ĐẠI HỌC QUỐC GIA HÀ NỘI

ĐẠI HỌC CÔNG NGHỆ

* *

*

BÁO CÁO:

BẢO MẬT – HỆ MẬT MÃ KHÓA CÔNG KHAI VÀ ỨNG DỤNG HỆ MẬT MÃ KHÓA CÔNG KHAI

Người thực hiện: Hà Đức Hiệp – 17020732

Hà Nội, năm 2020.

Chương A. Tổng quan về bảo mật

I. Tổng quan về bảo mật

a) Khái niệm bảo mật

Bảo mật trong hệ thống thông tin được hiểu là các biện pháp nhằm ngăn chặn, giảm thiểu đến mức tối đa khả năng bị xâm hại tới thông tin/tài nguyên. Trong đó, khả năng bị xâm hại là bất kỳ điểm yếu nào có thể bị lợi dụng để tấn công hệ thống hoặc thông tin trong hệ thống đó

b) Các hình thức tấn công

Để hiểu rõ hơn về các cơ chế bảo mật, ta cần phải tìm hiểu các phương thức tấn công. Có nhiều cách để phân chia, dựa theo hình thức thực hiện, có thể chia thành hai nhóm lớn sau: tấn công Chủ động và tấn công Bị động

i. Tấn công chủ động

Là hình thức tấn công dưới dạng tác động tới tài nguyên hay dữ liệu của hệ thộng và làm ảnh hưởng tới hoạt động của hệ thống. Tấn công chủ động có thể chia thành các nhóm nhỏ sau: tấn công giả mạo, tấn công lặp lại, tấn công sửa đổi thông tin và tấn công từ chối dịch vụ.

ii. Tấn công bị động

Tấn công bị động là hình thức tấn công dưới dạng thu thập, phân tích thông tin trên đường truyền mà không gây nguy hoại tới tài nguyên hay dữ liệu của hệ thống. Tấn công bị động rất nguy hiểm vì nó không làm thay đổi thông tin truyền. Có hai loại tấn công bị động phổ biến là theo dõi lưu lượng và nghe lén.

c) Các dịch vụ bảo mật

Dựa vào hình thức tấn công trên, chuẩn kiến trúc bảo mật cho mô hình OSI X.800 đưa ra 5 dịch vụ bảo mật sau:

- Bảo mật dữ liệu
- Dịch vụ xác thực
- Dịch vụ điều khiển truy cập
- Toàn vẹn dữ liệu
- Chống chối bỏ

II. Tổng quan về mã hóa

a) Định nghĩa

Mã hóa là việc ứng dụng toán học vào việc biến đổi thông tin thành dạng khác với mục đích che giấu nội dung, ý nghĩa thông tin cần được bảo vệ. Một phương pháp mã hóa đầy đủ, có thể áp dụng cho một hệ thống thông tin được gọi là hệ mã hóa.

Có hai loại mã hóa chính: mã hóa đối xứng và mã hóa bất đối xứng

b) Mã hóa đối xứng

Mã hóa đối xứng là loại mã hóa mà quá trình mã hóa và giải mã sử dụng cùng một mã khóa gọi là khóa bí mật hay khóa đối xứng. Mã hóa đối xứng gồm các loại mã hóa sau:

- Mã hóa Affine
- Mã hóa Caesar
- Mã hóa Hill
- Mã hóa Vigenere

c) Mã hóa bất đối xứng

Trong mã hóa đối xứng, có một vấn đề phát sinh là việc quy định chung mã khóa k giữa người gửi và người nhận. Khóa này cần được thay đổi để đảm bảo bí mật và mỗi khi thay đổi, nó phải được trao đổi bí mật giữa hai bên. Điều này khiến cho mã hóa đối xứng trở nên không an toàn trong hiện nay.

Mã hóa bất đối xứng sử dụng một ý tưởng khác so với mã hóa đối xứng. Trong mã hóa bất đối xứng, nó sử dụng một cặp khóa gồm: khóa công khai được công bố rộng rãi để thực hiện việc mã hóa thông tin và khóa bí mật được giữ kín để thực hiện việc giải mã thông tin được mã hóa bằng khóa công khai. Loại mã hóa này đặc biệt ở chỗ với khóa công khai, việc giải theo chiều thuận (mã hóa) là dễ dàng trong khi giải theo chiều ngược lại (giải mã) lại rất khó khăn. Chiều ngược lại chỉ thực sự dễ giải khi và chỉ khi có khóa bí mật.

Hiện nay, có hai phương pháp mã hóa công khai phổ biến được sử dụng là RSA và ECC.

Chương B. Hệ mật mã công khai

Mật mã hóa khóa công khai là một dạng mật mã hóa cho phép người sử dụng trao đổi các thông tin mật mà không cần phải trao đổi các khóa chung bí mật trước đó. Điều này thực hiện bằng cách sử dụng một cặp khóa có quan hệ toán học với nhau là khóa công khai và khóa bí mật.

I) Hệ mật mã công khai RSA

Đây là thuật toán đầu tiên phù hợp với việc rạo ra chữ ký điện tử đồng thời với việc mã hóa. Nó đánh dấu một sự tiến bộ vượt bậc của lĩnh vực mật mã học trong việc sử dụng khóa công khai. RSA đang được sử dọng phổ biến trong thương mại điện tử và được cho là đảm bảo an toàn với điều kiện độ dài khóa đủ lớn.

a) Lịch sử

Thuật toán được Ron Rivest, Adi Shamir và Len Adleman mô tả lần đầu vào năm 1977 tại MIT. Tên của thuật toán được ghép từ 3 chữ cái đầu của tên 3 tác giả.

b) Hoạt động

i. Mô tả sơ lược

Sơ đồ chung của hệ mật mã khóa công khai được cho bởi S = (P, C, K, E, D)[(1)]. Trong đó P là tập kí tự bản rõ; C là tập kí tự bản mã; K là tập các khóa gồm 2 phần K = (K', K"), K' là khóa công khai dành cho việc tạo mã, K" là khóa bí mật dành cho việc giải mã; Với mỗi ký tự bản rõ x ϵ P, thuật toán lập mã E cho ta ký tự mã tương ứng y = E(K', x) ϵ C; và với ký tự mã y thuật toán giải mã D sẽ cho ta lại ký tự bản rõ x: D(K", y) = D(K", (K', x)) = x.

Ta có thể mô phỏng trực quan một hệ khóa RSA như sau: Bob muốn gửi cho Alice một thông tin mật mà chỉ Alice có thể đọc được. Để làm được điều này, Alice gửi cho Bob một chiếc hộp có khóa mở sẵn và giữ lại chìa. Bob nhận chiếc hộp và cho thông tin mật vào hộp và khóa lại (như loại khóa chỉ cần sập chốt lại, và sau khi sập thì ngay cả Bob cũng không mở được). Sau đó Bob gửi lại hộp cho Alice. Alice dùng khóa của mình mở hộp và đọc thông tin mật.

ii. Tạo khóa

Để xây dựng hệ mật mã khóa công khai RSA, bước đầu tiên ta cần phải tạo khóa công khai và khóa bí mật theo các bước sau:

1. Chọn 2 số nguyên tố lớn p và q với p≠q, lựa chọn là ngẫu nhiên và độc lập.
2. Tính n = pq.
3. Tính giá trị hàm số Ơle $\emptyset(n) = (p-1)(q-1)$.
4. Chọn ngẫu nhiên một số tự nhiên e thỏa mãn: $1 < e < \emptyset(n)$ và $\gcd(e, \emptyset(n)) = 1$.
5. Tính d thỏa mãn: $de \equiv 1 \ mod \ \emptyset(n)$.

Khi đó, ta được khóa công khai K' = (n,e) và khóa bí mật K" = d.

iii. Mã hóa

Giả sử, Bob muốn gửi đoạn thông tin M cho Alice. Đầu tiên, Bob chuyển M thành một số x < n theo một hàm có thể đảo ngược được thỏa thuận trước.

Lúc này, Bob có m và cũng biết khóa công khai K' = (n,e) do Alice gửi. Bob sẽ tính y ϵ C là bản mã hóa của m theo công thức:

$$y = x^e \ mod \ n$$

Cuối cùng Bob gửi y cho Alice.

iv. Giải mã

Alice nhận c từ Bob và biết khóa bí mật K" = d. Alice có thể tìm được x từ y theo công thức:

$$x = y^d \ mod \ n$$

Biết m, Alice tìm lại M theo phương pháp thỏa thuận trước.

v. Chứng thực

Quá trình giải mã hoạt động vì ta có:

$$y^d mod \ n = (x^e)^d mod \ n = x^{ed} mod \ pq$$

Do $\begin{cases} ed \equiv 1 \ (mod \ p-1) \\ ed \equiv 1 \ (mod \ q-1) \end{cases} => \begin{cases} x^{ed} \equiv x \ mod \ p \\ x^{ed} \equiv x \ mod \ q \end{cases}$ (theo định lý Fermat nhỏ)

$=> x^{ed} \equiv x \; mod \; pq \; hay \; y^d \equiv x \; mod \; n$

vi. Tổng quát

Như vậy, sơ đồ chung của hệ mật RSA được định nghĩa bởi danh sách (1). Trong đó:

1. P = C = Z_n, trong đó n là số nguyên Blum, tức là tích của hai số nguyên tố.
2. K = {K = (K', K"): K' = (n,e); K" = d; gcd(e, Ø(n)) = 1; $de \equiv 1 \; mod \; Ø(n)$}
3. E(K', x) = $x^e \; mod \; n$ với mọi x ϵ P.
4. D(K",y) = $y^d \; mod \; n$ với mọi y ϵ C.

c) An ninh

Độ an toàn của hệ thống RSA dựa trên 2 vấn đề của toán học: bài toán phân tích một số nguyên lớn ra số nguyên tố và bài toán RSA. Nếu 2 bài toán trên là khó (không tìm được giải thuật hiệu quả để giải chúng) thì không thể thực hiện được việc thám mã toàn bộ với RSA.

Bài toán RSA là bài toán tính căn bậc e môđun n (với n là hợp số): tìm số x sao cho $x^{\,e} = c \; mod \; n$, trong đó (e, n) là khóa công khai và c là bản mã. Hiện nay, phương pháp triển vọng nhất là phân tích n ra thừa số nguyên tố. Khi thực hiện điều này, kẻ tấn công sẽ tìm được 2 số nguyên tố p và q. Khi đó sẽ tìm được dễ dàng giá trị Ơle Ø(n) và từ đó tìm được d thông qua e. Chưa có một phương pháp nào được tìm ra trên máy tính để giải bài toán này trong thời gian đa thức. Người ta cũng chưa chứng minh được điều ngược lại (sự không tồn tại của thuật toán).

Tuy nhiên, trong thực tế vẫn tồn tại nhiều sơ hở mà kẻ tấn công có thể tận dụng để đe dọa tính bảo mật của các hệ mã RSA.

II) Hệ mật mã công khai ECC

Nếu RSA là hệ mật mở đầu trong việc sử dụng khóa công khai thì các hệ mật trên đường cong EC đánh dấu một bước tiến mới về tốc độ xử lý và độ an toàn trong việc mã hóa và giải mã.

a) Tổng quan về đường cong Elliptic

i. Tập điểm $E(Z_p)$ trên đường cong Elliptic

Để đơn giản, ta xét đường cong Elliptic trên trường hữu hạn Z_p trong đó p là một số nguyên tố lớn hơn 3. Khi đó, một đường cong Elliptic (E) trên Z_p được xác định bởi phương trình dạng:

$$(E)y^2 = x^3 + ax + b\ (mod\ p)\ ^{(1)}$$

Ở đây a,b thuộc Z_p và $4a^3 + 27b^2 \neq 0$ (mod p), và một điểm đặc biệt O, gọi là điểm vô cực. Tập $E(Z_p)$ bao gồm tất cả các điểm (x,y) với x, y ϵ Z_p thỏa mãn phương trình (1), cùng với điểm vô cực O.

ii. Quy tắc cộng trên các điểm thuộc đường cong Elliptic $E_p(a,b)$

Trên các điểm thuộc $E_p(a,b)$ ta dựa vào phép cộng hai điểm thuộc $E_p(a,b)$. Với phép cộng này, $E_p(a,b)$ trở thành một nhóm cộng. Phép toán được xác định như sau:

- P + O = P với mọi P ϵ $E_p(a,b)$.
- Nếu P = (x,y) ϵ $E_p(a,b)$ thì tồn tại điểm – P = (x,-y) ϵ $E_p(a,b)$ thỏa mãn: P + (-P) = O, -P được gọi là điểm đối của P.
- Cho P = (x_1,y_1) ϵ $E_p(a,b)$ và Q = (x_2,y_2) ϵ $E_p(a,b)$. Ở đây, P ≠ -Q. Khi đó, P + Q = (x_3,y_3), ở đây:

$$x_3 = \lambda^2 - x_1 - x_2$$

$$y_3 = \lambda(x_1 - x_3) - y_1$$

Trong đó:

$$\begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1}, & khi\ P \neq Q \\ \dfrac{3{x_1}^2 + a}{2y_1}, & khi\ P = Q \end{cases}$$

b) Hệ mật Elgamal trên đường cong Elliptic

Hệ mật Elgamal làm việc với nhóm Cyclic hữu hạn. Năm 1978, Kobliz đã đưa một hệ mật ECC dựa trên hệ Elgamal. Cũng như RSA, Alice cần gửi tin nhắn cho Bob để người khác không đọc được. Để làm như vậy, cả hai sử dụng thuật toán sau:

1. Bob thiết lập khóa công khai dựa trên đường cong Elliptic E trên trường hữu hạn F_p sao cho bài toán logarithm rời rạc là khó giải trong $E(F_p)$. Bob chọn $G \epsilon E(F_p)$ làm điểm cơ sở và một số nguyên s làm khóa riêng của Bob. Khi đó, Bob tính B = s.G và công khai tập khóa K'=(E, F_p, P, B).

i.Mã hóa

Alice để gửi bản tin của mình sẽ làm như sau:

2. Downloads khóa công khai K' của Bob.
3. Biểu diễn bản tin m như điểm $M \epsilon E(F_p)$.
4. Chọn ngẫu nhiên số nguyên k bí mật và tính $M_1 = k.P$.
5. Tính $M_2 = M + kB$.
6. Gửi M_1 và M_2 cho Bob.

ii. Giải mã

Bob giải mã bản tin M bằng cách tính: $M = M_2 – s.M_1$

iii. Chứng thực

Bob có thể nhận được M bởi vì:

$M_2 - s.M1 = (M + kB) – s(kP) = M + k(sP) – s(kP) = M$

Do người khác không có khóa riêng s của Bob nên anh ta khó thể tính M. Chỉ có cáh để tìm ra s là giải bải toán logarithm rời rạc dựa trên B và P biết rằng B = sP.

c) An ninh

Độ an toàn của ECC dựa vào bài toán logarithm rời rạc trên nhóm Cyclic các điểm của đường cong Elliptic (ECDLP). Đối với bài toán logarit rời rạc trên trường hữu hạn hoặc bài toán phân tích số, tồn tại các thuật toán dưới dạng dạng hàm mũ để giải các bài toán này (tính chỉ số hoặc sàng trường số). Tuy nhiên, đối với bài toán ECDLP cho đến nay vẫn chưa tìm được thuật toán dưới hàm mũ để giải. Nhà toán học nổi tiếng J. Silverman cũng như nhiều nhà thám mã khác đã nghiên cứu các thuật toán tương tự, như thuật toán tính chỉ số để áp dụng cho ECDLP nhưng đều không thành công. Hiện nay, thuật toán tốt nhất để giải bài toán ECDLP là Pollard với độ phức tạp cỡ, trong đó G là nhóm

điểm đường cong elliptic. Điều này đã tạo ra những ưu việt của hệ mật ECC so với các hệ mật khóa công khai khác.

Mật mã ECC cung cấp tính an toàn tương đương với các hệ mật khóa công khai truyền thống, trong khi độ dài khóa nhỏ hơn nhiều lần. Người ta đã ước lượng rằng cỡ khoá 3248 bit của hệ mật RSA cho cùng một độ an toàn như 256 bit của hệ mật ECC. Điều đó có nghĩa là việc cài đặt ECC sử dụng tài nguyên hệ thống ít hơn, năng lượng tiêu thụ nhỏ hơn.... Với ưu thế về độ dài khóa nhỏ, ECC đang được ứng dụng rộng rãi trong nhiều lĩnh vực.

Chương C. Ứng dụng của hệ mã công khai – Giao thức SSL

Giao thức SSL (Secure Sockets Layer) là giao thức an toàn được sử dụng rộng rãi nhất trên Internet hiện nay. SSL là giao thức dùng để thiết lập bảo mật giữa server và client. SSL mã hoá tất cả dữ liệu truyền qua lại giữa chủ và khách ở mức IP sockets.

SSL thực hiện mã hoá, xác định quyền và bảo vệ thông tin toàn vẹn khi trao đổi dữ liệu trong một môi trường mạng vốn có nhiều người dùng và không an toàn.

Hai thành phần chính của hệ mã khóa công khai là giao thức bắt tay SSLHP (SSL Handshake protocol) và giao thức lớp ghi SSLRP (SSL Record Layer protocol). SSLHP cho phép server và client thực hiện thoả thuận về bộ mã sử dụng, xác nhận lẫn nhau và thiết lập một khoá chính chia sẻ (thường sử dụng các thuật toán mã hoá công khai). SSLRP được xuất phát từ khoá chính chia sẻ và sử dụng chúng cùng với các thuật toán công khai để mã hoá các dữ liệu.

I) Quá trình bắt tay đầy đủ dựa vào thuật toán RSA

Thuật toán RSA là một thuật toán mã hoá công khai được sử dụng khá rộng rãi.(Thuật toán mã hoá công khai là thuật toán mã hoá sử dụng cặp khoá riêng (private key) và khoá công khai (public key) nhằm hai mục đích: khoá riêng dùng để chứng thực còn khoá công khai được dùng để mã hoá dữ liệu).

Đầu tiên, client gửi một bản tin ngẫu nhiên tới server để thoả thuận bộ mật mã hoá thông qua việc trao đổi các bản tin ClientHello và ServerHello.Sau đó, server sẽ gửi lại khoá công khai RSA cho client trong bản tin ServerCertificate. Do client cũng có khoá công khai của server; nó sẽ kiểm tra khoá công khai này. Nếu phù hợp, client sẽ dùng chính khoá công khai này để mã hoá một số ngẫu nhiên 48 byte (còn được gọi là premaster). Số sau khi được mã hoá sẽ được gửi kèm bản tin ClientKeyExchange tới server. Server sẽ dùng khoá mật RSA để giải mã bản tin premaster này. Cả hai phía sẽ sử dụng bản tin premaster để tạo ra bản tin master đồng thời với việc trao đổi bản tin lúc đầu trước đó sẽ được dùng để trao đổi khoá mã hoá; vector khởi đầu và khoá MAC (Message Authentication Code) nhằm phục vụ cho mã hoá và chứng thực bởi SSLRP.

II) Quá trình bắt tay đầy đủ dựa vào mã hoá ECC

Thông qua hai bản tin đầu (được xử lí giống như RSA); client và server thoả thuận bộ khoá mật ECC. Bản tin ServerCertificate chứa khoá ECDH công khai của server được xác nhận bởi một nhà cấp quyền ECDSA. Sau khi đã chứng thực chữ kí ECDSA, client sẽ chuyển khoá chung ECDH của nó tới server thông qua bản tin ClientKeyExchange. Tiếp theo, mỗi bên lại sử dụng khoá riêng ECDH của chính nó và khoá chung của bên kia để thực hiện mã hoá và giải mã ECDH và đạt đến việc chia sẻ một khoá mật premaster secret. Việc kế thừa khoá mật chính và khoá đối xứng không thay đổi so với RSA.

III) Mã hóa công khai trong SSL trong hai chế độ

Hoạt động mã hoá công khai trong SSL được thực hiện bởi một client và server trong các chế độ khác nhau của quá trình bắt tay SSL như sau:

- Bắt tay dựa vào RSA: client thực hiện hai hoạt động mã hoá công khai: một để kiểm chứng việc chứng thực của server và một để mã hoá các khoá mật premaster với khoá công khai của server. Server thực hiện một hoạt động với khoá mật RSA để giải mã bản tin ClientKeyExchange và khôi phục lại premaster.
- Bắt tay dựa vào ECDH-ECDSA: client thực hiện chứng thực ECDSA để kiểm chứng lại chứng thực của server và sau đó thực hiện ECDH bằng việc sử dụng khoá riêng ECDH và khoá chung ECDH của server để tính ra premaster chia sẻ. Tất cả các server cần thực hiện cùng một hoạt động ECDH để thu được mã mật tương ứng.

IV) So sánh

Để thực hiện việc so sánh việc sử dụng RSA và ECC trong quá trình bắt tay của SSL, người ta sử dụng hai bộ mã khác nhau TLS_RSA_WITH_RC4_128_SHA và TLS_ECDH_ECDSA_WITH_RC4_128_SHA. Với mỗi bộ mã người ta sử dụng ba cấp độ an ninh khác nhau. Đối với RSA người ta sử dụng khoá có độ dài 1024 bit, 1536 bit và 1048 bit, Với ECC sử dụng khoá có độ dài 160 bit, 192 bit và 224 bit. Người ta sử dụng một công cụ để thực hiện nhiều phiên giao dịch đồng thời một lúc nhằm đo hai thông số trên một server. Và kết quả được thể hiện ở bảng 1.

Bảng 1: So sánh sử dụng RSA và ECC trong quá trình bắt tay của SSL

	ECC-160	**RSA-1024**	**ECC-192**	**RSA-1536**	**ECC-224**	**RSA-2048**
Thời gian(ms)	3.69	8.75	3.87	27.47	5.12	56.18
Số lệnh thực hiện / s	271.3	114.3	258.1	36.4	195.5	17.8
So sánh thời gian thực hiện	2.4:1		7.1:1		11:1	
So sánh độ dài khoá	1:6.4		1:8		1:9.1	

V) Kết luận

Ta có thể nhận thấy rằng ECC mang lại hiệu quả hơn hẳn so với RSA trên những phương diện như: tốc độ, khả năng tính toán, dải thông đường truyền, hiệu quả lưu trữ,… Các ưu thế trên của ECC có thể phát huy mạnh trong các ứng dụng mà đường truyền, khả năng tính toán, tốc độ hay lưu trữ bị hạn chế. Điều đó chứng tỏ rằng trong một vài năm tới có thể ECC sẽ là giải thuật trao đổi khóa phổ biến nhất trong thời gian dài.