INSTITUTO SUPERIOR TÉCNICO

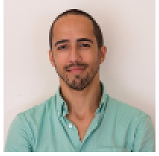
Medical Records Database

NETWORK AND COMPUTER SECURITY

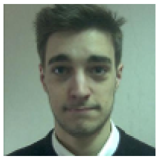
GROUP A05

António MONTEIRO apedrocruz@tecnico.ulisboa.pt 80990

Bernardo CORDEIRO bernardo.f.cordeiro@tecnico.ulisboa.pt 78778



Pedro LINDEZA pedro.lindeza@tecnico.ulisboa.pt 80831



December 7, 2017

1 Problem

With the rising of technology usage in hospitals, the sharing of patients' clinical data (such as blood tests results, X-Rays, etc.) between health care institutions has proven to be very important at providing doctors with precise information to make the best possible diagnostic. This data has to be publicly accessible, though data repositories in the Internet, by any institution, to allow patients to be attended anywhere.

To achieve this, it is essential that these communications are done in a way such that third parties cannot intercept any message content, so, it's necessary to guarantee confidentiality when fetching external clinical data. To ensure that only authorized personnel is allowed to read patient data, it is imperative to correctly implement an access control policy, so that only patient's doctors' are authorized to read sensitive clinical data.

2 Requirements

- confidentiality - no third parties between the data repository and the doctor are able to read patient records sent through the Internet;
- authentication - all patient records must be authenticated by doctor responsible for creating it;
- integrity - there has to be a way to attest if patient records were modified from its original form;
- access control policies - a doctor can only access its own patient data records, with patient's consent.

3 Solution

There is a central authority entity which is National Health Service (NHS). This entity certificates hospitals, which in their turn can certificate the doctors that work there. Patients are entities that are directly trusted by the NHS.

To meet the security goals we're committing to, the patient has to give his doctor authorization (in form of a digital signature) for him to fetch his clinical data from the NHS central data repository (that saves and manages all patient records). The doctor passes this authorization to the hospital where he works to obtain a signature whether the hospital authorizes this request or not. To ensure data integrity we must ensure that the hospital can only sign if the patient has allowed data access request or if it's a case provided for in the exceptions (e.g: life or death situation).

After having an authorized request from the hospital, the doctor fetches patient records NHS with the hospital permission (asserted by the digital signature).

This solution implements an implicit chain of trust:

- NHS→Hospital→Doctor
- NHS→Patient

The access control model RBAC is also implicitly presented in this solution, enforced with digital signatures.

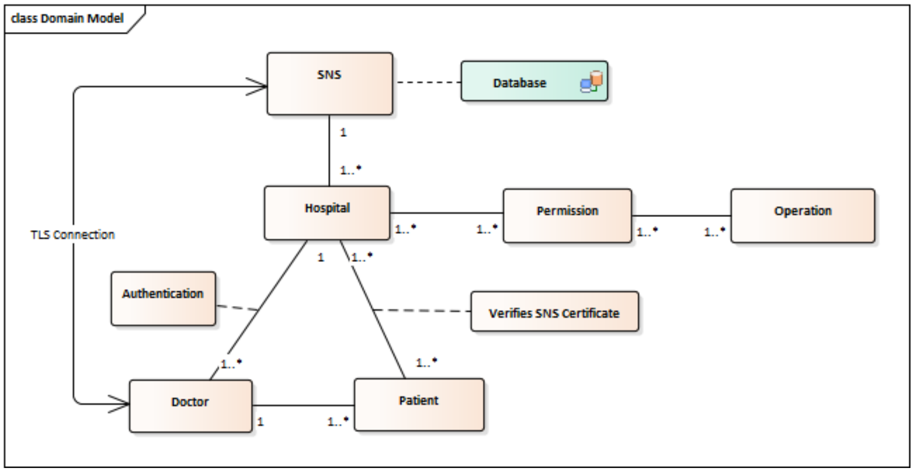


Fig. 1: UML Domain Model of Solution

3.1 Basic

In this solution we propose a system that match our practical goal. Keys distribution implementation.

3.2 Intermediate

This solution will take confidentiality into account, and so, all external communications will be confidential. TLS implementation.

3.3 Advanced

The final solution will also feature the fail-safe feature, in which patient records can be obtained in an emergency situation.

3.4 Key distribution

It is assumed that each entity enrolled in this system has already got its own private key, and a certificate signed by the responsible entity. Doctors' keys are trusted by the health care institution where they work, the health care institution's key is trusted by the national health service (which is self-signed). Patients' keys are trusted by the national heath service.

3.5 Tools Used

- Java Keytool - to generate certificate structure and a symmetric key so that the NHS can cipher records;
- Java RMI with TLS support - to provide confidential communication between external entities;

4 Results

The proposed solution provides confidential and authenticated communication between the Hospital and the NHS. The communication between the doctor/patient/hospital was not a main concern in this solution, as we assumed it to be a closed secure network.

Our solution fulfills the proposed requirements. Every entity participating in the National Heath System possesses an asymmetric RSA key pair, and a signed certificate. The NHS itself is the top of the hierarchy, the root certificate authority, and delegates the power to hospitals to issue their own certificates, which means hospitals are also certificate authorities, and certificates the patients. Hospitals sign their doctors certificates.

The key distribution was not a main concern when designing this system. We developed a generic solution that could be implemented in a real case scenario with the usage of certificates in smart cards.

The storage of patient records by the central repository was not foreseen in terms of security. We decided to encrypt all records in the NHS, which was done using 128-bit AES-CBC, generating different IVs for each record.

Communication between the NHS and the Hospital is done using TLS 1.2, with both server and client authentication.

To ensure access control policies, NHS only accepts requests authorized by the hospital, thus meaning that control access is done by the hospital (which the NHS trusts).

Integrity and authentication were a continuous focus in this implementation. Patient records are signed by the doctors who wrote it. The authorization that is sent from the Hospital to the NHS is also signed.

5 Evaluation

Having met all proposed requirements, this system serves its purpose, which was to create a central data repository to store patient records in the Internet. Having records signed by doctors means they can't be forged, by using TLS to communicate between parties over the Internet implicates that no third-party will have access to the patient record, and finally by encrypting patient records in the NHS, means that the chosen data store may be compromised, but will never leak patient information.

6 Conclusion

Our solution aims to provide a secure implementation of a medical records database. It assures that records are secure, communication between entities is ciphered and protected and that the database is secure against outside attacks and known vulnerabilities.

References

[1] Oracle, Java SE Documentation *Using Java RMI with SSL:* https://docs.oracle.com/javase/8/docs/technotes/guides/rmi/socketfactory/SSLInfo.html

[2] William Grosso *"Java RMI"*, O'Reilly - First Edition October 2001.