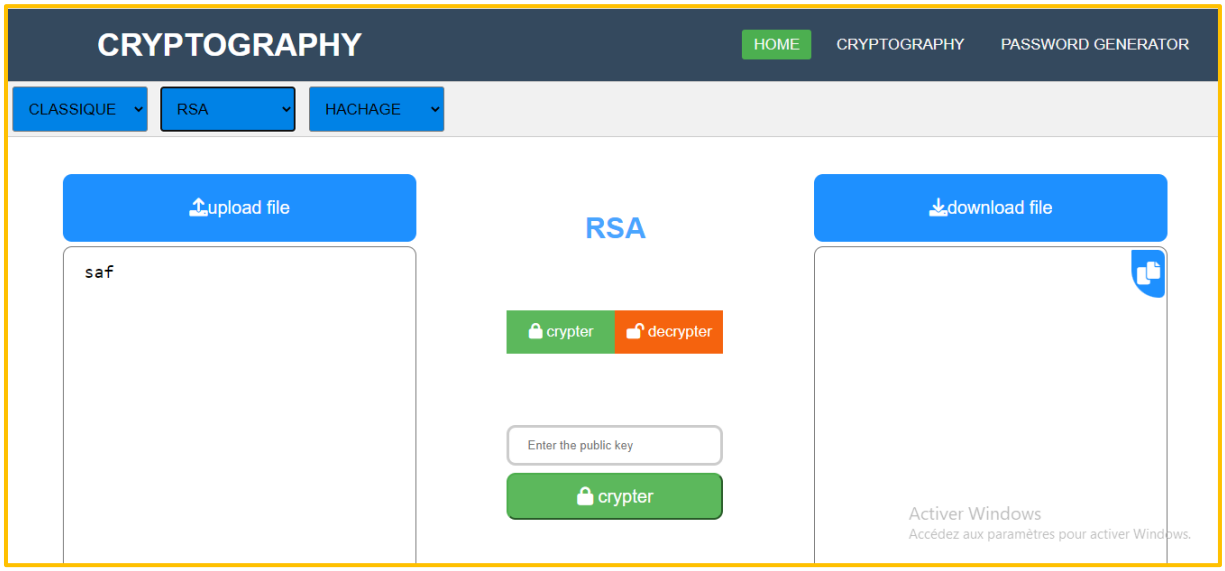
2. عن طريق الأسطر والأعمدة :



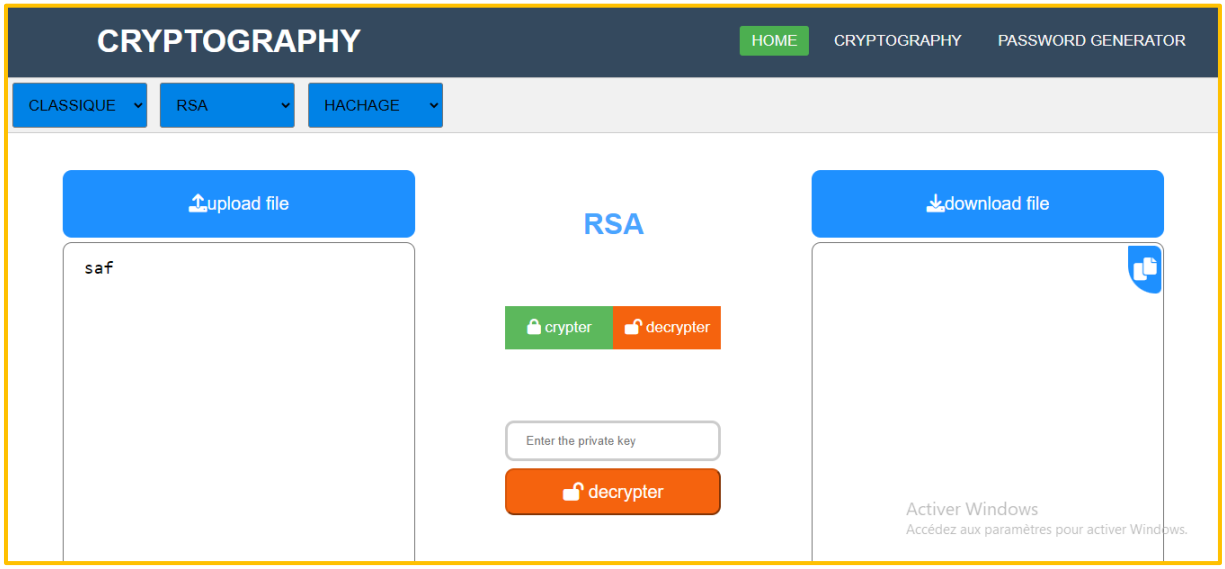
صورة 30:التحويل عن طريق الأسطر والأعمدة

التشفير الحديث Modern :

16-5 RSA :

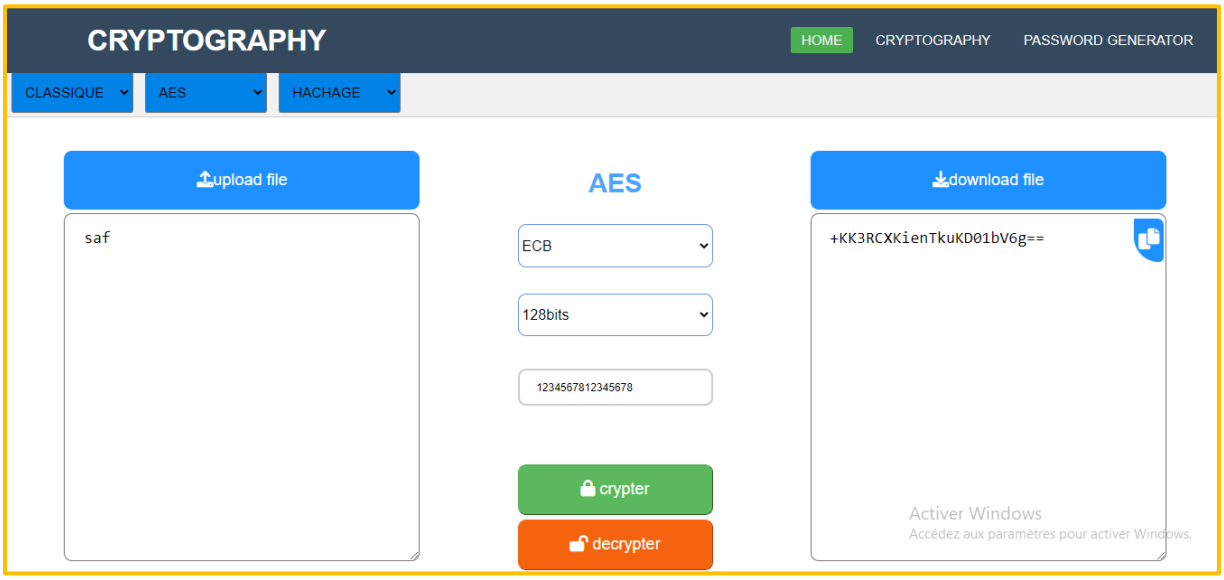


صورة 31:تشفيرRSA



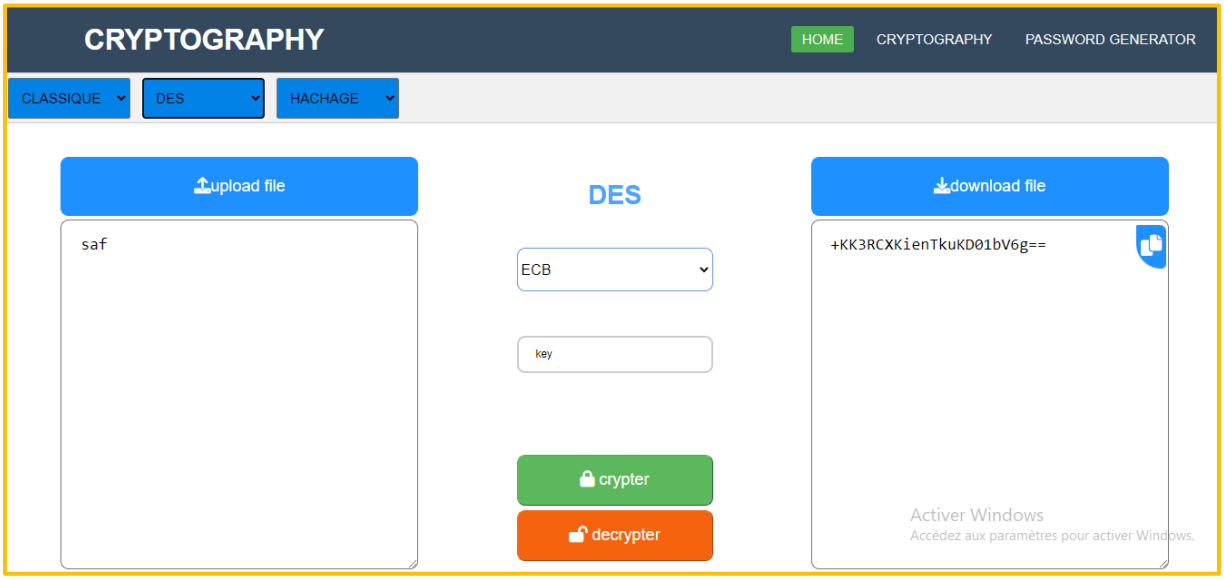
صورة32: تشفيرRSA

17-5 : AES



صورة 33:تشفير وفك تشفير AES

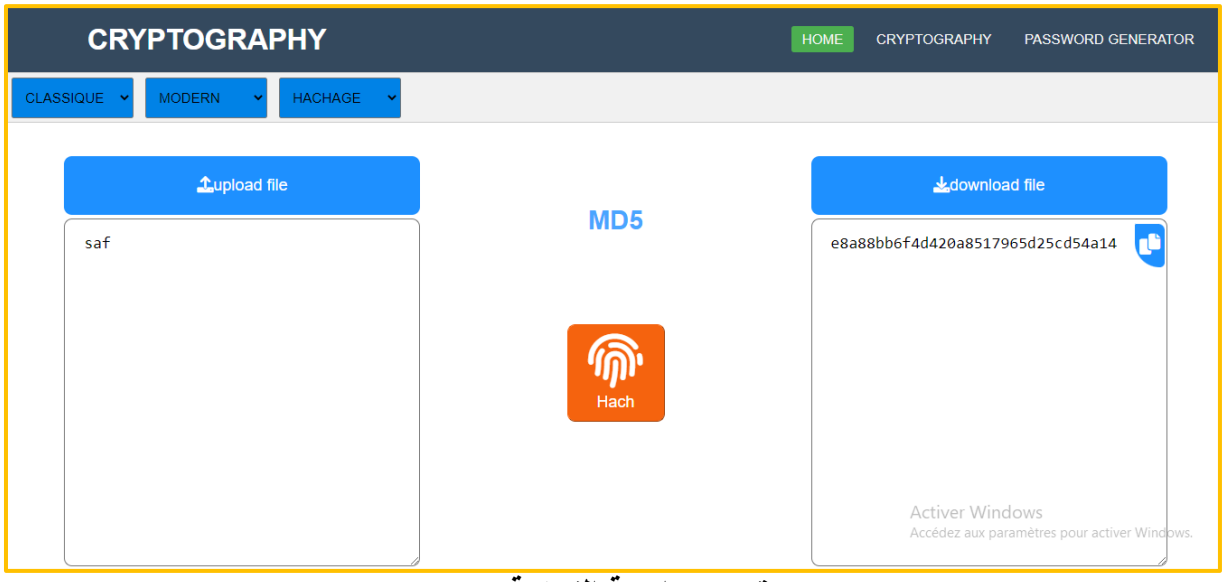
18-5 : DES



صورة 34:تشفير وفك تشفير DES

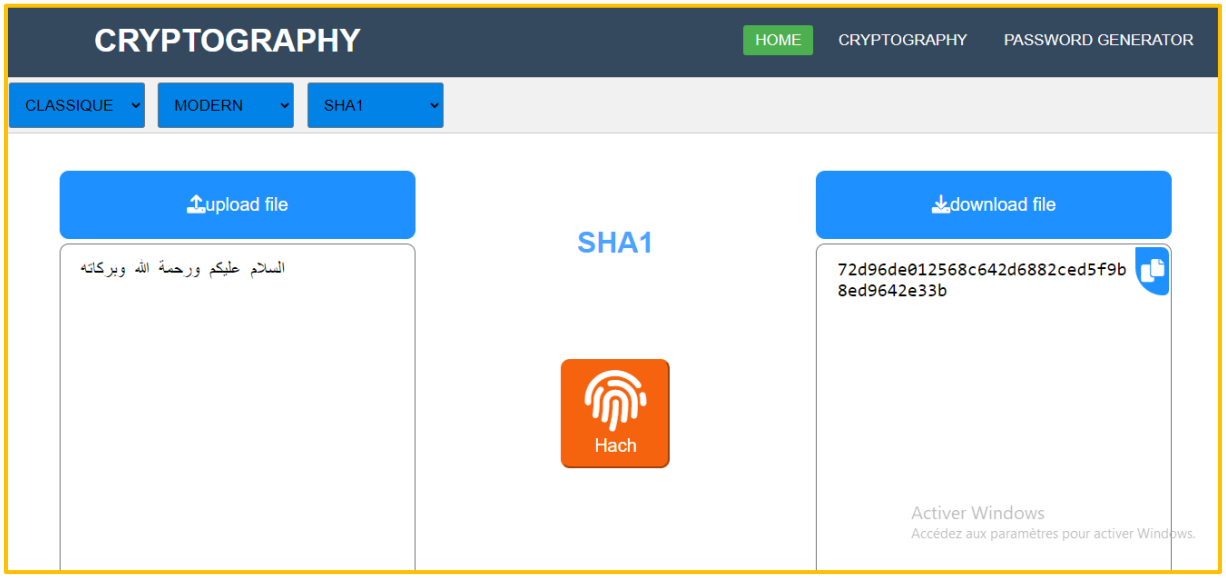
: HACHAGE

: MD5 19-5



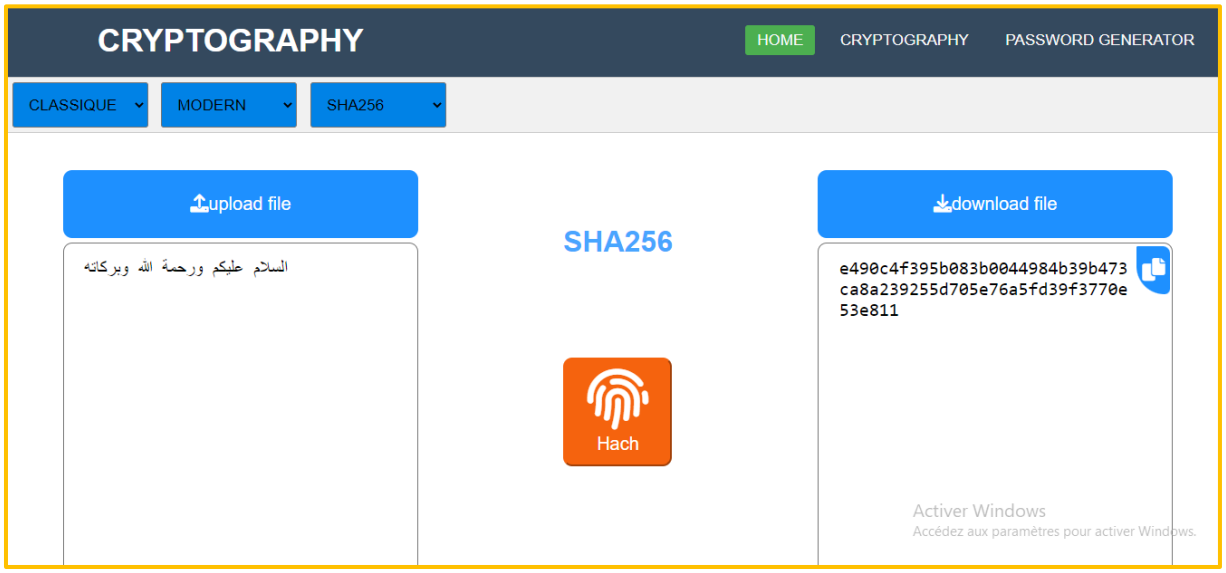
صورة 35:واجهة التجزئة بMD5

: SHA1 20-5



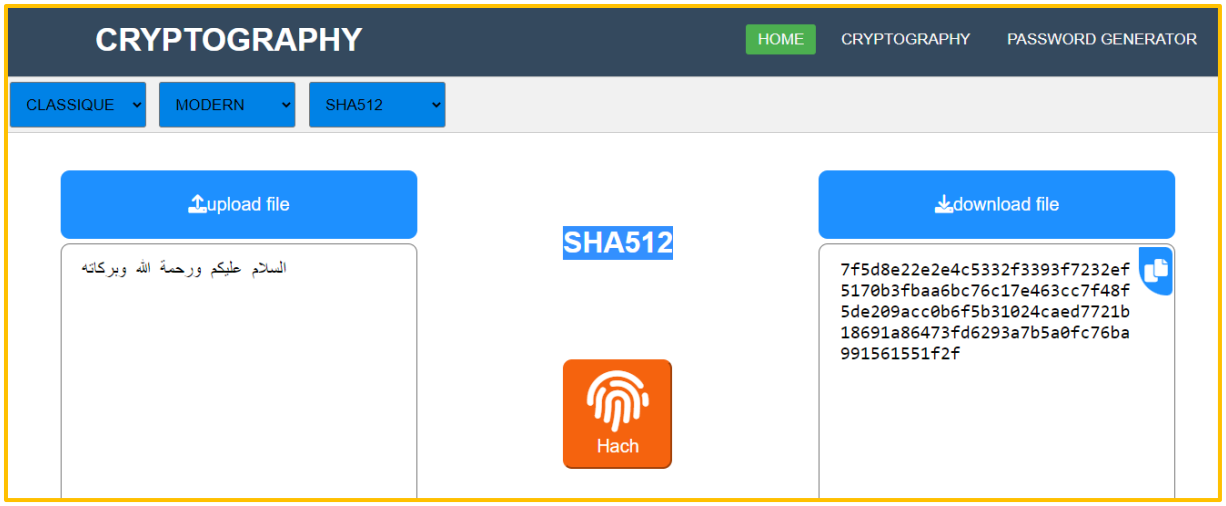
صورة 36:واجهة التجزئة بSHA

21-5 : SHA256



صورة 37:واجهة التجزئة بSHA256

22-5 : SHA512

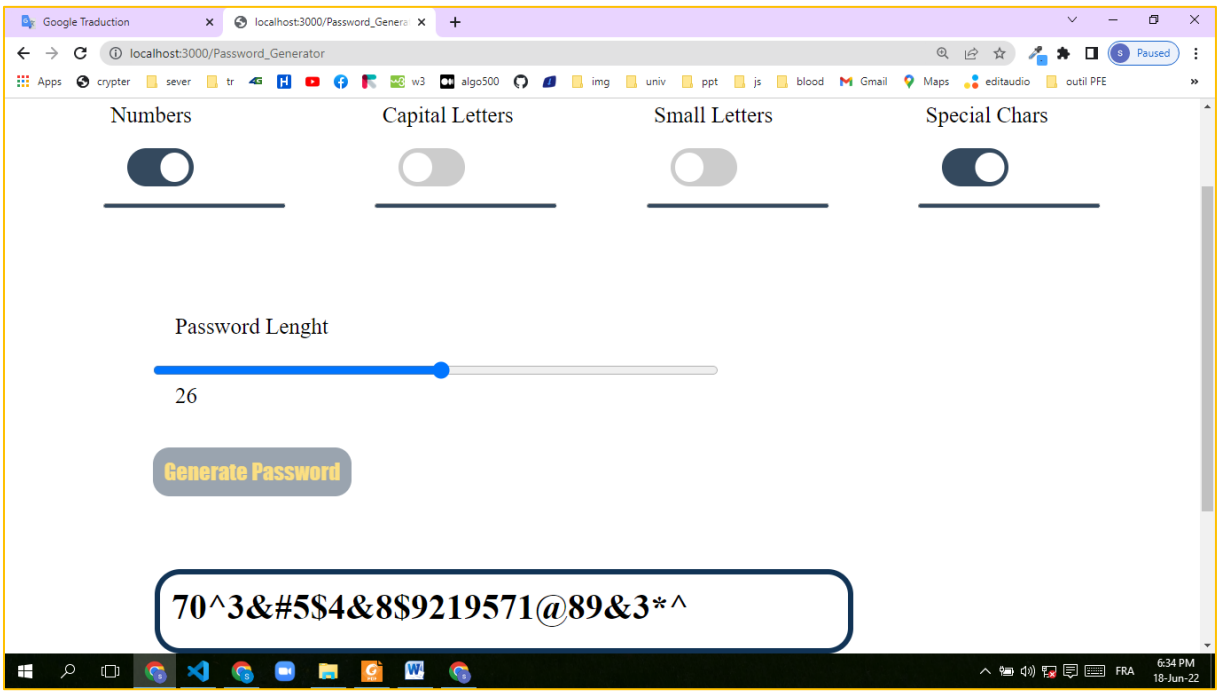


صورة 38:واجهة التجزئة بSHA512

23-5 واجهة توليد الكلمات السرية :

في واجهة توليد الكلمات السرية يمكن للمستخدم ان يتحصل على كلمة سر قوية و يمكنه ايضا اختيار مما تتكون(احرف او رموز او ارقام) و ايضا يمكنه اختيار الطول

ملاحظة : الطول يكون اقل من خمسين حرفا



صورة 39: واجهة توليد الكلمات السرية