Tạp chí Khoa học ĐHQGHN, Khoa học Tự nhiên và Công nghệ 24 (2008) 63-69

Thử nghiệm xây dựng hệ thống cung cấp và quản lý chứng chỉ số

Trịnh Nhật Tiến[1,*], Trương Thị Thu Hiền[1], Vũ Văn Triệu[2], Đào Ngọc Phong[3]

[1]*Khoa Công nghệ Thông tin, Trường Đại học Công nghệ, ĐHQGHN, 144 Xuân Thủy, Hà Nội, Việt Nam*
[2]*Trung tâm dịch vụ giá trị gia tăng, 4 Láng Hạ, Hà Nội, Việt Nam*
[3]*Sở thông tin Viễn thông Hà Nội, 185 Giảng Võ, Hà Nội, Việt Nam*

Nhận ngày 2 tháng 4 năm 2007

Tóm tắt. Nhu cầu trao đổi thông tin trên mạng máy tính ngày một gia tăng kéo theo những yêu cầu cấp thiết về bảo đảm an toàn truyền tin trên mạng. Ví dụ hai người từ xa không nhìn thấy nhau, không nghe được giọng nói của nhau, nhưng vẫn có thể trao đổi thông tin trên mạng máy tính công khai (thoả thuận, thanh toán hợp đồng, ký kết hợp tác, thi vấn đáp, …) mà yên tâm rằng: họ đang làm việc với đúng đối tác của mình, người thứ ba "khó" thể biết họ đang làm việc gì.
Một trong những cách để giải bài toán trên là xây dựng Hạ tầng cơ sở mật mã khoá công khai (PKI – Public Key InfraStucture). Trên đó có Hệ thống cung cấp và quản lý chứng chỉ số.
Báo cáo trình bày thử nghiệm xây dựng hệ thống trên, trong đó sử dụng công nghệ SSL và IAIK. Hệ thống đã được dùng tại một số cơ quan, và đã được xác nhận có hiệu quả.

Từ khóa: PKI, CA, SSL, IAIK.

1. Hệ thống cung cấp và quản lý chứng chỉ số

1.1. Nhu cầu về Hệ thống cung cấp và quản lý chứng chỉ số

1.1.1. Khái niệm Chứng chỉ số

Luật giao dịch điện tử (trong đó có "chữ ký số") ở Việt nam đã có hiệu lực từ 3/2006. Theo luật này, chúng ta có thể "ký" từ xa qua mạng máy tính, không phải gặp nhau.

Thông thường để xác thực chủ nhân của tài liệu người ta dùng "chữ ký tay" đánh dấu phía dưới tài liệu đó. "Chữ ký tay" của người ta được công khai cho mọi người biết để kiểm tra.

"Chữ ký số" không phải là một dấu hiệu như "chữ ký tay". Đó là một bản mã hoá tài liệu. Để kiểm tra "chữ ký số", chủ nhân của nó phải thông báo "khoá công khai" cho mọi người biết."Khoá công khai" còn dùng để lập mã bằng hệ mã hóa khóa công khai và nhiều ứng dụng khác.

Nhưng để bảo đảm tính pháp lý của "khoá công khai", phải có cơ quan thẩm quyền chứng thực chìa khoá này. Hiện nay cơ quan như vậy gọi là trung tâm cung cấp và quản lý "chứng chỉ số"

(CA: Certification Authority) [1].

* Corresponding author. Tel.: 84-4-7547064
E-mail: tientn@vnu.edu.vn

"Chứng chỉ số" (CCS) là giấy chứng nhận chủ nhân của "khóa công khai". CCS giống như chứng minh thư, được dùng khi giao dịch trên mạng, nhằm đảm bảo nhận diện đúng đối tượng giao dịch và góp phần bảo đảm an toàn cho các nội dung giao dịch. Rõ ràng CCS là thành phần rất quan trọng trong "giao dịch điện tử".

1.1.2. Hệ thống cung cấp và quản lý chứng chỉ số

Để bảo đảm cho trung tâm CA hoạt động nhanh và thuận lợi, ta phải xây dựng hệ thống cung cấp và quản lý chứng chỉ số. Đó là phần mềm hỗ trợ CA thực hiện các nhiệm vụ sau:

♦ Xét duyệt đề nghị và cấp "chứng chỉ số".

♦ Quản lý các "chứng chỉ số" đã được CA cấp, các "chứng chỉ số" còn hiệu lực pháp lý, các "chứng chỉ số" hết hiệu lực pháp lý (các "chứng chỉ số" bị thu hồi).

♦ Cung cấp bằng chứng pháp lý khi xảy ra tranh chấp trong "giao dịch điện tử".

Hệ thống cung cấp và quản lý chứng chỉ số là bộ phận quan trọng của Hạ tầng cơ sở mật mã khoá công khai (PKI), trên đó người ta có thể thực hiện được các giao dịch điện tử an toàn [2,3].

1.2. Các thành phần của Hệ thống ung cấp và quản lý chứng chỉ số

1.2.1. Các yêu cầu đối với Hệ thống cung cấp và quản lý chứng chỉ số

♦ Đối với khách hàng, hệ thống thực hiện được các yêu cầu: cấp mới, gia hạn, thay thế, thu hồi, nhận thông tin về chứng chỉ số, tìm kiếm chứng chỉ số, gửi chứng chỉ số cho khách hàng.

♦ Hệ thống có thể hoạt động trên các môi trường thông dụng (UNIX, Windows, OS/2…), và có giao tiếp đồ họa (GUI).

♦ Hệ thống phải mở rộng thêm được các chức năng mới, hay kết nối được với hệ thống khác.

1.2.2. Các thành phần của Hệ thống cung cấp và quản lý chứng chỉ số

♦ Bộ phận cung cấp và quản lý chứng chỉ số (Certification Authority: CA).

♦ Bộ phận xác thực khách hàng xin cấp chứng chỉ số (Registration Authority: RA).

RA duyệt yêu cầu xin cấp chứng chỉ số. Sau đó gửi yêu cầu sang bộ phận CA.

CA sẽ có những đáp ứng cụ thể như: cấp mới, gia hạn, thay thế, thu hồi chứng chỉ số.

CA có quyền tạo danh sách thu hồi (CRL), tạo chuỗi chứng thực và quản trị người dùng

(phân quyền, tạo mới, sửa, xoá).

2. Công nghệ SSL

2.1. Giới thiệu công nghệ SSL (Secure Socket Layer)

SSL của Nestcape (1994) là công nghệ phổ biến bảo đảm an toàn thông tin trên Internet. SSL có thể bảo vệ thông tin bằng mật mã, kiểm tra sự toàn vẹn thông tin, chứng thực,…

SSL có độ an toàn cao và "trong suốt" đối với tầng ứng dụng. Hiện nay có khoảng hơn 300.000 địa chỉ Internet chấp nhận giao dịch điện tử, thì gần như tất cả các địa chỉ này đều sử dụng SSL, đồng thời hầu hết các webserver dùng trên Internet đều hỗ trợ SSL.

Giao thức SSL đóng vai trò như một tầng trong mô hình TCP/IP mở rộng. SSL được đặt giữa tầng ứng dụng (Application Layer) và tầng giao vận (Transport Layer). Các giao thức ứng dụng (Application Protocols) dùng SSL thường được viết thêm hậu tố "-s:" (ví dụ: https, ftps,...).

SSL dùng hệ mã hoá khoá đối xứng để mã hoá dữ liệu trước khi truyền tin. Để thoả thuận khoá đối xứng giữa hai bên truyền tin, SSL phải

thực hiện quá trình "bắt tay". Có thể chia hoạt động của giao thức SSL thành hai tầng:

♦ Tầng 1: SSL Handshake Protocol và các giao thức con SSL khác (SSL subprotocols), cho phép Client và Server xác thực lẫn nhau, thoả thuận hệ mã hoá và khoá bí mật trước khi giao dịch.

♦ Tầng 2: SSL Record Protocol (RP) được đặt trên các tầng truyền thông tin cậy (như TCP). SSL RP được dùng để đóng gói các giao thức ở tầng cao hơn (gói dữ liệu trước khi truyền đi). Trước khi mã hoá để truyền đi, gói tin có thể được nén để tiết kiệm băng thông đường truyền.

Một số khả năng của SSL:

♦ SSL là giao thức cho phép thiết lập kênh truyền tin an toàn, tin cậy, có xác thực.

♦ SSL sử dụng chứng chỉ số hoặc giao thức thoả thuận khoá bí mật chung.

♦ SSL sử dụng các hệ mã hoá đối xứng để đảm bảo yêu cầu về tốc độ. Thông tin trước khi truyền đi được nén nhằm tiết kiệm tài nguyên đường truyền.

♦ SSL được thiết kế độc lập với các chương trình ứng dụng. Nói cách khác, SSL được dùng để thiết lập kênh truyền tin an toàn và "trong suốt" đối với các ứng dụng trên nó. Ví dụ trong khi truyền tin giữa client và server, dữ liệu được mã hoá, nhưng người dùng cuối không có cảm nhận về sự chuyển dạng dữ liệu. Do "tính trong suốt" này, mà gần như mọi giao thức hoạt động trên TCP có thể chạy được trên SSL chỉ với một chút sửa đổi.

♦ SSL có khả năng tận dụng trạng thái phiên đã thiết lập, để tạo kênh truyền mới được nhanh chóng, giảm thời gian "bắt tay".

♦ SSL thích hợp cho các ứng dụng viết bằng ngôn ngữ C và C++.

2.2. Tạo lập kết nối (bằng Handshake Protocol)

2.2.1. Giao thức "bắt tay" (Handshake Protocol)

SSL Handshake Protocol có nhiệm vụ tạo kết nối giữa Client và Server. Nó thiết lập các thông số của một phiên làm việc SSL. Cụ thể:

♦ Client gửi thông điệp Client_hello tới Server, nội dung gồm có: chứng chỉ số của Client, hệ mã hoá, hàm băm, thuật toán nén, chuỗi byte ngẫu nhiên của Client dùng để tạo khoá chung, thời gian gửi thông điệp. Tất cả được sắp xếp theo thứ tự ưu tiên sử dụng của Client.

♦ Server gửi thông điệp Server_hello tới Client, nội dung gồm có: chứng chỉ số của Server, hệ mã hoá, thuật toán nén, định danh phiên làm việc, chuỗi byte ngẫu nhiên của Server dùng để tạo khoá chung, thời gian gửi thông điệp.

Nếu một bên nào đó thiếu chứng chỉ số hoặc khoá công khai trong chứng chỉ số chỉ có chức năng kiểm tra chữ ký, thì gửi thông điệp yêu cầu chứng chỉ số của bên kia. Mỗi bên sẽ gửi cho đối tác chứng chỉ số hay thông điệp cảnh báo nếu bên kia yêu cầu.

♦ Server thực hiện các việc sau:

Gửi thông điệp Server_key_exchange để trao đổi khoá với Client. Gửi thông điệp Sever_done để báo rằng Server kết thúc phần trao đổi khoá.

♦ Client thực hiện các việc sau:

- Gửi thông điệp Client_key_exchange để trao đổi khoá với Server. Client cần tạo khoá mật chung với Server, thì phải dùng thông tin trong Server_key_exchange.

- Đóng gói thông tin tạo khoá chung vào Client_key_exchange và gửi cho Server.

- Gửi thông điệp finished, thông báo kết thúc bắt tay.

- Server và Client gửi Change_cipher_spec

để báo tin cho bên kia biết các gói tin sau này sẽ được bảo vệ (mã hoá) bằng các khoá chung đã thoả thuận.

2.2.2. Trạng thái phiên và giao thức "bắt tay lại" (Rehandshake)

Khi Client và Server "bắt tay" xong, thì một phiên làm việc mới được thiết lập, kênh truyền bảo mật được tạo thành, tầng SSL Record protocol đi vào hoạt động. Hãy xem kỹ thuật lưu trữ thông số phiên làm việc, và cách "bắt tay lại" (*Rehandshake*) để tạo phiên làm việc mới.

a) Trạng thái phiên

Để quá trình "bắt tay" mới thực hiện nhanh chóng, ngay sau lần "bắt tay" đầu tiên, SSL thiết lập và lưu trữ trạng thái của phiên hoạt động (Session state) và trạng thái kết nối, để có thể dùng lại tối đa các thông số đã thiết lập.

Thông tin về trạng thái phiên làm việc:

- Session identifier: chuỗi các byte do Server tạo ra, để định danh duy nhất phiên làm việc.

- Peer certificate: chứng chỉ số đối tác truyền tin.

- Compression method: Phương pháp nén dữ liệu trước khi mã hoá và truyền đi.

- Cipher spec: hệ mã hoá và hàm băm quy ước dùng giữa Client và Server.

- Master secret: 48 byte thông tin mật dùng chung giữa Client và Server.

- Is resumable: cờ trạng thái, cho thiết lập hay không phiên làm việc mới từ phiên hiện thời.

Thông tin về trạng thái kết nối:

- Server and Client random: các chuỗi byte ngẫu nhiên trong Client_hello và Server hello.

- Server write MAC secret: chuỗi byte bí mật dùng để tính MAC của thông điệp gửi đi.

- Server write key: khoá để mã hoá dữ liệu trước khi truyền của Server.

- Client write key: khoá để mã hoá dữ liệu trước khi truyền của Client.

- Initialization vector: vector khởi tạo dùng trong thuật toán mã hoá khối.

- Sequence number: mỗi bên truyền tin có một số, để đếm số thông điệp truyền và nhận tin.

b) Giao thức Rehandshake

Khi Client và Server cần khởi tạo lại phiên làm việc trước đó, hoặc nhân đôi phiên làm việc hiện tại (thay vì phải tạo phiên làm việc mới), giao thức Rehandshake thực hiện như sau:

- Client gửi thông điệp Client_hello của phiên làm việc trước đó. Server tìm Session identifier tương ứng (trong kho lưu trữ).

- Nếu tìm được Session identifier, thì Server gửi cho Client: Server_hello với Session identifier.

Vào thời điểm này, Client và Server phải gửi cho nhau thông điệp Change_cipher_spec.

Trực tiếp xử lý thông điệp finished, trạng thái phiên và trạng thái liên kết tương ứng với session identifier đó sẽ được sử dụng lại. Việc khởi tạo lại được thực hiện xong và dữ liệu tại tầng ứng dụng có thể tiếp tục được trao đổi.

- Nếu không tìm được Session identifier, Server tạo ra Session identifier mới. Server và Client thực hiện lại đầy đủ việc "bắt tay" từ đầu.

Sau khi trạng thái phiên và trạng thái liên kết được thiết lập, dữ liệu được đóng gói, nén, mã hoá và truyền đi qua SSL Record protocol. Công việc được chia 3 giai đoạn:

- SSL Plaintext: dữ liệu được phân thành các khối có kích thước nhỏ hơn 2^{14} và thêm các thông tin như loại dữ liệu gửi đi: change_cipher_spec, alert, handshake, application data. Các dữ liệu điều khiển được ưu tiên gửi đi trước dữ liệu ứng dụng (application data).

- SSL Compressed: dữ liệu sẽ được nén, theo phương pháp quy ước trong trạng thái kết nối.

- SSL Ciphertext: Tính MAC thêm vào gói tin, mã hoá toàn bộ gói tin đóng gói, gửi đi.

2.3. Đóng gói và truyền dữ liệu (bằng Record Protocol)

Giao thức "bản ghi" (Record Protocol-RP) dùng để đóng gói dữ liệu trước khi truyền đi: nén dữ liệu để tiết kiệm băng thông và thời gian truyền tin, mã hoá gói tin nén để bảo mật.

Cụ thể giao thức RP thực hiện các công việc:

- Phân mảnh (Fragmentation): Thông điệp tầng trên được phân nhỏ thành các gói $\leq 2^{14}$ byte.

- Nén tin (Compresion): Nén từng gói tin trên, nhận được gói tin nén $\leq$ 1024 byte.

- Mã hoá tin nén (Encrypt): Mã hoá từng gói tin nén trên, để bảo mật dữ liệu.

- Tính mã xác thực (MAC - Message Authentication Code): Tính MAC từng gói tin nén trên, để bảo toàn dữ liệu.

3. Công nghệ IAIK

3.1. Giới thiệu công nghệ IAIK

Giống như SSL, công nghệ IAIK (Institute for Applied Information Processing and Communication) cũng dùng để xây dựng hệ thống cung cấp và quản lý chứng chỉ số. Nhưng SSL thì thích hợp cho các ứng dụng viết bằng ngôn ngữ C và C ++, trong khi IAIK lại thích hợp cho các ứng dụng viết bằng ngôn ngữ Java [4].

Trong thử nghiệm chúng tôi sử dụng IAIK-JCE, ở đây JCE (The Java Cryptography Extension) là kế thừa từ JCA (Java Cryptography Architecture-Kiến trúc về mật mã của Java).

Một số đặc điểm của IAIK- JCE:

- IAIK-JCE gồm 3 gói chính:

package javax.crypto, package javax.crypto.spec, package javax.crypto.interfaces.

- IAIK-JCE mở rộng các công cụ an ninh, nhưng vẫn tuân theo các chuẩn về kiến trúc của JCA.

- IAIK-JCE hỗ trợ hầu hết các cấu trúc cơ bản trong ASN.1 (Abstract Syntax Notation One):

BOOLEAN, INTEGER, BITSTRING, OCTETSTRING, NULL, OBJECTIDENTIFIER, ENUMERATED, SEQUENCE, SET, SEQUENCE OF, SET OF, UTCTime, GeneralizedTime, hầu hết các kiểu String.

3.2. Công nghệ IAIK hỗ trợ các chuẩn mật mã

+ IAIK là một tiện ích bảo mật (Crypto Toolkit)

(được viết bằng ngôn ngữ Java):

+ Cung cấp một tập các API (Application Programming Interface) cho lập trình Crypto.

+ Hỗ trợ các lĩnh vực: Hạ tầng mật mã khoá công khai (Public Key Infrastructure), các loại an toàn: Communication, Messaging, XML, Mobile.

IAIK-JCE hỗ trợ các chuẩn PKCS:

(Public-Key Cryptography Standards):

+PKCS#1: RSA Encryption Standard (Chuẩn mã hoá RSA).

+PKCS#3: Diffie Hellman Key Agreement Standard (Chuẩn thoả thuận khoá).

+PKCS#5: Password-Based Encryprion Standard

+PKCS#7: Cryptographic Message SyntaxStandard

+PKCS#8: Private-Key Information Syntax Standard

+PKCS#9: Selected Attribute Types

+PKCS#10: Certification Request Syntax Standard

+PKCS#12: Personal Information Exchange Syntax Standard

Hỗ trợ các kiểu sinh số ngẫu nhiên theo chuẩn ANSI X9.17, FIPS PUB 186-2.

IAIK hỗ trợ các chuẩn về chứng chỉ số (CCS):

+ Phân chia thành JCA certificate / crl API.

+ Kế thừa từ JCA certificate/crl API, tạo mới chứng chỉ số.

+ Hỗ trợ X.509 public key certificate (chứng chỉ khoá công khai).

+ Hỗ trợ X.509 certificate revocation list (CRLs-Danh sách chứng chỉ số bị thu hồi).

+ Hỗ trợ X.509 qualified certificate, X.509 attribute certificate.

+ Cài đặt X.509 certificate và crl extension, private Netscape cert extenion.

+ Cài đặt qualified, attribute, and OCSP certificate extension.

+ Cài đặt OCSP (Online certificate status protocol).

+ Giao thức kiểm tra trạng thái CCS trực tuyến.

+ Hỗ trợ Client và Server: tạo, ký, phân tích, kiểm tra các OCSP request và OCSP response.

+ Cài đặt các OCSP Client và Server mở rộng.

+ Hỗ trợ OCSP thông qua giao thức HTTP, gồm các tiện ích để tạo các OCSP response từ các CRL và các OCSP response chứng thực chứng chỉ.

4. Kết quả thử nghiệm ứng dụng

Luật giao dịch điện tử ở Việt Nam có hiệu lực từ 3/2006, trước đó 2 năm chúng tôi đã thử nghiệm xây dựng hệ thống cung cấp và quản lý chứng chỉ số, nhằm đón chờ cơ hội sử dụng.

Hệ thống đã được dùng thật trong hệ thống thanh toán tiền lương tại một công ty và một ngân hàng, được dùng trong hệ thống chuyển khoản trực tuyến tại một ngân hàng khác. Đó cũng là một phần kết quả của đề tài nghiên cứu khoa học - công nghệ tại Sở Khoa học- Công nghệ Hà Nội.

Chúng tôi đã nghiên cứu ưu nhược điểm của từng công nghệ, cụ thể:

IAIK có nhược điểm là: do viết bằng Java, nên các chương trình mã hóa và giải mã là chậm.

Nhưng nó lại có tất cả các ưu điểm của ngôn ngữ Java. Đối với việc phát triển các ứng dụng liên quan đến bảo mật bằng ngôn ngữ Java, thì IAIK là lựa chọn hàng đầu.

SSL có nhược điểm trong truyền thông là tốc độ chậm hơn so với IAIK. Nhưng nó lại thích hợp cho các ứng dụng viết bằng C và C++.

Hiện nay trong một số ứng dụng bảo mật truyền tin tại nước ta, người ta dùng một trong hai công nghệ: SSL hay IAIK, nhưng chúng tôi dùng cả hai để xây dựng hệ thống cung cấp và quản lý chứng chỉ số, vì thế đã tận dụng được mặt mạnh của cả hai. Cụ thể là:

+ Sử dụng IAIK trong việc tạo chứng chỉ số và các chương trình mã hóa.

+ Sử dụng công nghệ SSL trong việc bảo mật WebServer và xác thực người dùng.

Tài liệu tham khảo

[1] John Wiley – PKI Security Solutions for the Enterprise, 2003.

[2] IBM – *Deploying a Public Key Infrastructure*, 2000.

[3] NIST – Introduction to the Federal PKI Infrastructure, 2001.

[4] TTTP://JCE.IAIK.TUGRAZ.AT.

An experiment of building a system providing and administrating digital certificates

Trinh Nhat Tien[1], Truong Thi Thu Hien[1], Vu Van Trieu[2], Dao Ngoc Phong[3]

[1] *Faculty of Information Technology, College of Technology, VNU, 144 Xuan Thuy, Hanoi, Vietnam.*

[2] *Value Added Service Center, 4 Lang Ha, Hanoi, Vietnam*

[3] *Hanoi Department of Information and Telecomunication, 185 Giang Vo, Hanoi, Vietnam*

The increasing demand of communicating via Internet has resulted in the imperative need for information security on the Internet. For instance, two people of long distance cannot see each other or hear other's voice, yet, can exchange information via Internet publicly (negotiating, signing contract, taking oral test, etc) feeling self-assured that they are working with their true partners and the third person can hardly know what they are doing.

One of the methods to solve the above problem is building Public Key Infrastructure, including the System providing and administrating digital certificates. The report demonstrates the experiment of building the above system, involving the use of SSL and IAIK technology. The system has been applied in several offices and recognized to be effective.