المستخلص

يتناول هذا البحث موضوع تأمين عملية تبادل المعلومات بين الأنظمة المربوطة والمختلفة عبر شبكات الإتصال المحلية والواسعة،وذلك سعياً لتفادي الإختراقات الأمنية الناتجة عن الربط المباشر بين تلك الأنظمة.

ولتحقيق هذا الهدف تم إستخدام مبدأ التراسل كوسيلة للربط بين الأنظمة وجعل الرسائل هي الوسيلة لنقل المعلومات فيما بينها، ومن ثمّ تم القيام بتأمين النظام المسئول عن التراسل.

لتبادل رسائل المعلومات بين الأنظمة المربوطة تقوم شيفرة نقطة الإتصال لوسيط التراسل بإستخلاص رسالة طلب المعلومة من قاعدة بيانات أحد أطراف الإتصال ومن ثمّ تأمين تلك الرسالة وإرسالها عبر شبكة الإتصال؛ وبالمقابل يقوم طرف الإتصال المعني برسالة الطلب بإستقبالها ومعالجتها ومن ثمّ يقوم بالإستعلام عن المعلومات المطلوبة من قاعدة البيانات الخاصة به وإرسال النتيجة في شكل رسالة مرة أخرى بعد تأمينها إلى الطرف الأول.

تم تحسين نظام للتراسل من الأنظمة مفتوحة المصدر (Message Queue For C Plus Plus) وذلك بإضافة شيفرة للإتصال بقواعد بيانات Microsoft SQL Server مدعومة بنموذج لتأمين الرسائل والذي استخدم فيه عدة وسائل وهى تشمل الهيكلة بإستخدام متجهات الصفوف، الترجمة بالتحويل إلى الصورة الثنائية، الضغط بإستخدام خوارزمية ZIP وأخيرًا التشفير بإستخدام خوارزمية Rijndael القياسية.أيضاً تم التحكم في صلاحيات الوصول لنظام التراسل عبر منح صلاحية الوصول فقط للمستخدمين المصرّح لهم، بالإضافة إلى ذلك تم تسجيل حركة كل الرسائل في ملفات المراقبة الخاصة بالنظام.

تم تطوير النظام بإستخدام لغة البرمجة C++، بالإضافة إلى ذلك تم تطوير واجهة للتخاطب بإستخدام بيئة تطوير التطبيقات Delphi وهذا بهدف تشغيل النظام؛ ومن ثمّ تمت تجربته في بيئة Windows XP.

عند تجربة النظام تم إستخدام رسائل عديدة بأطوال مختلفة، وبالتالي تمت دراسة مدى تاثير نموذج التشفير – الذي يعتبر الجزء الأهم في نموذج التأمين المقدّم – على أطوال الرسائل وعلى زمن إرسال الرسائل في حالة وجود نموذج التشفير وفي حالة عدم وجوده. وبالإختبار وجِدَ أن وجود نموذج التأمين ككل أضاف درجة كبيرة من التعقيد والسرية على الرسالة بالإضافة إلى وجود نظام التشفير ضمن نموذج التأمين ليس ذا تأثير كبير على معدّل زمن إرسال الرسائل ويزيد على الأكثر 15 بايت لكل رسالة مرسلة مهما كان طولها.

Abstract

This research addresses the issue of securing the exchange of information between different systems interconnected via wide networks, in order to avoid security penetrations resulting from the direct link between those systems. To achieve this goal, the principle of messaging has been used as a way to integrate the systems and this make the messages as a media for transferring information among them, and then the messaging system which is responsible for this has been secured.

For the exchange of information messages between integrated systems, the end-point code of the messaging middleware extracts the requesting information message from the database of one of the communication parties and thereby secures that message and sends it over the network; on the other hand the communication party who is concerned with the request information message will receive and process this message and then queries the requested information from the database and sends the result in the form of a message again after securing it to the first party.

The open-source system (Message Queue For C Plus-Plus) has been enhanced as secure messaging system by adding a code to contact the Microsoft SQL Server database and this code is supported by a model for securing messages using several means, including a restructuring using vector classes, transforming to binary format, compressing using ZIP algorithm and Finally, encoding using standard Rijndael encryption algorithm. Also the access permissions to the messaging system has been controlled, so the authorized users just have the right to access; in addition to all the messages traffic has been registered into the log files of the system.

The system has been developed using C++ programming language, in addition to that a front-end has been developed using Delphi development tool in order to operate the system, and thereby it was tested under Windows XP operating system.

While testing the system there was numerous messages with different lengths are used, and therefore were to examine the impact of model Encryption - which is the most important part of the provided security model - at the length of messages and at the sending time of them in the case of existence and absence of the encryption model. The experiment found that the presence of the whole security model added a high complexity and confidentiality into the messages; in addition to that, the existence of encryption system within the security model doesn't have a big effect on the rate of the messages' sending time and its affection on the message length not more than 15 bytes for each one sent regardless of its length.