مجلة جامعة دمشق للعلوم الهندسية
المجلد 40- العدد 1 (2024) 148-164

Damascus university Journal for engineering sciences
Vol 40 No.1 (2024): 148-164

دراسةٌ مُقارِنةٌ بين عائلات المُعمِّيات الكتليّة

جعفر سلطان[1]، د. محمد إياد الخياط[2]

[1]طالب دراسات عليا في قسم النُّظم والشبكات الحاسوبيّة – كليّة الهندسة المعلوماتيّة – جامعة دمشق.

[2]مُدرِّسٌ في قسم النُّظم والشبكات الحاسوبيّة – كليّة الهندسة المعلوماتيّة – جامعة دمشق.

المُلخَّص

تُعَدّ التعمية الآليّة الأمنيّة المُستخدَمة لتوفير خدمة سرّيّة البيانات، تُعَدّ المُعمِّيات التناظريّة النوع الأكثر استخداماً كونها أسهل في التنجيز وتستهلك مواردا أقلّ مُقارَنةً بالمُعمِّيات اللّاتناظريّة، وهي آمنةٌ طالما أنّ المفتاح يبقى سرّيّاً. تُعَدّ المُعمِّيات الكتليّة التكراريّة النوع الأكثر شيوعاً من المُعمِّيات التناظريّة بسبب فعاليتها في التنجيزات البرمجيّة والتنجيزات العتاديّة المُخصَّصة. تختلف المُعمِّيات الكتليّة التكراريّة من حيث هيكليّة الجولات التي تُجريها، وتُعَدّ هيكليّة الجولة مجالاً بحثيّاً مفتوحاً ومُتطوِّراً باستمرار. بناءً على هيكليّة الجولة؛ يُمكِن تمييز عائلاتٍ مختلفةٍ تُوفِّر ميزاتٍ فريدةً من حيث الأداء والأمان وسهولة التنجيز. تُظهِر الدراسة المرجعيّة أنّه تُوجَد عديدٌ من الأوراق البحثيّة التي تُناقِش أداء وأمان المُعمِّيات وتُجري المُقارَنة فيما بينها على منصّاتٍ مناسبةٍ لبيئاتٍ معيَّنة. على الرغم من ذلك؛ لا تُوجَد دراساتٌ سابقةٌ تُقارِن بين هيكليّات العائلات المختلفة وميزاتها الفريدة. تهدف هذه الورقة البحثيّة إلى شرح هيكليّات عائلات المُعمِّيات الكتليّة التكراريّة، والتعريف بميزاتها الفريدة، وإجراء المُقارَنة فيما بينها، وتقديم الاستنتاجات بناءً على تلك الميزات، وبالتالي فهي تُزوِّد القارئ بالقدرة على فهم الميزات المُشتَرَكة للمُعمِّيات الكتليّة التي تنتمي إلى نفس العائلة، وهي تُمثِّل نقطة انطلاقٍ جيّدةً تُسهِّل على الباحث في مجال تطوير خوارزميّات التشفير اختيار الهيكليّة المناسبة لبناء المُعمِّي الكتليّ المطلوب.

الكلمات المفتاحيّة: المُعمِّي الكتليّ، عائلة شبكة فيستيل (FN)، عائلة شبكة الاستبدال والتقليب (SPN)، عائلة مُخطَّط Lai–Massey (LM)، عائلة عمليّات الجمع/التدوير XOR/ (ARX)، المُعمِّيات الهجينة.

تاريخ الإيداع: 2022/6/29
تاريخ القبول: 2022/10/31

A Comparative Study between Block Cipher Families

Jafar Sultan[1], Dr. Mhd. Iyad Alkhayat[2]

[1]Master student in the Department of Computer Systems and Networking - Faculty of Information Technology Engineering - Damascus University.

[2]Lecturer in the Department of Computer Systems and Networking - Faculty of Information Technology Engineering - Damascus University.

Received: 29/6/2022
Accepted: 31/10/2022

Abstract

Cryptography is the security mechanism used to provide data confidentiality service. Symmetric ciphers are the most widely used type because they are easier to implement and consume less resources compared to asymmetric ciphers, and they are considered secure as long as the key is kept secret. Iterating block ciphers are the most popular type of symmetric ciphers because of their effectiveness in software implementations and dedicated hardware implementations. Iterating block ciphers differ in terms of the structure of the rounds they perform, and the round structure is an open and constantly evolving field of research. Depending on the structure of the round, different families can be distinguished that provide unique features in terms of performance, security and ease of implementation. The reference study shows that there are many research papers that discuss and compare the performance and security of ciphers on platforms that are suitable for particular environments. However, there are no previous studies that compare the structures of different families and their unique features. This paper aims to explain the structures of families of iterating block ciphers, identify their unique features, compare them, and provide conclusions based on those features. Thus, it provides the reader with the ability to understand the common features of the block ciphers belonging to the same family, and it is a good starting point that makes it easier for the researcher in the field of developing encryption algorithms to choose the appropriate structure to build the required block cipher.

Keywords: Block cipher, Feistel Network family (FN), Substitution-Permutation Network family (SPN), Lai–Massey scheme family (LM), Addition/Rotation/XOR family (ARX), Hybrid ciphers.

1- المُقدِّمة

يُعرَّف التشفير (encryption) بأنَّه تحويل البيانات إلى شكلٍ مُختلِفٍ يُخفي المعنى الأصليّ للبيانات ويمنع استخدام النموذج الأصليّ. التحويل العكسيّ المُقابِل هو فكّ التشفير (decryption) الذي يُعيد البيانات المُشفَّرة إلى شكلها الأصليّ (Shirey, 2007, 119).

يُعَدّ التشفير الآليّة الأمنيّة المُستخدَمة لتوفير خدمة سرّيّة البيانات (confidentiality) التي تنطوي على حماية البيانات من الهجمات السلبيّة[1] التي تشمل كشف محتوى الرسائل وتحليل حركة مرور البيانات (Stallings, 2017, 27).

يتكوَّن نظام التعمية (cryptosystem) من فضاء النصوص الصريحة (plaintexts)، فضاء النصوص المُشفَّرة (ciphertexts)، فضاء مفاتيح التشفير (keys)، خوارزميّة التشفير التي يُشار لها أيضاً بالمُشفِّر (encryptor)، خوارزميّة فكّ التشفير التي يُشار لها أيضاً بفاكّ التشفير (decryptor).

يُشار إلى الخوارزمية المُستخدَمة في التشفير وفكّ التشفير بالمُعمِّي (cipher) (Shirey, 2007, 61).

بناءً على عدد المفاتيح المُستخدَمة؛ يُمكِن تمييز نوعين من المُعمِّيات (Stallings, 2017, 89):

- المُعمِّيات التناظريّة (Symmetric ciphers): تقوم بتشفير البيانات وفكّ تشفيرها باستخدام نفس المفتاح ويُسمّى المفتاح السرّيّ (secret key) كونه يجب أن يبقى سرّيّاً بين المُرسِل والمُستقبِل.
- المُعمِّيات اللّاتناظريّة (Asymmetric ciphers): تستخدم زوجاً من المفاتيح، يُستخدَم أحدهما في عمليّة التشفير لدى المُرسِل ويُسمّى المفتاح العام (public key)؛ ويُستخدَم الآخر في عمليّة فكّ التشفير لدى المُستقبِل ويُسمّى المفتاح الخاصّ (private key).

يعتمد الأمان في أنظمة التعمية التناظريّة بشكلٍ جوهريٍّ على توزيع المفتاح بشكلٍ آمنٍ على كلٍّ من المُرسِل والمُستقبِل، تحلّ أنظمة التعمية اللّاتناظريّة مشكلة توزيع المفاتيح (Schneier, 1996, 59)، حيث يبقى المفتاح الخاصّ سرّيّاً لدى مالكه ويقوم بمُشارَكة المفتاح العامّ مع العموم ضمن شهادةٍ رقميّةٍ تحمل هويّته ليُمكِّن جميع الأطراف التي ترغب بإرسال البيانات المُشفَّرة له من استخدامه.

يتطلَّب استخدام أنظمة التعمية اللّاتناظريّة تحقُّقاً صارِماً من الشهادات الرقميّة لدعم الثقة بين الطرفين، كما أنّ المُعمِّيات اللّاتناظريّة تُعَدّ أكثر تعقيداً في التنجيز وتستهلك مواردا أكبر مُقارَنةً بالمُعمِّيات التناظريّة. تُعَدّ المُعمِّيات التناظريّة النوع الأكثر استخداماً كونها أسهل في التنجيز وتستهلك مواردا أقلّ وهي آمنةٌ طالما أنّ المفتاح يبقى سرّيّاً.

في التطبيقات العمليّة؛ يُستخدَم التشفير اللّاتناظريّ لتأمين وتوزيع مفاتيح الجلسة (session keys) التي تُستَخدَم من قبل المُعمِّيات التناظريّة لتأمين حركة الرسائل، وهذا ما يُسمّى أحياناً نظام التعمية الهجين (hybrid cryptosystem) (Schneier, 1996, 61).

بناءً على طريقة مُعالَجة النصّ الصريح؛ تُصنَّف المُعمِّيات التناظريّة إلى نوعين أساسيَّين (Stallings, 2017, 120):

- المُعمِّيات الدفقيّة (Stream ciphers): تقوم بتحويل دفقٍ من النصّ الصريح إلى دفقٍ من النصّ المُشفَّر باستخدام دفقٍ موازٍ من مفتاح التشفير، يجب

[1] تُصنَّف الهجمات من حيث القصد إلى صنفين، الصنف الأوّل هو الهجمات السلبيّة (passive attacks) التي تُحاول التعرُّف على المعلومات أو الاستفادة منها من أحد الأنظمة ولكنّها لا تُؤثِّر على الموارد الخاصة بهذا النظام، الصنف الثاني هو الهجمات النشطة (active attacks) التي تُحاوِل تغيير موارد النظام أو التأثير على عمليّاته (Shirey, 2007, 23).

توفير دفق المفتاح لكلا المُستخدِمَين مُسبَقاً عبر قناةٍ مُستقِلّةٍ وآمنة، من الناحية العمليّة؛ يتمّ تنجيز مُولِّد تدفُّق بتات المفتاح كإجراءٍ حسابيٍّ؛ بحيث يحتاج المُرسِل والمُستقبِل فقط إلى مُشارَكة مفتاح التوليد.

- المُعمِّيات الكتليّة (Block ciphers): تقوم بمُعالَجة كتلةٍ من النصّ الصريح ككلّ وليس كعناصر مُفرَدةٍ واستخدامها لإنتاج كتلةٍ من النصّ المُشفَّر مُساويةٍ لها بالطول، حيث يتمّ تقسيم النصّ الصريح إلى كتلٍ متساوية الطول، ويقوم المُعمِّي بمُعالَجة هذه الكتل باستخدام نفس المفتاح للحصول على الكتل المُقابِلة من النصّ المُشفَّر.

تتميَّز المُعمِّيات الكتليّة بقابليّة التطبيق على نطاقٍ أوسع من التطبيقات قياساً بالمُعمِّيات الدفقيّة (Stallings, 2017, 121)، وتُعَدّ مُفضَّلةً بسبب إمكانيّة استخدامها لتحقيق نفس وظيفة المُعمِّيات الدفقيّة في بعض أنماط عملها.

يُمكِن تصميم المُعمِّيات الكتليّة الحديثة بطرقٍ مختلفة، يُعَدّ المُعمِّي الكتليّ التكراريّ (iterative block cipher) التصميم الأكثر استخداماً.

يُعرَّف المُعمِّي الكتليّ التكراريّ بأنّه تطبيق عددٍ من التقليبات المنطقيّة المُعتمِدة على المفتاح بشكلٍ مُتكرِّر عبر عدّة جولات (rounds)؛ يُشار إلى تلك التقليبات المنطقيّة بتحويل الجولة (round transformation)، يُعبَّر عن المُعمِّي الكتليّ التكراريّ $\beta(k)$ كما يلي (Daemen *et al.*, 2002, 24):

$$\beta(k) = \rho^r(k^r)\, \rho^{r-1}\left(k^{r-1}\right) \cdots \rho^1\left(k^1\right)$$

يُعبِّر المُتحوِّل r عن عدد الجولات التي يُجريها المُعمِّي الكتليّ التكراريّ، ويُعبِّر المُتحوِّل ρ^i عن تحويل الجولة رقم i، ويُعبِّر المُتحوِّل k^i عن مفتاح الجولة رقم i. يتمّ اشتقاق مفاتيح الجولات من مفتاح التشفير عن طريق خوارزميّة اشتقاق المفاتيح التي تُولِّد جدول المفاتيح (key schedule) الذي يُشار له أيضاً بالمفتاح المُوسَّع K (expanded key) حيث أنّ:

$$K = k^1 \mid k^2 \mid \cdots \; k^r$$

يُعَدّ المُعمِّي الكتليّ التكراريّ ذو المفتاح (key-iterating block cipher) التصميم الأكثر شيوعاً، ويُعرَّف بأنّه التطبيق المُتتالي لتحويل الجولة المُستقِلّ عن المفتاح متبوعاً بإضافة مفتاح الجولة بواسطة عمليّة XOR بسيطة، حيث أنّه يُجري نفس تحويل الجولة في جميع الجولات عدا الجولة الابتدائيّة والأخيرة، ويُعبَّر عنه كما يلي (Daemen *et al.*, 2002, 26):

$$\beta(k) = \sigma(k^r)\, \rho\, \sigma\left(k^{r-1}\right) \rho \cdots \sigma\left(k^1\right) \rho\, \sigma\left(k^0\right)$$

يُعبِّر المُتحوِّل r عن عدد الجولات التي يُجريها المُعمِّي الكتليّ التكراريّ، ويُعبِّر المُتحوِّل ρ عن تحويل الجولة، ويُعبِّر التعبير $\sigma(k^i)$ عن تحويل إضافة مفتاح الجولة رقم i، يتمّ اشتقاق مفاتيح الجولات من مفتاح التشفير عن طريق خوارزميّة اشتقاق المفاتيح التي تُولِّد جدول المفاتيح K حيث أنّ: $K = k^0 \mid k^1 \mid k^2 \mid \cdots \; k^r$

تجري إضافة مفتاحٍ أوليّة قبل الجولة الأولى، الدافع وراء ذلك هو أنّه في سياق هجمات النصّ الصريح المعروف (known-plaintext attacks) يُمكِن ببساطةٍ إزالة أيّة عمليّاتٍ تسبق إضافة المفتاح الأوليّة أو تلي إضافة المفتاح النهائيّة دون معرفة المفتاح، وبالتالي فهي لا تُساهِم في أمان التشفير، ولذا فإنّ عديداً من التصميمات تقوم بتطبيق إضافة المفتاح الأوليّة أو النهائيّة (Daemen *et al.*, 1999, 8).

تُعَدّ المُعمِّيات الكتليّة التكراريّة ذات المفتاح النوع الأكثر استخداماً من المُعمِّيات الكتليّة وهي تصلح للتنجيزات الفعّالة.

ففي التنجيزات العتاديّة المُخصَّصة؛ يُمكِن تنجيز تحويل الجولة وإضافة المفتاح باستخدام التوصيلات

الفيزيائيّة، وبالتالي يُمكِن إجراء تشفير الكتلة ببساطةٍ عن طريق تكرار تحويل الجولة بالتناوب مع مفاتيح الجولة المناسبة، وفي التنجيزات البرمجيّة؛ هناك حاجةٌ إلى كتابة كود تحويل جولةٍ واحدٍ واستخدامه ضمن حلقة، وبالتالي يُمكِن إجراء تشفير الكتلة ببساطةٍ عن طريق تنفيذ هذه الحلقة بالعدد المطلوب من التكرارات (Daemen *et al.*, 2002, 26).

بهدف إحباط تحليل الشيفرة (cryptanalysis) المبنيّ على التحليل الإحصائيّ؛ اقترح كلود شانون Claude Shannon وظيفتَين أصبحتا حجر الزاوية في تصميم المُعمِّيات الكتليّة؛ هما وظيفة الخلط (confusion) ووظيفة النشر (diffusion) (Stallings, 2017, 124).

تسعى وظيفة الخلط لجعل العلاقة بين إحصائيّات النصّ المُشفَّر وقيمة مفتاح التشفير مُعقَّدةً قدر الإمكان بهدف إحباط مُحاوَلات استنتاج المفتاح (Stallings, 2017, 125). يتحقّق ذلك بجعل رموز النصّ الصريح تُؤثِّر على رموز النصّ المُشفَّر بطريقةٍ مُعقَّدةٍ.

تسعى وظيفة النشر لجعل العلاقة الإحصائيّة بين النصّ الصريح والنصّ المُشفَّر مُعقَّدةً قدر الإمكان بهدف إحباط مُحاوَلات استنتاج المفتاح، وهذا يتطلَّب تبديد البنية الإحصائيّة للنصّ الصريح إلى إحصاءاتٍ منتشرةٍ على كامل مدى النصّ المُشفَّر. يتحقّق ذلك بجعل كلّ رمزٍ من النصّ الصريح يُؤثِّر على العديد من رموز النصّ المُشفَّر؛ وهذا يُكافِئ أن يتأثَّر كلّ رمزٍ من النصّ المُشفَّر بالعديد من رموز النصّ الصريح. (Stallings, 2017, 124).

عادةً ما تُوفِّر الجولات في المُعمِّيات الكتليّة التكراريّة وظيفة الخلط باستخدام عمليّة الاستبدال (substitution) اللّاخطّيّة، وتُوفِّر وظيفة النشر باستخدام عمليّة التقليب (permutation) الخطّيّة (Avanzi, 2016, 25).

تختلف المُعمِّيات الكتليّة التكراريّة من حيث هيكليّة الجولات التي تُجريها على كتلة البيانات التي تُسمّى الحالة (state)، بناءً على هيكليّة الجولة يُمكِن تمييز العائلات التالية:

- عائلة شبكة فيستيل Feistel Network (FN).
- عائلة شبكة الاستبدال والتقليب Substitution-Permutation Network (SPN).
- عائلة مُخطَّط Lai–Massey Lai–Massey scheme (LM).
- عائلة عمليّات الجمع/ التدوير/ XOR Addition/ Rotation/ XOR (ARX).

تُعَدّ هيكليّة الجولة مجالاً بحثيّاً مفتوحاً ومُتطوِّراً باستمرار، تُظهِر الدراسة المرجعيّة أنّه تُوجَد عديدٌ من الأوراق البحثيّة التي تُناقِش أداء وأمان المُعمِّيات وتُجري المُقارَنة فيما بينها على منصّاتٍ مناسبةٍ لبيئاتٍ معيَّنة. على الرغم من ذلك؛ لا تُوجَد دراساتٌ سابقةٌ تُقارِن بين هيكليّات العائلات المختلفة وميزاتها الفريدة.

تهدف الورقة البحثيّة إلى شرح هيكليّات عائلات المُعمِّيات الكتليّة، والتعريف بميزاتها الفريدة، وإجراء المُقارَنة فيما بينها، وتقديم الاستنتاجات بناءً على تلك الميزات، وبالتالي فهي تُزوِّد القارئ بالقدرة على فهم الميزات المُشتَرَكة للمُعمِّيات الكتليّة التي تنتمي إلى نفس العائلة، وهي تُمثِّل نقطة انطلاقٍ جيّدةً تُسهِّل على الباحث في مجال تطوير خوارزميّات التشفير اختيار الهيكليّة المناسبة لبناء المُعمِّي الكتليّ المطلوب.

تمّ تنظيم باقي هذه الورقة على النحو التالي: يستعرض القسم الثاني أبرز الدراسات السابقة التي تُناقِش أداء وأمان المُعمِّيات وتُجري المُقارَنة فيما بينها، يُعرِّف القسم الثالث بعائلة شبكة فيستيل وأصنافها الأساسيّة، يُعرِّف القسم الرابع بعائلة شبكة الاستبدال والتقليب، يُعرِّف القسم الخامس بعائلة مُخطَّط Lai-Massey، يُعرِّف القسم السادس بعائلة ARX، يُلقي القسم السابع الضوء على

المُعمِّيات الهجينة، وأخيراً يُجري القسم الثامن مُقارَنةً بين عائلات المُعمِّيات الكتليّة ويُقدِّم الاستنتاجات بما يتعلّق بميزاتها الفريدة.

2- الدراسات السابقة

تُوجَد عديدٌ من الأوراق البحثيّة التي تُناقِش أداء وأمان المُعمِّيات التي تنتمي إلى عائلاتٍ مختلفةٍ وتُجري المُقارَنة فيما بينها على منصّاتٍ مناسبةٍ لبيئاتٍ معيَّنة.

قام Eisenbarth *et al.* (2012) بتقييم أداء مجموعةٍ متنوّعةٍ من المُعمِّيات الكتليّة التقليديّة وخفيفة الوزن التي يُمكِن استخدامها في الأجهزة محدودة الموارد، تمّ إجراء المُقارَنة فيما بينها من حيث الطاقة (energy) واستهلاك الذاكرة وعدد دورات المُعالَجة، ولم تشمل المُقارَنة مُعمِّياتٍ من عائلة LM.

قام Batina *et al.* (2013) بتقييم مجموعةٍ من المُعمِّيات الكتليّة خفيفة الوزن ومُقارَنتها بالمُعمِّي AES من حيث منطقة الرقاقة (chip area) مُعبَّراً عنها بعدد البوّابات المُكافِئة (Gate Equivalent) واستهلاك الطاقة والقدرة (power) وعدد دورات المُعالَجة دون أن تشمل المُقارَنة مُعمِّياتٍ من عائلة ARX و LM.

قام Cazorla *et al.* (2013) بتقييم أداء التنجيزات البرمجيّة لمجموعةٍ من المُعمِّيات الكتليّة التقليديّة وخفيفة الوزن التي يُمكِن استخدامها في بيئة شبكات الحسّاسات اللّاسلكيّة وتنتمي إلى عائلات FN و SPN و LM، وجرت مُقارَنتها من حيث استهلاك الذاكرة وعدد دورات المُعالَجة.

أجرى Kushwaha *et al.* (2014) مُقارَنةً بين عددٍ من المُعمِّيات التي يُمكِن استخدامها في العقد مُقيَّدة الموارد وتنتمي إلى عائلتَي FN و SPN، وجرت مُقارَنتها من حيث منطقة الرقاقة (GEs) وعدد دورات المُعالَجة.

قدَّم Beaulieu *et al.* (2015) المُعمِّي SIMON ذا الهيكليّة FN المُناسِب للتنجيزات العتاديّة والمُعمِّي SPECK ذا الهيكليّة ARX المُناسِب للتنجيزات البرمجيّة، وأجرى الباحثون المُقارَنة بين عدّة مُعمِّياتٍ خفيفة الوزن إضافةً إلى المُعمِّي AES. أُجريَت مُقارَنة التنجيزات البرمجيّة من حيث المَردود (throughput) وحجم الذاكرة، وأُجريَت مُقارَنة التنجيزات العتاديّة من حيث المَردود ومنطقة الرقاقة (GEs).

قدَّم Albermany *et al.* (2016) نظرةً عامّةً مُوجَزةً حول مجموعةٍ من المُعمِّيات الكتليّة مع تصنيفها ضمن ثلاث فئاتٍ أساسيّةٍ هي FN و SPN و UFN دون إجراء مُقارَنةٍ بين ميزات الهيكليّات.

أجرى Appel *et al.* (2016) دراسةً استقصائيّةً للمُعمِّيات الكتليّة خفيفة الوزن المُناسِبة للاستخدام في عقد إنترنت الأشياء (Internet of Things). استفاد الباحثون من المُقارَنة التي أجراها Beaulieu *et al.* (2015) وأضافوا معيار الكفاءة لتقييم أداء التنجيزات العتاديّة الذي يُراعي المَردود ومنطقة الرقاقة (GEs). كما استفاد الباحثون من المُقارَنة التي أجراها Eisenbarth *et al.* (2012) وبعض الدراسات الأخرى لتقييم أداء التنجيزات البرمجيّة بما يشمل قياسات عدد دورات المُعالَجة وحجم الذاكرة المطلوبة.

قدَّم Avanzi (2016) لمحةً شاملةً عن أكثر من ثمانين مُعمِّياً كتليّاً معروفاً مع مُقارَنة أداء التنجيزات البرمجيّة والعتاديّة وشرح ثلاث عائلاتٍ أساسيّةٍ من المُعمِّيات وهي SPN و FN و LM.

قيّم Hosseinzadeh *et al.* (2017) أداء التنجيزات البرمجيّة في بيئة شبكة الحساسات اللّاسلكيّة لخمسةٍ من المُعمِّيات الكتليّة خفيفة الوزن التي تنتمي إلى عائلات ARX و SPN و FN من حيث استهلاك الذاكرة واستهلاك الطاقة.

ركّز الباحثون في Acosta *et al.* (2017) اهتمامهم على التشفير خفيف الوزن الذي يجمع بين الأمان واستهلاك القدرة بشكلٍ أساسيّ، واستنتج الباحثون أنّ مُعمِّيات العائلة SPN لديها أفضل أداءٍ من حيث استهلاك الطاقة مُقابِل البت، كما ناقش الباحثون المعماريّات المختلفة لتنجيز صناديق الاستبدال في العائلة SPN.

قام Rana *et al.* (2019) بإجراء دراسةٍ استقصائيّةٍ لمجموعة متنوّعةٍ من المُعمِّيات الكتليّة التقليديّة وخفيفة الوزن، وتقييمها من حيث استهلاك الذاكرة وعدد دورات المُعالَجة.

بناءً على الدراسة المرجعيّة السابقة يتَّضِح أنّه لا تُوجَد دراساتٌ سابقةٌ تُقارِن بين هيكليّات عائلات المُعمِّيات الكتليّة التكراريّة المختلفة وميزاتها الفريدة، ومن هذه النقطة تسعى هذه الورقة البحثيّة إلى شرح هيكليّات عائلات المُعمِّيات الكتليّة التكراريّة، والتعريف بميزاتها الفريدة، وإجراء المُقارَنة فيما بينها، وتقديم الاستنتاجات بناءً على تلك الميزات.

3- عائلة شبكة فيستيل

سُمّيَت عائلة شبكة فيستيل تيمُّناً بعالم التشفير هورست فيستيل Horst Feistel، يأخذ مُشفِّر شبكة فيستيل كتلة نصٍّ صريحٍ بطول $2w$ بت ومفتاح التشفير k كمُدخَلات.

يقوم المُشفِّر بتقسيم كتلة النصّ الصريح إلى نصفين كلّ منهما بطول w بت: نصفٌ أيسر يُشار له بالرمز $_0$ ونصفٌ أيمن يُشار له بالرمز RE_0، وتجري مُعالَجة نصفَي الكتلة عبر n جولةً لها نفس البنية، ثم يَتَّحِد نصفا الكتلة لإنتاج كتلة النصّ المُشفَّر (Stallings, 2017, 125).

يكون دخل الجولة رقم i هو نصفا الكتلة $_{i-1}, RE_{i-1}$ الناتجان عن الجولة السابقة بالإضافة إلى مفتاح الجولة k_i المُشتَقّ من المفتاح K باستخدام خوارزميّة جدولة المفاتيح، وينتج عن الجولة نصفا الكتلة LE_i, RE_i، وتُجري الجولة عمليّتين أساسيّتين هما (Stallings, 2017, 125):

- عمليّة الاستبدال: تجري على النصف الأيسر من الكتلة $_{i-1}$ ، يتمّ ذلك عن طريق تطبيق تابع الجولة F على النصف الأيمن من الكتلة RE_{i-1} ثمّ إجراء عمليّة XOR ما بين خرج تابع الجولة والنصف الأيسر من الكتلة.
- عمليّة التقليب: بعد عمليّة الاستبدال السابقة تتمّ المُبادَلة (swap) بين نصفَي الكتلة.

كما هو مُوضَّحٌ في الشكل 1 (أ)؛ يُمكِن التعبير عن ناتج الجولة رقم i في المُشفِّر كما يلي (Stallings, 2017, 128):

$$(LE_i, RE_i) = \left(RE_{i-1}\,, LE_{i-1} \oplus F(RE_{i-1}, K_i)\right)$$

كما هو مُوضَّحٌ في الشكل 1 (ب)؛ يُمكِن التعبير عن ناتج الجولة رقم i في فاكّ التشفير كما يلي (Stallings, 2017, 128):

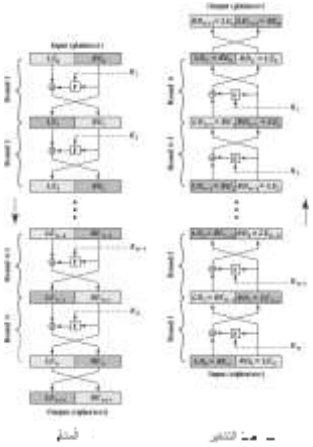
$$(LD_i, RD_i) = \left(RD_{i-1}, LD_{i-1} \oplus F(RD_{i-1}, K_i)\right)$$

يُوجَد تماثلٌ بين هيكليّة فاكّ التشفير والمُشفِّر وكلاهما يستخدمان نفس تابع الجولة. تجري مُعالَجة النصّ المُشفَّر بنفس الطريقة المُتَّبعة في المُشفِّر لكن مع استخدم المفاتيح الفرعيّة بترتيبٍ عكسيٍّ، تُغني هذه الميزة عن الحاجة إلى خوارزميّتين مختلفتين للتشفير وفكّ التشفير (Stallings, 2017, 127) مما يُخفِّض الكلفة الإجماليّة لتنجيز المُعمِّي برمجيّاً وعتاديّاً.

بما أنّه لا حاجة لأن يكون تابع الجولة قابلاً للانعكاس؛ يُمكِن تصميم تابع الجولة بشكلٍ مُستقلٍّ ليكون مُعقَّداً بأيّ شكلٍ كان (Avanzi, 2016, 28).

يُطلَق على هيكليّة شبكة فيستيل أيضاً التسمية LR-Construction تيمّناً بالعالِمين Michael Luby و Charles Rackoff اللّذَين قاما بتحليل بنية مُعمِّيات شبكة فيستيل، وأثبتا أنّه إذا كان تابع الجولة وظيفةً

تشفيريّةً شبه عشوائيّةٍ (pseudo-random) آمنةً؛ فإنّه – مع استخدام مفاتيح فرعيّةٍ كبذرة– تكون ثلاث جولاتٍ كافيةً لجعل المُعمِّي الكتليّ تقليباً شبه عشوائيّ، في حين تكفي أربع جولاتٍ لجعله تقليباً شبه عشوائيٍّ قويّاً (Avanzi, 2016, 30).



الشكل (1) هيكليّة المُشفِّر وفاكّ التشفير في عائلة FN

تُوجَد أنواعٌ مختلفةٌ من التصاميم المُمكِنة لمُعمِّيات شبكة فيستيل، ويُمكِن تصنيفها بشكلٍ عامٍّ ضمن ثلاثة أصناف أساسيّة (Schneier *et al.*, 1996, 1):

- شبكة فيستيل المُتوازِنة (Balanced FN).
- شبكة فيستيل غير المُتوازِنة (Unbalanced FN).
- شبكة فيستيل غير المُتوازِنة المُعمَّمة (Generalized Unbalanced FN).

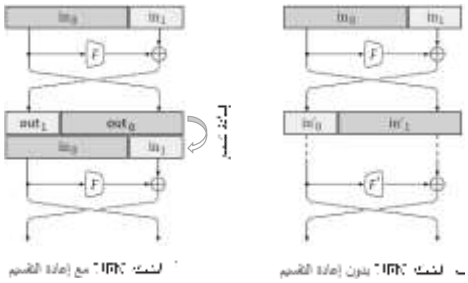
3–1– شبكة فيستيل المُتوازِنة

تُعَدّ شبكة فيستيل المُتوازِنة (BFN) الهيكليّة الأساسيّة لشبكات فيستيل، تقوم بتقسيم الحالة إلى جزئين (فرعين) لهما نفس الطول وتُعالِجهما وفق ما تمّ شرحه سابقاً. من أمثلة المُعمِّيات التي لها هيكليّة BFN: المُعمِّي DES، والمُعمِّي FEAL، والمُعمِّيَين Khufu و Khafre، والمُعمِّي Blowfish، والمُعمِّي ICE، والمُعمِّي CAST-128، والمُعمِّي LOKI97، والمُعمِّي Twofish، والمُعمِّي E2، والمُعمِّي DFC، والمُعمِّي Camellia، والمُعمِّي SEED، والمُعمِّي GOST، والمُعمِّي SIMON، وشبكة المستوى الأوّل في كلٍّ من المُعمِّيَين MISTY-1 و DEAL.

3–2– شبكة فيستيل غير المُتوازِنة

تقوم شبكة فيستيل غير المُتوازِنة (UFN) بتقسيم الحالة إلى فرعين مُختلِفي الطول، يجب إعادة تقسيم (repartition) الحالة الناتجة في خرج الجولة بالشكل المُناسِب قبل تمريرها إلى الجولة التالية بحال كانت الشبكة تطبِّق تابع جولةٍ ثابتاً خلال جميع جولات المعالجة (Avanzi, 2016, 32)؛ كما هو مُوضَّحٌ في الشكل 2 (أ).

تُوجَد أشكالٌ من شبكة UFN لا تتطلَّب عمليّة إعادة تقسيم الحالة، بدلاً من ذلك فهي تستخدم تابِعَي جولةٍ مختلفَين يُستخدَمان بالتناوب عبر الجولات، يُطبَّق أحدهما على الفرع الأصغر لتوسيعه ويُطبَّق الآخر على الفرع الأكبر لضغطه (Avanzi, 2016, 32)؛ كما هو مُوضَّحٌ في الشكل 2 (ب). مثالها شبكة المُعمِّيات LION, BEAR, LIONESS وشبكة المستوى الثالث في كلٍّ من المُعمِّيَين MISTY-1, MISTY-2، وشبكة المستوى الثالث من المُعمِّي KASUMI.



الشكل (2) شبكات فيستيل غير المُتوازِنة (UFN)

3–3– شبكة فيستيل غير المُتوازِنة المُعمَّمة

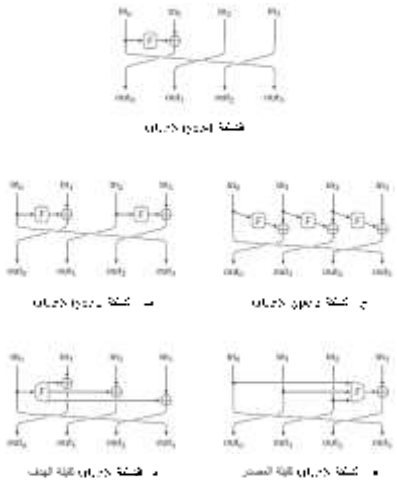
تقوم شبكة فيستيل غير المُتوازِنة المُعمَّمة (GUFN) بتقسيم الحالة إلى عدّة فروعٍ متساوية الطول؛ يُستخدَم منها فرعٌ مصدرٌ (source) (أو أكثر) لتعديل فروعٍ أخرى هدف (target).

تُوجَد عدّة أنواع من شبكات GUFN أبرزها (Avanzi, 2016, 32):

- الشبكة type-1 التي تُنتِج في نهاية الجولة فرعاً واحداً مُعدَّلاً؛ الشكل 3 (أ)، مثالها شبكة المُعمِّي CAST-256.
- الشبكة type-2 التي تُنتِج في نهاية الجولة فرعَين مُعدَّلَين؛ الشكل 3 (ب)، مثالها شبكة المُعمِّي RC6 وشبكة المُعمِّي CLEFIA.
- الشبكة type-3 التي تُنتِج في نهاية الجولة ثلاثة فروعٍ مُعدَّلة؛ الشكل 3 (ج).

تُوجَد حالتان فريدتان من شبكات GUFN هما الشبكة ثقيلة الهدف (target-heavy) والشبكة ثقيلة المصدر (source-heavy) (Avanzi, 2016, 33).

تستخدِم الشبكة ثقيلة الهدف فرعاً واحداً لتعديل الفروع الأخرى؛ الشكل 3 (د)، مثالها شبكة جولات الطور الثاني في المُعمِّي MARS التي تُعَدّ ثقيلة الهدف من النوع type-3. على العكس؛ تقوم الشبكة ثقيلة المصدر بتعديل فرعٍ واحدٍ باستخدام الفروع الأخرى؛ الشكل 3 (هـ)، مثالها شبكة المُعمِّي التجريبيّ MacGuffin.



الشكل (3) شبكات فيستيل غير المُتوازِنة المُعمَّمة (GUFN)

تُعَدّ شبكة فيستيل مُتجانِسةً (homogeneous) إذا كان تابع الجولة مُتطابِقاً في جميع الجولات، وهي الحالة الأكثر استخداماً في المُعمِّيات الكتليّة التكراريّة، وبخلاف ذلك تُعَدّ الشبكة غير مُتجانِسةٍ (heterogeneous) (Schneier *et al.*, 1996, 3).

تُعَدّ الشبكات غير المُتجانِسة أكثر تعقيداً من حيث التحليل والتنجيز، مثالها شبكة المُعمِّي Khufu التي يتغيّر فيها صندوق الاستبدال المُستخدَم في تابع الجولة كلّ ثماني جولات. في حين أنّ الشبكات المُتجانِسة تتميَّز بتنجيزٍ عتاديٍّ أقلّ كلفةً وتنجيزٍ برمجيٍّ أصغر حجماً وأسهل (Schneier *et al.*, 1996, 3).

تُعدّ شبكة UFN مُتّسِقةً (consistent) إذا بقيت أحجام الفروع المصدر والهدف ثابتةً خلال جميع الجولات، وبخلاف ذلك تُعَدّ الشبكة غير مُتّسِقةٍ (inconsistent)؛ مثالها شبكة المُعمِّيات LION, BEAR, LIONESS. تجدر المُلاحَظة بأنّ شبكة UFN غير المُتّسِقة هي حتماً غير مُتجانِسة، في حين أنّه من المُمكِن أن تكون الشبكة غير مُتجانِسةٍ ولكنّها مُتّسِقة (Schneier *et al.*, 1996, 4).

من المُمكِن أن يتغيّر تصميم شبكة UFN من جولةٍ إلى أخرى أو خلال مجموعةٍ من الجولات؛ كما في المُعمِّي MARS المُكوَّن من ثلاثة أطوار تختلف جولاتها من حيث الهيكليّة وتابع الجولة، والمُعمِّي Skipjack المُكوَّن من أربع دفعاتٍ من الجولات يتناوب فيها استخدام نوعين من هيكليّة الجولة.

تُعَدّ شبكة UFN كاملةً (complete) إذا كانت جميع بتّات الحالة تُستخدَم خلال الجولة الواحدة (أي أنّها إمّا جزءٌ من فروعٍ مصدر أو جزءٌ من فروعٍ هدف)، وبخلاف ذلك تُعَدّ الشبكة غير كاملةٍ (incomplete) (Schneier *et al.*, 1996, 4)؛ مثالها شبكة المُعمِّيَين Khufu, Khafre التي تستخدِم ناتج

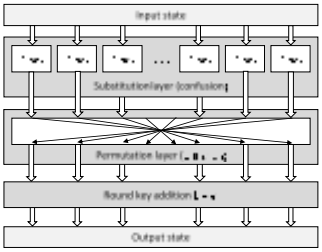
استبدال جزءٍ فقط من نصف الحالة في تعديل نصف الحالة الآخَر.

تسمح تصميمات شبكة فيستيل غير المُتوازِنة ببناء مُعمِّياتٍ كتليّةٍ ذات ميزاتٍ جيّدة، إذ تتمتَّع تصميمات UFN بمُقاوَمتها الموروثة لتحليل الشيفرة الفرقيّ، وإنّ تزويدها بتابع جولةٍ يتمتَّع بمُقاوَمة تحليل الشيفرة الخطّيّ يُنتِج مُعمِّياً مُقاوِماً لكلٍّ من تحليل الشيفرة الخطّيّ والفرقيّ[2] (Schneier *et al.*, 1996, 17).

تتمتَّع شبكة UFN ثقيلة الهدف بمُقاوَمة تحليل الشيفرة الخطّيّ، كذلك تتمتَّع شبكة UFN ثقيلة المصدر بمُقاوَمة تحليل الشيفرة الفرقيّ. وبالتالي فإنّ بناء مُعمٍّ كتليٍّ يُجري عدّة جولات UFN ثقيلة الهدف متبوعةً بعدّة جولات UFN ثقيلة المصدر يُوفِّر ميزاتٍ مُقاوِمةً لتحليل الشيفرة الخطّيّ والفرقيّ على حدٍّ سواء (Schneier *et al.*, 1996, 17).

4- عائلة شبكة الاستبدال والتقليب

يُجري تحويل الجولة في مُعمِّيات عائلة شبكة الاستبدال والتقليب (SPN) ثلاثة تحويلاتٍ مختلفةٍ قابلةٍ للعكس على كامل الحالة يُشار لها بالطبقات (layers). وهي طبقة الاستبدال وطبقة التقليب وطبقة إضافة مفتاح الجولة التي عادةً ما تستخدم عمليّة XOR بسيطة؛ كما هو مُوضَّحٌ في الشكل 4.



الشكل (4) هيكليّة الجولة في مُعمّي عائلة SPN

عادةً ما يتمّ تنجيز طبقة الاستبدال كعمليّاتٍ تجري على جدول بحثٍ (lookup table) يُعيّن قِيَم أجزاء من الحالة مع قِيَم أخرى مُقابِلة (mapping)، كما يُمكِن أن يكون التنجيز الفعليّ بطرق أخرى مثل استخدام دارةٍ بسيطةٍ أو برنامجٍ قصيرٍ يُجري عمليّاتٍ لاخطّيّة (Avanzi, 2016, 25).

يُشار إلى الوظيفة البسيطة اللّاخطّيّة التي تُعدِّل جزءاً فقط من الحالة بصندوق الاستبدال S-box، وتتكوَّن طبقة الاستبدال الكاملة من تطبيقاتٍ مُتعدِّدةٍ ومتوازيةٍ لصناديق الاستبدال (التي قد لا تختلف بالضرورة عن بعضها البعض) على أجزاء من الحالة (Avanzi, 2016, 26).

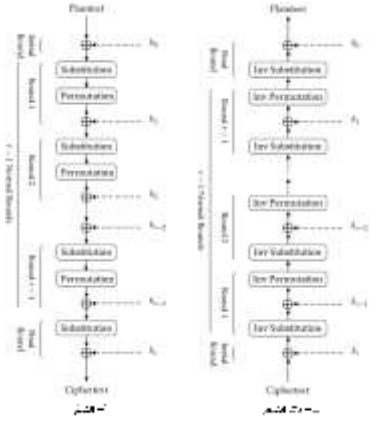
يُمكِن تمثيل طبقة التقليب كتبديل مَواقع ثابتٍ وبسيطٍ لبتّات الحالة أو كتحويلٍ خطّيٍّ عكوسٍ يُعالِج الحالة بالكامل، يُشار إلى وظيفة التقليب بصندوق التقليب P-box، من حيث المبدأ؛ لا يُوجَد ما يمنع أن يكون تحويل النشر لاخطّيّاً (Avanzi, 2016, 26).

يسمح الفصل بين وظيفتَي الخلط والنشر ببناء جولاتٍ مُكوَّنةٍ من عمليّاتٍ أبسط يُمكِن تحليلها رياضيّاً بشكلٍ مُستقِلّ، كما أنّه يُؤدّي إلى بساطةٍ في نهج التصميم وتحليله ممّا يُفيد الأمان بشكلٍ مؤكَّد، حيث أنّ معظم التصميمات التي صمدت بشكلٍ أفضل أمام تحليل الشيفرة هي تصميماتٌ منتظمةٌ جدّاً ومبنيّةٌ باستخدام كتلٍ بسيطة، في حين أنّ عديداً من التصميمات التي كانت تبدو ذكيّةً والتي مزجت أنواعاً مختلفةً من العمليّات المُعقَّدة والغامِضة

[2] يُعَدُّ كلٌّ من تحليل الشيفرة الفرقيّ (differential cryptanalysis) وتحليل الشيفرة الخطّيّ (linear cryptanalysis) هجوم نصٌّ صريحٍ مُختار (chosen-plaintext attack)؛ يستخدم عدداً كبيراً من أزواج النصّ الصريح والنصّ المُشفَّر لتحديد بتّات المفتاح (Daemen *et al.*, 2002, 83, 85). يسعى تحليل الشيفرة الفرقيّ لإيجاد احتمال انتشار فرقٍ في بتّات الدخل يُحدِّده نمط فرق (difference pattern) الدخل إلى فرقٍ في بتّات الخرج يُحدِّده نمط فرق الخرج. يهتمّ تحليل الشيفرة الخطّيّ بالعلاقة الخطّيّة (علاقة التكافؤ parity) التي تربط مجموعةً من البتّات يتمّ اختيارها باستخدام نمط اختيار (selection pattern)، ويسعى لإيجاد سعة الارتباط الخطّيّ (linear correlation amplitude) بين تكافؤ الدخل وتكافؤ الخرج.

تـــمّ كســـرها بشـــكلٍ مفـــاجئٍ خـــلال وقـــتٍ قصـــير (Avanzi, 2016, 25).

لجعل المُشفِّر وفاكّ التشفير أكثر تشابهاً من حيث الهيكليّة؛ لا تحتوي الجولة الأخيرة على طبقة التقليب كما في سائر الجولات الأخرى؛ كما هو موضَّحٌ في الشكل 5، لا يُحسِّن ذلك أو يقلّل من أمان التشفير بأيّ شكلٍ من الأشكال (Daemen *et al.*, 1999, 8).



الشكل (5) هيكليّة المُشفِّر وفاكّ التشفير في عائلة SPN

يتمّ فكّ التشفير ببساطةٍ عن طريق عكس العمليّات (باستخدام تحويلات الاستبدال والتقليب العكسيّة) وتطبيق مفاتيح الجولات بترتيبٍ عكسيّ؛ كما هو مُوضَّحٌ في الشكل 5 (ب).

تجدر الإشارة إلى أنّه يُمكِن أن تكون لجولات فاكّ التشفير نفس بنية جولات المُشفِّر بحال كانت وظيفة النشر في طبقة التقليب خطّيّة؛ حيث يُمكِن المُبادَلة ما بين طبقة التقليب العكسيّة وطبقة إضافة المفتاح كونهما خطّيّتان مع تطبيق تحويل النشر على المفاتيح المُقابِلة في جدول المفاتيح (Stallings, 2017, 198).

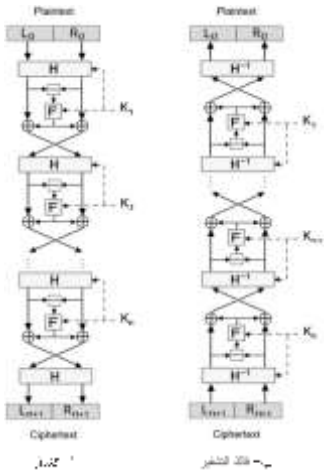
من أمثلة المُعمِّيات التي لها هيكليّة SPN: المُعمِّي K-64 SAFER، والمُعمِّي 3-Way، والمُعمِّي SQUARE، والمُعمِّي Serpent، والمُعمِّي CRYPTON، والمُعمِّي AES، والمُعمِّي KHAZAD، والمُعمِّي ARIA، والمُعمِّي PRESENT، والمُعمِّي PRINT، والمُعمِّي LED، والمُعمِّي KLEIN، والمُعمِّي PRINCE، والمُعمِّي Midori.

5- عائلة مُخطَّط Lai–Massey

سُمّيَت عائلة مُخطَّط Lai–Massey تيمُّناً بالباحِثَين Xuejia Lai و James L. Massey اللّذَين قاما بتقديمها، تجمع هيكليّة LM بين مزايا كلٍّ من هيكليّتي شبكة فيستيل وشبكة الاستبدال والتقليب، حيث يتمّ تقسيم الحالة إلى جزئين لهما نفس الطول ويجري خلط الجزء الأيسر والأيمن من الحالة في نفس الوقت ممّا يُسرِّع وظائف الخلط والنشر (Avanzi, 2016, 37).

بشكلٍ مُبسَّط؛ يأخذ مُشفِّر LM كتلة نصٍّ صريحٍ بطول $4X$ بت والمفتاح k كمُدخَلات، يتمّ تقسيم كتلة النصّ الصريح إلى نصفين كلّ منهما بطول $2X$ بت: يُشار إلى النصف الأيسر بالرمز $L_0 = (X_1, X_2)$، ويُشار إلى النصف الأيمن بالرمز $R_0 = (X_3, X_4)$، ثمّ تتمّ مُعالَجتهما عبر n جولةً بنفس الطريقة باستخدام المفاتيح الفرعيّة $(K_1, K_2, .., K_n)$، ثمّ يتّحِدان لإنتاج كتلة النصّ المُشفَّر.

في كلّ جولةٍ؛ يُمرَّر نصفا الكتلة إلى تابع نصف الجولة H مع جزءٍ من المفتاح الفرعيّ K_i ويتمّ الحصول على $(L_i', R_i') = H(L_i, R_i)$، ثمّ يتمّ تمرير الفرق بين L_i', R_i' إلى تابع الجولة F جنباً إلى جنب مع باقي أجزاء المفتاح الفرعيّ K_i، ثمّ تجري إضافة نتيجة تطبيق تابع الجولة إلى كلٍّ من L_i', R_i' باستخدام عملية XOR وتنتهي الجولة بإجراء عمليّة مُبادَلةٍ بين نصفَي الكتلة والحصول في الخرج على نصفَي الكتلة L_{i+1}, R_{i+1}.



الشكل (6) هيكليّة المُشفِّر وفاكّ التشفير في عائلة LM

كما هو مُوضَّحٌ في الشكل 6 (أ)؛ يُمكِن التعبير عن ناتج الجولة $1 \le i \le n$ في المُشفِّر كما يلي:

$$(L_i', R_i') = H(L_i, R_i, some\ of\ K_i)$$
$$T_i = F(L_i' - R_i', rest\ of\ K_i)$$
$$(L_{i+1}, R_{i+1}) = (R_i' \oplus T_i, L_i' \oplus T_i)$$

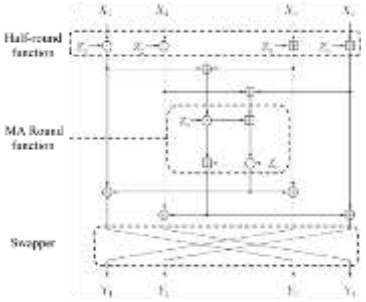
تكون لفاكّ التشفير نفس هيكليّة المُشفِّر؛ وتجري مُعالَجة النصّ المُشفَّر بنفس الطريقة المُتَّبعة في المُشفِّر لكن مع استخدام المفاتيح الفرعيّة بترتيبٍ عكسيّ، وهناك ضرورةٌ لعكس تابع نصف الجولة H^{-1} دون الحاجة إلى عكس تابع الجولة؛ ممّا يتيح بناء فاكّ التشفير بسهولةٍ أكبر؛ مع إمكانيّة تصميم تابع الجولة بشكلٍ مُستقِلٍّ ليكون مُعقَّداً بأيّ شكلٍ كان.

كما هو مُوضَّحٌ في الشكل 6 (ب)؛ يُمكِن التعبير عن ناتج الجولة $1 \le i \le n$ في فاكّ التشفير كما يلي:

$$(L_i', R_i') = H^{-1}(L_i, R_i, some\ of\ K_i)$$
$$T_i = F(L_i' - R_i', rest\ of\ K_i)$$
$$(L_{i+1}, R_{i+1}) = (R_i' \oplus T_i, L_i' \oplus T_i)$$

قدَّم Xuejia Lai (1992) المُعمِّي IDEA واقترح استخدام تابع جولةٍ مُركَّبٍ من عمليّات الضرب موديولو $(2^{16} + 1)$ يُرمَز لها بالرمز $\odot$ وعمليّات الجمع موديولو 2^{16} يُرمَز لها بالرمز $\boxplus$، ويُشار لتابع الجولة بالاسم MA للدلالة على هيكليّة الضرب والجمع (Multiplication-Addition) التي يستخدمها.

يتكوَّن مفتاح الجولة من ستّ كلماتٍ $(Z_1 \ .. \ Z_6)$ تُستخدَم الكلمات Z_1, Z_2, Z_3, Z_4 في تابع نصف الجولة والكلمتان Z_5, Z_6 في تابع الجولة؛ كما هو مُوضَّحٌ في الشكل 7 (Lai, 1992, 22).



الشكل (7) هيكليّة الجولة في المُعمِّي IDEA من العائلة LM

في عمليّة فكّ التشفير تُستخدَم نفس المفاتيح ولكن بترتيبٍ عكسيٍّ بحيث يتكوَّن المفتاح من الكلمات نفسها مع تعديلٍ بسيطٍ يشمل استخدام المعكوس الضربيّ للكلمتين Z_1, Z_2 والمعكوس الجمعيّ للكلمتين Z_3, Z_4 (Lai, 1992, 23). يُغني استخدام معكوس كلمات مفتاح الجولة Z_1, Z_2, Z_3, Z_4 عن الحاجة لعكس تابع نصف الجولة وبالتالي يُمكِن استخدام نفس الخوارزميّة في كلٍّ من المُشفِّر وفاكّ التشفير.

تُحقِّق هيكليّة المُعمِّي IDEA النشر الكامل اعتباراً من الجولة الأولى، حيث يعتمد كلّ بتٍّ في الخرج على جميع بتّات الدخل وباستخدام جميع بتّات المفتاح (Lai, 1992, 32).

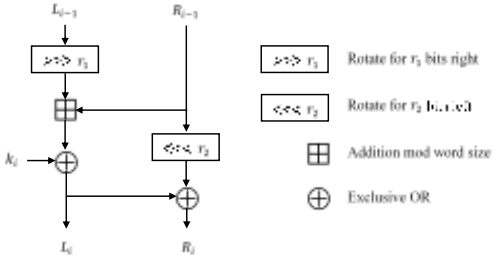
تُوجَد أمثلةٌ أخرى على مُعمِّياتٍ لها هيكليّة LM. على سبيل المثال؛ المُعمِّي MESH الذي يستخدم تابع الجولة ذا هيكليّة MA المُستخدَم في المُعمِّي IDEA، والمُعمِّي Akelarre الذي يستخدم تابع جولةٍ مُركَّباً من عمليّات الجمع موديولو 2^{32} وعمليّات التدوير المُعتمِدة على البيانات (data dependent rotation)، ويُشار له بالاسم

AR للدلالة على هيكليّة الجمع والتدوير (Addition-Rotation) المُستخدَمة.

6- عائلة عمليّات الجمع/ التدوير/ XOR

يستخدم مُعمِّي عائلة ARX العمليّات البسيطة التالية فقط: الجمع المعياريّ (modular addition)، والتدوير على مستوى البتّ (bitwise rotation)، وعمليّة XOR المنطقيّة. يعتمد تصميم مُعمِّيات عائلة ARX على الجمع المعياريّ مصدراً وحيداً يُوفِّر اللّاخطّيّة (Dinu *et al.*, 2016, 2).

يُعَدّ المُعمِّي SPECK مثالاً على مُعمِّيات ARX يُوفِّر أداءً أمثليّاً في التنجيزات البرمجيّة مع الحفاظ على مستوى أمانٍ مقبول، يُوضِّح الشكل 8 بنية الجولة في المُعمِّي SPECK (Beaulieu *et al.*, 2015, 16).



الشكل (8) بنية الجولة في المُعمِّي SPECK من العائلة ARX

يُوفِّر تصميم عائلة ARX ثلاث إيجابيّاتٍ رئيسيّة (Dinu *et al.*, 2016, 2):

1- يُؤدّي التخلُّص من عمليّات البحث في الجدول المُرتبِطة بالتصميمات القائمة على صناديق الاستبدال إلى زيادة المرونة ضدّ هجمات القناة الجانبيّة (side-channel attacks)، إضافةً إلى اختصار متطلبات الذاكرة عند التنجيز.

2- يُؤدّي استخدام العمليّات البسيطة فقط إلى تقليل عدد العمليّات الإجماليّة التي يُجريها المُعمِّي ممّا يسمح بتنجيزاتٍ برمجيّةٍ وعتاديّةٍ سريعة.

3- الكود البرمجيّ المطلوب لتنجيز المُعمِّي يكون صغيراً جداً.

يُعَدّ تصميم عائلة ARX الأنسب للاستخدام في الأنظمة مُقيَّدة الموارد لما يُوفِّره من ميزاتٍ تجعله مُناسِباً للاستخدام في المُعمِّيات الكتليّة خفيفة الوزن (light-weight) خاصّةً تلك التي تكون فيها مُتطلَّبات الذاكرة هي الأكثر صرامة.

تبقى المشكلة الأساسيّة المفتوحة للبحث في عائلة ARX هي التوصُّل إلى تصميمٍ يجعل المُعمِّي آمناً بشكلٍ مُؤكَّدٍ ضدّ تحليل الشيفرة الفرقيّ والخطّيّ (Dinu *et al.*, 2016, 2).

7- المُعمِّيات الهجينة

بناءً على دراسة عائلات المُعمِّيات الكتليّة التي تمّ استعراضها في الأقسام السابقة؛ اخترنا تسمية المُعمِّيات الهجينة (hybrid ciphers) للدلالة على المُعمِّيات ذات الهيكليّات المُختلطة بين تلك العائلات. تسعى المُعمِّيات الهجينة للاستفادة من مزايا الهيكليّات للحصول على أداءٍ مناسبٍ لتطبيقاتٍ خاصّة.

تُوجَد عدّة أمثلةٍ على المُعمِّيات الهجينة مثالها: المُعمِّي LEA ذو الهيكليّة العامّة GUFN type-3 ويستخدم عمليّات ARX، والمُعمِّي TEA والمُعمِّي XTEA اللّذان لهما الهيكليّة العامّة BFN ويستخدمان عمليّات ARX، والمُعمِّي FOX ذو الهيكليّة العامّة LM ويستخدم تابع جولةٍ ذو هيكليّة SPN، والمُعمِّي SIT ذو هيكليّة GUFN type-2 مُعدَّلةٍ ويستخدم تابع جولةٍ ذو هيكليّة SPN.

8- مُقارَنة عائلات المُعمِّيات الكتليّة والاستنتاجات

على الرغم من أنّ جميع عائلات المُعمِّيات الكتليّة تُحقِّق وظائف الخلط والنشر التي اقترحها كلود شانون لبناء مُعمٍّ كتليٍّ آمنٍ؛ إلّا أنّه تُوجَد اختلافاتٌ جوهريّةٌ فيما بينها

تجعل استخدام هيكليّةٍ مُعيَّنةٍ منها أكثر مُلاءمةً وقابليّةً للتطبيق في تنجيزاتٍ مُعيَّنة.

بناءً على ما تمّ شرحه في الفقرات السابقة سنقوم بالمُقارَنة بين عائلات المُعمِّيات الكتليّة من حيث قابليّة التنفيذ المُتوازي واستهلاك موارد المُعالَجة والذاكرة وإمكانيّة الاكتفاء بخوارزميّةٍ واحدةٍ في المُعمِّي وسرعة الخلط والنشر في تحويل الجولة.

تمتلك هيكليّة الجولة في مُعمِّيات SPN بنيةً تكوينيّةً تدعم قابليّة التنفيذ المُتوازي على مستوى حزم الحالة التي يُطبَّق عليها صندوق الاستبدال S-box، وبالتالي يُمكِن مُعالَجة جميع حزم الحالة على التوازي وبشكلٍ مُستقِلٍّ تماماً؛ كما هو الحال في المُعمِّي AES.

تكون مُعمِّيات FN أكثر قابليّةً للتنفيذ المُتوازي قياساً بمُعمِّيات LM وARX، حيث أنّ هيكليّة الجولة في مُعمِّيات LM وARX تتطلَّب تنفيذ العمليّات بالتناوب على نصفَي الحالة وفق مسارات مُعالَجةٍ تحدّ بشكلٍ كبيرٍ من قابليّة التنفيذ المُتوازي كما هو مُبيَّن في الشكل 7 والشكل 8.

بناءً على قابليّة التنفيذ المُتوازي؛ فإنّ توفُّر وحدات المُعالَجة المركزيّة CPUs (Central Processing Units) التي تحتوي على وحدات تنفيذٍ عديدةٍ يجعل العائلة SPN مُفضَّلةً للاستخدام مُقارَنةً بباقي العائلات.

تستهلك مُعمِّيات ARX موارد مُعالَجةٍ وذاكرةً قليلةً جدّاً لاعتمادها على العمليّات البسيطة واستغنائها عن عمليّات البحث في الجداول وكون الكود البرمجيّ المطلوب لتنجيز المُعمِّي صغيرٌ جدّاً.

ترتبط الموارد المُستهلَكة في باقي العائلات بنوع العمليّات المُنفّذة، ففي عائلة SPN تتعلَّق الموارد المُستهلَكة بطريقة تنجيز وظيفة الخلط في طبقة الاستبدال وتعقيد وظيفة النشر في طبقة التقليب.

تتعلَّق الموارد المُستهلَكة في عائلة FN بتعقيد العمليّات المُنفَّذة في تابع الجولة F، أمّا في عائلة LM عادةً ما تكون العمليّات المُنفَّذة في تابع نصف الجولة H وتابع الجولة F مُعقَّدةً نسبيّاً وتستهلك موارد مُعالَجةٍ كبيرةً قياساً بباقي العائلات؛ كما هو الحال في المُعمِّي IDEA والمُعمِّي MESH والمُعمِّي Akelarre.

بناءً على موارد المُعالَجة والذاكرة المطلوبة؛ تُعَدّ مُعمِّيات ARX الأنسب للاستخدام في الأنظمة مُقيَّدة الموارد من حيث قدرة المُعالَجة والتي تكون فيها مُتطلَّبات الذاكرة هي الأكثر صرامة، لذلك فهي مُناسِبةٌ لبناء المُعمِّيات الكتليّة خفيفة الوزن (light-weight).

تتمتّع مُعمِّيات FN بميزةٍ تلقائيّةٍ هي عدم الحاجة إلى خوارزميَّتَين مُختلفتين للتشفير وفكّ التشفير، حيث تسمح مُعمِّيات FN ببناء تابع الجولة F بصورةٍ مُستقِلّةٍ كوظيفةٍ أحاديّة الاتجاه مُعقَّدةٍ بأيّ شكلٍ كان؛ مع التركيز على خصائص الخلط المرغوبة دون أيّة قيودٍ تفرضها قابليّة الانعكاس.

في مُعمِّيات LM هناك ضرورةٌ لعكس تابع نصف الجولة H فقط دون الحاجة إلى عكس تابع الجولة F في المُشفِّر؛ ممّا يُتيح إمكانيّة تصميم تابع الجولة بشكلٍ مُستقِلٍّ ليكون مُعقَّداً بأيّ شكلٍ كان.

وبالتالي فإنّه تُوجَد حاجةٌ إلى خوارزميَّتَين مُختلفتَين للتشفير وفكّ التشفير في مُعمِّيات LM، يُمكِن الاكتفاء بخوارزميّةٍ واحدةٍ للتشفير وفكّ التشفير عند استخدام نفس تابع نصف الجولة في فاكّ التشفير، يقتضي ذلك استخدام معكوس كلمات مفتاح الجولة المُستخدَمة في تابع نصف الجولة في عمليّة فكّ التشفير؛ كما هو الحال في المُعمِّي IDEA.

تتطلَّب مُعمِّيات SPN وARX أن تكون وظائف الخلط والنشر في المُشفِّر قابلةً للعكس (من أجل فكّ التشفير)،

وبالتالي فإنّه بشكلٍ عامّ تُوجَد حاجةٌ إلى خوارزميَّتَين مُختلفتَين للتشفير وفكّ التشفير.

بما أنّ طبقة الاستبدال مُستقِلّةٌ وظيفيّاً عن طبقة التبديل في مُعمِّيات SPN؛ يُمكِن الاكتفاء بخوارزميّةٍ واحدةٍ للتشفير وفكّ التشفير بحال كانت تحويلات الخلط والنشر معكوسةً ذاتيّاً (involution) كما هو الحال في المُعمِّي KHAZAD.

إنّ عدم الحاجة إلى خوارزميَّتَين مُختلفتَين للتشفير وفكّ التشفير يُؤدِّي إلى اختصار منطقة الرقاقة المطلوبة لتنجيز المُعمِّي ككلّ، وبالتالي تتفرّد مُعمِّيات FN بكون منطقة الرقاقة المطلوبة لتنجيز المُعمِّي ككلّ أقلّ ما أمكن.

تُجري مُعمِّيات LM عمليّات خلط النصف الأيسر والأيمن من الحالة بنفس الوقت ممّا يُسرِّع وظائف الخلط والنشر، وبإمكانها أن تُحقِّق النشر الكامل اعتباراً من الجولة الأولى؛ كما هو الحال في المُعمِّي IDEA.

تستخدم مُعمِّيات ARX الجمع المعياريّ مصدراً وحيداً يُوفِّر اللّاخطّيّة ويُمكِنها أن تُجري عمليّات الخلط والنشر على نصفَي الحالة بنفس الوقت؛ كما هو الحال في المُعمِّي SPECK. لكنّ زيادة طول الحالة تعني زيادة طول المُعامِلات؛ وبالتالي تُصبِح موارد المُعالَجة المطلوبة لتنجيز عمليّات الجمع المعياريّ أكبر. لذلك تُستخدَم عمليّات ARX بشكلٍ هجينٍ مع هيكليّاتٍ أخرى كما هو الحال في المُعمِّي LEA والمُعمِّي TEA والمُعمِّي XTEA بما يُحقِّق التوازن بين سرعة النشر من جهةٍ وكلفة العمليّات المُنفَّذة من جهةٍ أخرى.

يُطبِّق تحويل الجولة في مُعمِّيات SPN طبقات الاستبدال والتقليب التي تُوفِّر وظائف الخلط والنشر على الحالة بالكامل، وترتبط خصائص النشر بتعقيد وظيفة النشر المُستخدَمة في طبقة التقليب.

يُعالِج تحويل الجولة في مُعمِّيات FN نصف الحالة (في شبكات BFN) أو عدداً من الفروع الهدف من الحالة (في شبكات GUFN)، وبالتالي تبقى وظائف الخلط والنشر مَحصورَةً في جزءٍ من الحالة خلال الجولة الواحدة؛ الأمر الذي يُبطِئ خصائص النشر.

تنعكس زيادة سرعة الخلط والنشر في تحويل الجولة إيجاباً على تصميم المُعمِّيات، بحيث تزداد مُقاوَمة المُعمِّي لتحليل الشيفرة الفرقيّ والخطّيّ عند إجراء نفس العدد من الجولات، وبالتالي ينخفض عدد الجولات المطلوب تنفيذها في المُعمِّي.

يُلخِّص الجدول 1 نتائج المُقارَنة بين ميزات عائلات المعمِّيات الكتليّة من حيث قابليّة التنفيذ المُتوازي واستهلاك موارد المُعالَجة والذاكرة وإمكانيّة الاكتفاء بخوارزميّةٍ واحدةٍ في المُعمِّي وسرعة الخلط والنشر في تحويل الجولة.

الجدول (1) مُقارَنةٌ بين ميزات عائلات المُعمِّيات الكتليّة /1/

الميزات / عائلة المُعمِّي	قابليّة التنفيذ المُتوازي	استهلاك موارد المُعالَجة والذاكرة	الاكتفاء بخوارزميّةٍ واحدةٍ في المُعمِّي	سرعة الخلط والنشر في تحويل الجولة
FN	مُتوسِّطة	مُتوسِّط	تلقائيّة	مُنخفِضة
SPN	تلقائيّة	مُتوسِّط	مُمكِنة	مُتوسِّطة
LM	ضعيفة	كبير	مُمكِنة	عالية
ARX	ضعيفة	مُنخفِض	صعبة	مُتوسِّطة

بناءً على المُناقَشة السابقة؛ يُمكِن التعبير عن ميزات عائلات المُعمِّيات الكتليّة المُبيَّنة في الجدول 1 بالتأثير المُقابِل لكلّ منها، كما هو مُبيَّنٌ في الجدول 2.

الجدول (2) مُقارَنةٌ بين ميزات عائلات المُعمِّيات الكتليّة /2/

الميزات / عائلة المُعمِّي	الاستفادة من وحدات CPU مُتعدِّدة وحدات التنفيذ	مُلاءَمة بناء المُعمِّيات خفيفة الوزن	تقليل منطقة الرقاقة عند تنجيز المُعمِّي ككلّ	عدد الجولات التي تُحقِّق مستوى الأمان المطلوب
FN	مُتوسِّطة	مُتوسِّطة	عالية	كبير
SPN	عالية	مُتوسِّطة	مُمكِنة	مُتوسِّط
LM	ضعيفة	ضعيفة	مُمكِنة	قليل
ARX	ضعيفة	عالية	ضعيفة	مُتوسِّط

يُفضَّل استخدام هيكليّة ARX في بناء المُعمِّيات الكتليّة خفيفة الوزن التي تُناسِب الأنظمة مُقيَّدة الموارد، فهي تُوفِّر أداءً أمثليّاً في التنجيزات البرمجيّة والعتاديّة مع الحفاظ على مستوى أمانٍ مقبول. ولكنّها لا تصلح للتطبيقات التي تتطلَّب مستويات أمانٍ عاليةً مضمونةً كونه يصعب التوصُّل إلى تصميمٍ يجعل مُعمِّي ARX آمِناً بشكلٍ مُؤكَّدٍ ضدّ تحليل الشيفرة الفرقيّ والخطّيّ.

تسمح تصميمات شبكة فيستيل غير المُتوازِنة المُعمَّمة GUFN ببناء مُعمِّياتٍ كتليّةٍ ذات ميزاتٍ جيّدةٍ من حيث مُقاوَمة تحليل الشيفرة الخطّيّ والفرقيّ، حيث يُمكِن استخدام هيكليّة GUFN المُكوَّنة من عدّة مراحل مختلفةٍ في الشبكة وتابع الجولة لتوفير مستوى أمانٍ عالٍ جدّاً ولكن على حساب سرعة التنفيذ وبساطة التصميم؛ كما هو الحال في المُعمِّي MARS.

تتميَّز مُعمِّيات SPN بتصميمها البسيط، فهي من ناحيةٍ أولى تُتيح وصفاً بسيطاً للمُعمِّي وبالتالي تقييماً أمنيّاً أكثر دقّةٍ، كما أنّها من ناحيةٍ أخرى تمنح المُصمِّمين مزيداً من الحرّيّة في تصميم طبقات الخلط والنشر في المُعمِّي.

يُمكِن استنتاج أنّ هيكليّة SPN هي الأنسب لبناء مُعمٍّ كتليٍّ مُعَدٍّ للاستخدامات العامّة يُوفِّر مستوى أمانٍ عالياً وسرعةً في الأداء وبساطةً في التصميم وسهولةً في التحليل تُساهِم في تطوير مستوىً أعلى من التيقُّن فيما يتعلّق بقوّة الخوارزميّة.

أخيراً؛ يُمكِن تصميم مُعمِّياتٍ هجينةٍ تستفيد من الهيكليّات المُقترَحة في العائلات الأساسيّة لتُوفِّر مزايا الأمان والأداء المناسبة، وتُعَدّ المُعمِّيات الهجينة التي تدمج بين هيكليّة ARX وهيكليّات FN الأكثر استخداماً بين المُعمِّيات الهجينة المعروفة.

التمويل: هذا البحث ممول من جامعة دمشق وفق رقم التمويل (501100020595).

References

[1] Acosta, A. J., Sánchez, E. T., Jiménez, C. J., & Mora, J. M. (2017). **Power and Energy issues on lightweight cryptography.** Journal of Low Power Electronics, Vol. 13, No. 3, pp. 326-337.

[2] Albermany, S. A. K. & RadiHamade, F. (2016). **Survey: Block cipher Methods.** International Journal of Advancements in Research & Technology (IJOART), Volume 5, Issue 11, pp. 11-22.

[3] Appel, M., Bossert, A., Cooper, S., Kuϭmaul, T., Löffler, J., Pauer, C., *et al.* (2016). **Block ciphers for the IoT – SIMON, SPECK, KATAN, LED, TEA, PRESENT, and SEA compared.** Report, ATHENE, National Research Center for Applied Cybersecurity.

[4] Avanzi, Roberto. (2016). **A Salad of Block Ciphers.** IACR ePrint Archive: Report 2016/1171, pp. 25.

[5] Batina, L., Das, A., Ege, B., Kavun, E. B., Mentens, N., Paar, C., *et al.* (2013). **Dietary Recommendations for Lightweight Block Ciphers: Power, Energy and Area Analysis of Recently Developed Architectures.** International Workshop on Radio Frequency Identification: Security and Privacy Issues (RFIDSec), Springer, Berlin, Heidelberg, LNCS (8262) pp. 103-112.

[6] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., & Wingers, L. (2015). **The SIMON and SPECK Lightweight Families of Block Ciphers.** Proceedings of the 52nd ACM/EDAC/IEEE Design Automation Conference (DAC), Article No. 175.

[7] Cazorla, M., Marquet, k., & Minier, M. (2013). **Survey and Benchmark of Lightweight Block Ciphers for Wireless Sensor Networks.** International Conference on Security and Cryptography (SECRYPT), published by IEEE, August 2015, Electronic ISBN: 978-9-8975-8131-1

[8] Daemen, J. & Rijmen, V. (1999). **AES Proposal: Rijndael.** Submission to NIST, Version2.

[9] Daemen, J. & Rijmen, V. (2002). **The Design of Rijndael: AES- The Advanced Encryption Algorithm.** Springer, Berlin, Heidelberg.

[10] Dinu, D., Perrin, L., Udovenko, A., Velichkov, V., Großschädl, J., & Biryukov, A. (2016). **Design Strategies for ARX with Provable Bounds: SPARX and LAX.** International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT), Springer, Berlin, Heidelberg, LNCS (10031) pp. 484-513.

[11] Eisenbarth, T., Gong, Z., Güneysu, T., Heyse, S., Indesteege, S., Kerckhof, S., *et al.* (2012). **Compact Implementation and Performance Evaluation of Block Ciphers in ATtiny Devices.** International Conference on Cryptology in Africa (AFRICACRYPT), Springer, Berlin, Heidelberg, LNCS (7374) pp. 172-187.

[12] Hosseinzadeh, J. & Bafghi, A. G. (2017). **Software Implementation And Evaluation Of Lightweight Symmetric Block Ciphers Of The Energy Perspectives And Memory.** International Journal of Engineering Education (IJEE), Vol. 9, No. 2.

[13] Kushwaha, P. K., Singh, M. P., & Kumar, P. (2014). **A Survey on Lightweight Block Ciphers.** International Journal of Computer Applications, Vol. 76, No. 17, pp. 1-7.

[14] Lai, Xuejia. (1992). **On the Design and Security of Block Ciphers.** Doctoral Thesis, ETH Zürich

[15] Rana, S., Wadud, M. A. H., Azgar, A., & Kashem, M. A. (2019). **A Survey Paper of Lightweight Block Ciphers Based on Their Different Design Architectures and Performance Metrics.** International Journal of Computer Engineering and Information Technology, Vol. 11, No. 6, pp. 119-129.

[16] Schneier, B. & Kelsey, J. (1996). **Unbalanced Feistel Networks and Block Cipher Design.** International Workshop on Fast Software Encryption (FSE), Springer, Berlin, Heidelberg, LNCS (1039) pp. 121-144.

[17] Schneier, Bruce. (1996). **Applied Cryptography: Protocols, Algorithms, and Source Code in C.** John Wiley & Sons, Inc. ISBN: 0471128457.

[18] Shirey, Robert W. (2007). **Internet Security Glossary, Version 2.** Internet Engineering Task Force (IETF), Network Working Group, RFC 4949 (Informational).

[19] Stallings, William. (2017). **Cryptography and Network Security Principles and Practice.** Seventh edition global edition, Pearson Education.