i

Integration Of TLS In An Internet/Intranet-based Application

[1]Lutz Vorwerk, [1]Ch. Meinel
[1] *Institut für Telematik e.V, Bahnhofstrasse 30-32, D-54292 Trier, Germany*

Keywords: telemedicine, radiology, multimedia, security, transport layer security, Java

Abstract: In telemedicine a growing need for security is obvious. There are several approaches to integrate security in standards like DICOM (Digital Imaging and Communication in Medicine) or HIPAA (Health Insurance Portability and Accountability Act). These definitions are independent from the kind of implementation or use of programming languages. Here an implementation will be described which integrates a secure transfer of data in an existing system by using the transport layer security (TLS) and Java.

1. INTRODUCTION

This paper describes how to integrate encryption in the communication process of a radiological department with local physicians. As a helping tool in the development serve the Java security extension[8] and the Java Cryptix extension[7]. An example is the workflow of the creation of reports in radiology. Let us have a closer look at this workflow.

2. CREATING REPORTS IN RADIOLOGY

The field of application radiology has different techniques. The choice of the appropriate method should depend on keeping the hassle for the patient with x-rays as low as possible. The approaches used today are improved

with regard to the emission of x-rays, but when using techniques like computer tomography (CT), which produce a series of images, one ought to think about how often this technique should be applied to a patient. The CT uses x-rays to produce cross-section images of the patient being examined. The technique to use magnetic fields instead of x-rays is called Magnet Resonance Tomography (MRT). This technique produces images from the resonance of hydrogen atoms evoked by magnetic fields. It is less dangerous with regard to x-ray burden, but expensive and dangerous if pieces of metal are contained in the body of the patient. If these pieces are not discovered in time and the technique is applied to the patient, these pieces will be influenced by the magnetic field and can be responsible for internal injuries. Another technique are ultrasonic waves to produce an image. This technique is appropriate for the pre-examinations or examinations which do not need high-resolution capture systems. The techniques described here can be combined with the use of a contrast medium which is also inconvenient for the patient. It is obvious that with respect to the patient, the techniques and the use of a contrast medium should only be applied if necessary. Redundancy can be prevented if the digital images of a patient are archived in an image archive, which is central or de-central. If a physician ordered an radiological examination, the order and the digital images are stored together in a database. If the patient changes his physician, the images are available for the other physician. This method saves a lot of costs.

3. SITUATION IN A RADIOLOGICAL DEPARTMENT

Assume the working conditions in a radiological department with single place systems without a connection to the internet: One important feature of an application used in this environment is that this application has an easy and fast operating user interface which is easy to understand. This is because findings have to be performed quick and under time pressure.

To realize a system which finds the approval of most physicians, it has to be developed in JAVA. This programming language allows to connect to the order database where the physician can deposit his or her order. The order is then processed by the radiological department, which orders, in turn, existing images of the patient and decides if there are available images and if these images are sufficient to answer the requested order of the physician. If yes, the radiologists perform the findings by using these images and send the

report to the physician. If not, they tell the patient to make an appointment to produce new images. This procedure serves the physical health of the patient.

There are some kinds of impediments. The law in Germany is very restrictive concerning the transfer of digital medical images and patient data and to do produce reports based on data received online.

Among the existing components of this system is a multimedia editor[4,5,9] which will be connected via sockets to a database. A socket is one end-point of a two-way communication link between two programs running on the network. These sockets provide a secure protocol mechanism capable to store the reports and orders done by physicians in a secure way.

The new system component presented here uses the javax - API (application programming interface) to implement a secure transfer between two locations[8]. One location has an applet (i. e. the multimedia-editor or a text editor) which uses the implemented TLS (transport layer security)[1] to communicate with a servlet which works on a database to store incoming requests and process them. Incoming requests are URLs that refer to objects to be processed. Those URLs contain keys to decrypt the data of the referenced object.

The first thing you have to consider here is the definition of the TLS protocol definition. After that, a transfer of the TLS protocol to the structure of the hierarchical ordered security classes form the javax-API is necessary. To perform the transfer we have restricted the used encryption algorithms to those which are actually considered as highly secure. The management of used algorithms will be performed by a database which stores the compiled Java classes. The database will be used by a Java Servlet. The communication between the database and the Servlet is handled in a secure way to prevent interior attacks. Information send to the database will be encrypted by PGP (Pretty Good Privacy). The Servlet governs a key table stored as an pgp-encrypted object in the database. During initialization the servlet asks the user for login and password in order to be capable to decrypt the key table. If the key table is available the servlet is able to read and store objects in the database. New keys will be created for new objects. We are using the Cryptix Java Security extension as a helping tool. It provides the majority of the encryption algorithms and is a valuable basis for our development. How to integrate a TLS connection in the system is shown in Figure 1.

Cryptix supports encryption algorithms to perform symmetric encryption, asymmetric encryption, block ciphers, and digital signatures. The following list contains the encryption algorithms we use to implement our TLS-Protocol.

3.2 Symmetric encryption

DES-EDE3 is chosen for symmetric encryption because it is considered as secure: This is 3-key Triple DES. The conformance of this algorithm used in Cryptix have to be checked, yet. The data, checked against SSLeay only and therefore needs further comparisons.

3.3 Asymmetric encryption

The asymmetric encryption method 'RSA' (Rivest, Shamir, Adleman) has been patented in the United States but not in Europe. Within Europe, differing methods are used[3,6]. Key encoding allows to generate RSA key pairs with a key length of less than 768 Bit. RSA will only be useful in the future if it is possible to generate RSA keys of a length of 1024 Bit or more.

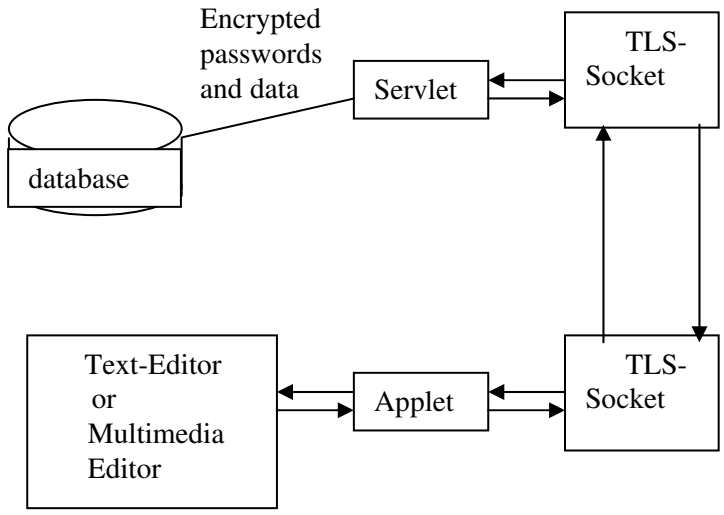

Figure 1. Integrating a TLS connection

3.4 Cipher Modes

CBC (cipher block chaining) is appropriate because it allows block ciphering for predictable connections[2]. There are other block ciphers but they are not convenient enough. They are either insecure and require additional treatment to guarantee security or they have features not needed for the implementation of TLS.

3.5 Signature Modes and Message Digests

As a hash function for signatures it is most common to use MD2 (MD = message digest) or MD5. We cannot recommend MD2, MD4 and MD5 though since they have been proven insecure. Appropriate hash functions are SHA-1(Secure Hash Algorithm 1) or RIPEMD-160 (RACE integrity primitives evaluation).

4. BREAKING DOWN THE TLS DEFINITION FOR OUR PURPOSES

First the Server sends the client the public key included in a certificate. Next, the client sends the server the public key (RSA key generation) included in a certificate as a response to a request of the server. The client and the server both verify the certificate. After positive verification the key can be used to exchange messages. First, the symmetric key algorithm is negotiated. A symmetric key, generated from 3DES, is exchanged between client and server. This key is generated from the server.

5. CONCLUSION

The system presented here is capable to use a TLS connection for the transfer of individual data which is regarded as confidential and therefore have to be encrypted. The transport layer security is chosen to guarantee high security and for testing out this protocol with the most popular and as secure regarded encryption algorithms. As a test environment serves the workflow of a radiological department and the multimedia editor. Influence on the choice of encryption algorithms used together with the TLS protocol is influenced by the Report of Teletrust[3] and experiences gained in a project that deals with the description assessment of a secure infra structure[3].

REFERENCES

1. Transport Layer Security Working Group, Tim Dierk Christopher Allen, 1997, The TLS-Protocol Version 1.0.
2. Alfred J.Menezes, Paul C. van Oorschot, Scott A. Vanstone, 1996, Handbook of Applied Cryptography.
3. TeleTrust Deutschland e.V., 1998, Kryptographische Verfahren im Gesundheits- und Sozialwesen.
4. Vorwerk, L., Meinel, Ch., 2000, A multimedia editor for making findings in radiology. In *CBMS (Computer based medical systems).*
5. von Gehlen, S.,Vorwerk, L.,Eichelberg, M., Jensch, P. , 1999, A multimedia editor for radiological reports based on the DICOM Supplement Structured Reporting. *In CARS `99, Computer Assisted Radiology and Surgery,*(H.U. Lemke, M.W. Vannier, K. Inamura and A.G. Farman), 438-442, Elsevier, Paris.
6. Vorwerk, L., Lüpken, B., Radke, C., Engel, T., Meinel, Ch., 2000, Bericht zur Bestandsaufnahme zur Sicherheitsinfrastruktur in DIZ. Institut für Telematik.
7. Systemics Ltd on behalf of the Cryptix Development Team, 1997, Cryptix is a trademark of Systemics Ltd., (http://cryptix.moremagic.com/).
8. Sun Microsystems, 2000, JavaTM Secure Socket Extension 1.0.1, Inc. 901 San Antonio Road, Palo Alto, California, 94303, U.S.A.
9. Lutz Vorwerk, Entwicklung eines Multimedia-Editors für die Befunderhebung in der Radiologie auf der Basis des DICOM Supplement Structured Reporting, (Master Thesis), Carl von Ossietzky Universität Oldenburg, November 1998.

---

[i] Medicom 2000, Kluwer Academic Plenum Publishers New York USA, p 143-149, Remagen Deutschland, 2000.