الخلاصة

من امثلة الانظمة القياسية الحالية في أنظمة تشفير ذات المفتاح المشترك هي (Serpent, Two Fish, Rc6, Mars, Rijndeal) كـ (final list candidate). رغم ان خوارزمية الـ (Serpent) تعطي السرية لكن تم ادخال التحسينات على هذه الخوارزمية مما يؤمن قوتها للنسخة اللاحقة (Next Generation).

في هذه الاطروحة ، الخوارزمية المطورة مصممة للاستفادة من القوة التي تدعمها خوارزمية الـ (Serpent) مع التغلب على نقاط الضعف الموجودة فيها. لكي تكون النتائج المتوخاة من هذه الخوارزمية ، مما يضمن تحسن في السرية الاداء من بين طرق التشفير الحالية و افضل مما متوفر في خوارزمية الـ (Serpent).

حجم الكتلة يزداد الى 256 bits بدلاً من 128 bits باستعمال دالة الدورة (Round Function) في بناء (Feistel) ، و كذلك تستعمل الخوارزمية المطورة الدالة الخاضعة للمفتاح قبل وبعد كل دورة بدلاً من الطرق التقليدية باستخدام التبديل الاولي و النهائي وبالاستعانة بجداول ثابتة مما يعطي للخوارزمية المقترحة الحماية ضد الـ(Cryptanalysis) بنوعيه (Linear and Differential).

النتائج التي تم الحصول عليها توضح بان الخوارزمية المقترحة تملك نفس خصائص خوارزمية التشفير من ناحية التشفير وفك الشفرة باستعمال نفس خوارزمية تقسيم المفتاح. وكذلك تكون أمينة أكثر وانها تستعمل الـ(Permutation) من نوع (Key-Dependent) مما يضمن الحماية ضد الـ(Cryptenalysis) بنوعيه (Linear, Differential).

وقت التنفيذ في خوارزمية الـ(Serpent) كان (23.9) دقيقة في (100000) بايت، بينما بعد التطوير كان (20.77) دقيقة في (100000) بايت.

كذلك معدل (Avalanche Effect)في خوارزمية الـ (Serpent) كان 61.1, بينما معدل (Avalanche Effect) في الخوارزمية بعد التطوير كان 132.6.

وكذلك بسبب كبر المفتاح و حجم الكتلة الداخلة للخوارزمية المقترحة لذلك (Exhustive Key Search) و الـ (Matching Cipher Text Attack) غير ممكن.

تم بناء الخوارزمية باستخدام لغة البرمجة (Visual C++) مما يضمن الحصول على افضل واجهات للمستفيد (GUI).

PDF created with pdfFactory trial version www.pdffactory.com

Abstract

The current standards for shared –key encryption are: Two Fish, Serpent, Mars, Rc6 and Rijndeal as final list candidates.

Although Serpent has provided a secure encryption algorithm an improvement will be submitted and implemented for this Serpent algorithm which is well suited for the next generation.

In this thesis a improved algorithm is designed to take advantage of the powerful algorithm, which is supported by Serpent algorithm with overcoming weakness, resulting in a much improved security /per formance trade off over existing chiphers.

As a result, this improved algorithm offers more secure than Serpent. The block size can be increased to 256 bits instead of 128 bits by using round function in a Feistel construction, also the improved algorithm uses key dependent function before and after each round instead of initial and final permutation which uses fixed tables. This give the algorithm, aprotection against differential and linear cryptanalysis.

The results obtained illustrate that the improved algorithm uses the same algorithm criteria for encryption and decryption with some key schedules and adopts key –dependent permutation and substitution to provide protection against differential and linear cryptanalysis.

The execution time of the circuit algorithm was 23.9 second for 100000 bytes, while after the improved was 20.77 second for 100000 bytes.

Also the average of avalanche effect of the circuit algorithm is 61.1, while the average avalanche effect of the algorithm after improvement is 132.6 . The large key and input block size of the improved algorithm cause exhaustive key search and the matching chiphertext attack are infeasible.

The algorithm was implemented using a programming of visual C++ to provide a good user interface.

PDF created with pdfFactory trial version www.pdffactory.com