فصل –2 التشفير بالمفتاح الخاص

2.1 الأمنية الاحتسابية

- ذكرنا في الفصل السابق ، ان النص المشفّر في مناهج التشفير تامّة الأمنية لايكشف اي معلومة عن النص الصريح (على افتراض سرية المفتاح).
- نقدّم في هذا الفصل مستوى جديد للامنية يعرف **بالامنية الاحتسابية** computational security والتي تعد هدف مناهج التشفير الحديثة ، حيث تسمح للخصم بكسر الشفرة عند توفر الوقت والاحتساب الكافيين
  - تحت بعض الافتراضات ، تتجاوز كمية الاحتسابات المطلوبة لكسر هذه المناهج العمر باكمله ، حتى في حالة استخدام اسرع الحواسيب ، مما يجعل مستوى امنيتها كافيا.
- يعتبر مستوى الأمنية الاحتسابية "اضعف" من مستوى الأمنية التامة. ويعود سبب استخدامه حاليا الى كفائته ، حيث لايشترط ان يكون طول مفتاح التشفير مساوٍ لطول النص الصريح ، كما هو الحال في الأمنية التامة.

تعريف(2.1): تكون f دالة **ضئيلة جدا negligible** اذا كان لجميع متعددات الحدود $p(.)$ يوجد N ، بحيث لكل قيم $n > N$ يتحقق $f(n) < \frac{1}{p(n)}$.

مثال(2.1): تعتبر الدالة 2^{-n} ضئيلة جدا ، حيث لجميع قيم $n \geq 20$ تكون $2^{-n} < 10^{-6}$.

- تتضمن مناهج التشفير ذات الأمنية الاحتسابية تحديد **معامل أمنية** security parameter على شكل عدد صحيح n ، بحيث يكون معروف لدى جميع الاطراف بما فيهم الخصم. يكون وقت التنفيذ للخصم (ووقت الاطراف الاخرى) وكذلك احتمالية نجاح الخصم بشكل دالة بالنسبة الى n.
- يحدّد معامل الأمنية طول المفتاح الخاص ، وبالتالي حجم فضاء المفتاح ، وعليه فإن استخدام معامل أمنية كبير يجعل من الصعب اجراء مهاجمة brute force وبالتالي يوفر أمنية أكبر للنظام.

2.2 منهج التشفير الانسيابي

قبل الشروع بتصميم مناهج التشفير الآمنة التي توفر الأمنية الاحتسابية. نقدّم تعريف **المولّد شبه العشوائي** pseudorandom generator.

المولد شبه العشوائي. هو خوارزمية محددة deterministic تستلم بذرة seed قصيرة عشوائية تماما وتوسّعها الى خيط رمزي طويل شبه عشوائي. ليكن n هو طول البذرة التي تعتبر ادخال المولّد وليكن $\ell(n)$ هو طول الاخراج.

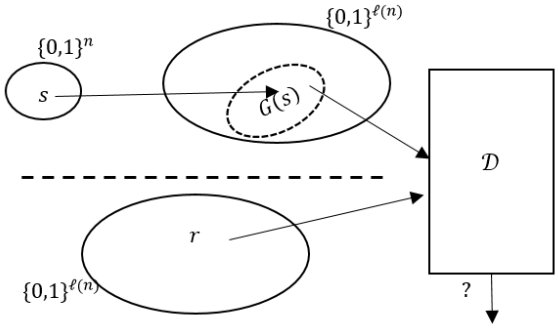
تعريف(2.3): ليكن $\ell(.)$ متعددة حدود وليكن G خوارزمية محدّدة متعددة الحدود بحيث عند استلام $s\epsilon\{0,1\}^n$ ، تخرج الخوارزمية G خيط رمزي بطول $\ell(n)$. نقول ان G هو مولّد شبه عشوائي اذا تحقق الشرطين التاليين:

1. **شرط التوسيع**: لكل n يكون $\ell(n) > n$.
2. **شرط شبه العشوائية**: لجميع خوارزميات التمييز متعددة الحدود D ، توجد دالة ضئيلة $negl$ ، بحيث يكون:

$$|Pr[D(r) = 1]| - |Pr[D(G(s)) = 1]| \leq negl(n)$$

بحيث r متغير عشوائي حقيقي يتم اختياره من $\{0,1\}^{\ell(n)}$ ، والبذرة s يتم اختيارها بصورة عشوائية من $\{0,1\}^n$. تعرف الدالة $\ell(n)$ **بعامل التوسيع** للمولّد G.

- ينص التعريف باختصار على انه لايوجد **مميّز** D يستطيع ان يميّز بين الاخراج شبه العشوائي للمولّد $G(s)$ والخيط العشوائي الحقيقي r الّا بنسبة نجاح ضئيلة جدا.
- يوضّح الشكل (2.1) تجربة تمييز المولّد شبه العشوائي.



شكل (2.1): تمييز المولّد شبه العشوائي

- في هذا الفصل ، سوف نفترض وجود المولّدات العشوائية. اما كيفية تصميمها ومناقشة امنيتها فسوف نؤجلها الى فصول اخرى.

التشفير الانسيابي لرسائل ثابتة الطول. اصبحنا الأن جاهزين لتصميم منهج تشفير ثابت الطول بحيث لايمكن تمييز نصوصه المشفّرة من قبل الخصم.

- يشبه المنهج المقترح تماما منهج تشفير one-time pad باستثناء استخدام خيط شبه عشوائي بدلا من المفتاح الطويل. وبما ان الخيط شبه العشوائي "يبدو عشوائيا" لاي خصم ، فأنه يمكن اثبات أمنية المنهج عن طريق "افتراض" وجود مولّد شبه عشوائي.
- يوضّح المنهج (2.1) منهج التشفير ثابت الطول.

Construction (2.1): Fixed–length stream cipher scheme

Assumption. Let G be pseudorandom generator with expansion factor ℓ

Input: message, $m \in \{0,1\}^{\ell(n)}$

Output: ciphertext, $c \in \{0,1\}^{\ell(n)}$

Key generation: Generate the key $k \leftarrow \{0,1\}^n$ randomly.

Encryption: $c = m \oplus G(k)$.

- لاحظ ان منهج (2.1) يقوم ايضا بتطبيق عملية XOR بين ثنائيات الرسالة والمفتاح كما هو الحال في منهج (one time pad). الفرق الرئيسي بينهما هو طول المفتاح: في منهج (2.1) يكون المفتاح اقصر بكثير من المفتاح في منهج (one time pad) ، حيث تمثّل البذرة مفتاح المنهج الجديد ويتم توسيعها لتكون بطول الرسالة من قبل المولّد G.
- المنهج الذي ذكرناه اعلاه يعرف عادة **بالشفير الانسيابي** stream cipher ، وهو يعتمد على توليد سلسلة من الثنائيات شبه العشوائية ومن ثم تطبيق عملية XOR بين السلسلة الناتجة وسلسلة ثنائيات النص الصريح. ونقصد بالتشفير الانسيابي هو "الاداة" المستخدمة لتوليد سلسلة المفتاح وليس طريقة تشفير مستقلة.
- تعتبر بعض المصادر ان التشفير الانسيابي هو تشفير مستقل وهو اعتبار خاطئ كما ذكرنا.

التشفير الانسيابي لرسائل متغيرة الطول. يقتصر المنهج (2.1) على تشفير رسائل ثابتة الطول. يمكن تجاوز هذه المشكلة بتصميم مولّدات شبه عشوائية متغيرة الطول ، بحيث تنتج خيوط شبه عشوائية بأي طول مرغوب.

- تستلم هذه المولّدات ادخالان: البذرة s وطول الاخراج المرغوب ℓ. يرمز للمولد بالشكل $G(s, \ell)$. وفق هذا السياق ، تتم عملية تشفير الرسالة m بالمفتاح k بالشكل $c = G(k, |m|) \oplus m$ ، اما عملية فك الشفرة فتتم بالشكل $m = G(k, |c|) \oplus c$.

2.3 الأمنية تحت مهاجمة (CPA)

- لحد الأن ، تم افتراض وجود خصم ضعيف يشاهد فقط النص المشفر.
- في هذا المقطع ، سيتم التعامل مع مهاجمة اقوى تعرف بمهاجة chosen-plaintext attack(CPA). فكرة هذه المهاجمة هي ان الخصم يستطيع ان يطلب تشفير اي نص صريح من اختياره ، بمعنى ان لدى الخصم القدرة على الوصول الى خوارزمية التشفير.

تصميم مناهج تشفير من نوع CPA

- يتم تصميم مناهج التشفير من نوع CPA بالاستعانة بأداة تعرف **بالدالة شبه العشوائية** pseudorandom function ، وهي دالة ثنائية الادخال $F: \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$ ادخالها الاول هو المفتاح والادخال الأخر هو الرسالة.
- للبساطة ، سنفترض ان هذه الدالة حافظة للطول ، بمعنى ان المفتاح ، الادخال ، والاخراج لهم جميعا نفس الطول.
- عند اختيار المفتاح $k \in \{0,1\}^n$ تقوم الدالة $F_k(.)$ بتحويل ادخال بطول n ثنائية الى اخراج بطول n ثنائية ايضا.
- تسمى الدالة $F_k(.)$ بدالة شبه عشوائية اذا لايوجد مميّز يستطيع ان يميّزها عن دالة اخرى f يتم اختيارها بصورة عشوائية بحيث كلا الدالتين لهما نفس المجال والمجال المقابل.

تعريف(2.4): لتكن الدالة $F: \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$ دالة مفتاحية كفوءة. تدعى F **بالدالة شبه العشوائية** اذا كان لجميع خوارزميات التمييز D ، توجد دالة ضئيلة جدا $negl$ بحيث:

$$|Pr[D(F_k(n)) = 1]| - |Pr[D(f_k(n)) = 1]| \leq negl(n)$$

بحيث ان $k \leftarrow \{0,1\}^n$ يتم اختياره عشوائيا وتمثل $f_k(.)$ دالة عشوائية حقيقية لخيوط بطول n من الثنائيات.

- يوضّح المنهج(2.2) طريقة التشفير من نوع CPA ، والذي يُعد منهج احتمالي probabilistic ، حيث يطبّق F_k على القيمة العشوائية r ومن ثم تطبيق عملية XOR

Construction (2.2): CPA–secure encryption scheme

Assumption. Let F be pseudorandom function of length n

Input: message, $m \in \{0,1\}^n$

Output: ciphertext, $c \in \{0,1\}^n$

Key generation: Generate the key $k \leftarrow \{0,1\}^n$ randomly.

Encryption: Generate $r \leftarrow \{0,1\}^n$ randomly and compute

$$c =< r, m \oplus F_k(r) >.$$

Decryption: On receiving $c =< r, s >$, and k, compute

$$m = s \oplus F_k(r).$$

بين الناتج مع النص الصريح.

- تعتمد أمنية هذا المنهج على كون F_k دالة شبه عشوائية وذلك لكي تحمي النص الصريح كما هو حال طريقة one-time pad ، شريطة عدم تكرار قيمة r.
- مما يعيب هذا المنهج هو قلّة الكفاء ، حيث ان حجم النص المشفّر المرسل هو ضعف حجم النص الصريح.
- نقدّم الأن حلا اكثر كفاءة وذلك باستخدام **دوال البعثرة شبه العشوائية** pseudorandom permutation functions ، والتي تمثل المدخل **للتشفير الكتلي** block cipher ، حيث يتم تقسيم النص المطلوب تشفيره الى **كتل** (بدلا من تشفير سلسلة).
- يضمن هذا الاسلوب ان يكون طول النص المشفّر مساوٍ لطول النص الصريح (بدلا من ضعف الطول) اضافة الى توفير أمنية تجاه CPA.
- ليكن لدينا الدالة $F: \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$ دالة كفؤة ، حافظة للطول ، ومعتمدة على مفتاح.
- تعرف الدالة F بالبعثرة المفتاحية اذا كان لجميع قيم k ، تكون $F_k(.)$ من نوع واحد لواحد ، بمعنى كونها دالة تقابلية يوجد فيها لكل عنصر في المجال هناك عنصر وحيد في المجال المقابل.
- لكي تكون $F_k(.)$ دالة بعثرة شبه عشوائية فانه يشترط عدم وجود مميز distinguisher يستطيع تمييزها عن البعثرة العشوائية الحقيقية $f_k(n)$ كما يشترط ان يكون للدالة $F_k(.)$ دالة معكوس $F(.)^{-1}$.
- لاحظ ان الدالة $F_k(.)$ تمثل عملية التشفير $Enc_k(.)$ والدالة $F(.)^{-1}$ تمثل عملية فك الشفرة $Dec_k(.)$.

تعريف (2.5): لتكن الدالة $F: \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ **دالة بعثرة مفتاحية كفؤة**. تدعى F بالبعثرة العشوائية اذا كان لجميع خوارزميات التمييز D ، توجد دالة ضئيلة جدا negl بحيث:

$$\left|\Pr[D(F_k(n), F_k^{-1}(n)) = 1]\right| - \left|\Pr[D(f_k(n), f_k^{-1}(n)) = 1]\right| \leq \text{negl}(n)$$

بحيث ان $k \leftarrow \{0,1\}^n$ يتم اختياره عشوائيا و تمثل $f_k(.)$ دالة بعثرة عشوائية حقيقية لخيوط بطول n من الثنائيات.

- نكرر هنا ، ان دوال البعثرة شبه العشوائية $F_k(.)$ تستخدم كأدوات لتصميم مناهج التشفير الكتلي. وكما ذكرنا في التشفير الانسيابي ، فإن التشفير الكتلي $F_k(.)$ بحد ذاته ليس منهج تشفير مستقل بل اداة تستخدم لتصميم مناهج تشفير آمنة.
- على سبيل المثال ، فإن منهج التشفير الذي يعمل بمجرد حساب $c = F_k(.)$ لايعتبر منهج أمن بحد ذاته لانه لايقاوم المهاجمة من نوع CPA.
- في تكملة هذا الفصل فسوف نفترض وجود دوال البعثرة شبه العشوائية $F_k()$ ونؤجل تصميمها الى الفصل القادم. نفترض ان اطوال كتل المدخلات والمخرجات سيكونان متساويان وهو مايعرف بطول الكتلة block size ، في حين ان طول المفتاح سيكون مختلف عن طول الكتلة.
- عند تشفير نص معيّن بطريقة التشفير الكتلي فأنه يتم تقسيمه الى كتل وتشفّر هذه الكتل بتطبيق الدالة $F_k()$ على كل كتلة. تتم عملية التشفير وفق نمط عمل معيّن. يبين الجزء التالي انماط العمل المستخدمة في التشفير الكتلي.

2.4 انماط العمل

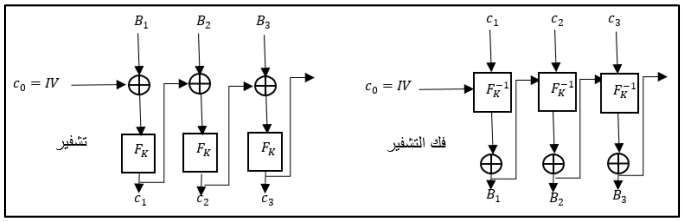
نمط العمل هو طريقة اساسية لتشفير رسائل مختلفة الطول. يتم تحشية pad الرسالة ليصل طولها الى احد مضاعفات طول الكتلة وذلك باضافة الثنائية 1 متبوعة بعدد من الاصفار (10^*) ، وبذلك سوف نفترض ان طول الرسالة هو بالضبط عدد يقبل القسمة على طول الكتلة. نفترض وجود دالة بعثرة شبه العشوائية بطول n وان عدد كتل الرسالة المطلوب تشفيرها هو ℓ ،كل كتلة بطول n.

النمط الاول: Electronic Code Mode (ECM). يعتبر ابسط نمط لتشفير البيانات. ليكن لدينا الرسالة $B = B_1, B_2, \dots, B_\ell$ فأنه ينتج النص المشفّر عن تشفير كل كتلة بصورة مستقلة. نعني بالتشفير تطبيق دالة البعثرة شبه العشوائية للكتلة الصريحة. حيث نحصل على $c =< F_k(B_1), F_k(B_2), \dots, F_k(B_\ell) >$. يتم فك الشفرة بنفس الطريقة لكن باستخدام الدالة F^{-1}. لاحظ ان طول النص المشفّر مساوٍ لطول النص الصريح ، وهو اقصر من النص المشفّر الناتج من منهج (3.2)

- يمتاز هذا النمط بكون التشفير محدد deterministic وبالتالي فإن هذا النمط لا يمكن ان يكون آمنا بالنسبة لمهاجمة CPA ، حيث ان تكرار نفس الكتلة في النص الصريح يسبب تكرار نفس النص المشفر ، لذا يجب ان لايستخدم هذا النمط.

النمط الثاني: Cipher Block Chaining (CBC) mode. في هذا النمط يتم اختيار **متجّه عشوائي ابتدائي** IV بطول n. تشفّر الكتلة الأولى بحساب $IV \oplus B_1$. نحصل على بقية الكتل المشفرة بتطبيق عملية XOR بين الكتلة المشفرة رقم $i-1$ مع الكتلة الصريحة رقم i. بمعنى اخر ، يتم اولا اسناد $c_0 = IV$ ، بعدها نحسب $c_i = F_k(c_{i-1} \oplus B_i), \forall i = 1, \dots, \ell$. يكون النص المشفر النهائي بالصورة $< IV, c_1, \dots, c_\ell >$. لايعتبر IV قيمة سرية بل يجب ان يرسل للطرف المستلم كي تتم عملية فك الشفرة.

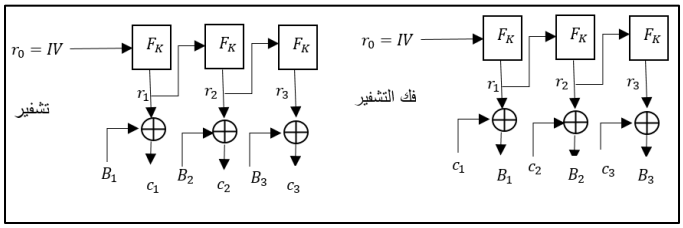
- يمتاز هذا النمط بكونه عشوائيا وبالتالي يحقق أمنية CPA. المشكلة الرئيسة لهذا النمط هي ان التشفير يجب ان يتم بصورة متسلسلة لأن النص المشفّر c_i يجب ان يكون متوفرا لتشفير الكتلة B_{i+1}. يوضّح الشكل (2.2) كيفية تشفير وفك شفرة كتل البيانات باستخدام نمط CBC.



شكل (2.2): عمل نمط التشفير CBC

النمط الثالث: Output Feedback (OFB) mode. يرتكز هذا النمط على استخدام التشفير الكتلي لتوليد سلسلة مفتاح شبه عشوائية ومن ثم تطبيق عملية XOR بين تلك السلسلة والرسالة الصريحة. اولا ، يتم اختيار المتغير العشوائي $IV \leftarrow \{0,1\}^n$ ومن ثم يتم توليد سلسلة عشوائية بمعزل عن النص الصريح. ليكن $r_0 = IV$ ، تحسب بقية الكتل $r_i = F_k(r_{i-1})$. يتم تشفير الكتلة رقم i بتطبيق عملية XOR ، $c_i = r_i \oplus B_i$. يتضمن النص المشفر الكتل المشفرة بالاضافة الى قيمة IV.

- يمتاز هذا النمط بكونه عشوائي ، لايحتاج الى معكوس الدالة F ، وتتم عمليات التشفير و فك الشفرة بصورة متسلسلة ايضا كما هو الحال بالنمط CBC. يتفوّق هذا النمط بأن توليد سلسلة المفتاح ، والتي تتطلب عمليات مكلفة زمنيا ، يتم بمعزل عن النص الصريح وبالتالي يمكن توليد المفتاح قبل تشفير الرسالة مما يسمح بان تجري عمليات التشفير بصورة سريعة للغاية. يوضّح الشكل (2.3) كيفية تشفير وفك شفرة كتل البيانات باستخدام نمط OFB.

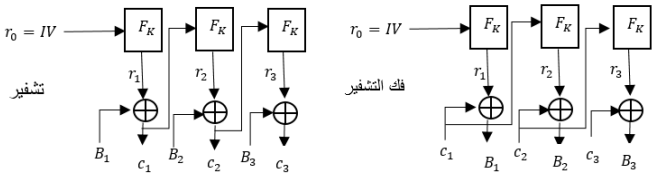


شكل (2.3): عمل نمط التشفير OFB

النمط الرابعة: Ciphertext Feedback (CFB) mode، يعمل هذا النمط على توليد سلسلة عشوائية اولا. نبدأ بـ $r_0 = IV$ ومن ننتج كتلة المفتاح r_i وذلك بتشفير النص المشفّر السابق. بمعنى ان

$$r_i = F_k(c_{i-1}): i \geq 1$$

يتم تشفير الكتل باستخدام المعادلة $c_i = B_i \oplus r_i$. يستخدم هذا النمط ايضا F_k لعملية فك الشفرة ، كما هو الحال في نمط OFB. يوضّح الشكل (2.4) كيفية تشفير وفك شفرة كتل البيانات باستخدام نمط CFB.



شكل (2.4): عمل نمط التشفير CFB

النمط الخامس: Counter (CTR) mode، يعمل هذا النمط ايضا على توليد سلسلة مفتاح شبه عشوائية. اولا ، يتم اختيار المتغير العشوائي $IV \leftarrow \{0,1\}^n$ ويرمز للمتغير IV بالرمز ctr ، ثم يتم توليد السلسلة العشوائية بحساب

$$.r_i = F_k(ctr + i)$$

اخيرا ، تشفّر الكتلة رقم i بالشكل: $c_i = r_i \oplus B_i$. يتضمن النص المشفر: الكتل المشفرة بالاضافة الى قيمة IV. لاحظ ، ان هذا النمط لايحتاج الى معكوس الدالة F.

- يتميز هذا النمط بكون التشفير فيه يكون عشوائي ، كما يمكن تشفير الكتل وفك شفرتها بصورة متوازية ، اضافة الى امكانية توليد سلسلة المفتاح بصورة مستقلة عن النص المطلوب تشفيره. اخيرا ، يسمح هذا النمط بفك شفرة الكتلة رقم i بدون اي كتلة اخرى وهذا مايعرف بالوصول العشوائي random access. هذه الصفات ، تجعل نمط CTR اختيارا جذّابا.

طول الكتل ومستوى الأمنية. جميع انماط التشفير السابقة ، باستثناء نمط ECB ، تستخدم متغير عشوائي IV. يؤثر هذا المتغير على عشوائية عملية التشفير ، وبالتالي يضمن تشفير نص جديد في كل مرة. هناك عامل اخر يتحكم بمستوى الأمنية في التشفير الكتلي ، بالاضافة الى مفتاح التشفير ، وهو طول الكتلة. كلما زاد طول الكتلة ، كلما زادت الأمنية.

التشفير الانسيابي والتشفير الكتلي

كما رأينا في انماط التشفير ، انه من الممكن الحصول على نمط التشفير الانسيابي بتوليد سلسلة مفتاح عشوائية باستخدام التشفير الكتلي ومن ثم تطبيق عملية XOR بين ثنائيات الرسالة الصريحة والسلسلة العشوائية. يمتاز التشفير الانسيابي بكونه اكفأ من التشفير الكتلي مما يجعله مفضّلا في الاجهزة محدودة الموارد كاجهزة الموبايل. بالمقابل ، فإن التشفير الانسيابي يكسر باستمرار وبالتالي تكون امنيته اقل ، لذا لاينصح باستخدامه مالم يكن هناك حاجة فعلية.

2.5 الأمنية تحت مهاجمة CCA

لحد الأن ، تم تعريف الأمنية تجاه نوعين من الخصوم: النوع الاول هو نوع خامل يقوم فقط بمشاهدة النص المشفر ، بينما الثاني هو نوع نشط يقوم بمهاجمات من نوع CPA. النوع الثالث من الخصوم هو الخصم الذي يجري مهاجمات من نوع chosen-ciphertext attacks (CCA) ، حيث يكون اقوى من النوعين السابقين. في مهاجمة CCA يكون لدى الخصم القدرة على تشفير اي نص من اختياره وله القدرة ايضا على فك شفرة اي نص مشفّر. بصورة ادق ، له الحق في الوصول الى خوارزميتي التشفير وفك الشفرة.

سوف يتم تأجيل عرض انظمة التشفير ذات أمنية CCA الى حين دراسة ادوات تصميمها ، وهي شفرات توثيق الرسائل message authentication codes في الفصول القادمة.