مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (5) العدد(8) 2021

Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (5) No. (8) 2021

تصميم وبناء بروتوكول أمني لحماية الوثائق الرقمية

د. بسيم صالح برهوم *

(تاريخ الإيداع 27 /7 /2021 . قُبِل للنشر في 20 /9 /2021)

□ ملخّص □

تزداد حجم الهجمات السبيرانية وتزداد فعاليتها يوماً بعد يوم ولاسيما مع استخدام منطق عمل الحوسبة الكمومية Quantum Computing (QC)، وازدياد أعداد الأجهزة المرتبطة بإنترنت الأشياء Internet of thing (IOT) الأمر الذي يفرض البحث عن آليات حماية تستطيع مقاومة الهجمات الحديثة، ويمكنها التعامل مع البيئات المحدودة الموارد (زمن التنفيذ، والطاقة اللازمة للعمل ومساحة التخزين)، الأمر الذي سيؤدي إلى فشل معظم خوارزميات التشفير Encryption Algorithms في تأمين الحماية اللازمة، وسينتج عن ذلك ارتفاع الخسائر الأمنية في مختلف نواحي الحياة، من هنا تظهر الحاجة إلى وجود آلية لتأمين الحماية اللازمة للمعلومات الرقمية من تلك الهجمات و آثارها المدمرة.

يقترح هذا البحث بروتوكولاً أمنياً متكاملاً يعتمد بشكل أساسي على خوارزمية المنحنيات الإهليليجية ECC Elliptic curve cryptography في مرحلة التوقيع الرقمي Digital Signature، وفي مرحلة ضمان أمن المفاتيح المستخدمة في خوارزمية التشفير المتناظرة Symmetric Encryption Algorithm، وذلك بهدف ضمان العمل مع الحوسبة الكمومية، و في البيئات المحدودة الموارد، لأن معظم خوارزميات التشفير الحالية لا تستطيع الصمود أمام آليات عمل الحوسبة الكمومية، من هنا جاء التركيز على خوارزمية ECC التي تمتلك إمكانيات مميزة في فضاء المفاتيح والهامش الأمني، وفي مجال السرعة، إضافة إلى محدودية الطاقة التي تحتاجها.

الكلمات المفتاحية : حماية المستندات، خوارزمية ECC، المشفر العياري AES، خوارزمية SHA512، إثبات الهوية، سلامة الرسائل، مقاومة الحوسبة الكمومية، التوقيع الرقمي ، بروتوكول أمني.

* عضو هيئة تدريس في قسم البرمجيات ونظم المعلومات – كلية الهندسة المعلوماتية – جامعة تشرين - سوريا.

مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (5) العدد(8) 2021

Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (5) No. (8) 2021

Design and build a security protocol to protect digital documents

Dr: Baseem Saleh Barhoum *

(Received 27 / 7/ 2021 . Accepted 20 / 9 / 2021)

□ ABSTRACT □

The size of cyber- attacks is increasing and their effectiveness is increasing day by day, especially with the use of the logic of quantum computing (QC), and the increase in the number of devices connected to the Internet of things (IOT), which necessitates the search for protection mechanisms that can resist modern attacks, and can it deal with environments that have limited resources from execution time, the power needed to work and storage space, which will lead to most encryption algorithms faltering in securing the necessary protection, and this will result in high security losses in various aspects of life, hence the need for a mechanism to secure the necessary protection digital information from those attacks and their devastating effects.

This paper proposes an integrated security protocol based mainly on the ECC elliptic curve cryptography algorithm in the digital signature stage, and in the stage of ensuring the security of the keys used in the Symmetric Encryption Algorithm, with the aim of ensuring work with quantum computing, and in limited environments. Resources, because most current encryption algorithms cannot withstand the workings of quantum computing, hence the focus on the ECC algorithm, which has distinct capabilities in key space and security margin, and in the field of speed, in addition to the limited power it needs.

Keywords: document protection, ECC algorithm, AES standard cipher, SHA512, proof of identity, message integrity, quantum computing resistance, digital signature , security protocol.

* Lecturer – Department of Software and Information Systems – Faculty of Informatics engineering – Tishreen University – Syria.

1. مقدمة

لقد تطورت التكنولوجيا على مر السنين، من الرسائل المكتوبة بخط اليد إلى الهواتف المحمولة، ومن مكالمات الفيديو إلى الهواتف الذكية، ولكن مع ظهور تلك الابتكارات الجديدة، زادت أيضًا إمكانية تعرض المستخدمين لهجمات متنوعة، ولاسيما في ظل ازدياد حجم الهجمات السبيرانية المرتبطة بإنترنت الاشياء وازدياد فعاليتها، وذلك من خلال استخدام منطق عمل الحوسبة الكمومية التي تسمح بزيادة سرعة المعالجة حيث يأخذ البت قيم متعددة بين الصفر والواحد باحتمالات مختلفة وسيؤدي هذا إلى انهيار معظم خوارزميات التشفير أمام الهجمات التي ستقودها الحواسيب الكمومية، وسينتج عن ذلك ارتفاع الخسائر الأمنية في مختلف نواحي الحياة (خسائر الهجمات السيبرانية هذا العام وفق تقرير الأمم المتحدة تجاوزت 6 تريليون دولار)، وبالتالي فإن الحاجة إلى مزيد من الأمان وأشكال الاتصال الموثوقة أصبحت الآن ضرورية أكثر من أي وقت مضى. يلعب التشفير Encryption بأشكاله وأساليبه المتنوعة دورًا رئيسيًا في عمليات حماية البيانات و الاتصالات الأمنة و في ضمان سلامة المستندات والتحقق من صحة مصدرها.

نقوم في هذا البحث بتصميم وبناء بروتوكولاً أمنياً متكاملاً يقوم على استخدام هجين لتقنيات التشفير المتناظر Symmetric واللامتناظر Asymmetric لتحقيق الأهداف الأمنية الأساسية، من سلامة و تكاملية Integrated، ووثوقيه Authentication، وسرية Security. ويستخدم هذا البروتوكول أقل ما يمكن من الموارد الحسابية، ويستطيع مقاومة الهجمات الحديثة، حيث يسمح لنا استخدام خوارزمية المنحنيات الإهليلجية ECC زيادة أطوال المفاتيح المستخدمة وفقاً للسويات الأمنية المطلوبة [7]، ويضمن أقل ما يمكن من التكلفة الزمنية والعتادية وذلك في عملية إثبات الهوية من خلال التوقيع الرقمي على البصمة Digest الناتجة عن خوارزمية SHA512 والتي تضمن سلامة الوثيقة من التعديل، وعملية ضمان أمن المفاتيح السرية المشتركة التي تستخدمها الخوارزمية التناظرية القياسية AES.

2. مشكلة البحث

التحدي الأكبر الآن يتمثل في عملية استخدام أنظمة التشفير بنوعيه المتماثل Symmetric وغير المتماثل Asymmetric بشكل يضمن الوقوف في وجه الهجمات السبرانية في عصر انترنت الاشياء بما تحويه من أعداد كبيرة من الأجهزة المختلفة والتي تنمو وتزداد بشكل كبير ويومي ولاسيما مع ظهور عصر الحوسبة الكمومية التي ستفرض تغيير في غالبية خوارزميات التشفير المستخدمة، تتلخص مشكلة البحث الأساسية في استخدام توليفة من خوارزميات التشفير التي تستطيع التكيف مع الاحتياجات الأمنية الجديدة في بيئات محدودة الموارد ولاسيما الأجهزة الصغيرة والبطاقات الذكية.

3. أهداف البحث

يهدف البحث بشكل أساسي إلى تصميم وبناء بروتوكولاً أمنياً باستخدام توليفة من خوارزميات التشفير المتناظر Symmetric واللامتناظر Asymmetric مع خوارزميات إنتاج البصمة الرقمية Digital Digest، وخوارزميات التوقيع الرقمي Digital Signature لتحقيق الأهداف الأساسية لأمن المعلومات وتطبيقها على الوثيقة المطلوبة بشكل يحقق تكلفة حسابية منخفضة وبالتالي سرعة إنجاز أعلى، وبمستوى أمني عالي ومناسب للبيئات

المحدودة الموارد وللأجهزة المحمولة وإنترنت الأشياء Internet of thing (IOT)، حيث سيتم التركيز على اختيار خوارزميات قادرة على مقاومة القدرات العالية للحواسيب الكمومية التي بدأت بالظهور في العديد من بلدان العالم، لذلك سنقوم وعلى غير العادة المتبعة باستخدام الخوارزميات الإهليليجية ضمن أكثر من خطوة من خطوات البناء وصولا إلى تحقيق الأهداف الأمنية ككتلة واحدة وليس ككتل متلاحقة.

4. سناريو العمل :

1- دراسة مختصرة لكل من الخوارزميات المستخدمة وفق ترتيب استخدامها في بناء البروتوكول وهي :

a) خوارزمية SHA512 لإنتاج بصمة الوثيقة الرقمية من أجل ضمان سلامتها من التعديل.

b) خوارزمية المنحنيات الإهليليجة ECC بشكلها المناسب لعملية التوقيع الرقمي (إثبات الهوية).

c) خوارزمية التشفير المعيارية AES128 لتأمين سرية حزمة البيانات مع التوقيع.

d) خوارزمية ECC بشكلها المناسب لإجراء تشفير للمفتاح السري المستخدم.

2- تصميم البروتوكول وتحديد توضع الخوارزميات والمفاتيح المستخدمة.

3- بناء البروتوكول وتحقيقه عمليا على مثال يوضح الفاعلية في تحقيق الأهداف الأمنية.

1- الدراسة المرجعية

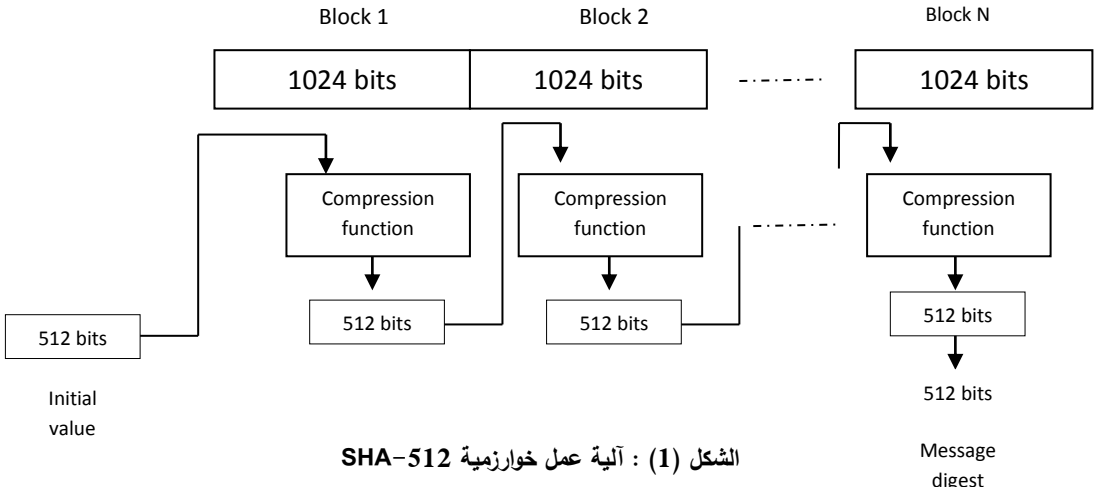
توجد العديد من الابحاث التي استخدمت خوارزمية المنحنيات الإهليليجية ECC ذات المفتاح العام، وتوجد العديد من الابحاث التي استخدمت خوارزمية AES في ضمان السرية [5]، و ابحاث اخرى استخدمت ديفي هيلمان في إنتاج المفاتيح السرية. لكن لا يوجد استخدام لهذه التوليفة المقترحة والتي ستؤدي إلى زيادة في المستوى الأمني مع تقليل في عمليات المعالجة المطلوبة، وبالتالي زيادة في سرعة الانجاز، وذلك من خلال استخدام أطوال مفاتيح قصيرة نسبيا، وهذا سيكون ذات أثر واضح ولاسيما في البيئات المحدودة الموارد[2]، مع إمكانية زيادة أطوال المفاتيح المستخدمة من أجل زيادة السوية الأمنية وفقا لطبيعة الحالة المطلوبة، ولاسيما عند تطبيق مبادئ الحوسبة الكمومية الذي سيؤدي إلى إضعاف فاعلية العديد من خوارزميات التشفير الحالية، إضافة إلى تعطيل قدرة غالبية هذه الخوارزميات.

1.1 خوارزمية إنتاج البصمة SHA-512 :

تُنتج هذه الخوارزمية بصمة بطول 512 بت، وتعمل على تقسيم النص إلى كتل طول كل منها 1024 بت ومن ثم إضافة حقل من 128 بت يمثل طول الرسالة الأصلية و إضافة بتات الحشو اللازمة، حيث يتم مزج قيمة ابتدائية مكونة من 512 بت مع أول كتلة من النص لإنتاج البصمة الأولى بطول 512 بت ومن ثم يتم مزج هذه البصمة مع الكتلة الثانية لإنتاج البصمة الثانية وهكذا يتم تكرار هذه الآلية 80 مرة حتى الكتلة

الأخيرة حيث يتم مزج البصمة رقم N−1 مع الكتلة رقم N لإنتاج البصمة الأخيرة ذات الرقم N والتي تمثل بصمة الرسالة ككل[5],[1].

توضح آلية عمل هذه الخوارزمية بالشكل (1) .



الشكل (1) : آلية عمل خوارزمية SHA−512

المنحنيات الإهليلجية Elliptic Curve:

تعتمد فكرة التشفير باستخدام المنحني الإهليلجي على المنحنيات التي تكون متحولاتها وعواملها مقيدة بعناصرر من حقل منتهي حيث تستخدم معادلة من الشكل

$$y^2 \bmod p = (x^3 + ax + b) \bmod p \qquad (1)$$

تتألف المجموعة E(a,b) من كل أزواج الأعداد الصحيحة (x,y) التي تحقق المعادلة (1)، مع نقطة اللانهاية O

نعرف عملية الجمع + على المجموعة E(a,b)، حيث a و b تحقق المعادلة [14] :

$$y^2 = x^3 + ax + b$$

من وجهة نظر هندسية كما يلي :

1. نقطة اللانهاية O

تحقق $O = -O$، ومن أجل أية نقطة e من المنحني الإهليلجي فإن : $e + O = e$

2. عندما $e = (x, y)$

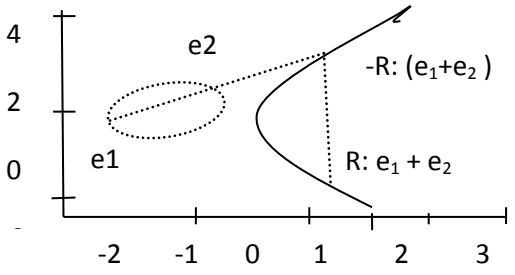
فإن $-e = (x, -y)$ ، وبالتالي فإن : $e + (-e) = e - e = 0$

3. لجمع نقطتين e_1

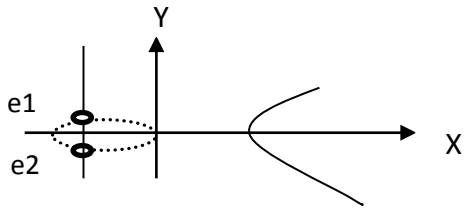
و e_2 مختلفتين بالإحداثي X، نرسم خطّاً مستقيماً بينهما و نوجد R نقطة تقاطعه مع المنحني، نعرف الجمع $e_1 + e_2$ على أنه نظير نقطة التقاطع بالنسبة للمحور X [15] . كما هو موضح بالشكل (2).

4. يمكن

توضيح عملية جمع النقطتين e و e– المشتركتين بالإحداثي x هندسيا بمستقيم عمودي يقطع المنحني في اللانهاية. لذلك نقول أن $O = (-e) + e$ ، و يوضح الشكل (3) هذه الحالة.

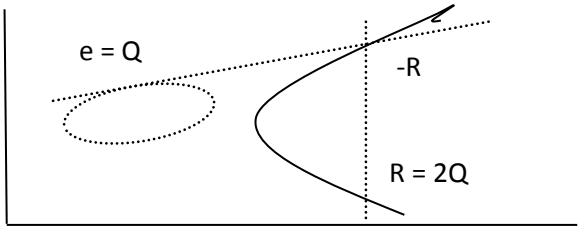


الشكل (2) : بنية الجمع على المنحني $y^2 = x^3 - x$



الشكل (3): جمع نقطتين كل منهما عكس الاخرى (e + (– e) = o)

5. لمضاعفة النقطة Q، نرسم المستقيم المماس للمنحني في هذه النقطة، ونوجد نقطة تقاطع أخرى R. والتي سيكون عندها $Q + Q = 2Q = -R$ ، مبين في الشكل (4).



الشكل (4) : يوضح عملية مضاعفة نقطة على المنحني

خوارزمية المنحني الإهليليجي ECC :

خوارزمية غير تناظرية Asymmetric تعمل مع مفتاحين عام وخاص، ويمكن أن تستخدم في التوقيع الرقمي، و التشفير، وفي تبادل المفاتيح key exchange ، تم تصميمها من قبل شركة (IBM) و بالتعاون مع جامعة واشنطن، و تعتمد على المنحنيات الإهليليجية، والتي هي عبارة عن معادلات تكعيبية بمتحولين من الشكل[15] $y^2 = x^3 + a\,x + b \bmod P$ ، حيث $4a^3 + 27\,b^2 \neq 0$

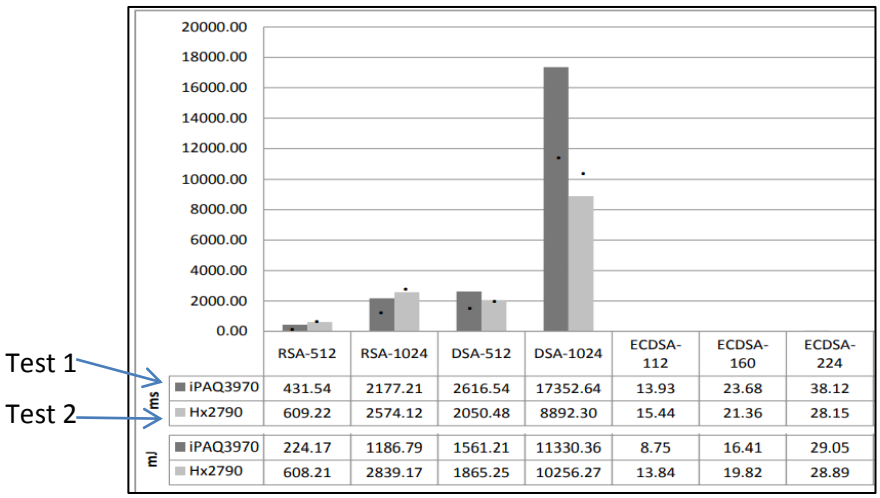
وبالتالي فإننا نحتاج إلى معرفة المعاملات a و b و p وG، حيث a و b يستخدمان في معادلة المنحنى (1) و G النقطة المولدة.

1.2 استخدام خوارزمية ECC في التوقيع (ECDSA)

- نختار المنحني الإهليليجي $E_p(a,b)$، حيث p عدد اولي.
- نختار المفتاح الخاص d_A، ونحسب : $e_2(x_2,y_2) = d_A \times e_1(x_1,y_1)$ (نقطة ثانية على المنحني).
- نختار عدد اولي q اخر، فيكون (a,b,p,q,e_1,e_2) المفتاح العام e_A.

- نختار عدد سري عشوائي r قيمته بين 1 و q−1.
- نحسب : $P(u,v) = r \times e_1(x_1,y_1)$ (نقطة ثالثة على المنحني)، مع ملاحظة أن ضرب نقطة من المنحني بعدد ثابت k يعني إضافة النقطة إلى نفسها k مرة.
- نأخذ قيمة X للنقطة P(u,v) على انها S_1 ($S_1=u \bmod q$).
- لحساب قيمة S_2 نحتاج إلى بصمة الرسالة أو الوثيقة H(M) (ناتج تطبيق خوارزمية SHA)، و يمكننا التعبير عنها بعدد صحيح كبير H، ثم نحسب S_2 ($S_2 = r^{-1} (H + d_A \times S_1) \bmod q$)، حيث d_A هو المفتاح الخاص للطرف A و S_1 هو الإحداثي X للنقطة ($P(u,v) = r \times e_1(x_1,y_1$.
- يرسل الطرف A كل من التوقيع (S_1,S_2)، والرسالة M للطرف الاخر B.
- للتحقق من صحة التوقيع يحتاج الطرف B إلى المفتاح العام $e_2 = d_A \times e_1(x_1,y_1)$ للطرف A، ثم نحسب $P(x,y) = [S_2^{-1} \times H \times e_1(x_1,y_1) + S_2^{-1} \times S_1 \times e_2(x_2,y_2)] \bmod q$
- إذا كان الإحداثي x للنقطة P يساوي S_1 ($x=S_1 \bmod q$) ، يتم قبول التوقيع، وخلاف ذلك نرفض.[4]

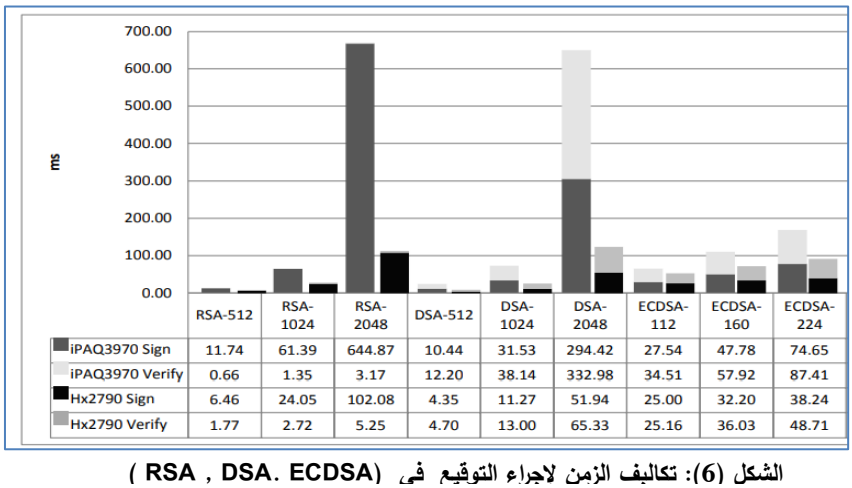
ويبين الشكل (5) مقارنة بين أهم خوارزميات التوقيع الرقمي من حيث التكلفة الزمنية وتكلفة الطاقة لتوليد المفاتيح.



	RSA-512	RSA-1024	DSA-512	DSA-1024	ECDSA-112	ECDSA-160	ECDSA-224
ms iPAQ3970	431.54	2177.21	2616.54	17352.64	13.93	23.68	38.12
ms Hx2790	609.22	2574.12	2050.48	8892.30	15.44	21.36	28.15
mJ iPAQ3970	224.17	1186.79	1561.21	11330.36	8.75	16.41	29.05
mJ Hx2790	608.21	2839.17	1865.25	10256.27	13.84	19.82	28.89

الشكل (5): تكايف الزمن والطاقة لتوليد المفاتيح في (RSA , DSA. ECDSA)

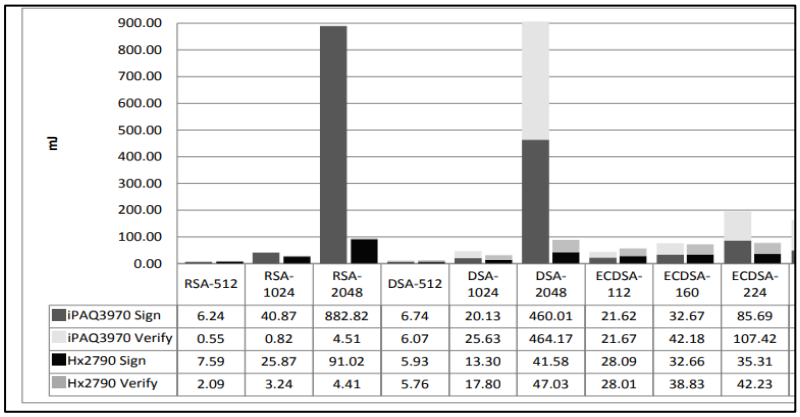
نقدم في الشكل (6) مقارنة بين أهم خوارزميات التوقيع الرقمي من حيث التكلفة الزمنية اللازمة للتوقيع وللتحقق من صحة التوقيع (في بيئتي اختبار مختلفة) [6]



	RSA-512	RSA-1024	RSA-2048	DSA-512	DSA-1024	DSA-2048	ECDSA-112	ECDSA-160	ECDSA-224
iPAQ3970 Sign	11.74	61.39	644.87	10.44	31.53	294.42	27.54	47.78	74.65
iPAQ3970 Verify	0.66	1.35	3.17	12.20	38.14	332.98	34.51	57.92	87.41
Hx2790 Sign	6.46	24.05	102.08	4.35	11.27	51.94	25.00	32.20	38.24
Hx2790 Verify	1.77	2.72	5.25	4.70	13.00	65.33	25.16	36.03	48.71

الشكل (6): تكاليف الزمن لاجراء التوقيع في (RSA , DSA. ECDSA)

بينما يغرض الشكل (7) مقارنة بين أهم خوارزميات التوقيع الرقمي من حيث تكلفة الطاقة اللازمة للتوقيع وللتحقق من صحة التوقيع (في بيئتي اختبار مختلفة) [6]، وهذا سوف يتم مناقشته في الدراسة التحليلية.

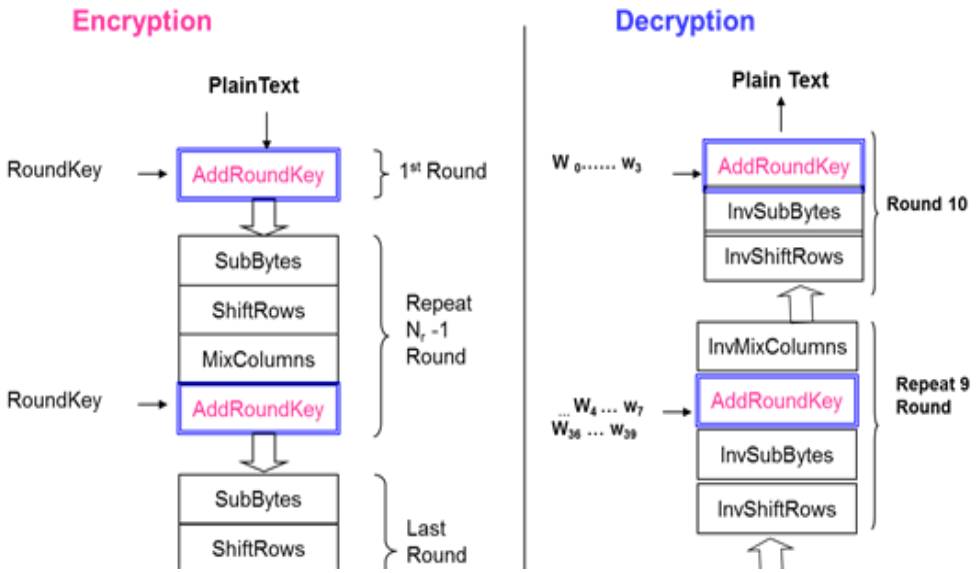


	RSA-512	RSA-1024	RSA-2048	DSA-512	DSA-1024	DSA-2048	ECDSA-112	ECDSA-160	ECDSA-224
iPAQ3970 Sign	6.24	40.87	882.82	6.74	20.13	460.01	21.62	32.67	85.69
iPAQ3970 Verify	0.55	0.82	4.51	6.07	25.63	464.17	21.67	42.18	107.42
Hx2790 Sign	7.59	25.87	91.02	5.93	13.30	41.58	28.09	32.66	35.31
Hx2790 Verify	2.09	3.24	4.41	5.76	17.80	47.03	28.01	38.83	42.23

الشكل (7): تكاليف الطاقة لإجراء التوقيع في RSA , DSA.

1.3 المشفر المعياري Rijndael

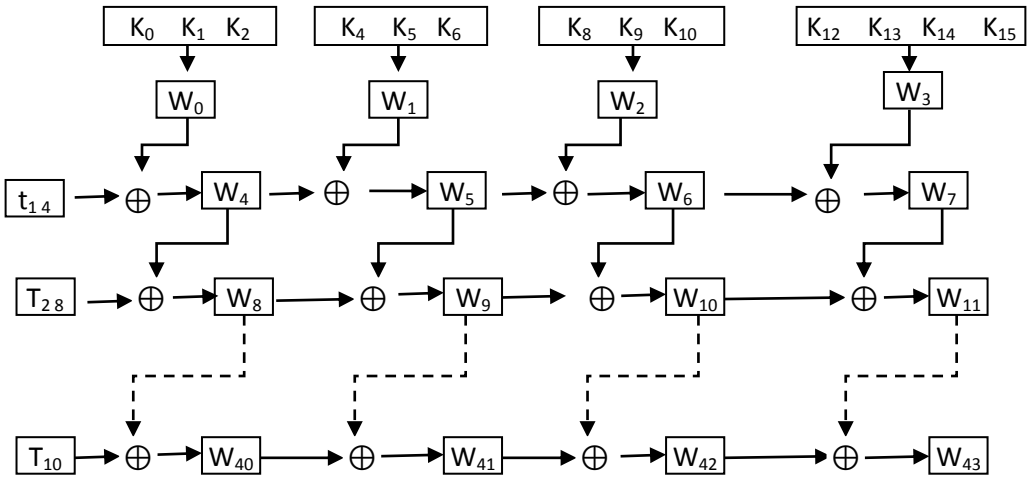
مشفر معياري متقدم AES: Advanced Encryption Standards ، يعتمد على بعض العمليات البسيطة والخاصة في الحقل ($GF(2^8)$ ، ويعمل بكفاءة عالية برمجيا وعتاديا وعلى منصات عمل مختلفة[8]، وهو عبارة عن خوارزمية كتلية Block Cipher تعمل بطول مفتاح وكتلة متغيرين (128، 192، 256) بت، وتقوم على تكرار عدة تحويلات، الشكل (8) يوضح بنيتها الأساسية .



الشكل (8): يبين بنية خوارزمية AES

توليد المفاتيح الجزئية :

يلزم N+1 مفتاح مع N تكرار، حيث يتم استخدام كلمات المفتاح الأساسي في صناعة المفاتيح الفرعية اللازمة من خلال الحصول على 4 X (N+1) كلمة. $W_0, W_1, W_2, \ldots, W_{4(N+1)-1}$، فمثلا يلزمنا 44 كلمة مع النسخة AES-128 بتكراراتها العشر [1]، يتم الحصول على هذه الكلمات وفق الشكل (9).



الشكل (9) :آلية توليد كلمات المفاتيح الفرعية في AES-128

التشفير باستخدام AES128 : تتم عملية التشفير بمعالجة ثمانيات النص (بايتات) وفق أربع تحويلا أساسية :

1- تحويل ننثر البايتات SubByte في صندوق التعويض S-box الخاص بالخوارزمية.
2- تحويل إزاحة الأسطر ShiftRows بشكل دوراني نحو اليسار بمقادير تابعة لرقم السطر .
3- تحويل مزج الأعمدة MixColumns من خلال ضرب كل عمود بمصفوفة مربعة ثابتة.
4- تحويل إضافة مفتاح التكرار AddRoundKey .
5- نكررهذه الخطوات عشر مرات، مع إلغاء تحويل مزج الأعمدة في التكرار الأخير.

تتمتع خوارزمية AES المعيارية بهامش أمني عالي وبقدرات مميزة في مجال البيئات المحدودة الموارد[13].

1.4 استخدام خوارزمية ECC في التشفير Encryption:

- يختار الطرف A مفتاحه الخاصd_A، ويحسب مفتاحه العام $e2(x2,y2) = d_A \times e_A$، وينشره.
- يختار الطرف B مفتاحه الخاصd_B ، ويحسب مفتاحه العام $e2(x2,y2) = d_B \times e_B$، وينشره.
- الطرف المرسل يرمز النص الأصلي على شكل نقاط على المنحني باستخدام خوارزميات الترميز المناسبة.
- يختار المرسل عدد r و يحسب النص المشفر : $C2 = M + r \times e_2$, $C1 = r \times e1$، حيث e_2 مفتاح المستقبل العام، و M الرسالة المطلوب تشفيرها، و e1 نقطة من المنحني $E_P(a, b)$ [9].

و يتم فك التشفير Decryption وفق الآلية الآتية:

- يستخلص الطرف المستقبل النص الأصلي M من النص المشفر C2 , C1، بتطبيق العلاقة الآتية :

M= C2 – (d × C1)، طبعاً عملية الطرح هنا تمثل عملية جمع مع inverse، و يتم إيجاد النص الأصلي انطلاقاً من النقطتين C2 , C1 وباستخدام مفتاحه الخاصّ d. مثلاً إذا أخذنا المنحني E_{13} (1,1)، وكان لدينا العلاقة: [2− (11−4) 5] mod 13، فإنه وبإجراء الحساب نحصل على الناتج 2،

في الجدول (1) نبين زمن التشفير وفك التشفير لكل من خوارزمية RSA و خوارزمية ECC [9].

الجدول (1) : يبين زمن التشفير وفك التشفير بالثانية مع 256 bit

Input: 256 bits						
Security Bit Level	**Encryption**		**Decryption**		**Total Time**	
	ECC Enc. Time	RSA Enc. Time	ECC Dec. Time	RSA Dec. Time	ECC Total Time	RSA Total Time
80	7.9240	0.5596	22.8851	19.3177	30.8091	19.8772
112	39.7008	0.5815	26.3331	102.0337	66.0339	102.6153
128	58.4386	0.5611	27.4060	209.6086	85.8446	210.1697
144	77.5034	0.5718	32.1522	311.0649	109.6556	311.6368

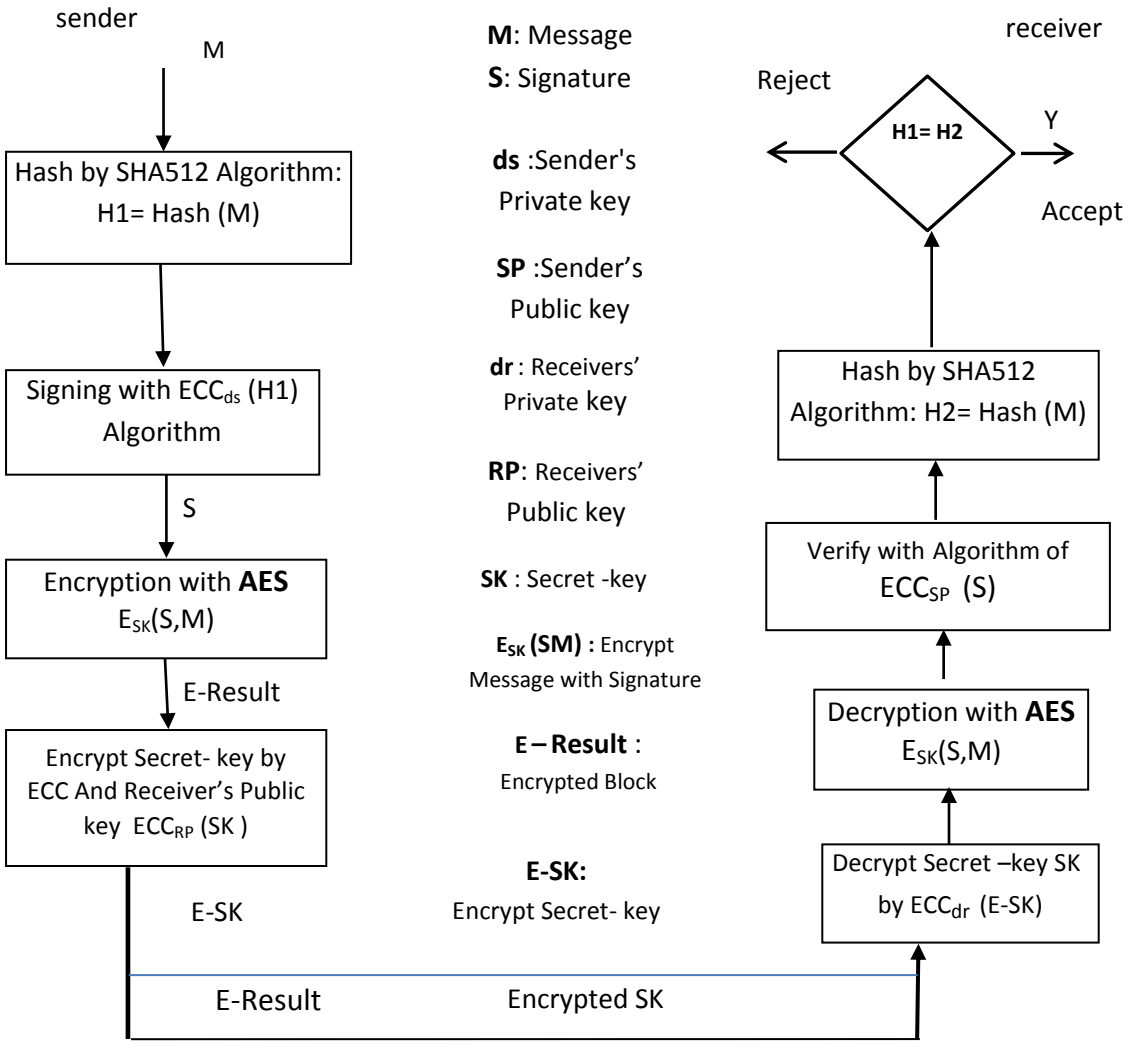
2- تصميم البروتوكول المقترح :

من القضايا الرئيسية التي يجب العمل عليها عند تصميم البروتوكول استخدام توليفة من الخوارزميات خفيفة الوزن لتحقيق سرعة عالية وذاكرة أقل، حيث ينصب التركيز الرئيسي على تقليل مساحة التخزين، وتقليل استهلاك الطاقة ولاسيما للأجهزة الذكية خفيفة الوزن، لذلك قمنا باختيار خوارزمية تشفير المنحنى الإهليجي ECC، حيث أن حجم مفتاحها أقل بكثير من حجوم مفاتيح خوارزميات المفتاح العام الاخرى، وتحقق مستوى عالي من الامن مقارنة بخوارزميات التشفير الأخرى، وهذا يجعلها قادرة نسبيا على الصمود عند استخدام مبادىء الحوسبة الكمومية لأنها تستطيع زيادة أطوال المفاتيح المستخدمة وبسرعات جيدة وباستهلاك طاقة أقل، ومن أجل تحقيق ذلك :

- نستخدم خوارزمية SHA-512 لإنتاج بصمة رقمية Digest بطول 512 بت مهما كان طول الوثيقة.
- نستخدم خوارزمية ECC من أجل إجراء عملية توقيع بصمة الوثيقة.
- نستخدم الخوارزمية المعيارية AES128، مع إمكانية زيادة طول المفتاح السري ولاسيما عند الحديث عن الحوسبة الكمومية وقدراتها العالية، وهذا سيتطلب زيادة طول مفتاح الخوارزميات الأخرى المستخدمة في عمليات التوقيع وتشفير المفاتيح المستخدمة في عمليات ضمان السرية.
- نستخدم خوارزمية المنحني الإهليليجيECC من أجل تشفير المفتاح السري الذي سيتم استخدامه في التشفير وذلك من أجل إرساله بشكل آمن إلى الطرف المستقبل.

3- بناء البروتوكول وتحقيقه عمليا

نبين في الشكل (10) البنية الأساسية للبروتوكول المقترح.



الشكل (10): البنية الأساسية للبروتوكول المقترح

دراسة تحليلية لاستخدام البروتوكول المقترح

أجرينا دراسة تحليلية ومقارنة لعدة أبحاث تهتم بموضوع تقييم أداء خوارزميات التشفير المستخدمة في صناعة التوقيع الرقمي من أجل الوصول إلى نتائج دقيقة تؤيد استخدام توليفة الخوارزميات في البروتوكول المقترح، وتم

التركيز على فكرة زيادة أطوال المفاتيح لمجابهة المقتضيات الأمنية لعصر الحوسبة الكمومية وتوفير الأمن للأجهزة المتعددة والمختلفة في إنترنت الأشياء من خلال تقليل الموارد اللازمة للحصول على المستويات الأمنية المطلوبة.

اولاً : من الناحية الأمنية ومقاومة الحوسبة الكمومية

يعتمد الحاسوب الكمومي على بت الكم (كيوبت)، وعلى عكس البت الثنائي(حالة 1 أو 0 في كل مرة)، يمكن للكيوبت، أن يكون في حالتين مختلفتين في نفس الوقت ، يُشار إليهما بـ | 0 > و | 1 >، ويعبر عن ذلك بالصيغة $|\psi> = \alpha | 0> + \beta | 1>$، ويقوم الحاسب الكمي بإجراء الحسابات باستخدام دائرة كمومية تحتوي على بوابات كمومية تحل محل البوابات الرقمية (OR , NOT , AND).

يكمن سر الأمان العالي لخوارزمية ECC من أنه إذا كان لدينا نقطة R = k * P، وكنا نعرف R ونعرف P، فلا توجد طريقة لمعرفة قيمة k، حيث لا توجد عملية طرح نقطي أو قسمة نقطية على المنحنيات الإهليليجية، وبالتالي لا يمكن العثور على النقطة R، و نلاحظ أننا بحاجة إلى r و d_A لحسابS_2 ، ونحتاج إلى S_1 و e2 للتحقق من التوقيع، وبما أن S_1 هو X النقطة $P(u,v) = r \times e1(x1,y1)$، و e_2 $= d \times e_1$ فإنه لا يمكننا حساب d أوr من معرفة e_2 وS_1 ، وهذا يجعل الخوارزمية آمنة، حيث لا توجد طريقة للعثور على المفاتيح الخاصة، ولا لتزوير التوقيع[15].

- تتمتع خوارزمية ECC بإمكانيات كبيرة في مجال هامش طول المفتاح مقارنة مع الخوارزميات الأخرى [3]، كما هو موضح بالجدول (2).

الجدول (2) : يبين أطوال المفاتيح الممكنة

Security Strength	Key size	
Symmetric Key Size	ECC	RSA/DSA/DH
80 bits	160 bits	1024 bits
112 bits	224 bits	2048 bits
128 bits	256 bits	3072 bits
192 bits	384 bits	7680 bits
256 bits	512 bits	15360 bits

من الجدول (2) نجد أن الخوارزمية ECC تتطلب مفتاح أقصر لتحقيق نفس المستوى الأمني الذي تحققه الخوارزميات الأخرى [8]، فمثلا ECC256 تؤمن نفس السوية الأمنية التي تؤمنها RSA3072 أو DSA3072 وعند تطبيق مبادىء الحوسبة الكمومية ستكون الخوارزمية ECC قادرة على زيادة السوية الأمنية لأنها تمتلك هامش أمني عالي للمفاتيح، بمعنى أخر يمكنها زيادة طول المفتاح إلى 384 أو إلى 512 بت دون أثر كبير على الاداء، ولكن الخوزميات الأخرى ستحتاج إلى زيادة طول المفتاح إلى 15360 بت لتحقيق نفس السوية الأمنية وهذا سيؤثر بشكل كبير على الأداء ويجعلها غير مناسبة عملياً.

ثانياً: من ناحية سرعة توليد المفاتيح والطاقة اللازمة

بدراسة أزمنة توليد المفاتيح ,والطاقة اللازمة لذلك في الشكل5 (Test1) نجد أن الزمن اللازم في الخوارزمية ECCوفقاً للسوية الأمنية القياسية 80 bit أقل بكثير من الزمن الذي تحتاجه الخوارزميات الأخرى (من 91 إلى 732 ضعف)، إضافة إلى أن الطاقة اللازمة لتوليد المفاتيح في الخوارزمية ECC وفقاً للسوية الأمنية 80 bit تقل عن الطاقة اللازمة في الخوارزميات الأخرى بـ 72 إلى 690 ضعف كما هو مبين في الجدول (3) .

الجدول (3) : يبين زمن توليد المفاتيح والطاقة اللازمة لذلك وفقاً للسوية الأمنية 80 bit

Security Strength	Total power of key generation (mj)			Total time of key generation (ms)		
Symmetric Key–80 bits	ECC160	RSA1024	DSA1024	ECC160	RSA1024	DSA1024
	16.41	1186.79	11330.36	23.68	2177.21	17352.64

ثالثاً: من ناحية سرعة صناعة التوقيع والتحقق من صحته

بدراسة التكاليف الزمنية لصناعة التوقيع في الشكل (6)، وتكاليف الطاقة من الشكل (7) (Test2) علماً أن الزمن الكلي اللازم بالميلي ثانية، وعلى الطاقة اللازمة لذلك بالميلي جول، كما هو مبين في الجدول (4) .

الجدول (4) : يبين زمن التوقيع والتحقق والطاقة اللازمة وفقا للسويتين الأمنيتين 80 – 112 بت

Security Strength	Key size		Process	Total time Sign and Verify		
Symmetric Key Size	ECC	RSA/DSA	(ms , mj)	ECC	RSA	DSA
80 bits	160 bits	1024 bits	Sin & Verify	68.23	26.77	24.27
			Power	71.49	28.91	31.10
112 bits	224 bits	2048 bits	Sin & Verify	86.25	107.33	117.27
			Power	77.54	95.43	88.61

نلاحظ أنه لتحسين المستوى الأمني من 80 بت إلى 112 بت فإن حاجة الخوارزمية RSA من الزمن تزداد من 26.77 ms إلى 107.33 ms أي بنسبة 300%، وحاجتها من الطاقة تزداد من 28.91 mj إلى 95.43 mj أي بنسبة 230%، وفي خوارزمية DSA ايضاً يزداد الزمن من 24.27,ms إلى 117.27 msأي بنسبة 381% وتزداد الطاقة من 31.10 mj إلى 88.61 mj أي بنسبة 185%، بينما في خوارزمية ECC قإن الزمن يزداد من 68.23 ms إلى 86.26 ms أي بنسبة 26%، وكذلك حاجة الخوارزمية من الطاقة تزداد من 71.49 mj إلى 77.54 mj اي بنسبة 8.5%، أي من أجل السوية الأمنية 112 بت فإن حاجة الخوارزمية ECC من الطاقة لصناعة التوقيع الرقمي أقل بنسبة من 14% إلى 23%، وتزداد أكثر مع المفاتيح الأطول

رابعاً: من ناحية سرعة التشفير وفك التشفير

بدراسة النتائج المبينة في الجدول (1) (دخل 256 يت) ووفقا للأطوال القياسية للمفاتيح في الجدول (2)، نجد أن خوارزمية ECC تحتاج زمن تشفير وفك تشفير يتزايد بشكل بسيط مع تزايد طول المفتاح (66 ثانية مع مفتاح

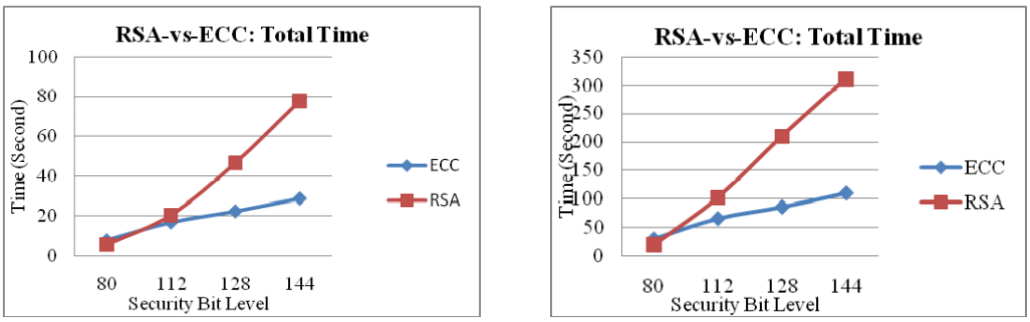
112 بت ويصبح 85 ثانية من أجل مفتاح 128 بت) اي بنسبة 29% ، بينما حاجة الخوارزمية RSA يتزايد بشكل كبير مع تزايد طول المفتاح (102 ثانية من أجل مفتاح 112 بت ويصبح الزمن 210 ثانية من أجل مفتاح 128 بت) أي بنسبة 106%.

الجدول (5) يوضح زمن التنفيذ وفقاً للسويات الأمنية القياسية وبحزمتي دخل مختلفتين (64 – 256bit).

الجدول (5): يبين زمن التشفير وفك التشفير وفقا للسويات الأمنية القياسية

Security Strength	Key size		Total time(S)– (64 bit) Encrption – Decrption		Total time(S) –(256 bit) Encrption – Decrption	
Symmetric Key Size	ECC	RSA/DSA/DH	ECC	RSA	ECC	RSA
80 bits	160 bits	1024 bits	8,0784	5.6738	30.8091	19.8772
112 bits	224 bits	2048 bits	16.9188	20.5743	66.0339	102.6153
128 bits	256 bits	3072 bits	22.4466	46.6454	85.8446	210.1697
144 bits	384 bits	7680 bits	28.7093	77.9027	109.6556	311.6368

بالمقارنة نلاحظ أنه ومع تزايد طول المفتاح تصبح خوارزمية المنحنيات الإهليليجية ECC الاسرع وبمقدار يتجاوز الضعف (مع سوية أمنية 144 بت، حيث زمن التشفير في خوارزمية ECC يكون 109 ثانية مقابل 311 ثانية في خوارزمية RSA)، من الواضح تتفوق ECC من حيث الكفاءة التشغيلية الزمنية والأمان [9]، الشكل (11) يوضح ذلك.



الشكل (11) يبين زمن التشفير الكلي بالثانية مع دخل – 256 بت و 64 بت

نلاحظ أن الخوارزمية ECC تتمتع بكفاءة أعلى مع أطوال مفاتيح أقصر و هذا يعني أن الأجهزة تتطلب طاقة معالجة أقل لتشفير البيانات وفك تشفيرها، مما يجعل ECC مناسبًا جدًا للأجهزة المحمولة وإنترنت الأشياء وحالات الاستخدام الأخرى ذات الموارد الحسابية المحدودة، وفي حالات الاستخدام الأكثر تقليدية مثل خوادم الويب توفر المفاتيح الأصغر أمانًا عالي مع مصافحة SSL أسرع، وهذا يترجم إلى سرعة تحميل أعلى.

من خلال تحليل أداء خوارزميات التشفير المختلفة في أجهزة صغيرة ومقارنة التكاليف الأساسية تبين أن أفضل خوارزمية متماثلة، فيما يتعلق بتكاليف الوقت والطاقة، هي AES، أما في مجال التشفير اللامتناظر، فأن أداء خوارزمية المنحنيات الإهليليجية ECC (سواء في التوقيع أو في التشفير) هو الأفضل من بين الخوارزميات الاخرى، ولاسيما عند الحاجة إلى زيادة طول المفتاح بهدف زيادة السوية الأمنية مع الحفاظ على السرعة والموارد المستهلكة وهذا ما تفتقر إليه أهم خوارزميات المفتاح العام (DSA , RSA) [6].

إضافة إلى إمكانية تضمين خوارزمية ECDSA في أجهزة ذات سعة تخزين وحوسبة محدودة بهدف تقليل التكاليف الحسابية [10].

النتائج والتوصيات

إن استخدام خوارزمية ECC (من أجل التوقيع) في البروتوكول المقترح سيحسن زمن توليد المفاتيح بمقدار 91 ضعف، ويقلل استهلاك الطاقة بأكثر من 72 مرة عن استخدام خوارزمية أخرى وهذا المقادير ستزداد مع زيادة طول المفتاح المستخدم ، أما الزمن اللازم لعملية التوقيع سيتحسن بنسبة تفوق 27%، والطاقة اللازمة للتوقيع ستتحسن بنسبة تفوق 175%، أما في مرحلة تشفير المفتاح المستخدم مع خوارزمية AES فإن الزمن يتحسن بنسبة 77%، أي أن استخدام خوارزمية ECC يحافظ على تزايد بسيط في الزمن وفي الطاقة بينما الخوارزميات الأخرى تصبح غير مجدية ، و التحسين الأكبر سيكون عندما نحتاج إلى مفتاح أطول من أجل زيادة السوية الأمنية المطلوبة لمواجهة الهجمات المتعلقة بالحوسبة الكمومية حيث يمكن للخوارزمية ECC العمل مع مفتاح بطول 512 بت، ولكن هذا المستوى الأمني يتطلب من الخوارزميات الأخرى مفتاحاً بطول 15360 بت وهذا غير ممكن عملياً في البيئات المحدودة الموارد كالأجهزة الصغيرة والبطاقات الذكية، ايضا الخوارزمية المعيارية AES هي الأفضل ولاسيما في مجال البيئات المحدودة الموارد، من هنا نؤكد على قدرة البروتوكول المقترح المكون من الخوارزميات السابقة على تحقيق الأمن اللازم في عصر الحواسيب الكمومية وإنترنت الأشياء بزمن أقل بقدار 27% مضافاً لها نسبة 77% في تشفير المفتاح، وموارد طاقة أقل ايضا بنسبة 175% مضافاً لها بالحد الأدنى نسبة 14%، إضافة إلى أن زمن توليد المفاتيح أقل بحد أدنى 91 ضعف، وطاقة توليد المفاتيح أقل بحد أدنى 72 ضعف.

يمكن العمل مستقبلا من أجل تخفيض تكاليف ضرب النقاط القياسي على المنحنيات الإهليلجية، والعمل على توطين استخدام التواقيع المختومة وتضمينها داخل الوثائق الرقمية.

المراجع

1. BEHROUZ.A.F, 2008, *Introduction to Cryptography and Network Security*, *McGrawHill, United States*,721.

2. NURZHAN, Z. A; DAVOR ,S.M, 2017, *Security and Privacy in Decentralized Energy Trading through Multi-Signatures, Blockchain and Anonymous Messaging Streams ,IEEE*, 15.

3. VAHDATI, Z; YASIN, S; GHASEMPOUR, A; SALEHI, M, 2019,*COMPARISON OF ECC AND RSA ALGORITHMS IN IOT DEVICES*. *Journal of Theoretical and Applied Information Technology*, Vol.97. No 16,17.

4. ALI, I, 2015,*COMPARISON AND EVALUATION OF DIGITAL SIGNATURE SCHEMES EMPLOYED IN NDN NETWORK*. *International Journal of Embedded systems and Applications(IJESA)*,Vol.5, No.2, 15.

5. BOS, J; Özen, O; , Stam, M, 2011,*Efficient Hashing Using the AES Instruction Set*, *International Association for Cryptologic Research*,, *University of Bristol* ,United Kingdom, 16.

6. Chang ,Z; Woźniak, M. 2020, *Encryption technology of voice transmission in mobile network based on 3DES-ECC algorithm , University of Posts &Telecommunications, China,* 11.

7. RIFA, H.; HERRERA. J. 2011,*Computational and Energy Costs of Cryptographic Algorithms on Handheld Devices*, *Universitat Oberta de Catalunya, Spain*, 18.

8. NISHAAL, J; VERMA,K, 2017, *A Comparative Evaluation of Algorithms in the Implementation of an Ultra-Secure Router-to-Router Key Exchange System*, *Hindawi Security and Communication Networks , University of Oklahoma, USA*, Article ID 1467614 ,8.

9. MAHTO, D; YADAV, D.K, 2018, *Performance Analysis of RSA and Elliptic Curve Cryptography, International Journal of Network Security*, Vol.20, No.4,11.

10. MAHTO, D; YADAV, D.K. 2017, *RSA and ECC: A Comparative Analysis*, *International Journal of Applied Engineering Research* , Vol 12, NO 19 ,9.

11. ALAA D. ; ASMAA F. ; SUZAN B. ; MASHAEL, S. A; KAR ,J ,2015,*Conventional and Improved Digital Signature Scheme: A Comparative Study, King Abdulaziz University, Jeddah, KSA,*10

12. برهوم، بسيم. 2020، أمن نظم المعلومات. جامعة تشرين، سوريا، 310.

13. دبش محمد; الخير عدنان; شعار محمود; برهوم بسيم، 2011، طر يقة جديدة في تقييم خوارزمية التشفير المتناظرة، مجلة جامعة حلب، العدد 75.

14. Said.M. A. 2019, *Efficient Elliptic Curve Cryptography Software Implementation on Embedded Platforms, University of Sheffield, United Kingdom ,140.*

15. AJAYKUMAR, N; SARVAGYA,M; PARANDKAR.P *,2020,A novel security algorithm ECC-L for wireless sensor network, , Reva University, Bangalore, India,6.*