|  |
| --- |
| BAN CƠ YẾU CHÍNH PHỦ  **HỌC VIỆN KỸ THUẬT MẬT MÃ**  ¯¯¯¯¯¯¯¯¯¯¯¯¯¯¯¯ |
| ĐỒ ÁN TỐT NGHIỆP  **XÂY DỰNG PHẦN MỀM KÝ SỐ, MẬT MÃ CMS, JSON VÀ XML TRÊN WINDOWS** |
| Ngành: An toàn thông tin  Mã số: 7.48.02.02  *Sinh viên thực hiện*:  **Võ Lê Huy – AT131315**  Lớp: AT13P  *Cán bộ hướng dẫn*:  **Tiến sĩ Lê Quang Huy**  Cục chứng thực số và Bảo mật thông tin |
| Thành phố Hồ Chí Minh, 2021 |
| BAN CƠ YẾU CHÍNH PHỦ  **HỌC VIỆN KỸ THUẬT MẬT MÃ**  ¯¯¯¯¯¯¯¯¯¯¯¯¯¯¯¯ |
| ĐỒ ÁN TỐT NGHIỆP  **XÂY DỰNG PHẦN MỀM KÝ SỐ, MẬT MÃ CMS, JSON VÀ XML TRÊN WINDOWS** |
| Ngành: An toàn thông tin  Mã số: 7.48.02.02  *Sinh viên thực hiện*:  **Võ Lê Huy – AT131315**  Lớp: AT13P  *Cán bộ hướng dẫn*:  **Tiến sĩ Lê Quang Huy**  Cục chứng thực số và Bảo mật thông tin |
| Thành phố Hồ Chí Minh, 2021 |

**Mục lục**

[Chương 1: Hạ tầng mật mã khóa công khai và chữ ký số 5](#_Toc88414618)

[1.1. Hạ tầng mật mã khóa công khai 5](#_Toc88414619)

[1.1.1. Khái niệm 5](#_Toc88414620)

[1.1.2. Sự cần thiết của PKI 5](#_Toc88414621)

[1.1.3. Các thành phần của hạ tầng khoá công khai 6](#_Toc88414622)

[1.1.4. Chứng thư số 7](#_Toc88414623)

[1.1.5. Danh sách thu hồi chứng thư số 17](#_Toc88414624)

[1.1.7. Mô hình hoạt động của PKI 23](#_Toc88414625)

[1.1.8. Luật pháp và chính sách về PKI 25](#_Toc88414626)

[1.2. Chữ ký số 26](#_Toc88414627)

[1.2.1. Khái niệm 26](#_Toc88414628)

[1.2.2. Đặc trưng 27](#_Toc88414629)

[1.2.3. Quy trình tạo và kiểm tra chữ ký số 27](#_Toc88414630)

[1.2.4. Các thuật toán chữ ký số 29](#_Toc88414631)

**LỜI CẢM ƠN**

Lời nói đầu tiên, em xin gửi lời cảm ơn đến quý thầy cô Học Viện Kỹ Thuật Mật Mã cũng như quý thầy cô khoa an toàn thông tin đã truyền đạt kiến thức, kinh nghiệm quý báu, quan tâm, hỗ trợ em trong suốt thời gian học tại trường.

Đặc biệt, em xin chân thành cảm ơn Tiến sĩ Lê Quang Huy, người đã hướng dẫn, tạo điều kiện thuận lợi cho em thực hiện khóa luận này.

Mặc dù, em đã cố gắng hoàn thành khóa luận nhưng do hạn chế về thời gian và kiến thức nên không tránh khỏi những sai sót. Em kính mong nhận được sự thông cảm và ý kiến đóng góp từ quý thầy cô để em có thể hoàn thiện và phát triển khóa luận này.

Em xin chân thành cảm ơn.

Thành phố Hồ Chí Minh, 2021

# Chương 1: Hạ tầng mật mã khóa công khai và chữ ký số

## 1.1. Hạ tầng mật mã khóa công khai

### 1.1.1. Khái niệm

Tên tiếng Anh là Public Key Infrastructure, viết tắt PKI là một cơ chế để cho một bên thứ 3 (thường là nhà cung cấp chứng thực số) cung cấp và xác thực định danh các bên tham gia vào quá trình trao đổi thông tin. Cơ chế này cũng cho phép gán cho mỗi người sử dụng trong hệ thống một cặp khóa công khai/khóa bí mật. Các quá trình này thường được thực hiện bởi một phần mềm đặt tại trung tâm và các phần mềm phối hợp khác tại các địa điểm của người dùng. Khóa công khai thường được phân phối trong chứng thực khóa công khai.

Khái niệm hạ tầng khóa công khai (PKI) thường được dùng để chỉ toàn bộ hệ thống bao gồm nhà cung cấp chứng thực số (CA) cùng các cơ chế liên quan đồng thời với toàn bộ việc sử dụng các thuật toán mật mã hóa khóa công khai trong trao đổi thông tin. Tuy nhiên phần sau được bao gồm không hoàn toàn chính xác bởi vì các cơ chế trong PKI không nhất thiết sử dụng các thuật toán mã hóa khóa công khai.

Nhu cầu mã hóa, định danh, xác thực thông tin đã có từ rất lâu đời, những cách thủ công lạc hậu đã triển khai nó được ứng dụng ví dụ ở con người thời trung cổ được dánh dấu để trên tráng để phân biệt phạm nhân (người bị tước quyền công dân) và người dân bình thương hoặc các trường hợp công văn được lưu truyền,…PKI là công nghệ đằng sau chứng chỉ số. Chứng chỉ kỹ thuật số đáp ứng mục đích tương tự như giấy phép lái xe hoặc hộ chiếu - đó là một phần nhận dạng để chứng minh danh tính và cung cấp một số khoản hỗ trợ nhất định. Chứng chỉ kỹ thuật số cho phép chủ sở hữu của nó mã hóa, ký và xác thực. Theo đó, PKI là công nghệ cho phép bạn mã hóa dữ liệu, ký điện tử vào tài liệu và xác thực bản thân bằng các chứng chỉ.

PKI là khuôn khổ cơ bản cho toàn bộ công nghệ; nó không phải là một thực thể vật lý duy nhất. PKI đóng gói các khác nhau tạo nên công nghệ, bao gồm phần cứng, phần mềm, con người, chính sách và thủ tục cần thiết để tạo, quản lý, lưu trữ, phân phối và thu hồi chứng chỉ kỹ thuật số. Một phần quan trọng của công nghệ PKI là CA, là cơ quan cấp giấy chứng nhận. CA là thực thể cấp chứng chỉ kỹ thuật số.

### 1.1.2. Sự cần thiết của PKI

Mã hoá: giữ bí mật thông tin và chỉ có người có khoá bí mật mới giải mã được.

Tạo chữ ký số: cho phép kiểm tra một văn bản có phải đã được tạo với một khoá bí mật nào đó hay không.

Thoả thuận khoá: cho phép thiết lập khoá dùng để trao đổi thông tin bảo mật giữa 2 bên.

PKI là một phần quan trọng của xương sống chiến lược CNTT. PKI rất quan trọng vì công nghệ dựa trên chứng chỉ giúp các tổ chức thiết lập chữ ký, mã hóa và danh tính đáng tin cậy giữa con người, hệ thống và mọi thứ.

Với việc các mô hình kinh doanh đang phát triển ngày càng phụ thuộc nhiều hơn vào các giao dịch điện tử và tài liệu kỹ thuật số, và với nhiều thiết bị nhận biết Internet hơn được kết nối với mạng công ty, vai trò của cơ sở hạ tầng khóa công khai không còn giới hạn ở các hệ thống biệt lập như email an toàn, thẻ thông minh để truy cập vật lý hoặc lưu lượng truy cập web được mã hóa. PKI ngày nay được kỳ vọng sẽ hỗ trợ số lượng lớn hơn các ứng dụng, người dùng và thiết bị trên các hệ sinh thái phức tạp. Và với các quy định nghiêm ngặt hơn về bảo mật dữ liệu của chính phủ và ngành, các hệ điều hành chính thống và các ứng dụng kinh doanh đang trở nên phụ thuộc hơn bao giờ hết vào một tổ chức PKI để đảm bảo sự tin cậy.

### 1.1.3. Các thành phần của hạ tầng khoá công khai

Thông thường về mặt tổng quan PKI bao gồm phần mềm máy khách (client), phần mềm máy chủ (server), phần cứng (như thẻ thông minh), … các quy trình hoạt động liên quan.

#### 1.1.3.1. Chứng chỉ khóa công khai

Mục tiêu của việc trao đổi khoá bất đối xứng là phát một cách an toàn khoá công khai từ người gửi (mã hoá) đến người nhận (giải mã). PKI hỗ trợ tạo điều kiện cho việc trao đổi khoá an toàn để đảm bảo xác thực các bên trao đổi với nhau.

Public key Certificate được phát bởi Certificate Authority (CA). Để CA phát public key certificate cho đáp ứng mật mã đầu cuối thì đầu cuối đầu tiên phải đăng ký với CA. Quá trình đăng ký gồm: sự đăng ký, sự kích hoạt, và sự chứng nhận của mật mã đầu cuối với PKI (CAs và RAs). Quá trình đăng ký như sau:

1. Mật mã đầu cuối đăng ký với CA hoặc RA. Trong quá trình đăng ký, mật mã đầu cuối đưa ra cách nhận biết đến CA. CA sẽ xác thực đầu cuối, phát public key đến đầu cuối .
2. Các đầu cuối bắt đầu khởi tạo phase bằng cách tạo ra một public/private keypair và public key của keypair được chuyến đến CA.
3. CA viết mật hiệu lên public key certificate cùng với private key để tạo một public key certificate cho mật mã đầu cuối.
4. Lúc này các mật mã đầu cuối có thể yêu cầu public key certificate từ mật mã đầu cuối khác. Chúng có thể sử dụng CAs public key để giải mã public key certificate để thu được khoá thích hợp

#### 1.1.3.2. Kho lưu trữ chứng chỉ

Kho lưu trữ để có thể truy vấn, thao tác khi cần.

#### 1.1.3.3. Thu hồi chứng chỉ

Thu hồi chứng thư số là làm mất hiệu lực của chứng thư số một cách vĩnh viễn từ một thời điểm xác định.

#### 1.1.3.4. Sao lưu và khôi phục khóa

Lưu trữ khóa là một dịch vụ được cung cấp bởi nhiều tổ chức CA. Dùng để tránh trường hợp không giải mã được dữ liệu khi bị mất khóa. Để lưu trữ khóa, người dùng phải gửi khóa bí mật tới nơi lưu trữ. Bởi vì các yêu cầu lưu trữ hay khôi phục khóa đều phải được xác minh nên người sử dụng không thể thao tác trực tiếp đến nơi lưu trữ mà phải thông qua CA phát hành chứng nhận đó.

### 1.1.4. Chứng thư số

#### 1.1.4.1. Khái niệm

Chứng thư số là một dạng chứng thư điện tử được cấp bởi tổ chức cung cấp dịch vụ chứng thực chữ ký số. Chứng thư số có thể coi như giấy chứng minh nhân nhân (CMND) được sử dụng trong thế giới điện tử và mạng internet.

Loại chứng thư này được dùng như một công cụ điện tử giúp nhận diện cá nhân, máy chủ hoặc một số đối tượng khác. Nó gắn định danh đối tượng đó với một “khóa công khai” được cấp bởi tổ chức có thẩm quyền.

Chứng thư số chứa khóa công khai (public key), trong khi đó chữ ký số chứa khóa bí mật (private key). Chứng thư số và chữ ký số kết hợp lại sẽ tạo thành một cặp khóa. Bạn có thể sử dụng cặp khóa này để ký số. Khóa bí mật của chữ ký số được lưu trữ trong 1 USB (gọi là Token USB hoặc SmartCard) giúp các khóa này tránh bị sao chép hoặc bị tấn công bởi virus khiến hỏng hóc và mất dữ liệu). Chứng thư số chứa public key và các thông tin người dùng theo chuẩn X.509.

Trong chứng minh thư luôn có những thông tin như họ tên, ngày tháng năm sinh, quê quán, hộ khẩu thường trú… Chứng thư số cũng vậy, tuy nhiên dữ liệu trên chứng thư số không phải những thông tin như trên mà bao gồm các nội dung dữ liệu sau đây:

Tên của người dùng ở định dạng tên phân biệt (DN). DN chỉ định tên của người dùng và bất kỳ thuộc tính bổ sung nào cần thiết để nhận dạng duy nhất người dùng (ví dụ: DN có thể chứa số nhân viên của người dùng).

Khóa công khai của người dùng. Khóa công khai là bắt buộc để những người khác có thể mã hóa cho người dùng hoặc xác minh chữ ký số của người dùng.

Thời hạn hiệu lực (hoặc thời gian tồn tại) của chứng chỉ: ngày bắt đầu và ngày kết thúc.

Các hoạt động cụ thể mà khóa công khai sẽ được sử dụng để mã hóa dữ liệu, xác minh chữ ký số hay cả hai.

Chữ ký của CA trên chứng chỉ có nghĩa là mọi hành vi giả mạo nội dung của chứng chỉ sẽ dễ dàng bị phát hiện. Chữ ký của CA trên chứng chỉ giống như con dấu phát hiện giả mạo trên một chai thuốc - bất kỳ hành vi giả mạo nào đối với nội dung của chứng chỉ đều dễ dàng bị phát hiện. Miễn là có thể xác minh được chữ ký của CA trên chứng chỉ thì chứng chỉ đó có tính toàn vẹn. Vì tính toàn vẹn của chứng chỉ có thể được xác định bằng cách xác minh chữ ký của CA, chứng chỉ vốn đã an toàn và có thể được phân phối theo cách hoàn toàn công khai (ví dụ: thông qua hệ thống thư mục có thể truy cập công khai).Tất cả những thông tin này sẽ được sử dụng để kê khai thuế qua mạng, khai báo hải quan, dịch vụ công, phát hành hóa đơn điện tử,… và thực hiện các giao dịch trực tuyến khác.

#### 1.1.4.2. So sánh giữa PKI và SSL

PKI và SSL đều là các giải pháp dựa trên chứng chỉ thiết lập sự tin tưởng với các chứng chỉ do tổ chức phát hành chứng chỉ (CA) cấp - cho dù đó là ủy thác công khai (SSL) hay ủy thác riêng (PKI).

PKI là toàn bộ khuôn khổ bao gồm phần cứng, phần mềm, chính sách, v.v. PKI cũng bao gồm CA, là thứ cấp các chứng chỉ kỹ thuật số để thiết lập sự tin cậy. Thông thường, CA được quản lý nội bộ theo các chính sách và thủ tục phù hợp với mức độ bảo mật và đảm bảo cần thiết của tổ chức.

SSL là một trong những trường hợp sử dụng hàng đầu của PKI. Nó cũng liên quan đến một CA cấp chứng chỉ, nhưng nó phải được các trình duyệt công nhận là CA đáng tin cậy công khai. Và trong khi có nhiều trường hợp sử dụng cho PKI, mục đích của SSL là để bảo mật dữ liệu nhạy cảm được chuyển qua truyền thông trực tuyến, như ngân hàng trực tuyến hoặc giao dịch thương mại điện tử.

Các trường hợp sử dụng phổ biến cho PKI có thể được xác định bằng cách xem xét các ứng dụng sử dụng chứng chỉ kỹ thuật số phổ biến nhất, chẳng hạn như:

1. Các ứng dụng quan trọng trong kinh doanh này cho thấy rõ ràng rằng PKI là một phần chiến lược của xương sống CNTT cốt lõi.
2. Chứng chỉ SSL cho các trang web và dịch vụ công khai
3. VPN.
4. Các ứng dụng và dịch vụ dựa trên đám mây công cộng
5. Các ứng dụng dựa trên đám mây riêng
6. Bảo mật email
7. Xác thực người dùng doanh nghiệp
8. Xác thực thiết bị
9. Xác thực dựa trên đám mây riêng tư
10. Ký tài liệu / tin nhắn

#### 1.1.4.3. Tổng quan về chứng thư X509

Chứng nhận X.509 là chứng nhận khóa công khai phổ biến nhất. Hiệp hội Viễn thông quốc tế (International Telecommunications Union – ITU) đã chỉ định chuẩn X.509 vào năm 1988. Đây là định dạng phiên bản 1 của chuẩn X.509. Vào năm 1993, phiên bản 2 của chuẩn X.509 được phát hành với 2 trường tên nhận dạng duy nhất được bổ sung. Phiên bản 3 của chuẩn X.509 được bổ sung thêm trường mở rộng đã phát hành vào năm 1997.

Một chứng nhận khóa công khai kết buộc một khóa công khai với sự nhận diện của một người (hoặc một thiết bị). Khóa công khai và tên thực thể sở hữu khóa này là hai mục quan trọng trong một chứng nhận.

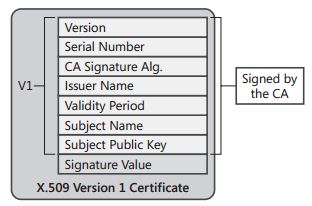
Về tổng quan ta đúc kết được các thông tin X509 gồm có các thành phần sau:

1. Version: Chỉ định phiên bản của chứng nhận X.509.
2. Serial Number: Số loạt phát hành được gán bởi CA. Mỗi CA nên gán một mã số loạt duy nhất cho mỗi giấy chứng nhận mà nó phát hành.
3. Signature Algorithm: Thuật toán chữ ký chỉ rõ thuật toán mã hóa được CA sử dụng để ký giấy chứng nhận. Trong chứng nhận X.509 thường là sự kết hợp giữa thuật toán băm (chẳng hạn như MD5 hoặc SHA-1) và thuật toán khóa công khai (chẳng hạn như RSA).
4. Issuer Name: Tên tổ chức CA phát hành giấy chứng nhận, đây là một tên phân biệt theo chuẩn X.500 (xem Phụ lục A). Hai CA không được sử dụng cùng một tên phát hành.
5. Validity Period: Trường này bao gồm 2 giá trị chỉ định khoảng thời gian mà giấy chứng nhận có hiệu lực. Hai phần của trường này là not-before và not-after, các giá trị thời gian này được đo theo chuẩn thời gian Quốc tế, chính xác đến từng giây.
6. Not-before chỉ định thời gian mà chứng nhận này bắt đầu có hiệu lực.
7. Not-after chỉ định thời gian mà chứng nhận hết hiệu lực.
8. Subject Name: là một X.500 DN, xác định đối tượng sở hữu giấy chứng nhận mà cũng là sở hữu của khóa công khai. Một CA không thể phát hành 2 giấy chứng nhận có cùng một Subject Name.
9. Public key: Xác định thuật toán của khóa công khai (như RSA) và chứa khóa công khai được định dạng tùy vào kiểu của nó.
10. Issuer Unique ID và Subject Unique ID: Hai trường này được giới thiệu trong X.509 phiên bản 2, được dùng để xác định hai tổ chức CA hoặc hai chủ thể khi chúng có cùng DN. RFC 2459 đề nghị không nên sử dụng 2 trường này.
11. Extensions: Chứa các thông tin bổ sung cần thiết mà người thao tác CA muốn đặt vào chứng nhận. Trường này được giới thiệu trong X.509 phiên bản 3.
12. Signature: Đây là chữ ký điện tử được tổ chức CA áp dụng. Tổ chức CA sử dụng khóa bí mật có kiểu quy định trong trường thuật toán chữ ký. Chữ ký bao gồm tất cả các phần khác trong giấy chứng nhận. Do đó, tổ chức CA chứng nhận cho tất cả các thông tin khác trong giấy chứng nhận chứ không chỉ cho tên chủ thể và khóa công khai. Những phần mở rộng của tên tập tin phổ biến cho chứng nhận X.509 bao gồm:
    1. Phần mở rộng tệp tin cer: chứng nhận được mã hóa theo luật mã hóa tiêu chuẩn (Canonical Encoding Rules – CER).
    2. Phần mở rộng tệp tin der: chứng nhận được mã hóa theo luật mã hóa phân biệt (Distinguished Encoding Rules – DER).
    3. Phần mở rộng tệp tin pem (Privacy-Enhanced Electronic Mail): định dạng mã hóa được sử dụng để lưu trữ các chứng nhận và khóa. Một tập tin được định dạng với chuẩn này có thể chứa các khóa bí mật (RSA và DSA), khóa công khai (RSA và DSA) và các chứng nhận X509. Định dạng này lưu trữ dữ liệu ở định dạng DER được mã hóa cơ sở 64, nằm giữa "-----BEGIN CERTIFICATE-----" và "-----END CERTIFICATE-----", phù hợp cho việc trao đổi ở dạng văn bản giữa các hệ thống.
    4. Phần mở rộng tệp tin p7b và p7c: PKCS #7 là một định dạng mã hóa cho việc lưu trữ một chứng nhận số và chuỗi chứng nhận của nó dưới dạng các ký tự ASCII. Định dạng này được sử dụng bởi CA để trả về các chứng nhận được phát hành cùng với chuỗi chứng nhận. Định dạng này có có thể được sử dụng như đầu vào cho yêu cầu gia hạn chứng nhận đến một CA.
    5. Phần mở rộng tệp tin pfx và p12: PKCS #12 là một định dạng mã hóa cho việc lưu trữ một chứng nhận số và kết hợp với khóa bí mật dưới dạng các ký tự ASCII. Định dạng này luôn luôn được trả về bởi CA khi CA phát sinh các khóa và phát hành chứng nhận đồng thời.

Đi sâu vào chi tiết có 3 phiên bản của chứng chỉ số được dùng trong một hệ tầng PKI là version 1,2, 3.

#### 1.1.4.4. Chứng thư X.509 phiên bản 1

Được định nghĩa vào năm 1988, X.509 version 1 giờ đây hầu như không còn được sử dụng nữa. Định dạng của loại chứng chỉ này được thể hiện như sau.



Một chứng chỉ X.509 version 1 bao gồm các trường sau:

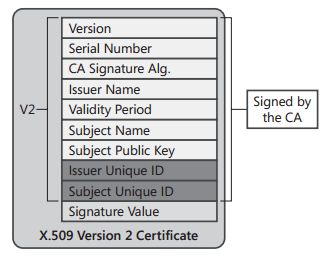
1. Version: chứa giá trị cho biết đây là chứng chỉ X.509 version 1
2. Serial Number: cung cấp một mã số nhận dạng duy nhất cho mỗi chứng chỉ được phát hành bởi CA
3. CA Signature Algorithm: tên của thuật toán mà CA sử dụng để ký lên nội dung của chứng chỉ số.
4. Issuer Name: tên phân biệt (distinguished name) của CA phát hành chứng chỉ. Thường thì tên phân biệt này được biểu diễn theo chuẩn X.500 hoặc định dạng theo đặc tả của X.509 và RFC 3280.
5. Validity Period: khoảng thời gian mà chứng chỉ được xem là còn hiệu lực, bao gồm 2 trường là: Valid From và Valid To.
6. Subject Name: tên của máy tính, người dùng, thiết bị mạng sở hữu chứng chỉ. Thường thì tên chủ thể này được biểu diễn theo chuẩn X.500 hoặc định dạng theo đặc tả của X.509, nhưng cũng có thể bao gồm các định dạng tên khác như được mô tả trong RFC 822.
7. Subject Public Key Info: khóa công khai của đối tượng nắm giữ chứng chỉ. Khóa công khai này được gửi tới CA trong một thông điệp yêu cầu cấp chứng chỉ (certificate request) và cũng được bao gồm trong nội dung của chứng chỉ được phát hành sau đó. Trường này cũng chứa nhận dạng của thuật toán được dùng để tạo cặp khóa công khai và khóa bí mật được liên kết với chứng chỉ.
8. Signature Value: chứa giá trị của chữ ký.

Các trường Issuer Name và Subject Name được cấu trúc để các chứng chỉ có thể được tổ chức thành một chuỗi các chứng chỉ mà bắt đầu bằng chứng chỉ được cấp cho người dùng, máy tính, thiết bị mạng, hoặc dịch vụ và kết thúc bằng chứng chỉ gốc của CA.

#### 1.1.4.5. Chứng thư X.509 phiên bản 2

Mặc dù chứng chỉ X.509 version 1 cung cấp khá đầy đủ những thông tin cơ bản về người nắm giữa chứng chỉ nhưng nó lại có ít thông tin về tổ chức cấp phát chứng chỉ khi chỉ bao gồm Issuer Name, CA Signature Algorithm và Signature Value. Điều này không giúp dự phòng trong trường hợp CA được thay mới.

Khi chứng chỉ của CA được thay mới, trường Issuer Name trong cả 2 chứng chỉ mới và cũ đều như nhau. Tương tự, có thể có một tổ chức khác muốn tạo một CA có trường Issuer Name trong chứng chỉ giống như vậy. Giải quyết vấn đề này để có thể sử dụng lại Issuer Name thì chứng chỉ X.509 version 2 đã được giới thiệu vào năm 1993. Trong định dạng của nó có thêm 2 trường mới như được thể hiện trong hình sau đây.



Hai trường mới được bổ sung là:

1. Issuer Unique ID: là một trường không bắt buộc, chứa chuỗi giá trị ở hệ 16, mang tính duy nhất và dành để nhận dạng CA. Khi CA thay mới chứng chỉ của chính nó, một Issuer Unique ID mới được khởi tạo cho chứng chỉ đó.
2. Subject Unique ID: là một trường không bắt buộc, chứa chuỗi giá trị ở hệ 16, mang tính duy nhất và dùng để nhận dạng chủ thể của chứng chỉ. Nếu chủ thể này cũng chính là CA thì trường này sẽ giống với Issuer Unique ID.

Ngoài việc đưa vào 2 trường mới ở trên thì trường Version trong chứng chỉ X.509 version 2 có giá trị là 2 để chỉ ra phiên bản của chứng chỉ.

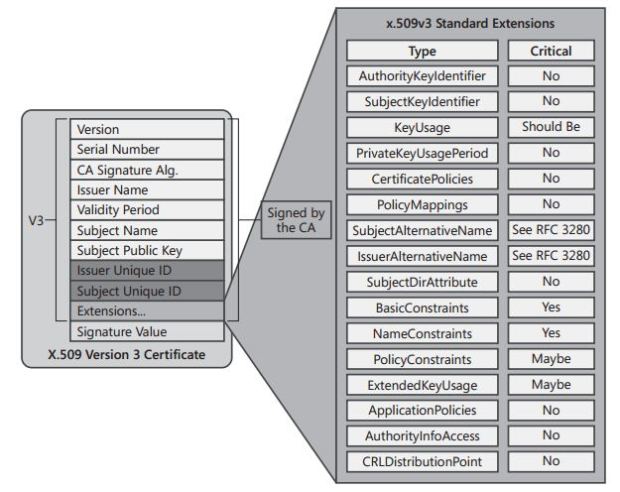
Các trường Issuer Unique ID và Subject Unique ID đã cải tiến quá trình xâu chuỗi chứng chỉ. Giờ đây việc tìm kiếm chứng chỉ của của CA sẽ là so khớp Issuer Name trong chứng chỉ được cấp phát với Subject Name trong chứng chỉ của CA và thực hiện thêm một bước kiểm tra thứ hai là so khớp Issuer Unique ID trong chứng chỉ được cấp phát với Subject Unique ID trong chứng chỉ của CA.

Bước so khớp thứ hai này cho phép phân biệt giữa các chứng chỉ của cùng một CA khi CA đó làm mới lại chứng chỉ của chính nó. Cách này cũng giúp phân biệt giữa các CA khác nhau nhưng trùng Subject Name.

Mặc dù định dạng X.509 version có cải tiến hơn version 1 nhưng chuẩn này cũng không còn được áp dụng rộng rãi. Và thực tế thì trong RFC 3280 đã khuyến cáo là bỏ qua việc sử dụng 2 trường mới trên của X.509 version 2 do lo ngại có thể có sự xung đột xảy ra nếu như hai chứng chỉ có cùng Subject Name và Subject Unique ID.

#### 1.1.4.6. Chứng thư X.509 phiên bản 3

Được ra đời vào năm 1996, định dạng X.509 version 3 được bổ sung thêm các phần mở rộng (extension) để khắc phục các vấn đề liên quan tới việc so khớp Issuer Unique ID và Subject Unique ID cũng như là các vấn đề về xác thực chứng chỉ. Một chứng chỉ X.509 version 3 co thể chứa một hoặc nhiều extension, như được thể hiện trong hình dưới đây:



Mỗi extension trong chứng chỉ X.509 version 3 gồm 3 phần:

1. -Extension Identifier: là một mã nhận dạng đối tượng (Object Identifier – OID) cho biết kiểu định dạng và các định nghĩa của extension.
2. -Criticality Flag: là một dấu hiệu cho biết thông tin trong extension có quan trọng (critical) hay không. Nếu một ứng dụng không thể nhận diện được trạng thái critical của extension hoặc extension không hề chứa giá trị nào thì chứng chỉ đó không thể được chấp nhận hoặc được sử dụng. Nếu mục criticality flag này không được thiết lập thì một có thể sử dụng chứng chỉ ngay cả khi ứng dụng đó không nhận diện được extension.
3. -Extension Value: là giá trị được gán cho extension. Nó phụ thuộc vào từng extension cụ thể.

Trong một chứng chỉ X.509 version 3 thì các extension sau có thể có là:

1. Authority Key Identifier: extension này có thể chứa một hoặc hai giá trị, chúng có thể là:

* Subject Name của CA và Serial Number của chứng chỉ của CA mà đã cấp phát chứng chỉ này.
* Giá trị băm của khóa công khai của chứng chỉ của CA mà đã cấp phát chứng chỉ này.

1. Subject Key Identifier: extension này chứa giá trị băm của khóa công khai của chứng chỉ.
2. Key Usage: một CA, người dùng, máy tính, thiết bị mạng hoặc dịch vụ có thể sở hữu nhiều hơn một chứng chỉ. Extension này định nghĩa các dịch vụ bảo mật mà một chứng chỉ có thể cung cấp như:

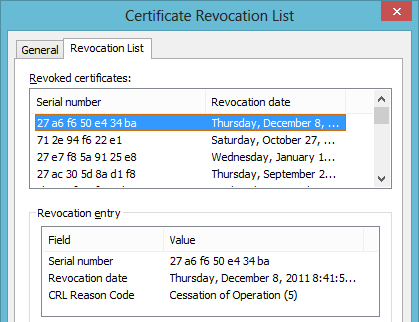
* Digital Signature: khóa công khai có thể được dùng để kiểm tra chữ ký. Khóa này cũng được sử dụng để xác thực máy khách và xác minh nguồn gốc của dữ liệu.
* Non-Repudiation: khóa công khai có thể được dùng để xác minh nhận dạng của người ký, ngăn chặn người ký này từ chối rằng họ không hề ký lên thông điệp hoặc đối tượng nào đó.
* Key Encipherment: khóa công khai có thể được dùng để trao đổi khóa, vú dụ như đối xứng (hoặc khóa phiên). Giá trị này được dùng khi một khóa RSA được dùng cho việc quản lý khóa.
* Data Encipherment: khóa công khai có thể được dùng để mã hóa dữ liệu một cách trực tiếp thay vì phải trao đổi một khóa đối xứng (hay khóa phiên) để mã hóa dữ liệu.
* Key Agreement: khóa công khai có thể được dùng để trao đổi khóa, ví dự như khóa đối xứng. Giá trị này được dùng khi một khóa Diffie-Hellman được dùng cho việc quản lý khóa.
* Key Cert Sign: khóa công khai có thể được dùng để kiểm tra chữ ký của chứng chỉ số.
* CRL Sign: khóa công khai có thể được dùng để kiểm tra chữ ký của CRL (danh sách chứa các chứng chỉ bị thu hồi).
* Encipher Only: giá trị này được dùng kết hợp với các extension Key Agreement và Key Usage. Kết quả là khóa đối xứng chỉ có thể được dùng để mã hóa dữ liệu.
* Decipher Only: giá trị này được dùng kết hợp với các extension Key Agreement và Key Usage. Kết quả là khóa đối xứng chỉ có thể được dùng để mã hóa dữ liệu.

1. Private Key Usage Period: extension này cho phép khóa bí mật có khoảng thời gian hiệu lực khác so với khoảng thời gian hiệu lực của chứng chỉ. Giá trị này có thể được đặt ngắn hơn so với khoảng thời gian hiệu lực của chứng chỉ. Điều này giúp khóa bí mật có thể được dùng để ký lên các tài liệu trong một khoảng thời gian ngắn (ví dụ, một năm) trong khi khóa công khai có thể được dùng để xác minh chữ ký trong khoảng thời gian hiệu lực của chứng chỉ là 5 năm.
2. Certificate Policies: extension này mô tả các chính sách và thủ tục được dùng để xác minh chủ thể của chứng chỉ trước khi chứng chỉ được cấp phát. Các chính sách chứng chỉ được đại diện bởi các OID. Ngoài ra, một chính sách chứng chỉ có thể bao gồm một đường dẫn (URL) tới trang web mô tả nội dung của chính sách và thủ tục.
3. Policy Mappings: extension này cho phép chuyển dịch thông tin về chính sách giữa hai tổ chức. Ví dụ, thử tưởng tượng rằng một tổ chức định nghĩa một chính sách chứng chỉ có tên là Management Signing mà trong đó các chứng chỉ được dùng để ký lên một lượng lớn các đơn đặt hàng. Một tổ chức khác có thể có một chính sách chứng chỉ tên là Large Orders mà cũng được dùng để ký lên một lượng lớn các đơn đặt hàng. Khi đó, Policy Mapping cho phép hai chính sách chứng chỉ này được đánh giá ngang nhau.
4. Subject Alternative Name: extension này cung cấp một danh sách các tên thay thế cho chủ thể của chứng chỉ. Trong khi định dạng cho Subject Name thường tuân theo chuẩn X.500 thì Subject Alternative Name cho phép thể hiện theo các dạng khác như User Principal Name (UPN), địa chỉ email, địa chỉ IP hoặc tên miền (DNS).
5. Issuer Alternative Name: extension này cung cấp một danh sách các tên thay thế cho CA. Mặc dù thường không được áp dụng nhưng extension này có thể chứa địa chỉ email của CA.
6. Subject Dir Attribute: extension này có thể bao gồm bất kỳ thuộc tính nào từ danh mục LDAP hoặc X.500 của tổ chức, ví dụ, thuộc tính country. Extension này có thể chứa nhiều thuộc tinh và với mỗi thuộc tính phải gồm OID và giá trị tương ứng của nó.
7. Basic Constraints: extension này cho biết chứng chỉ có phải của CA hay của các chủ thể như người dùng, máy tính, thiết bị, dịch vụ. Ngoài ra, extension này còn bao gồm một rằng buộc về độ dài của đường dẫn mà giới hạn số lượng các CA thứ cấp (subordinate CA) có thể tồn tại bên dưới CA mà cấp phát chứng chỉ này.
8. Name Constraints: extension này cho phép một tổ chức chỉ định không gian tên (namespace) nào được phép hoặc không được phép sử dụng trong chứng chỉ.
9. Policy Constraints: extension này có thể có trong các chứng chỉ của CA. Nó có thể ngăn cấm Policy Mapping giữa các CA hoặc yêu cầu mỗi chứng chỉ trong chuỗi chứng chỉ phải bao gồm một OID của chính sách chứng chỉ.
10. Enhanced Key Usage: extension này cho biết khóa công khai của chứng chỉ có thể được sử dụng như thế nào. Những cái này không có trong extension Key Usage. Ví dụ, Client Authentication (có OID là 1.3.6.1.5.5.7.3.2), Server Authentication (có OID là 1.3.6.1.5.5.7.3.1), và Secure E-mail (có OID là 1.3.6.1.5.5.7.3.4). Khi ứng dụng nhận được một chứng chỉ, nó có thể yêu cầu sự có mặt của một OID trong các OID kể trên.
11. CRL Distribution Points: extension này chứa một hoặc nhiều URL dẫn tới tập tin chứa danh sách các chứng chỉ đã bị thu hồi (CRL) được phát hành bởi CA. Nếu việc kiểm tra trạng thái thu hồi của chứng chỉ được cho phép thì một ứng dụng sẽ sử dụng các URL này để tải về phiên bản cập nhật của CRL. Các URL có thể sử dụng một trong các giao thức như HTTP, LDAP, FTP, File.
12. Authority Information Access: extension này có thể chứa một hoặc nhiều URL dẫn tới chứng chỉ của CA. Một ứng dụng sử dụng URL này để tải về chứng chỉ của CA khi xây dựng chuỗi chứng chỉ nếu như nó không có sẵn trong bộ nhớ đệm của ứng dụng.
13. Freshest CRL: extension này chứa một hoặc nhiều URL dẫn tới delta CRL do CA phát hành. Delta CRL chỉ chứa các chứng chỉ bị thu hồi kể từ lần cuối base CRl được phát hành. Nếu việc kiểm tra trạng thái thu hồi của chứng chỉ được cho phép thì một ứng dụng sẽ sử dụng các URL này để tải về phiên bản cập nhật của delta CRL. Các URL có thể sử dụng một trong các giao thức như HTTP, LDAP, FTP, File.
14. Subject Information Access: extension này chứa thông tin cho biết cách thức để truy cập tới các các chi tiết khác về chủ thể của chứng chỉ. Nếu đây là chứng chỉ của CA thì thông tin này có thể bao gồm các chi tiết về các dịch vụ xác minh chứng chỉ hay chính sách của CA. Nếu chứng chỉ được cấp cho người dùng, máy tính, thiết bị mạng, hoặc dịch vụ thì extension này có thể chứa thông tin về các dịch vụ được các chủ thể này cung cấp và cách thức để truy cập tới các dịch vụ đó.

### 1.1.5. Danh sách thu hồi chứng thư số

#### 1.1.5.1. CRL

Danh sách thu hồi chứng thư số (tiếng Anh: Certificate Revocation List, viết tắt CRL) là danh sách các certificate bị thu hồi và không còn được tin dùng nữa.



Mỗi một mục (entry) trong CRL tương ứng với một certificate và thường gồm 3 thông tin sau:

* Serial number của certificate.
* Thời điểm bị thu hồi.
* Lý do thu hồi.

Một CRL được tạo và phát hành (publish) định kỳ sau 1 khoảng thời gian nào đó do người quản trị CA chỉ định, ví dụ: 1 giờ, 1 ngày, 1 tuần, v.v. Một CRL cũng có thể được cập nhật và phát hành ngay sau khi một certificate nào đó bị thu hồi. Các CA sẽ đẩy CRL chứa các certificate do nó cấp phát và quản lý tới kho chứa là LDAP server hoặc Web server.

Để ngăn chặn nguy cơ CRL có thể bị làm giả dẫn đến certificate nào đó bị cố ý đưa vào hoặc bị loại bỏ khỏi CRL, các CRL đều có một chữ ký số được ký bởi CA đã phát hành nó. Và để xác thực chữ ký này trước khi có thể tin dùng CRL thì cần đến certificate của CA đã thực hiện việc ký trên. Thông thường certificate của các CA phổ biến đều được nạp sẵn bên trong các ứng dụng có hỗ trợ PKI như các trình duyệt web, đọc email hay hệ điều hành.

Khi ứng dụng PKI nhận được một certificate, thì bản thân certificate không chứa nội dung của CRL mà nó có một extension là CRL Distribution Points (CDP), cho biết địa chỉ URL của CRL (là file có đuôi .crl) mà nó cần tải về. Sau đó ứng dụng PKI phải phân tích (parse) file .crl này để xác định xem certificate đã bị thu hồi hay chưa, nói cách khác nếu serial number không có trong CRL thì certificate đó có thể được tin dùng.

Có thể thấy, CRL mắc phải một số hạn chế sau:

Nếu nhiều client cùng để tải về CRL từ kho chứa thì có nguy cơ làm tắc nghẽn, giảm hiệu suất mạng. Và nếu không thể kết nối tới kho chứa CRL do thì client không thể kiểm tra tính hiệu lực của certificate, dẫn đến certificate không được tin dùng.

Qua thời gian, khi số lượng các certificate được cấp phát cũng như thu hồi ngày một tăng dần, thì kích thước của file .crl cũng tăng theo (thường từ 200KB đến 20MB). Ứng dụng PKI phải tốn thời gian tải về và phân tích file .crl thường chứa một lượng rất lớn các certificate bị thu hồi, trong khi nó chỉ cần xác định trạng thái revocation của một (vài) certificate mà thôi.

Nếu certificate mà client cần kiểm tra đã bị thu hồi nhưng chưa được cập nhật vào CRL thì khi phân tích file .crl xong client vẫn chấp nhận certificate không còn hiệu lực đó!

Mặc định, các máy Windows có timeout là 15 giây khi cố gắng tải về CRL.

Ngoài ra, còn có các delta CRL chứa thông tin revocation cho các certificate bị thu hồi kể từ khi base CRL mới nhất được phát hành. Nhưng để kiểm tra trạng thái revocation, ứng dụng PKI vẫn cần phải có đủ cả base CRL và các delta CRL gần đây nhất. Dẫu vậy, cách này cũng sẽ giúp tiết kiệm thời gian và băng thông mạng vì nếu client đã có sẵn base CRL rồi thì nó chỉ cần tải thêm các delta CRL thôi.

#### 1.1.5.2. OCSP

Phương thức OCSP (tiếng Anh: Online Certificate Status Protocol) hiệu quả hơn CRL trong việc kiểm tra trạng thái thu hồi của certificate.

Như được mô tả trong RFC 2560, OCSP là một giao thức được sử dụng để nhận về trạng thái revocation của một certificate có chuẩn định dạng là X.509. Hoạt động theo mô hình client/server, các thông điệp OCSP (request, response) được mã hóa theo chuẩn ANS.1 và được truyền qua giao thức HTTP. Server cũng thường được gọi là OCSP responder. Cơ bản nó làm việc như sau:

Client gửi một request chứa serial number của certificate mà nó cần kiểm tra tới server.

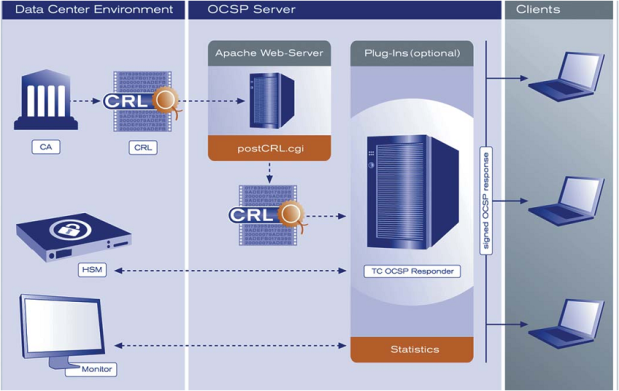
1. Nếu có sẵn một response được cache cho request trên thì server sẽ gửi ngay cho client. Còn không thì server sẽ kiểm tra xem có sẵn một CRL được cache chưa, nếu có thì nó sẽ dò tìm trong CRL cho serial number của certifcate rồi trả về kết quả cho client. Nếu chưa có file CRL, server sẽ tải về từ các vị trí CDP đã được cấu hình trước.
2. Response trả về cho client cho biết 1 trong 3 trạng thái có thể của certificate là:

* "good": không có trong CRL
* "revoked": bị thu hồi vĩnh viễn hoặc tạm thời (hold)
* "unknown": server không biết tới serial number có trong request

1. Response cũng được ký số bởi server sử dụng private key của một trong các thành phần:

* CA đã cấp phát certificate có trong request.
* Trusted Responder mà public key của nó đã được client tin tưởng
* CA Designated Responder (Authorized Responder) có certificate được cấp bởi CA mà OCSP server đang phục vụ cho nó.

1. Client nhận được kết quả và cache lại để lần sau không cần gửi request lên server để kiểm tra certificate đó nữa.
2. Nếu server không thể xử lý request, client sẽ nhận được response không được ký, chứa thông báo lỗi.



Rõ ràng, OCSP đã giải quyết được các vấn đề gặp phải với CRL là:

* Tiết kiệm băng thông do các request và response có kích thước nhỏ hơn nhiều (thường chỉ 4KB) so với file .crl.
* Tiết kiệm thời gian vì chỉ phải kiểm tra trạng thái của 1 certificate thay vì phải phân tích file .crl.
* Nếu thông tin revocation có sẵn trong cache tại client và server thì tiết kiệm được được cả thời gian lẫn băng thông.
* Hệ thống certificate validation với OCSP có thể dễ dàng được mở rộng, độ sẵn sàng cao khi cần xử lý một lượng lớn các request.
* OCSP responder đảm bảo luôn sử dụng các phiên bản CRL mới nhất làm cơ sở cho việc kiểm tra tính hiệu lực của certificate cũng như là khả năng phản hồi gần như lập tức (real-time) khi nó nhận được yêu cầu từ client.
* Một OCSP server có thể phục vụ công tác certificate validation cho nhiều CA. Điều này giúp client tránh phải lưu nhiều CRL.

Tuy nhiên, trong an toàn thông tin thì không có một giải pháp nào giải quyết được mọi khía cạnh rủi ro cả. OCSP không nằm ngoài quy luật đó, nó không phải là “viên đạn bạc” cho vấn đề certificate validation, bản thân nó cũng phải đối mặt với các nguy cơ khác nhau như:

* Availability: Nếu vì lý do nào đó mà client không thể liên lạc với OCSP server thì quá trình validation bị đổ vỡ, khi đó client có thể được cấu hình để quay lại cơ chế CRL.
* Replay attack: kẻ tấn công chụp lại các good response, chờ đến khi certificate bị thu hồi nhưng validity period vẫn còn hiệu lực thì hắn gửi lại response đó cho client.
* DoS/DDoS: kẻ tấn công cố gắng làm đầy khả năng xử lý của OCSP server bằng cách gửi request với tần suất lớn và liên tục. Việc server phải mất thời gian và năng lực để ký số cho mỗi response cũng khiến tình huống này thêm trầm trọng. Ngoài ra, việc các thông báo lỗi không được ký số cũng bị lợi dụng, kẻ tấn công sẽ gửi các thông báo lỗi giả này cho client và ngăn chặn các good response đến từ server khiến cho client không thể dùng được certificate này.
* Privacy: các thông điệp OCSP đều không được mã hóa nên việc phải gửi request tới OCSP server để kiểm tra certificate cho một domain nào đó khiến bộc lộ địa chỉ IP của client cũng như website mà client muốn ghé thăm.
* Compatibility: Một số ứng dụng và hệ điều hành cũ như Windows XP không hỗ trợ giao thức OCSP.

Nội dung sau đây sẽ chi tiết hơn về việc lý do thu hồi chứng thư số, lý do để hủy một chứng thư là do sự nhận diện được xác nhận bởi CA đã thay đổi. Danh sách hủy bỏ chứng nhận (Certificate Revocation List – CRL) chứa thông tin thời gian nhằm xác định thời điểm tổ chức CA phát hành nó. CA ký CRL với cùng khóa bí mật được dùng để ký các chứng thư. Các CRL thường được chứa trong cùng kho với các chứng thư nhằm dễ dàng cho việc rút trích. Các CA phát hành các CRL theo định kì, thường là hàng giờ hoặc hàng ngày.

Ngoài việc xác minh chữ ký của CA trên chứng chỉ thì hần mềm ứng dụng cũng phải chắc chắn rằng chứng chỉ vẫn đáng tin cậy tại thời điểm sử dụng. Các chứng chỉ không còn đáng tin cậy phải được CA thu hồi. Có rất nhiều lý do tại sao một chứng chỉ có thể cần phải bị thu hồi trước khi kết thúc thời hạn hiệu lực của nó. Ví dụ: khóa riêng tư (khóa ký hoặc khóa giải mã) tương ứng với khóa công khai trong chứng chỉ có thể bị xâm phạm. Ngoài ra, chính sách bảo mật của tổ chức có thể quy định rằng các chứng chỉ của nhân viên rời khỏi tổ chức phải bị thu hồi. Trong những tình huống này, người dùng trong hệ thống phải được thông báo rằng việc tiếp tục sử dụng chứng chỉ không còn được coi là an toàn nữa. Trạng thái thu hồi của chứng chỉ phải được kiểm tra trước mỗi lần sử dụng. Do đó, PKI phải kết hợp hệ thống thu hồi chứng chỉ có thể mở rộng. CA phải có khả năng công bố thông tin về trạng thái của mỗi chứng chỉ trong hệ thống một cách an toàn. Sau đó, phần mềm ứng dụng thay mặt người dùng phải xác minh thông tin thu hồi trước mỗi lần sử dụng chứng chỉ. Sự kết hợp giữa xuất bản và sử dụng nhất quán thông tin thu hồi chứng chỉ tạo thành một hệ thống thu hồi hoàn chỉnh. Phương tiện phổ biến nhất để phân phối thông tin thu hồi chứng chỉ là CA tạo danh sách thu hồi chứng chỉ an toàn (CRL) và xuất bản các CRL này lên hệ thống thư mục. CRLs chỉ định số sê-ri duy nhất của tất cả các chứng chỉ đã bị thu hồi. Trước khi sử dụng chứng chỉ, ứng dụng phía máy khách phải kiểm tra CRL thích hợp để xác định xem chứng chỉ có còn đáng tin cậy hay không. Các ứng dụng phía máy khách phải thay mặt người dùng kiểm tra các chứng chỉ đã thu hồi một cách nhất quán và minh bạch.

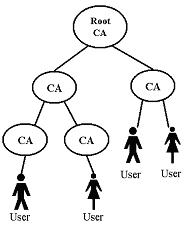
Có 2 trạng thái revocation được quy định trong RFC-3280 là:

* Revoked: một khi certificate đã bị thu hồi thì không thể khôi phục lại và sử dụng tiếp được nữa.
* Hold: certificate chỉ tạm thời bị mất hiệu lực. Ví dụ, nếu người dùng không chắc là private key đã bị mất hay chưa thì CA có thể đưa certificate vào trạng thái hold. Nếu sau đó tìm thấy private key và chắc rằng không ai đã đọc được nó thì trạng thái hold được gỡ bỏ và certificate có hiệu lực trở lại.

### 1.1.7. Mô hình hoạt động của PKI

Hiện tại có 3 mô hình PKI phổ biến: Hierarchical PKI, Mesh PKI, Single CA.

#### 1.1.7.1. Hierarchical PKI



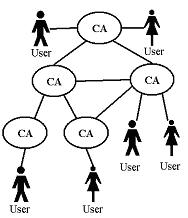
Đây là mô hình PKI được áp dụng rộng rãi trong các tổ chức lớn. Có một CA nằm ở cấp trên cùng gọi là root CA, tất cả các CA còn lại là các Subordinate CA (gọi tắt là sub. CA) và hoạt động bên dưới root CA. Ngoại trừ root CA thì các CA còn lại trong đều có duy nhất một CA khác là cấp trên của nó. Hệ thống tên miền DNS trên Internet cũng có cấu trúc tương tự mô hình này.

Tất cả các thực thể (như người dùng, máy tính) trong tổ chức đều phải tin cậy cùng một root CA. Sau đó các trust relationship được thiết lập giữa các sub. CA và cấp trên của chúng thông qua việc CA cấp trên sẽ cấp các chứng chỉ cho các sub. CA ngay bên dưới nó. Lưu ý, root CA không trực tiếp cấp chứng chỉ số cho các thực thể mà chúng sẽ được cấp bởi các sub. CA. Các CA mới có thể được thêm ngay dưới root CA hoặc các sub. CA cấp thấp hơn để phù hợp với sự thay đổi trong cấu trúc của tổ chức. Sẽ có các mức độ tổn thương khác nhau nếu một CA nào đó trong mô hình này bị xâm hại.

Trường hợp một sub. CA bị thỏa hiệp thì CA cấp trên của nó sẽ thu hồi chứng chỉ đã cấp cho nó và chỉ khi sub. CA đó được khôi phục thì nó mới có thể cấp lại các chứng chỉ mới cho người dùng của nó. Cuối cùng, CA cấp trên sẽ cấp lại cho nó một chứng chỉ mới.

Nếu root CA bị xâm hại thì đó là một vấn đề hoàn toàn khác, toàn bộ hệ thống PKI sẽ chịu ảnh hưởng. Khi đó tất cả các thực thể cần được thông báo về sự cố và cho đến khi root CA được phục hồi và các chứng chỉ mới được cấp lại thì không một phiên truyền thông nào là an toàn cả. Vì thế, cũng như single CA, root CA phải được bảo vệ an toàn ở mức cao nhất để đảm bảo điều đó không xảy ra và thậm chí root CA có thể ở trạng thái offline – bị tắt và không được kết nối vào mạng.

#### 1.1.7.2. Mesh PKI

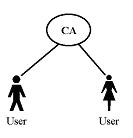


Nổi lên như một sự thay thế chính cho mô hình Hierarchical PKI truyền thống, thiết kế của Mesh PKI giống với kiến trúc Web-of-Trust trong đó không có một CA nào làm root CA và các CA sẽ có vai trò ngang nhau trong việc cung cấp dịch vụ. Tất cả người dùng trong mạng lưới có thể tin cậy chỉ một CA bất kỳ, không nhất thiết hai hay nhiều người dùng phải cùng tin một CA nào đó và người dùng tin cậy CA nào thì sẽ nhận chứng chỉ do CA đó cấp

Các CA trong mô hình này sau đó sẽ cấp các chứng chỉ cho nhau. Khi hai CA cấp chứng chỉ cho nhau thì một sự tin cậy hai chiều được thiết lập giữa hai CA đó. Các CA mới có thể được thêm vào bằng cách tạo các mối tin cậy hai chiều giữa chúng với các CA còn lại trong mạng lưới.

Vì không có một CA duy nhất làm cấp cao nhất nên sự tổn hại khi tấn công vào mô hình này có khác so với hai mô hình trước đó. Hệ thống PKI không thể bị đánh sập khi chỉ một CA bị thỏa hiệp. Các CA còn lại sẽ thu hồi chứng chỉ mà chúng đã cấp cho CA bị xâm hại và chỉ khi CA đó khôi phục hoạt động thì nó mới có khả năng cấp mới các chứng chỉ cho người dùng rồi thiết lập trust với các CA còn lại trong mạng lưới.

#### 1.1.7.3. Single CA



Đây là mô hình PKI cơ bản nhất phù hợp với các tổ chức nhỏ trong đó chỉ có một CA cung cấp dịch vụ cho toàn hệ thống và tất cả người dùng đặt sự tin cậy vào CA này. Mọi thực thể muốn tham gia vào PKI và xin cấp chứng chỉ đều phải thông qua CA duy nhất này. Mô hình này dễ thiết kế và triển khai nhưng cũng có các hạn chế riêng. Thứ nhất là ở khả năng co giãn – khi quy mô tổ chức được mở rộng, chỉ một CA thì khó mà quản lý và đáp ứng tốt các dịch vụ. Hạn chế thứ hai là CA này sẽ là điểm chịu lỗi duy nhất, nếu nó ngưng hoạt động thì dịch vụ bị ngưng trệ. Cuối cùng, nếu nó bị xâm hại thì nguy hại tới độ tin cậy của toàn bộ hệ thống và tất cả các chứng chỉ số phải được cấp lại một khi CA này được phục hồi.

### 1.1.8. Luật pháp và chính sách về PKI

Luật pháp đã quy định tại Thông tư 08 của Bộ Thông tin và Truyền thông để sử dụng chứng thư số.

#### 1.1.8.1. Điều kiện thu hồi mọi loại chứng thư số

Thời hạn sử dụng chứng thư số đã hết. Theo yêu cầu bằng văn bản từ thuê bao, có xác nhận của người quản lý thuê bao, theo Mẫu 09 của Phụ lục kèm theo Thông tư này trong các trường hợp: khóa bí mật bị lộ hoặc nghi bị lộ; thiết bị lưu khóa bí mật bị thất lạc hoặc các trường hợp mất an toàn khác; thiết bị lưu khóa bí mật bị hỏng.

Theo yêu cầu bằng văn bản giấy hoặc văn bản điện tử có ký số của chứng thư số có hiệu lực từ cơ quan tiến hành tố tụng, cơ quan an ninh.

Theo yêu cầu bằng văn bản giấy hoặc văn bản điện tử có ký số của chứng thư số có hiệu lực từ Người quản lý thuê bao.

Thuê bao vi phạm các quy định về quản lý, sử dụng thiết bị lưu khóa bí mật.

#### 1.1.8.2. Điều kiện thu hồi chứng thư số cá nhân

Các trường hợp quy định tại mục Đối với mọi loại chứng thư số.

Cá nhân thay đổi vị trí công tác mà thông tin về vị trí công tác mới không phù hợp với thông tin trong chứng thư số.

Cá nhân nghỉ hưu, thôi việc hoặc từ trần.

#### 1.1.8.3. Điều kiện thu hồi chứng thư số của cá nhân - người có thẩm quyền của cơ quan, tổ chức theo quy định của pháp luật về quản lý và sử dụng con dấu, người có chức danh nhà nước

Các trường hợp quy định tại mục Đối với mọi loại chứng thư số và mục Đối với chứng thư số của cá nhân.

Cá nhân thay đổi thẩm quyền của cơ quan, tổ chức theo quy định của pháp luật về quản lý và sử dụng con dấu, thay đổi chức danh nhà nước.

#### 1.1.8.4. Điều kiện thu hồi chứng thư số của cơ quan, tổ chức khi

Cơ quan, tổ chức giải thể, đổi tên hoặc địa chỉ hoạt động mà thông tin không phù hợp với thông tin trong chứng thư số.

#### 1.1.8.5. Điều kiện thu hồi chứng thư số của thiết bị, dịch vụ, phần mềm trong trường hợp

Các trường hợp quy định tại mục Đối với mọi loại chứng thư sốThiết bị, dịch vụ, phần mềm ngừng hoạt động, đổi tên mà thông tin không phù hợp với thông tin trong chứng thư số.

#### 1.1.8.6. Tầm quan trọng của việc thu hồi chứng thư số.

Chìa khóa cho sự thành công của một hệ thống PKI nằm ở chỗ việc cấp phát, thu hồi, kiểm tra hiệu lực certificate phải được tiến hành một cách chính xác và nhanh chóng.

CRL có thể phục vụ cho một môi trường nhỏ dưới 1000 certificate nhưng nếu số lượng certificate bị thu hồi lên tới hàng chục ngàn thì việc triển khai OCSP với độ sẵn sàng và tin cậy cao là được xem xét.

## 1.2. Chữ ký số

### 1.2.1. Khái niệm

Chữ ký số là những thông tin đi kèm với dữ liệu nhằm xác định chủ thể của dữ liệu được gửi. Cụ thể hơn chữ ký số được tạo ra từ việc sử dụng kết hợp giữa hàm băm và mật mã khóa công khai để đảm bảo tính toàn vẹn, giúp xác thực nguồn gốc của thông điệp và đồng thời bên gửi không thể chối từ việc đã tạo ra thông điệp đó. Nó là một giá trị băm của thông điệp được mã hóa bằng khóa bí mật của bên gửi rồi được đính kèm với thông điệp gốc. Bên nhận sẽ dùng khóa công khai của bên gửi để giải mã phần chữ ký ra được giá trị băm của thông điệp sau đó đối chiếu với giá trị mà nó thu được từ việc thực hiện lại hàm băm trên thông điệp gốc. Nếu hai giá trị đó giống nhau thì bên nhận có thể tin cậy được rằng thông điệp không bị thay đổi và nó chỉ được gửi từ bên sở hữu khóa công khai ở trên.

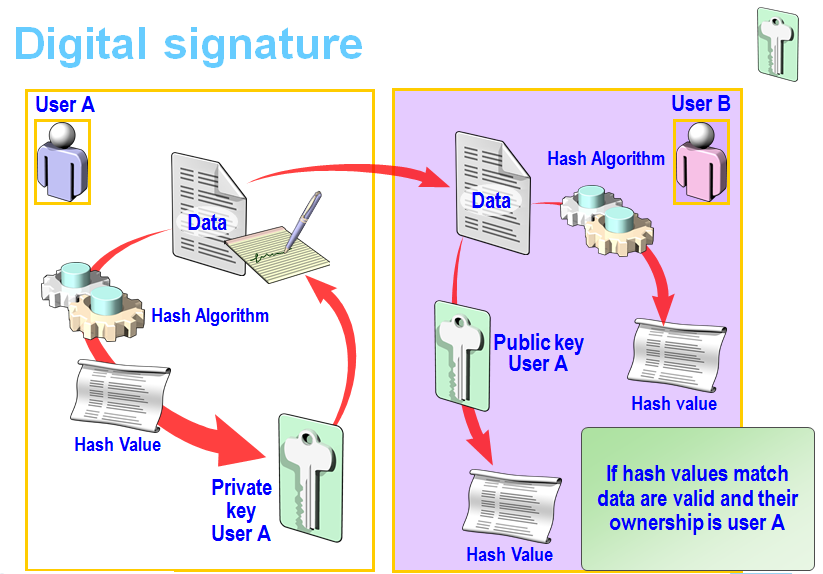
### 1.2.2. Đặc trưng

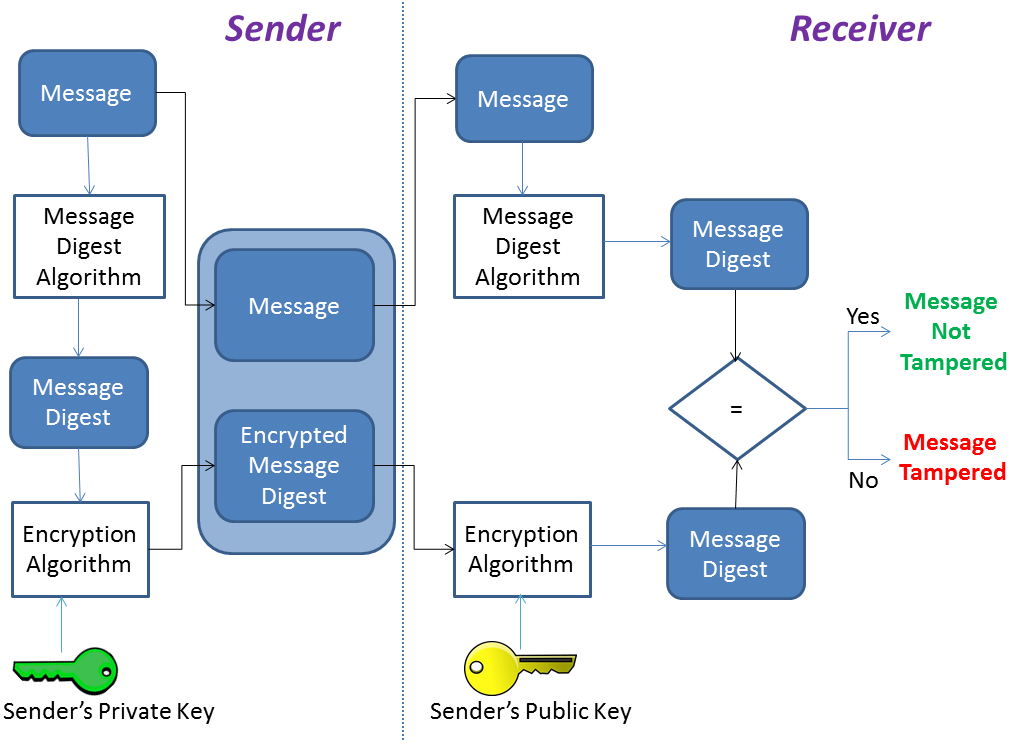
Đơn giản cho người sử dụng mà vẫn đảm bảo được tính bảo mật, kỹ thuật sử dụng chữ ký số là một trong những kỹ thuật được sử dụng phổ biến, đa dạng trong hầu hết các lĩnh vực, nhất là Tài chính, Ngân hang, Kế toán…

### 1.2.3. Quy trình tạo và kiểm tra chữ ký số

Giải thuật tạo ra chữ ký số (Digital Signature generation algorithm) là một phương pháp sinh chữ ký số.

Giải thuật kiểm tra chữ ký số (Digital Signature verification algorithm) là một phương pháp xác minh tính xác thực của chữ ký số, có nghĩa là nó thực sự được tạo ra bởi 1 bên chỉ định.





Một hệ chữ ký số (Figital Signature Scheme) bao gồm giải thuật tạo chữ số và giải thuật kiểm tra chữ kỹ số.

Quá trình tạo chữ ký số (Digital Signature signing process) bao gồm:

* Giải thuật tạo chữ ký số.
* Phương pháp chuyển dữ liệu thông điệp thành dạng có thể ký được.

Quá trình kiểm tra chữ ký số (Digital signature verification process) :

* Giải thuật kiểm tra chữ ký số.
* Phương pháp khôi phục dữ liệu từ thông điệp.

#### 1.2.3.1. Quá trình ký (bên gửi)

Tính toán chuỗi đại diện (message digest/ hash value) của thông điệp sử dụng một giải thuật băm (Hashing algorithm).

Chuỗi đại diện được ký sử dụng khóa riêng (Priavte key) của người gửi va 1 giải thuật tạo chữ ký (Signature/ Encryption algorithm). Kết quả chữ ký số (Digital signature) của thông điệp hay còn gọi là chuỗi đại diện được mã hóa (Encryted message digest).

Thông điệp ban đầu (message) được ghép với chữ ký số (Digital signature) tạo thành thông điệp đã được ký (Signed message).

Thông điệp đã được ký (Signed message) được gửi cho người nhận.

#### 1.2.3.2. Quá trình kiểm tra chữ ký (bên nhận)

Tách chữ ký số và thông điệp gốc khỏi thông điệp đã ký để xử lý riêng;

Tính toán chuỗi đại diện MD1 (message digest) của thông điệp gốc sử dụng giải thuật băm (là giải thuật sử dụng trong quá trình ký)

Sử dụng khóa công khai (Public key) của người gửi để giải mã chữ ký số -> chuỗi đại diện thông điệp MD2

So sánh MD1 và MD2:

* Nếu MD1 giống MD2 thì kết luận chữ ký kiểm tra thành công. Thông điệp đảm bảo tính toàn vẹn và thực sự xuất phát từ người gửi (do khóa công khai được chứng thực).
* Nếu MD1 khác MD2 thì kết luận chữ ký không hợp lệ. Thông điệp có thể đã bị sửa đổi hoặc không thực sự xuất phát từ người gửi.

### 1.2.4. Các thuật toán chữ ký số

#### 1.2.4.1. Giới thiệu

Hiện nay trên thế giới có rất nhiều thuật toán chữ ký số chẳng hạn như RSA, Rabin, Elgamal, Elliptic, Liên Xô Gost 34-10.94, Nga Gost 30.10.2001, … và rất nhiều thuật khác. Trong đó, họ nhà Elgamal còn được phân tách thành các thuật giải khác như Schnorr, DSS. Phần trình bày sau sẽ khai quát về những thuật toán phổ biến thường được dùng là RSA, Rabin và Elgamal.

#### 1.2.4.2. Sơ đồ chữ ký RSA

Sơ đồ chữ ký RSA được Diffie-Hellman đề xuất và được Ronald Linn Rivest, Adi Shamir và Leonard Adleman thực hiện.

Tạo khóa: quá trình tạo khóa cho sơ đồ chữ ký RSA giống như quá trình hình thành khóa của hệ mật RSA, tức là: Alice chọn cặp số nguyên tố đủ lớn p và q, với  , tính N=pq. Chọn số nguyên e thỏa mãn. Alice đi xác định số nguyên d, thỏa mãn phương trình . Số d là khóa mật của Alice.

Tạo chữ ký: để tạo ra chữ ký số của bức điện m Alice tạo ra số



Thẩm tra chữ ký: để thẩm tra chữ ký s có phải của Alice ký không thì Bob kiểm chứng bằng thủ tục sau:

Verify(N,e)(m,s)=TRUE, nếu như 

Rõ ràng chúng ta thấy rằng qúa trình tạo chữ ký và thẩm tra chữ ký giống với quá trình mã và giải mã của hệ mật RSA chỉ khác là quá trình tạo chữ ký Alice dùng khóa mật còn quá trình thẩm tra thì Bob dùng khóa công cộng.

Bàn luận về độ an toàn của sơ đồ chữ ký số RSA:

Nếu sơ đờ ký số thực hiện đơn giản như trên thì tội phạm (Oscar) dễ dàng lừa. Ví dụ như Oscar có thể chọn ngẫu nhiên  và tính toán độ lớn:



Thì rõ ràng quá trình thẩm tra (m,s) là hoàn toàn đúng. Ngoài ra do tính chất nhân của hàm RSA, tức là nếu có hai bức điện m1 và m2 tương ứng với nó là 2 chữ ký s1 và s2 thì dễ dàng hình thành chữ ký thứ ba s1s2 với bức điện thứ ba m1m2:



Để chống lại sự giả mạo chữ ký theo phương pháp trên thì một phương pháp đơn giản là thêm thông tin phụ vào bức điện M, có nghĩa là m=M||I, ở đây I là dấu hiệu nhận dạng ví dụ như I=”tên tác giả”.

Ngoài ra kết hợp với việc ký lên giá trị hàm hash của bức điện, tức là ký lên giá trị:

m=hash(M).

Vì những tính chất của hàm hash sẽ chống lại khả năng giả mạo trên trừ xác suất rất nhỏ, vì nếu khó có thể tìm được bức điện mà giá trị hash của nó trùng với giá trị hàm hash đã cho.

#### 1.2.4.3. Sơ đồ chữ ký Rabin

Sơ đồ chữ ký của Rabin rất giống sơ đồ chữ ký RSA. Sự khác nhau giữa chúng chỉ nằm ở quá trình thẩm tra chữ ký. Trong sơ đồ chữ ký RSA thì tham số e là số lẻ bởi vì thỏa mãn điều kiện gcd(e, )=1,  là số chẳn, còn trong sơ đồ Rabin e=2. Chúng ta xem cụ thể quá trình ký của sơ đồ Rabin.

Tạo khóa: quá trình tạo khóa của sơ đồ chữ ký Rabin giống như quá trình tạo khóa của sơ đồ chữ ký RSA. Tức là chọn hai số nguyên tố khác nhau p và q có độ lớn gần bằng nhau và tính N=pq. Số N là khóa mở của Alice, còn số p, q là khóa mật.

Tạo chữ ký: để tạo chữ ký  Alice tính giá trị:



ở đây chúng ta thấy để tạo nên chữ ký s thì m phải thuộc QRN. Alice có thể chọn cơ chế thích hợp để tạo nên m. Chú ý có đến ¼ số phần tử của nhóm thuộc về QRN nên Alice hình thành m là không khó.

Thẩm tra chữ ký: để thẩm tra chữ ký Bob xem thủ tục sau:

Verify(N)(m,s)=TRUE, nếu như 

Sơ đồ chữ ký Rabin có nhưng ưu điểm hơn so với sơ đồ RSA. Thứ nhất là giả mạo chữ ký là phức tạp như là phân tích số nguyên ra thừa số nguyên tố. Thứ hai là việc thẩm tra chữ ký hoàn thành nhanh hơn và hòan toàn thuận lợi thực thi các ứng dụng.

Và để chống lại các cách tấn công như trong RSA thì quá trình ký cũng thêm nhứng thông tin phụ và sử dụng hàm hash.

#### 1.2.4.4. Sơ đồ chữ ký Elgama

Sơ đồ chữ ký Elgamal được giới thiệu năm 1985. Sơ đồ này thiết kế dành riêng cho chữ ký số khác với sơ đồ RSA dành chung cho cả hệ thống mã công khai và chữ ký số.

Tạo khóa: quá trình tạo khóa giống như qúa trình tạo khóa của hệ mật Elgamal, tức là Alic chọn số nguyên tố p đủ lớn để bài toán logarith rời rạc trên Zp là khó giải, và chọn là phần tử nguyên thủy, chọn là số nguyên làm khóa mật và tính khóa công cộng yA= .

Tạo chữ ký: để ký lên thông điệp m Alice tạo ra số ngẫu nhiên k thỏa mãn  và UCLN(k,p-1)=1 và hình thành nên chữ ký là cặp (r,s), ở đây



Thẩm tra chữ ký: để thẩm tra chữ ký (r,s) Bob xem kết quả của hàm kiểm tra:

Verify(,yA,p)(m,(r,s))=TRUE, nếu như r < p và

Xem sự đúng đắn của phương trình thẩm tra chữ ký: thấy rằng Alice hình thành chữ ký với khóa mật xA và cả số nguyên ngẫu nhiên k. Việc thẩm tra chữ ký chỉ bằng thông tin công khai.