Chương trình KC-01:
Nghiên cứu khoa học
phát triển công nghệ thông tin
và truyền thông

Đề tài KC-01-01:
Nghiên cứu một số vấn đề bảo mật và
an toàn thông tin cho các mạng dùng
giao thức liên mạng máy tính IP

Báo cáo kết quả nghiên cứu

GIỚI THIỆU MỘT SỐ KẾT QUẢ MỚI TRONG BẢO MẬT MẠNG DÙNG GIAO THỨC IP, AN TOÀN MẠNG VÀ THƯƠNG MẠI ĐIỆN TỬ

Quyển 1B: "Nước Nga và chữ ký điện tử số"

HÀ NỘI-2003

Báo cáo kết quả nghiên cứu

GIỚI THIỆU MỘT SỐ KẾT QUẢ MỚI TRONG BẢO MẬT MẠNG DÙNG GIAO THỨC IP, AN TOÀN MẠNG VÀ THƯƠNG MẠI ĐIỆN TỬ

Quyển 1B: "Nước Nga và chữ ký điện tử số"

Chủ trì nhóm nghiên cứu
PGS, TS Hoàng Văn Tảo

Mục lục

Lời giới thiệu

Ngày 10 tháng 1 năm 2002, tổng thống Nga V. Putin đã ký sắc lệnh liên bang về chữ ký điện tử số. Trong chuyến đi công tác tại Cộng hoà Liên bang Nga, chúng tôi đã có được một số tài liệu có liên quan về vấn đề này. Cơ quan FAPSI là nơi chuẩn bị về cả khía cạnh kỹ thuật và pháp lý cho dự luật này. Trong tạp chí " DTR RFXTCNDF" cũng đã đăng bài "An toàn thông tin, thuật ngữ và định nghĩa" của IU.I.Kovalenko, trưởng bộ môn "Công nghệ và tổ chức bảo vệ thông tin" tại "Trung tâm học tập-phương pháp các công nghệ thông tin mới" của FAPSI. Ông Kovalenko cũng là người cung cấp bản dự thảo lần cuối của luật về chữ ký số trước khi đệ trình tổng thống Nga ký.

Để đi tới Luật về chữ ký điện tử số, nước Nga đã có một quá trình chuẩn bị kỹ càng từ trước. Trong tạp chí chuyên ngành về an ninh thông tin "CBCNTVS MTPJGFCYJCNB" số ra tháng 2-3 năm 2001 đã đăng bài viết của 3 chuyên gia FAPSI là tiến sĩ toán-lý A.C. Kuzmin, phó tiến sĩ kỹ thuật A.B. Korolkov và phó tiến sĩ toán-lý N.N. Murasov "Những công nghệ hứa hẹn trong lĩnh vực chữ ký điện tử số".

Về phía người sử dụng, cũng trong tạp chí "CBCNTVS MTPJGFCYJCNB", số ra tháng 8-9 năm 2001 đã đăng bài của các chuyên gia V. Miaxnhiankin và A. Mejutkov "Chữ ký điện tử hay con đường gian khổ thoát khỏi giấy tờ".

Vậy nước Nga đã dùng chuẩn chữ ký số nào? Chúng tôi cũng giới thiệu 2 chuẩn chữ ký số của Nga là GOST P 34.10-94 và GOST P 34.10-2001 cũng như chuẩn hàm băm GOST P.34.11-94. Nhưng do chuẩn hàm băm có sử dụng chuẩn mã khối GOST 24187-89, nên để cho đầy đủ, chúng tôi cũng mô tả cả thuật toán GOST 24187-89.

Cũng nên nhắc lại là tại Liên minh châu Âu, từ ngày 19/7/2001, luật chữ ký điện tử đã có hiệu lực và tại Mỹ là 9 tháng trước đó.

Để tiện so sánh, chúng tôi cũng giới thiệu thuật toán DSS cũng như họ các hàm băm SHA của Mỹ. Chuẩn này đã được công bố ngày 7 tháng 1 năm 2000 để thay cho chuẩn được đưa ra từ nhiều năm trước đây (1994). Chuẩn DSS mới bao gồm 3 thuật toán, trong đó 2 thuật toán đầu chính là chuẩn trước đó.

Chúng tôi cũng giới thiệu bài báo của 2 tác giả người Nga so sánh thuật toán mã khối GOST 24187-89 của Nga và thuật toán Rijndael là thuật toán sẽ được chấp nhận là chuẩn mã dữ liệu mới của Mỹ (AES) thay cho DES. Bên cạnh đó còn có một bài báo của 4 tác giả phương tây viết về chuẩn GOST 24187-89 của Nga.

Hiện nay, chúng tôi chưa có được thông tin về các chuẩn mật mã (mã khối, chữ ký số, hàm băm,..) của EU và các nước ở Châu Á như Hàn Quốc, Nhật Bản, các nước

ở Đông Nam Á như Singapore, Thái Lan. Nhưng thiết nghĩ việc tham khảo các chuẩn của hai nước Mỹ và Nga là quan trọng nhất.

Các công nghệ hứa hẹn trong lĩnh vực chữ ký điện tử số[1]

A.C. Kuzmin, tiến sĩ toán-lý
A.B. Korolkov, phó tiến sĩ kỹ thuật
N.N. Murasov, phó tiến sĩ toán-lý

Sự phát triển bùng nổ của hệ thống xử lý văn bản điện tử, thanh toán điện tử, thư tín điện tử, việc lan truyền rộng rãi các mạng máy tính lớn với số người sử dụng đông đã đặt ra bài toán tìm một công cụ tương tác giữa con người với sự trợ giúp của các phương tiện điện tử, qua đó, những người sử dụng không quen biết nhau có thể tin tưởng trao đổi thông tin với nhau, xác định được chính xác nguồn gốc thông tin đã được nhận theo các kênh điện tử, còn người phát tin không thể phủ nhận được trách nhiệm là tác giả.

Các phương pháp mật mã

Bài toán này được giải thành công với sự trợ giúp của các phương pháp mật mã. Các phương pháp mật mã có thể chia một cách qui ước ra làm 2 lớp. Lớp thứ nhất, đó là các phương pháp được thực hiện với sự trợ giúp của các thuật toán mã đối xứng. Để mã và giải mã nó trong các hệ thống với thuật toán mã đối xứng, người ta sử dụng cùng một khoá. Người gửi và người nhận giữa chìa khóa bí mật, không có chìa thì không thể đọc được thông tin (tất nhiên, nếu thuật toán mật mã là bền vững).

Lớp thứ hai, đó là các phương pháp được thực hiện với sự trợ giúp của các thuật toán phi đối xứng. Đặc điểm của các thuật toán này là để mã thông tin người ta sử dụng một khoá, để giải mã người ta sử dụng một khoá khác, nhận được bằng một cách đặc biệt từ khoá thứ nhất.

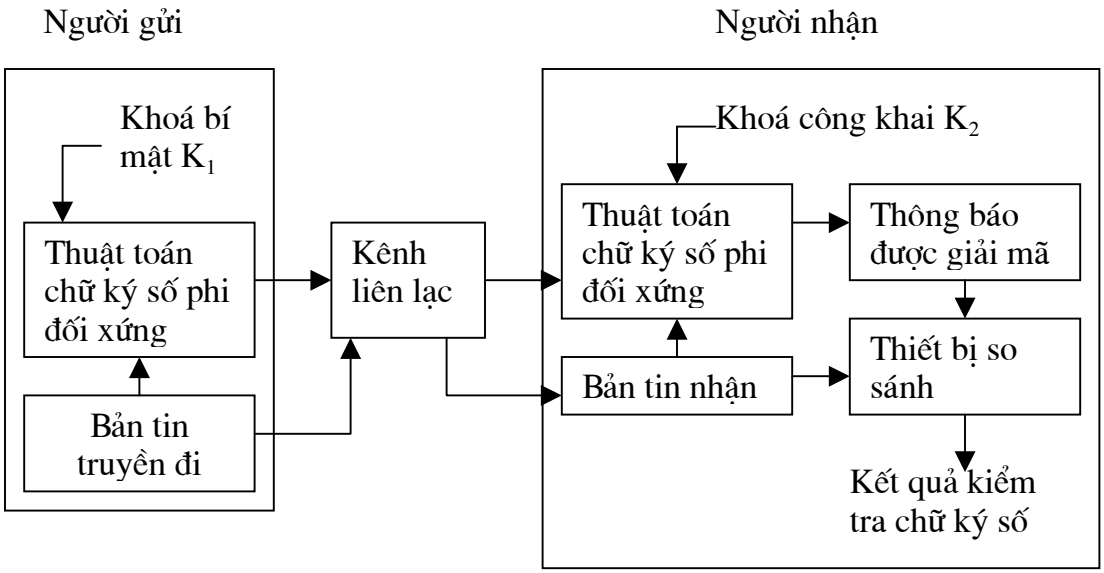
Theo các nguyên tắc của mật mã phi đối xứng mà các thuật toán chữ ký điện tử số được xây dựng. Khi đó, khoá thứ nhất (khoá dùng để mã) là "khoá bí mật", nó chỉ được biết bởi người ký, còn khoá thứ hai (khoá được sử dụng để giải mã), được gọi là "khoá không bí mật" (người ta còn gọi là "khoá công khai"), nó được biết bởi bất kỳ người nhận thông tin nào. Khoá công khai được người nhận thông tin sử dụng để kiểm tra tính chân thực của chữ ký điện tử số. Tất nhiên, thuật toán và khoá cần được chọn sao cho khi biết khoá công khai không thể nói gì về khoá bí mật.

Quá trình truyền thông tin có sử dụng chữ ký điện tử số

Quá trình truyền thông tin cùng với việc sử dụng chữ ký điện tử số như ở hình 1. Người gửi thông tin sử dụng khoá bí mật và một thuật toán phi đối xứng (thuật

[1] Tạp chí "CBCNTVS MTPJGFCYJCNB" số ra tháng 2-3 năm 2001, trang 72-75

toán chữ ký số) đã được chọn trước theo thoả thuận giữa những người sử dụng mã thông tin được truyền đi, biểu diễn nó ở dạng số. Sau đó người gửi thông tin theo một kênh liên lạc công khai gửi thông tin chưa được mã và chữ ký số nhận được bằng phương pháp mô tả ở trên.



Nguyên tắc làm việc của chữ ký số với bản tin ngắn

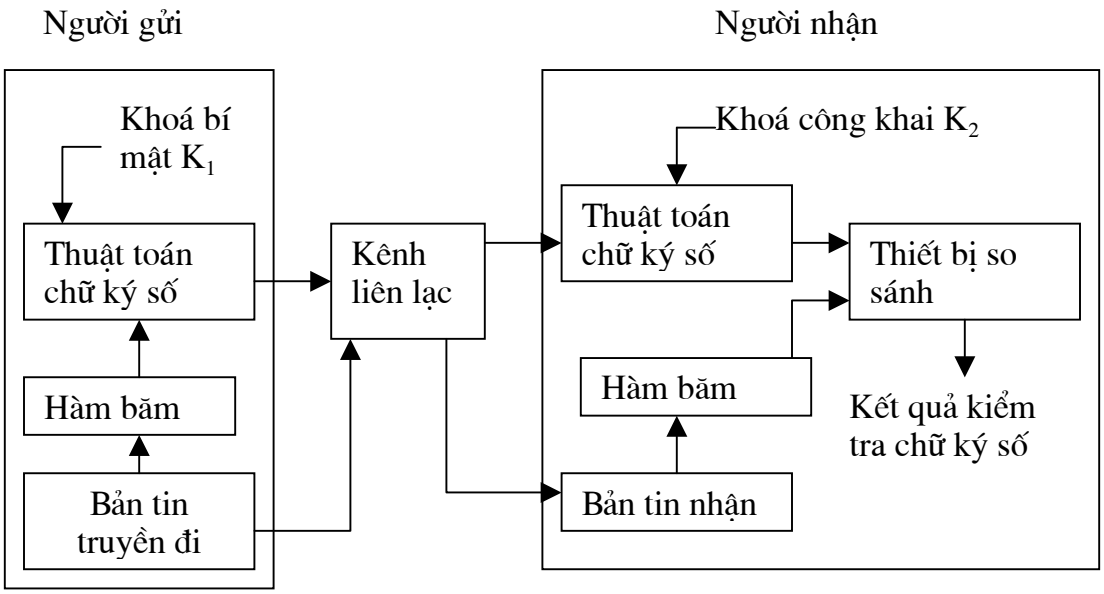
Người nhận bản tin với sự trợ giúp của khoá công khai và thuật toán chữ ký số đã được chọn trước theo thoả thuận giữa những người sử dụng giải mã chữ ký số. Sau đó anh ta so sánh thông tin không mã với thông tin nhận được khi giải mã chữ ký số. Nếu chữ ký số không bị làm giả và thông tin truyền theo kênh công khai không bị thay đổi thì chúng cần phải trùng với nhau hoàn toàn. Nếu chữ ký số là giả mạo thì hai thông tin đó sẽ khác nhau rất nhiều.

Phương pháp kiểm tra chữ ký số như vậy sẽ dẫn đến việc tăng gấp đôi độ dài bản tin cần truyền và được áp dụng chỉ trong trường hợp, nếu như thông tin được truyền ở dạng số có độ dài ngắn.

Trong trường hợp thông tin truyền đi có độ dài lớn, trước khi truyền nó được nén lại, có nghĩa là cần thực hiện một biến đổi toán học, được gọi là thuật toán băm.

Sau đó người gửi mã phiên bản băm của thông báo cùng với sự trợ giúp của khoá bí mật, giống như đã mô tả ở trên cho trường hợp thông báo ngắn, tức là tạo chữ ký số. Chữ ký số cùng với thông tin rõ được truyền cho người nhận. Tại đầu nhận, người nhận có được thông tin rõ và cả chữ ký số, đó là bản mã phiên bản băm của thông báo rõ. Biết khoá công khai và thuật toán mã (thuật toán chữ ký số), người nhận giải mã chữ ký số, đó chính là việc khôi phục phiên bản băm của thông báo

rõ. Do hàm băm cũng được người nhận biết từ trước, nên anh ta cũng có thể tạo ra được phiên bản băm của thông báo rõ mà chỉ cần sử dụng thông tin rõ mà anh ta đã nhận được. Bằng cách đó, người nhận có được 2 phiên bản băm của thông báo rõ (một nhận được khi giải mã chữ ký số, một có được khi băm thông tin rõ nhận được). Trong trường hợp trùng hoàn toàn 2 phiên bản đó có thể coi rằng thông tin nhận được không bị thay đổi và được truyền đi bởi người gửi cụ thể có khoá bí mật tương ứng với khoá công khai đã được dùng để giải mã chữ ký số.



Nguyên tắc làm việc của chữ ký số với việc băm bản tin

Các phương thức thể hiện chương trình

Đối với thuật toán chữ ký số cũng tồn tại nhiều phương pháp lập trình nó. Trong toán học cũng tồn tại nhiều hàm băm khác nhau. Với mục đích đảm bảo khả năng trao đổi thông tin giữa những người sử dụng cần phải củng cố một thuật toán chữ ký số duy nhất và một hàm băm, thoả mãn đủ các yêu cầu đã nêu ra ở trên. Trong các nước phát triển đã có thực tế việc đưa ra thuật toán chữ ký số và hàm băm ở dạng chuẩn quốc gia. Các chuẩn như vậy cũng tồn tại ở Liên bang Nga. Thuật toán chữ ký số và hàm băm đã được chọn là kết quả một công việc lớn của những nhà mật mã học thuộc các tổ chức chuyên ngành khác nhau, trong đó có các nhà mật mã học của FAPSI.

Từ lịch sử chuẩn của nước Nga

Chuẩn đầu tiên về chữ ký số được chấp nhận năm 1994 và đang hoạt động hiện nay thuộc về GOST P34.10-94 "Công nghệ thông tin. Bảo vệ thông tin bằng mật mã. Các thủ tục sinh và kiểm tra chữ ký điện tử số trên cơ sở thuật toán mật mã phi đối xứng". Trong chuẩn đó các thủ tục đã được chỉ ra là một phương án của lược đồ chữ ký số ElGamal, nó có tên của nhà toán học người Mỹ (gốc Ảrập) đã lập ra nó

năm 1984. Việc không thể giả mạo chữ ký trong trường hợp này được dựa trên độ phức tạp của việc giải bài toán logarithm rời rạc trong trường có **p** phần tử, tức là việc tìm theo một số nguyên tố lớn p cho trước và các số a, b một số x trong khoảng từ 2 đến p-1 sao cho $a^x = b \bmod p$.

Chuẩn gán cho p và a các giá trị sao cho độ dài ở dạng nhị của p là 512 hoặc 1024, còn bậc của phần tử a theo modulo p bằng một số nguyên tố q có 256 bit. Điều đó có nghĩa rằng q là số nhỏ nhất sao cho $a^q=1 \bmod p$. Các số p, q, a và b là công khai, còn x là khoá bí mật để ký. Hiện nay, để bảo vệ thông tin mật người ta thường sử dụng phương án chữ ký số với modulo nguyên tố p có 512 bit.

Thuật toán tính hàm băm được thiết lập bởi GOST P34.11-94 "Công nghệ thông tin. Bảo vệ thông tin bằng mật mã. Hàm băm", chuẩn này được chấp nhận đồng thời với chuẩn chữ ký số.

Cũng cần nói rằng, vào năm 1994 tại Mỹ cũng công nhận chuẩn chứ ký số (FIPS 186), trong số các lược đồ cho phép có một phương án khác của lược đồ ElGamal đã được nhắc đến.

Phương pháp sàng trường số

Trong 20 năm gần đây, bằng nỗ lực của các chuyên gia trong và ngoài nước đã đạt được tiến bộ đáng kể trong việc phát triển các phương pháp giải bài toán tính logarit rời rạc trong trường có p phần tử. Thành tựu cuối cùng trong lĩnh vực này là phương pháp sàng trường số, sử dụng công cụ toán học của lý thuyết số đại số. Các đánh giá lý thuyết và các kết quả thực nghiệm nhận được bằng cách sử dụng phương pháp vừa nêu, cũng như xu hướng phát triển kỹ thuật tính toán đã không cho phép trong một viễn cảnh dài hạn xem xét lược đồ chữ ký số P34.10-94 (ít nhất là trong trường hợp modulo p có 512 bit) như là một thuật toán mật mã tin cậy.

Dường như, lời giải đơn giản nhất trong trường hợp này là tăng độ dài cho phép của modulo nguyên tố p, tức là một cách tương ứng tăng độ phức tạp của việc tính logarit rời rạc và làm khó hơn việc giả mạo chữ ký. Thế nhưng việc thay đổi như vậy làm giảm đáng kể chất lượng khai thác thuật toán, bởi vì như thế sẽ tăng độ dài của khoá công khai, tức là cả thời gian tính và thời gian kiểm tra chữ ký. Mặt khác, một ưu điểm không nghi ngờ gì của lược đồ chữ ký số kiểu El-Gamal là khả năng (mang tính nguyên tắc) thể hiện nó không chỉ trong trường hữu hạn, mà cả ở các cấu trúc toán học khác, nơi có thể định nghĩa các tương tự của phép nhân và phép nâng lên luỹ thừa. Tất nhiên, cấu trúc này cần phải có các tính chất sau: phép tính logarit rời rạc trong nó, tức là tìm chỉ số mũ theo cơ số và kết quả nâng lên luỹ thừa với số mũ cần tìm, là một bài toán khó, còn chính phép nâng lên luỹ thừa có thể thực hiện tương đối nhanh. Phân tích kinh nghiệm kiến thiết và đánh giá các thuật toán mật mã trong và ngoài nước chỉ ra rằng, có thể chọn đường cong eliptic trên trường hữu hạn có p phần tử như là một cấu trúc như vậy, đó là tập các cặp số thoả mãn đồng dư thức $y^2 = x^3 + ax + b \bmod p$, với a và b cố định và thoả mãn thêm nào

đó, cho phép định nghĩa trên đường cong các phép toán tương tự như phép toán nhân và nâng lên luỹ thừa theo modulo p. Các số a và b được gọi là các hệ số của đường cong eliptic.

Dự án chuẩn quốc gia mới về chữ ký điện tử

Những điều nói trên đây là động cơ để thực hiện trong năm 2000 bởi các chuyên gia của FAPSI một dự án chuẩn quốc gia mới về chữ ký điện tử số. Dự án này xác định tiến trình lập và kiểm tra chữ ký trên cơ sở toán học của các đường cong eliptic. Chúng ta hãy nhắc tới một số đặc điểm cơ bản của lược đồ chữ ký điện tử số được đề nghị trong dự án.

1. Tính chấp nhận được tối đa trong quan hệ với chuẩn thực tại. Thứ nhất, lược đồ được đề nghị cũng là một phương án của lược đồ chữ ký El-Gamal, được sửa chữa thay cho các phép toán nhân và nâng lên luỹ thừa trong trường hữu hạn có p phần tử là các phép toán tương tự trên đường cong eliptic được xác định trên trường này. Thứ hai, nó cho phép sử dụng hàm băm đang có ở trong chuẩn hiện tại. Thứ ba, độ dài chữ ký trong dự án chuẩn mới không đổi so với trước. Các đặc tính đã được chỉ ra của lược đồ trong trường hợp chấp nhận nó như là chuẩn quốc gia mới sẽ làm dễ đáng kể việc thay đổi tương ứng các phương tiện chương trình cũng như thiết bị thực hiện việc sinh và kiểm tra chữ ký được xác định bởi chuẩn hiện tại.
2. Có các tính chất mật mã tốt, đảm bảo rằng khi giữ kín được khoá bí mật để ký thì không thể làm giả được chữ ký trong vòng nhiều chục năm ngay cả với sự tiến bộ của kỹ thuật tính toán và các thuật toán toán học tương ứng. Để so sánh, xin nói rằng nếu trong chuẩn chữ ký số đang được dùng lấy số nguyên tố làm modulo p có độ dài bằng 2048 bit thì độ phức tạp làm giả chữ ký như vậy cũng vẫn còn ít hơn hàng nghìn lần so với độ phức tạp làm giả chữ ký trong lược đồ mới.
3. Khả năng thực hiện với tốc độ cao thủ tục sinh và kiểm tra chữ ký trên các loại máy tính khác nhau. Dự án chuẩn xác định các công thức toán học cho các phép toán trên đường cong eliptic, tương tự như phép nhân và nâng lên luỹ thừa, nhưng không chỉ ra cụ thể các thuật toán nào được dùng để thực hiện các phép toán ấy. Kinh nghiệm của FAPSI chỉ ra khả năng xây dựng những thuật toán hiệu quả, cho phép, ví dụ, trên một máy tính cá nhân thông thường, thực hiện việc sinh và kiểm tra chữ ký số theo lược đồ mới với tốc độ gần như chuẩn đang dùng với độ dài số nguyên tố modulo p bằng 1024 bit. Trong khi đó, như đã nói ở trên, lược đồ mới có độ bền vững mật mã tốt hơn nhiều.

Các sửa đổi có thể

Việc sử dụng trong lược đồ chữ ký số công cụ toán học mới đề xuất sự cần thiết của các thủ tục hiệu quả mới cho phép sinh ra các tham số của lược đồ, trước hết là modulo p và các hệ số của đường cong eliptic. Dự án cũng đặt ra cho các tham số này các điều kiện dễ kiểm tra nhưng đảm bảo các tính chất mật mã cần thiết cho lược đồ. Điều này cho phép những người tạo ra các phương tiện chữ ký số có thể tự

lập trình các thủ tục cần thiết và làm cho chúng thích nghi với các kỹ thuật tính toán mà họ có trong tay. Vì các chỉ số chất lượng của các thiết bị chữ ký số nhìn chung phụ thuộc phần lớn vào chất lượng thể hiện các thủ tục đó, cho nên khuyến cáo người sử dụng dùng những phương tiện thực hiện chữ ký số có giấy chứng nhận của FAPSI.

Còn một ưu điểm nữa của việc sử dụng trong dự án chuẩn mới về chữ ký số lược đồ chữ ký ElGamal là tính trong sáng về mặt bản quyền, khác với lược đồ chữ ký Shnorr, được bảo vệ bởi bằng sáng chế của hàng loạt nước, trong đó có Mỹ. Việc sử dụng thuật toán được đăng ký sáng chế ở nước ngoài làm cơ sở của chuẩn quốc gia trong lĩnh vực an toàn thông tin sẽ kéo theo những vấn đề pháp lý và tài chính đủ nghiêm túc.

Tính đúng đắn của hướng đã chọn để nghiên cứu trong lĩnh vực tạo ra chuẩn chữ ký số mới một cách gián tiếp được khẳng định bởi sự kiện tại Mỹ đã chấp nhận chuẩn chữ ký số mới vào năm 2000, đó là FIPS 186-2, cho phép bên cạnh các thuật toán chữ ký số đã được chuẩn hoá trước đây trong chuẩn FIPS 186-1 một thuật toán mới dựa trên đường cong elipptic. Như đã biết, tại Mỹ, người ta dành nhiều sự chú ý cho việc tạo ra, đánh giá và chuẩn hoá các thuật toán mật mã.

Các phương tiện kỹ thuật thể hiện chữ ký điện tử số

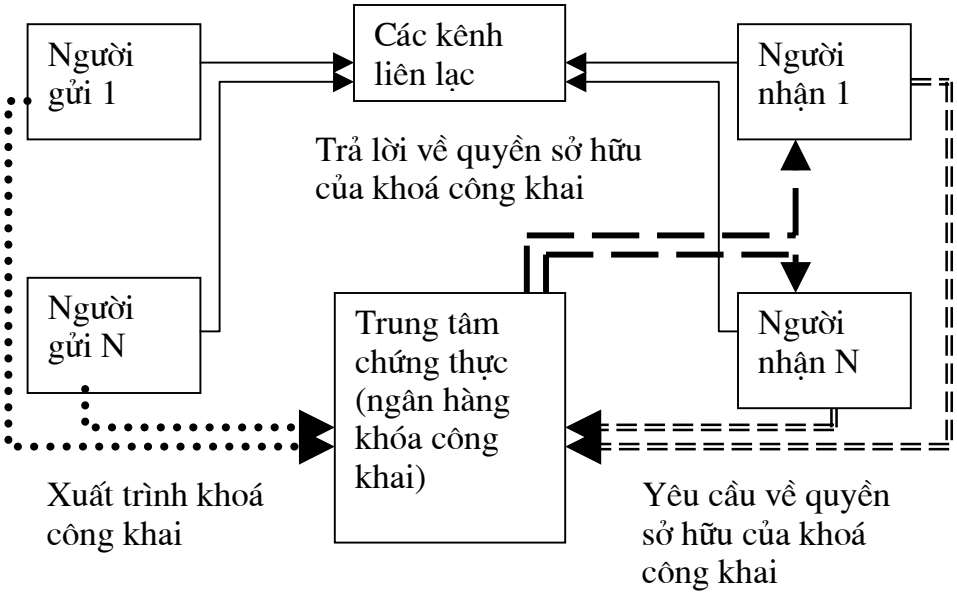
Việc thực hiện các biến đổi toán học phức tạp như đã nói đến trên đây (mã thông tin, băm nó, khẳng định tính chân thực của chữ ký số, chuẩn bị chìa khoá), cần phải được thực hiện trong một thời gian tương đối ngắn và được thực hành hoặc bằng chương trình phần mềm hoặc bằng một thiết bị lập trình được, chúng có tên chung là "phương tiện chữ ký số".

Rõ ràng, người sử dụng cần những đảm bảo chắc chắn rằng các thiết bị mà họ sử dụng có các chất lượng cần thiết đặc biệt, không những có thể đảm bảo việc tính và kiểm tra đúng chữ ký, mà còn có khả năng cần thiết bảo vệ thông tin về khoá bí mật trong quá trình vận hành phương tiện chữ ký số. Những chất lượng này chỉ có ở những thiết bị chữ ký số đã qua một cuộc thẩm định đặc biệt và có các kết quả tốt đối với các yêu cầu được đặt ra, các phương tiện này cần phải được cấp phép theo như luật pháp Liên bang Nga như đã từng làm với những phương tiện mật mã truyền thống.

Khoá cho chữ ký điện tử số

Bây giờ dành riêng nói về khoá. Như đã nói trên đây, người mà ký văn bản cần phải có khoá "bí mật" và chỉ anh ta biết khoá này; để tránh khả năng làm giả chữ ký số, khoá này, cũng như bất kỳ một khoá mã nào, cần phải thoả mãn các yêu cầu đã được công nhận về mặt mật mã. Nói riêng, cần phải loại bỏ khả năng vét cạn khoá. Trong mật mã hiện đại để chuẩn bị khoá, một thiết bị chuyên dụng được dùng, cho phép tạo khoá với xác xuất chọn ngẫu nhiên đúng khoá vào khoảng 10^{-70}-10^{-80}, về thực tế là không có thể.

Mỗi khoá bí mật tương ứng với một khoá công khai, người nhận thông tin phải sử dụng khoá công khai. Khoá công khai tương ứng với một khoá bí mật cụ thể được sinh bởi người gửi thông tin bằng một phần mềm đặc biệt trước đó và được gửi cho tất cả mọi người từ trước. Người sử dụng dùng khoá công khai để kiểm tra chữ ký của những người khác trong hệ thống, anh ta cần phải biết chắc rằng khoá nào trong số công khai thuộc về người nào. Trong trường hợp sai sót tại giai đoạn vận hành này của chữ ký số có thể xác định không đúng nguồn gốc thông tin cùng với tất cả những hậu quả sinh ra từ đó. Quan trọng là thông tin về tính sở hữu của khoá bí mật đối với người sử dụng cần phải được thể hiện văn bản hoá, và thể hiện đó cần phải được thực hiện bởi một cơ quan được phân công trách nhiệm.



Hình 3. Mạng liên lạc văn bản

Văn bản bảo đảm tính sở hữu của khoá công khai có cái tên là chứng nhận khoá công khai chữ ký số. Nó khẳng định tính sở hữu của khoá công khai đối với người chủ của khoá bí mật. Văn bản như vậy được phát hành bởi trung tâm chứng thực khoá công khai chữ ký số.

Sự hiện diện của một văn bản như vậy rất quan trọng khi giải quyết tranh cãi về việc tạo ra một tài liệu này hay tài liệu khác bởi một người cụ thể. Với mục đích loại bỏ khả năng đưa những thay đổi vào trong giấy chứng nhận khoá từ phía người sử dụng, trong khi truyền theo kênh liên lạc giấy chứng nhận ở dạng dữ liệu điện tử được ký bởi trung tâm chứng thực.

Như vậy, trung tâm chứng thực thực hiện các chức năng một công chứng viên điện tử, nó phải khẳng định tính chính thống của một văn bản điện tử được ký. Vì thế

một công chứng viên như vậy cần phải thực hiện các chức năng của mình trên cơ sở giấy phép được cấp bởi cơ quan nhà nước.

Cần phải nhấn mạnh rằng, việc thể hiện đúng đắn thuật toán chữ ký số là phương tiện mạng bảo vệ văn bản điện tử khỏi giả mạo, với việc sử dụng các cơ chế mật mã khác nữa sẽ bảo vệ các văn bản này khỏi việc truy cập trái phép.

Kết luận

Hiện tại đã hội đủ các điều kiện thuận lợi để giải quyết bài toán ứng dụng và sử dụng các hệ thống dựa vào chữ ký điện tử số. Tình hình hiện nay là, cùng một lúc, tại Đu-ma quốc gia Nga đang thảo luận dự luật "Về chữ ký điện tử số", tại Cơ quan chuẩn quốc gia Nga đang thực hiện những thủ tục cần thiết để chấp nhận và chuẩn bị đưa vào sử dụng chuẩn chữ ký điện tử số mới. Như vậy, chúng ta có điều kiện thuận lợi để nhận được cách giải quyết hỗ trợ lẫn nhau của các vấn đề pháp lý và tổ chức có liên quan đến việc áp dụng chữ ký điện tử số trong việc xử lý văn bản cũng như các vấn đề kỹ thuật, chất lượng có liên quan đến các tính chất mật mã của thủ tục chữ ký điện tử.

Chữ ký điện tử hay
con đường gian khổ tránh khỏi giấy tờ[2]

Tóm tắt nội dung: Chúng ta muốn gì ở chữ ký điện tử số và nó hơn chữ ký bình thường ở điểm gì? Trước hết, đó là khả năng nhận diện tính sở hữu của chữ ký trên cơ cở các chỉ số khách quan. Thứ hai, khả năng bảo vệ cao khỏi bị làm giả. Và thứ ba, mối liên quan mật thiết với văn bản được ký. Nếu như hai yêu cầu đầu còn có thể được thể hiện đối với chữ ký thông thường, thì việc thực hiện yêu cầu thứ ba chỉ có thể trong trường hợp áp dụng chữ ký điện tử số. Vấn đề áp dụng các tương tự khác của chữ ký tay không được xét đến ở đây.

Việc thực hiện cả 3 yêu cầu trở nên có thể là do chính nguồn gốc của chữ ký số, đó là một số đủ dài, nhận được như kết quả biến đổi thể hiện điện tử của văn bản cần phải bảo vệ cùng với việc sử dụng khoá bí mật (cá nhân) của người gửi. Một người bất kỳ có thể kiểm tra chữ ký điện tử số ở dưới văn bản với sự giúp đỡ của các biến đổi tương ứng với việc sử dụng một lần nữa thể hiện điện tử của văn bản, khoá công khai của người gửi và chính giá trị của chữ ký số. Khoá công khai và khoá bí mật liên quan với nhau từng đôi một, tuy nhiên không thể tính được khoá bí mật từ khoá công khai. Chính xác hơn, nếu phát biểu một cách thật chính xác, thì cho đến thời điểm hiện nay chưa tìm được thuật toán cho phép thực hiện các tính toán trong một thời gian chấp nhận được với trù tính mức độ phát triển hiện đại của kỹ thuật tính toán và độ dài của khoá được sử dụng.

Khoá bí mật được lưu giữ kín và chỉ biết bởi người chủ của nó, không một ai, trừ người chủ, có thể tạo ra chữ ký điện tử số dưới văn bản. Nhưng bất kỳ một ai cũng có thể kiểm tra (với sự giúp đỡ của khoá công khai đến tất cả mọi người), rằng văn bản được ký chính bởi người chủ và văn bản không bị thay đổi (vì giá trị của chữ ký số phụ thuộc vào nội dụng văn bản). Hệ quả lôgic là ở chỗ, việc đem chữ ký số từ một văn bản này sang một văn bản khác (theo tương tự việc quét hay tô lại chữ ký tay trên văn bản giấy hoặc sử dụng máy fax) là không thể. Như vậy, có thể nói rằng chữ ký số là một cái được đính kèm với từng văn bản điện tử cụ thể.

Tất nhiên, chỉ có thể nói về điều đó một cách chắc chắn, nếu như chính bản thân người chủ tạo ra chìa khoá hoặc là do trung tâm chứng thực sinh ra với sự có mặt của người chủ (khi người chủ không có những kỹ thuật cần thiết). Điều này gây ra một băn khoăn trong thực tế, có trong một số hệ thống, khi mà ***khoá được tạo ra từ trước*** bởi người tổ chức hệ thống, sau đó được phát cho người sử dụng.

---

[2] Đây là bản dịch bài báo của 2 tác giả V. Miaxnhiankin (chuyên gia bảo vệ viễn thông và mật mã của ngân hàng "Ngân khố phương Bắc", thành phố Ekaterinburg) và A. Mejutkov (trưởng phòng dịch vụ tự động hoá của Trung tâm thương mại thế giới và khách sạn Atrium Palace, thành phố Ekaterinburg) trên tạp chí tiếng Nga "Các hệ thống an ninh" số 40, tháng 8-9, năm 2001.

Cái gì ngăn cản việc thoát khỏi giấy và đồng loạt tiến tới trao đổi văn bản không dùng giấy

Như thông thường, các vấn đề ứng dụng và thực hiện ngăn cản chúng ta, điều này thấy rõ trên phương diện tổng thể.

Như đã biết, tất cả các hệ mã phi đối xứng ở nguyên dạng không chống lại được tấn công "người đứng giữa". Các khoá công khai của tất cả mọi người tham gia cần phải truy cập được bởi bất cứ ai để có thể kiểm tra được chữ ký sô. Có nghĩa là chúng có thể để ở trên máy chủ, truyền theo radio hay viết ở trên tường rào hoặc công bố ở mục rao vặt trên các báo. Nhưng cái gì đảm bảo rằng, khoá cùng với nhận dạng "Vaxia Pupkin" đúng là thuộc về Pupkin Vaxili Xtepanovic, sinh năm 19xx, có hộ chiếu số yyy được ai đó ký? Thực là không có gì cả. Không có ai ngăn cản việc tấn công ác ý của Surik Mrachnoukhov bằng cách sinh ra cặp khoá của mình cùng với nhận dạng của Vaxia Pupkin và đặt khoá công khai vào máy chủ công cộng với cái tên Vaxia. Bây giờ, Surik chặn bắt bản tin do Vaxia ký thông báo về việc anh ta bán tủ Tiệp và hãy gọi điện cho anh ta từ 17 đến 19 giờ, loại bỏ chữ ký số của Vaxia, thay đổi thông báo (gọi điện từ 1 giờ đêm đến 5 giờ sáng) và tạo chữ ký số mới bằng khoá bí mật trong cặp khoá của mình. Sau đó làm cho thông báo được mọi người biết (chúng ta giả thiết rằng Surik là người ác ý đến mức có khả năng lấy từ kênh thông tin những thông báo đi ra từ Vaxia). Kết quả là các thành viên của hội điện tử, khi nhận được thông báo thì hoàn toàn tin rằng nó do Vaxia viết, bởi vì việc kiểm tra chữ ký số thành công (nhắc lại rằng, trong thực tế khoá công khai nằm ở trên máy chủ). Có hai cách giải quyết có thể. Thứ nhất, Vaxia trực tiếp chuyển khoá công khai của mình tới tất cả những ai mà mình định trao đổi thư từ. Rõ ràng là điều này khó chấp nhận (không thể trực tiếp gặp tất cả được, không thể thấy trước tất cả các địa chỉ cần thiết). Cách giải quyết thứ hai là thành lập một trung tâm chứng thực. Tại vị trí đó chọn một người được tất cả mọi người tin cậy, ai cũng có thể gặp người ở vị trí trung tâm đó ít nhất một lần trực tiếp hoặc có một kênh liên lạc tin cậy (tức là không che phép thay đổi/giả mạo). Sau khi chọn được một người như vậy, mọi người tham gia vào hệ thống sinh ra cặp khoá của mình, xếp hàng ở trung tâm chứng thực, tại đó sẽ thu một lệ phí nào đó để chứng nhận về thông tin cá nhân của người đến và ký vào khoá công khai của anh ta bằng khoá bí mật của mình. Ngoài chính khoá công khai ra, trong các thông tin được ký còn có: tên của người chủ, các thông tin nhận dạng khác, thời hạn có hiệu lực của khoá, danh sách các hệ thống thông tin cho phép sử dụng chứng nhận và các thông tin khác (xem chuẩn X.509 hay điều 6 của dự luật "Về chữ ký điện tử số"). Tất cả những cái đó (khoá công khai, dữ liệu nhận dạng và chữ ký số) được gọi là giấy chứng nhận khoá công khai. Người chủ may mắn của khoá cầm trên tay giấy chứng nhận và ***khoá công khai của trung tâm.*** Bây giờ anh ta thuần tuý là may mắn- trung tâm chứng nhận là khoá đó đúng là thuộc về anh ta (vì vậy trong dự luật về chữ ký điện tử số những trung tâm này được gọi là trung tâm chứng thực). Vì những người tham gia khác của hệ thống cũng nhận được chứng nhận cùng với bản sao khoá công khai của trung tâm (nhận trực tiếp là kênh an toàn nhất), cho nên họ có thể tin cậy vào tính sở hữu của một khoá công khai

bất kỳ, bởi vì bây giờ khi thiết lập liên lạc, các thuê bao không trao đổi với nhau chỉ có khoá công khai nữa, mà là các giấy chứng nhận. Như vậy, vấn đề tổ chức được thêm vào cơ chế toán học gần như đã chặt chẽ của chữ ký điện tử sô. Thế là mọi vấn đề được giải quyết? Hãy đừng vội.

Ai có lỗi? và Làm gì?

Cơ sở pháp lý của việc sử dụng những cái tương tự với chữ ký viết tay (chữ ký điện tử số là một dạng như vậy) được công bố trong Luật dân sự. Tất nhiên, các công ty uy tín và nhà băng ký kết với nhau các thoả thuận tương ứng, trong đó các bên công nhận với nhau rằng các văn bản được ký bằng chữ ký số cũng có giá trị pháp lý giống như những văn bản trên giấy được ký bằng chữ ký thông thường và được đảm bảo bằng con dấu. Trong thoả thuận này các bên xác định rằng với sự trợ giúp của một phần mềm nào đó hay các thiết bị nào đó sẽ sinh ra các chữ ký số, quy tắc sử dụng nó (các biện pháp an toàn về tổ chức và kỹ thuật) và cái chính là quy định để giải quyết các tình huống tranh chấp.

Các tình huống tranh chấp có thể xảy ra đối với chữ ký số không nhiều:

- khoái thác khỏi trách nhiệm tác giả của thông báo (tôi không viết/tôi không gửi)
- khoái thác việc đã nhận thông tin (tôi chưa nhận được cái đó)
- tranh cãi về thời điểm nhận/gửi thông tin.

Việc xuất hiện của 2 tình huống sau được ngăn chặn từ đầu bởi việc xây dựng có chủ định giao thức trao đổi thông tin giữa 2 người dùng. Thứ nhất, trước khi ký mỗi thông báo người ta đều gắn dấu thời gian vào. Thứ hai, khi nhận mỗi thông báo người nhận phải gửi đi một khẳng định được ký bằng chữ ký số của mình. Về phía mình, khi nhận được khẳng định, người nhận lại gửi đi một biên nhận có kèm theo chữ ký số. Như vậy, mỗi khi có việc trao đổi thông tin thì xảy ra 3 lần gửi, tất nhiên, như thế hơi dư thừa, nhưng tránh được các vấn đề phức tạp nêu ở trên (tất nhiên, cả hai bên đều phải ghi nhật ký trong khoảng thời gian hội thoại về việc nhận/gửi cùng với chữ ký số). Trong nhiều trường hợp, việc trao đổi 3 bước như vậy cho phép dễ dàng giải quyết cả tình huống khoái thác về bản quyền tác giả của thông tin. Tình huống này cũng phải được xem xét trong thoả thuận để tránh khỏi việc hiểu lầm, cần phải viết ra theo từng bước: uỷ ban được thành lập như thế nào (thời hạn, số thành viên từ mỗi bên, sự cần thiết phải mời các chuyên gia độc lập), quy tắc thiết lập (sao từ bản mẫu) các phương tiện để thực hiện kiểm tra, các dấu hiệu hình thức theo đó thực hiện việc kiểm tra, quy định thiết lập kết quả. Không quên việc cần thiết phải lưu các bản sao giấy chứng nhận khoá công khai tại trung tâm chứng thực trong vòng một thời hạn nhất định được xác định theo thoả thuận của những người tham gia trao đổi thông tin. Tất nhiên, thời hạn lưu giữ không được ít hơn thời hiệu kiện tụng được xác định bằng Luật dân sự hoặc các điều luật khác đối với dạng quan hệ thoả thuận đang xét.

Phán xét "các vụ việc" được chuyển cho trọng tài

Như vậy, các hãng và ngân hàng quyết định tiếp tục giải quyết xung đột tại một toà án trung gian hoặc trọng tài.

Theo như luật "Về chữ ký điện tử số", trong trường hợp sử dụng các thiết bị không được cấp phép thì trách nhiệm sẽ do người phân phối các thiết bị đó gánh chịu (thí dụ như ngân hàng chẳng hạn). Cho nên, quan toà có thể không tiêu tốn thời gian vàng bạc của mình, lập tức chỉ tay về phía người có tội và đi giải quyết việc quan trọng hơn.

Tuy nhiên, giả thiết rằng những người tham gia là những người tuân thủ pháp luật: các thiết bị được họ sử dụng có giấy chứng nhận của FAPSI và nhà băng cũng có giấy phép của FAPSI. Trong trường hợp này, toà án, xem xét tất cả những tình tiết của sự việc, có thể quyết định thêm một cuộc giám định. Tất nhiên, để cho việc này cần có những cơ sở xác đáng. Ví dụ, người tham gia khoái thác khỏi bản quyền tác giả cần trình ra bằng chứng không thể bác bỏ rằng khoá bí mật của anh ta không bị tiết lộ, anh ta cùng với khoá bí mật luôn nằm ở một chỗ khác, nơi mà có nhiều người làm chứng. Sự việc phát triển tiếp theo 2 đường.

Thứ nhất, cuộc giám định độc lập phát hiện ra rằng thuật toán được cài đặt trong thiết bị không tuân theo chuẩn nhà nước Nga. Trong trường hợp này trách nhiệm đền bù thiệt hại sẽ quy cho phòng thí nghiệm nơi cấp chứng nhận cho thiết bị.

Con đường phát triển có thể thứ hai của quá trình: cuộc giám định phát hiện ra hư hỏng ở trong chính thuật toán (hay được công bố bởi một ai đó trong số những người nghiên cứu). Trong tình huống này trách nhiệm được quy cho người làm ra chuẩn, nhưng cố gắng theo kiện nhà nước là một việc ít triển vọng. Cho nên, một kết cục thường xảy ra là, nếu sự việc đã đến mức phải có cuộc giám định độc lập, thì sẽ kết thúc hoặc là cuộc giám định công nhận chữ ký số đang được tranh cãi là đúng (người tranh cãi thiệt về vật chất) hoặc là trong quá trình giám định sẽ tìm ra vi phạm quy tắc khai thác thiết bị bởi người thuê bao đang muốn chối bỏ trách nhiệm tác giả, từ đó có sự rò rỉ thông tin về khoá bí mật (những thiệt hại vật chất một lần nữa anh ta lại phải chịu).

Một số khía cạnh kỹ thuật không dễ nhìn thấy

Đấy chưa phải là những khó khăn cuối cùng trên con đường gian khổ của thông tin hoá toàn xã hội và sử dụng chữ ký số.

Trên đây chúng tôi đã nhiều lần nhắc đến việc "nhận một bản sao khoá công khai của trung tâm chứng thực". Có thể xuất hiện một câu hỏi hợp lý: đây chính là cái đinh để treo toàn bộ hệ thống (thực vậy, vì khoá công khai của trung tâm chứng thực không được ký bởi ai cả; trong trường hợp tốt nhất, đó là một giấy chứng nhận tự ký), như thế thì có thể tin vào tính toàn vẹn của nó như thế nào? Lối ra dễ nhận thấy- cùng với tệp (thể hiện điện tử của khoá) cần phải đưa ra một bản in khoá trên giấy (đây chính là "tính kinh tế" của giấy, được đảm bảo bằng chữ ký và dấu). Tương ứng, việc kiểm tra tính toàn vẹn của khoá được thực hiện bằng soát lại trực

giác khoá được đưa ra màn hình máy tính và so với cái được in trên giấy. Không tiện, nhưng nghĩ ra một cách khác thì thật là khó. Tất nhiên, nếu người sử dụng là thuê bao của một số hệ thống chữ ký điện tử số thì anh ta cần phải trực tiếp soát lại bằng mắt khoá công khai của mỗi trung tâm chứng thực so với bản in. Bài toán này có thể giải quyết nếu ở mức độ tổng thể có xây dựng một hệ thống hình cây các trung tâm chứng thực. Khi đó, khoá công khai của trung tâm chứng thực trong một hệ thống cụ thể (ví dụ, hệ thống "khách hàng-nhà băng") được ký bởi khoá công khai của trung tâm chứng thực cấp cơ quan chính quyền địa phương. Khoá công khai của trung tâm này lại được chứng thực bởi trung tâm chứng thực cấp chủ thể của liên bang, tiếp tục đến trung tâm chứng thực của vùng liên bang, ...cuối cùng đến trung tâm chứng thực chính nhất, trung tâm chứng thực liên bang. Như vậy, sẽ chỉ tồn tại một khoá công khai không được ai ký cả, và để tin vào tính toàn vẹn của nó thì chỉ có một cách là so bằng mắt. Khoá công khai (chính xác hơn là chứng nhận khoá công khai) của một trung tâm chứng thực khác bất kỳ luôn có thể kiểm tra theo chuỗi các chứng nhận.

Điều tinh tế kỹ thuật tiếp theo là ở chỗ, tất cả các hành động mà chúng ta nói tới ở trên, được thực hiện không phải trong bộ não con người, không phải ở trong đống giấy tờ, mà với sự trợ giúp của chương trình phần mềm, đến lượt nó, lại hoạt động trên một thiết bị nào đó. Tất cả những cái đó được làm bởi con người, mà con người thì thường nhầm lẫn. Việc chứng nhận của FAPSI đối với các thiết bị mật mã (trong trường hợp của chúng ta là các phương tiện chữ ký số) đảm bảo một độ tin cậy đủ cao việc không có lỗi và tính tuân thủ của các thao tác trong chương trình được viết theo các thuật toán chuẩn mã quốc gia. Nhưng lấy đâu ra đảm bảo rằng module chương trình mà người sử dụng nhận được như là phương tiện của chữ ký số tuân theo mẫu đã được cấp phép và không bị thay đổi? Khác với khoá công khai của trung tâm chứng thực, chương trình phần mềm có một kích thước lớn, việc in nó ra sẽ tốn nhiều chồng giấy, khó động viên được ai làm việc kiểm tra bản in với cái ở trên màn hình, chưa nói đến tính không hiệu quả của nó (lúc rảnh rang hãy thử hai dãy chữ số không có nghĩa độ dài khoảng vài trăm chữ số - lỗi là điều không thể tránh khỏi). Làm thế nào đây?

Một số phương án giải quyết bài toán

Phương án đầu tiên là tính tổng kiểm tra của chương trình phần mềm theo một thuật toán đã biết. Người sử dụng có thể tự tính tổng kiểm tra của phần mềm nhận được bằng một thiết bị mà anh ta tin tưởng và thực hiện theo thuật toán đã chỉ ra. Nếu tổng kiểm tra trùng với cái đã được công bố (và tất nhiên, trùng với cái ở trên giấy có chữ ký và dấu tròn), thì phần mềm có thể coi là tuân thủ mẫu. Dễ hiểu, để cho tin cậy nên dùng thuật toán có độ bảo đảm tốt hơn CRC32, ví dụ như MD5 hay SHA-1. Nếu chúng ta muốn là người yêu nước và sử dụng thuật toán băm nội địa theo như chuẩn GOST P 34.11 thì có một cái cản trở. Đó là do khi tính giá trị của hàm băm theo GOST P 34.11 thì cần phải sử dụng thuật toán mã khối theo GOST 28147-89 (mã vecto khởi điểm của phép băm ở chế độ thay thế đơn giản bằng các khoá được tạo từ thông tin cần bảo vệ). Nhưng để thực hiện được GOST 28147-89 ngoài khoá mã ra còn cần bảng các phép thế (theo tiếng Anh đó là các S-box), giá

trị của chúng không được chỉ ra trong chuẩn mà được chọn riêng biệt cho mỗi hệ thống. Do đó trong văn bản được đưa ra cùng với tổng kiểm tra (giá trị của hàm băm) còn phải chỉ ra xem nó được tính với các giá trị nào của bảng các phép thế.

Phương án thứ hai là ở việc người phân phát các thiết bị chữ ký số (đó là người tổ chức hệ thống xử lý văn bản điện tử- nhà băng, quỹ, người giữ tài nguyên dùng chung,...) đảm bảo phần mềm chữ ký số bằng chữ ký số của công chứng viên. Người sử dụng có thể tin vào tính toàn vẹn của phần mềm, tìm đến công chứng viên đã ký để kiểm tra chữ ký số hoặc đến trung tâm chứng thực nằm phía cao hơn trong cây kiến trúc. Theo một nghĩa nào đó điều này có lợi hơn nhiều so với phương án đầu: giấy vẫn là giấy, cần có một lượng đủ các giấy tờ được đảm bảo bằng chữ ký và con dấu cùng với tổng kiểm tra, còn việc nhân bản ở dạng điện tử phần mềm chữ ký số đã được ký là thoải mái.

Phương án thứ ba- nhận phần mềm chữ ký số từ nguồn gốc độc lập. Ví dụ, có thể có thể để mẫu phần mềm cho công chứng viên lưu giữ hoặc nhận trực tiếp nó từ người sản xuất. Phương án này logic hơn, nhưng mất nhiều chi phí để thực hiện.

Bàn một chút về pháp luật

Cách đây không lâu, tại Đuma quốc gia đã thảo luận lần đầu dự luật "Về chữ ký điện tử số", các cuộc tranh cãi xung quanh nó chưa hề tắt kể từ khi xuất hiện phiên bản đầu tiên của dự luật. Nếu như tính cần thiết của nó không có ai tranh cãi thì nội dung của nó gây ra hàng loạt trách móc. Chúng ta hãy chọn ra một số điểm gây tranh cãi từ đạo luật này, theo quan điểm của chúng tôi chúng có thể gây khó cho việc sử dụng thực tế chữ ký số.

Trong số những cái kèm theo bắt buộc của chứng nhận khoá công khai (điều 6, điểm 1) bao gồm yêu cầu chỉ ra các quan hệ pháp luật mà chứng nhận được dùng tróng đó. Ngoài việc không thống nhất với X.509, từ điều này suy ra rằng ông Vaxia Pupkin đã được nhắc đến ở phía trên hoặc cần phải tạo ra chìa khoá riêng cho mỗi dạng quan hệ và lấy chứng nhận tại trung tâm (sau đó không được dùng nhầm lẫn), hoặc ngay lập tức cố gắng hình dung tất cả các quan hệ cần cho bản thân và liệt kê chúng trong chứng nhận cách nhau bởi các dấu phẩy. Trong trường hợp thứ hai, cũng không thể đảm bảo tránh được trường hợp, sau này trong quá trình hoạt động cuộc sống sẽ xuất hiện quan hệ luật pháp mới chưa có trong chứng nhận. Cùng cần thấy rằng, thực tế trong mỗi thoả thuận có hàng loạt các quan hệ luật pháp khác nhau, và còn xa tất cả chúng là dễ thấy đối với người dùng. Thực tế, để thực hiện tinh thần và câu chữ của luật, trước khi tạo chữ ký cần phải tiến hành phân tích văn bản về mặt luật pháp, điều này chắc là bạn đồng ý rằng khó mà thực hiện được. Trong trường hợp ngược lại sẽ xuất hiện hàng loạt khả năng sử dụng ác ý từ mỗi bên.

Để điều tiết phần đáng kể các quan hệ tương hỗ trong luật đưa ra nhiều chỉ dẫn tới các điều luật chuẩn khác, nhưng chúng chưa có vào thời điểm này và chưa biết bao giờ mới có.

Trong luật còn chưa soạn đầy đủ vấn đề phân công trách nhiệm giữa những người tham gia trao đổi điện tử và các trung tâm chứng thực. Trên đây đã đưa ra các phương án giải quyết, mà theo quan điểm của chúng tôi là cho phép thực hiện phân định ranh giới. Nhưng cần thiết để cho những quan điểm đó tìm được sự phản ánh trong dự luật, chứ không sáng tạo ra một cách riêng biệt trong từng hoàn cảnh cụ thể với mạo hiểm rằng bỏ qua một cái gì đó.

Theo như điểm 1 điều 8 của dự luật, chủ yếu tổ chức thương mại có thể là trung tâm chứng thực. Yêu cầu này ngay lập tức làm tổn hại quyền của các tổ chức không thương mại, nhưng tỷ lệ của chúng trong tác động tương hỗ điện tử không ít.

Ngoài ra, không hoàn toàn rõ ràng yêu cầu này đối với việc sử dụng chữ ký số bởi các tổ chức nhà nước, chúng có thể thực hiện trao đổi điện tử trong các hệ thống dùng chung.

Trong phương án được đưa ra của dự luật đề nghị sử dụng chủ yếu các thiết bị chữ ký số được cấp phép, điều này không cho phép dễ áp dụng chữ ký số trong một số trường hợp. Dễ thấy là việc sử dụng các giấy chứng nhận nước ngoài, phương tiện để làm việc với chúng khó được cấp phép ở nước Nga.

Cũng cần phải đặt dấu chấm trên chữ "i" trong các vấn đề cấp phép. Theo luật pháp hiện hành, dịch vụ kỹ thuật của các thiết bị mật mã cần phải được cấp phép, nhưng điều này không có nghĩa là mỗi người tham gia trao đổi thông tin cần phải nhận được giấy phép - chỉ cần người tổ chức ra hệ thống có giấy phép là đủ.

Điểm 1 điều 6 yêu cầu rằng giấy chứng nhận khoá công khai của chữ ký số nhất thiết phải có họ, tên và tên đệm của người chủ. Luận điểm nguyên tắc này gây ra băn khoăn, vì làm cho không thể sử dụng chữ ký số để tham gia vào các cuộc thi dưới bí danh.

Yêu cầu về khả năng của trung tâm chứng thực có trách nhiệm dân sự không ít hơn 100 lần so với giới hạn trao đổi đã được chỉ ra trong chứng thực cũng gây ra tranh cãi. Có thể chỉ ra hàng loạt cuộc trao đổi, trong đó việc chỉ ra giá, và như vậy, giới hạn của trách nhiệm, là không thể. Ngoài ra, yêu cầu này trên thực tế xoá bỏ dạng ký hợp đồng ở dạng điện tử cùng với việc sử dụng chữ ký số- các cuộc trao đổi có giá trị lớn không thể ký bằng chữ ký số.

Luận điểm về việc cung cấp miễn phí dịch vụ khẳng định chữ ký (điều 9 điểm 4) một cách logic hơn được thay bằng một giới hạn cho mức phí tối đa. Tất nhiên, nếu chú ý đến việc điều chỉnh về kết luận từ thẩm quyền xét xử luật áp dụng chữ ký số phi thương mại, tự động vấn đề trả phí trong các hệ thống phi thương mại được loại bỏ.

Dự luật ngầm hiểu rằng việc sử dụng chữ ký số như là một cái tương tự với chữ ký của cá nhân, ngay cả khi khi anh ta đặt chữ ký không phải như một cá thể mà là người đại diện toàn quyền của tổ chức pháp lý. Để tránh sự hiểu nhầm, trong chứng nhậ khoá công khai cho người có chức quyền, cần phải chỉ ra rằng nó được cấp không đơn thuần cho Xiđôrốp, mà cho chủ tịch điều hành Xiđôrốp Ivan Pêtrôvic, người thực hiện trọng trách trên cơ sở tin cậy hay điều lệ của xí nghiệp. Dễ thấy rằng, trong trường hợp này, Xiđôrốp Ivan Pêtrôvic có hai chứng nhận: như một cá nhân và như một đại diện cho một tổ chức nào đó.

Như vậy, không nghi ngờ gì, dự luật cần thiết và được chờ đợi từ lâu cần phải được thảo luận kỹ hơn nữa, vì ở dạng hiện tại, phần lớn nó mới chỉ mang tính quảng cáo.

Chuẩn chữ ký số GOST P 34.10-94[3]

Chuẩn chữ ký số của Nga được lập sau phương án chuẩn của nước Mỹ, cho nên các tham số của thuật toán này được chọn với trù tính về khả năng tiềm tàng của người mã thám trong việc thám hệ mã. Nói riêng, việc tăng độ dài giá trị hàm băm làm giảm xác xuất đụng chạm, tương ứng với nó là bậc của phần tử sinh, điều này làm cho việc giải bài toán logarithm rời rạc sẽ khó hơn khi cần tìm khoá bí mật.

1. Chọn tham số

Để mô tả thuật toán sẽ sử dụng các ký hiệu sau;

B* -tập tất cả các từ hữu hạn trên bảng chữ cái B = {0,1};

|A| - độ dài từ A;

$V_k(2)$- tập tất cả các từ nhị phân độ dài k;

A||B - nối hai từ A và B (hay còn ký hiệu là AB);

A^k - nối k từ A liên tiếp;

<N>k - từ có độ dài k là kết quả phép tính N (mod 2k) với N là số nguyên không âm;

$\oplus$ - phép cộng từng bit theo modulo 2;

[+] - phép cộng theo quy tắc A [+] B = $<A+B>_k$ (k=|A|=|B|);

m - thông báo cần ký;

m_1- thông báo nhận được;

h - hàm băm, ánh xạ dãy m vào từ h(m) $\in$ V256(2);

p - số nguyên tố, $2^{509} < p < 2^{512}$, hoặc $2^{1020} < p < 2^{1024}$;

q - số nguyên tố, $2^{254} < q < 2^{256}$ và q là ước của (p-1);

a - số nguyên, $1 < a < p-1$, với $a^q \pmod p = 1$;

k- số nguyên, $0 < k < q$;

x - khoá bí mật của người sử dụng để ký, $0 < x < q$;

y - khoá công khai để kiểm tra chữ ký, $y = a^x \pmod p$.

Các số p, q và a là các tham số của hệ thống, được công bố công khai. Bộ giá trị cụ thể có thể là chung cho tất cả mọi người trong hệ thống. Số k được sinh trong quá trình ký thông báo cần phải giữ bí mậtvà được huỷ ngay sau khi ký. Số k được lấy từ bộ tạo ngẫu nhiên vật lý hoặc bởi dãy giả ngẫu nhiên với các tham số bí mật.

2. Tạo chữ ký

Việc tạo chữ ký số gồm những bước sau:

1) tính h(m) - giá trị hàm băm của thông báo m. Nếu h(m) (mod q)=0 thì gán cho h(m) giá trị $0^{255}1$;
2) chọn số nguyên k, $0 < k < q$;
3) tính hai giá trị $r'=a^k \pmod p$ và r=r' (mod q). Nếu r = 0, thì chuyển về bước 2 và chọn giá trị k khác;

[3] C.U.Mfhbxtd, D.D. Ujyxfhjd, H.T.Cthjd, Jcyjds cjdhtvtyyjq rhbgnjuhfabb, Vjcrdf, Ujhzxfz kbybz-Ntktrjv, 2002, cnh. 96-98.

4) người sử dụng dùng khoá bí mật x để tính giá trị s=(xr+kh(m)) (mod q). Nếu s=0 thì chuyển về bước 2, trong trường hợp ngược lại thì kết thúc thuật toán.

Chú ý rằng thông báo cho giá trị hàm băm bằng 0 không được ký. Trong trường hợp ngược lại phương trình ký sẽ được giản ước thành s=xr (mod q) và người ác ý sẽ tính được khoá bí mật.

3. Kiểm tra chữ ký

Phương trình kiểm tra là $r \equiv \left(a^{s.h(m_1)^{-1}} .y^{-r.h(m_1)^{-1}} (\mathrm{mod}\, p) \right) (\mathrm{mod}\, q)$. Thực vậy;

$$\left(a^{s.h(m)^{-1}} .y^{-r.h(m)^{-1}} (\mathrm{mod}\, p) \right) (\mathrm{mod}\, q) = \left(a^{s.h(m)^{-1}} .a^{-r.x.h(m)^{-1}} (\mathrm{mod}\, p) \right) (\mathrm{mod}\, q) =$$

$$= \left(a^{h(m)^{-1}(s-xr)} (\mathrm{mod}\, p) \right) (\mathrm{mod}\, q) = a^{h(m)^{-1}(xr+kh(m)-xr)} (\mathrm{mod}\, p)(\mathrm{mod}\, q) =$$

$$= a^{h(m)^{-1}.kh(m)} (\mathrm{mod}\, p)(\mathrm{mod}\, q) = a^{k} (\mathrm{mod}\, p)(\mathrm{mod}\, q) \equiv r$$

Việc tính toán có thể được thực hiện như sau:

1) Kiểm tra điều kiện: 0 < s < q và 0 < r < q. Nếu một trong những điều kiện đó không được thực hiện thì chữ ký được coi là không có hiệu lực.
2) Tính $h(m_1)$- giá trị của hàm băm đối với bản tin nhận được. Nếu $h(m_1) (\mathrm{mod}\, q)=0$ thì gán cho $h(m_1)$ giá trị $0^{255}1$.
3) Tính $v=(h(m_1))^{q-2} (\mathrm{mod}\, q)$, đó chính là giá trị $h(m_1)^{-1} (\mathrm{mod}\, q)$. Nói chung, việc tính $h(m_1)^{-1} (\mathrm{mod}\, q)$ bằng thuật toán Eucơlid mở rộng sẽ nhanh hơn việc nâng lên luỹ thừa
4) Tính $z_1=sv (\mathrm{mod}\, q)$ và $z_2= (q-r)v (\mathrm{mod}\, q)$
5) Tính $u = \left(a^{z_1} y^{z_2} (\mathrm{mod}\, p) \right) (\mathrm{mod}\, q)$
6) Kiểm tra điều kiện r = u. Nếu đẳng thức xảy ra thì công nhận chữ ký đúng và bản tin không bị thay đổi trong quá trình truyền.

Chuẩn chữ ký số GOST P 34.10-2001[4]

So sự tăng năng suất của các phương tiện tính toán và việc hoàn thiện thuật toán tính logarithm rời rạc trong trường hữu hạn nên xuất hiện nhu cầu tăng độ bền vững của thuật toán chữ ký điện tử số đối với nhiều dạng tấn công. Vì thế nên chuẩn GOST P 34.10-2001 đã được nghiên cứu trong dự án "Công nghệ thông tin. Bảo vệ thông tin bằng mật mã. Các quá trình tạo và kiểm tra chữ ký điện tử số". Nó sẽ được áp dụng từ ngày 1 tháng 7 năm 2002 thay cho chuẩn GOST P 34.10-94 trước đó.

Chuẩn bị tham số

Trong chuẩn này các phép toán của nhóm các điểm thuộc đường cong eliptic trên trường hữu hạn được sử dụng. Đường cong elipic E trên trường nguyên tố GF(p) ở dạng Veistrass được sử dụng, nó được cho bởi các hệ số a và b hoặc đại lượng J(E) được gọi là bất biến của đường cong: $J(E) = 1728\frac{4a^3}{4a^3 + 27b^2} \bmod p$. Các hệ số a và b của đường cong E được xác định thông qua hằng số bởi công thức:

a ≡ 3k mod p; b ≡ 2k mod p, trong đó $k = \frac{-J(E)}{1728 - J(E)} \bmod p$, J(E) ≠0, 1728.

Điểm Q được gọi là điểm bội k, k ∈ Z nếu như với một điểm P nào đó có đẳng thức Q=kP. Các tham số của lược đồ chữ ký số bao gồm:

- p - số nguyên tố là modul của đường cong eliptic, $p > 2^{255}$.
- đường cong eliptic được cho bởi bất biến J(E) hay các hệ số a và b
- số nguyên m - bậc của nhóm các điểm trên đường cong E
- số nguyên tố q - bậc của nhóm vòng con của nhóm các điểm thuộc đường cong eliptic E với điều kiện
$$\begin{cases} m = nq, n \in Z, n \geq 1 \\ 2^{254} < q < 2^{256} \end{cases}$$
- điểm cơ sở P ≠ 0 trên đường cong có bậc q, tức là pQ=0. Toạ độ của điểm này được ký hiệu qua (x_p, y_p);
- hàm băm ánh xạ thông báo có độ dài bất kỳ vào tập các vecto nhị phân độ dài 256. Hàm băm được định nghĩa bởi chuẩn GOST P 34.11-94.

Mỗi người sử dụng lược đồ chữ ký điện tử số cần có cặp khoá cá nhân sau:

- khoá bí mật của người sử dụng - số nguyên d, 0 < d < q;

---

[4] C.U.Mfhbxtd, D.D. Ujyxfhjd, H.T.Cthjd, Jcyjds cjdhtvtyyjq rhbgnjuhfabb, Vjcrdf, Ujhzxfz kbybz-Ntktrjv, 2002, cnh. 98-101.

- khoá công khai của người sử dụng - điểm Q với toạ độ (x_q, y_q), thoả mãn đẳng thức : dP=Q.

Các tham số của chữ ký điện tử số cần thoả mãn các điều kiện sau:

- $p^t \neq 1 \pmod q$ với mọi t=1,2,..,B với $B \geq 31$;
- $m \neq p$;
- $J(E) \neq 0$ hay 1728.

Chúng ta đặt tương ứng vecto nhị phân $\bar{h} = (\alpha_{255},.....,\alpha_0)$ với số $\alpha = \sum_{i=0}^{255} \alpha_i 2^i$.

Sinh chữ ký số

Chữ ký số cho bản tin M được tính như sau:

1. Tính hàm băm : $\bar{h} = h(M)$
2. Tính số nguyên α có dạng biểu diễn nhị phân là vecto $\bar{h}$ và xác định $e \equiv \alpha \pmod q$. Nếu e =0 thì lấy e = 1.
3. Sinh ra số giả ngẫu nhiên k thoả mãn bất đẳng thức $0 < k < q$.
4. Tính điểm thuộc đường cong eliptic C = kP và đặt $r \equiv x_C \pmod q$ với x_C là toạ độ của điểm C theo trục x. Nếu r =0 thì quay về bước 3.
5. Tính giá trị $s \equiv (rd+ke) \pmod q$. Nếu s=0 thì quay về bước 3.
6. Tính vecto nhị phân ứng với các số ra và s. Chữ ký số $\zeta = (\bar{r} \| \bar{s})$ là phép nối hai vecto nhị phân.

Kiểm tra chữ ký

Để kiểm tra chữ ký ta làm như sau:

1. Theo chữ ký nhận được ζ cần tính ra hai số nguyên r và s. Nếu các bất đẳng thức sau không đúng thì bác bỏ chữ ký: $0 < r < q$, $0 < s < q$.
2. Tính hàm băm của bản tin nhận được: $\bar{h} = h(M)$
3. Tính số nguyên α có dạng biểu diễn nhị phân là vecto $\bar{h}$ và xác định $e \equiv \alpha \pmod q$. Nếu e =0 thì lấy e = 1.
4. Tính $v \equiv e^{-1} \pmod q$
5. Tính $z_1 \equiv sv \pmod q$, $z_2 \equiv -rv \pmod q$
6. Tính điểm trên đường cong eliptic $C = z_1P + z_2Q$ và xác định $R \equiv x_C \pmod q$ với x_C là toạ độ của điểm C theo trục x
7. Chữ ký số được công nhận khi và chỉ khi R = r.

Hàm băm GOST P 34.11-94[5]

Khi mô tả hàm băm chúng tôi sẽ sử dụng các ký hiệu đã được sử dụng khi mô tả các thuật toán chữ ký số GOST P 34.10-94 và GOST P 34.10-2001, ngoài ra, gọi M là dãy nhị phân cần băm; h- hàm băm, ánh xạ dãy M vào từ $h(M) \in V_{256}(2)$; $E_K(A)$ - kết quả phép băm từ A với khoá K bởi thuật toán mã khối GOST 28147-89 ở chế độ thay thế đơn giản; H - vecto khởi điểm để băm.

Mô tả chung

Chúng ta hiểu hàm băm là một ánh xạ

$$h: B^* \to V_{256}(2).$$

Để định nghĩa hàm băm cần có:

- thuật toán tính băm theo từng bước κ
  $$\kappa: V_{256}(2) \times V_{256}(2) \to V_{256}(2)$$
- mô tả quá trình lặp để tính giá trị hàm băm

Thuật toán tính băm theo từng bước gồm 3 phần:

- tạo 4 khoá có 256 bit
- biến đổi mã: mã các từ con 64 bit của từ H bằng các khoá K_i (i=1, 2, 3, 4) bằng thuật toán GOST 28147-89 ở chế độ thay thế đơn giản
- ánh xạ xáo trộn kết quả mã

Tạo khoá

Xét $X=(b_{256}, b_{255},..., b_1) \in V_{256}(2)$.

Giả sử $X = x_4||x_3||x_2||x_1 = \eta_{16}||\eta_{15}||...||\eta_1=\xi_{32}||\xi_{31}||...||\xi_1$

với $x_i \in V_{64}(2)$, i=1..4; $\eta_j \in V_{16}(2)$, j=1..16; $\xi_k \in V_8(2)$, k=1..32.

Ký hiệu $A(X)= (x_1 \oplus x_2)||x_4||x_3||x_2$.

Biến đổi P: $V_{256}(2) \to V_{256}(2)$ chuyển từ $\xi_{32}||\xi_3 1||...||\xi_1$ thành từ $\xi_{\varphi(32)}||\xi_{\varphi(31)}||...||\xi_{\varphi(1)}$

với $\varphi(i+1+4(k-1))=8i+k$, i=0..3, k=1..8.

Để tạo khoá cần sử dụng các dữ liệu sau:

- các từ H, $M \in V_{256}(2)$;
- các hằng số: các từ C_i (i=2, 3, 4) có các giá trị $C_2=C_4=0^{256}$ và $C_3=1^80^81^{16}0^{24}1^{16}0^8(0^81^8)^21^80^8(0^81^8)^4(1^80^8)^4$ còn C_1??

Thuật toán tính khoá như sau:

1. Gán các giá trị i=1, U=H, V=M
2. Thực hiện $W=U \oplus V$, $K_1=P(W)$

[5] C.U.Mfhbxtd, D.D. Ujyxfhjd, H.T.Cthjd, Jcyjds cjdhtvtyyjq rhbgnjuhfabb, Vjcrdf, Ujhzxfz kbybz-Ntktrjv, 2002, cnh. 121-124.

3. Gán i=i+1
4. Kiểm tra điều kiện i=5. Nếu đúng nhảy đến bước 7, sai thì tiếp bước 5.
5. Tính U=A(U) $\oplus$ C_i, V=A(A(V)), W=U $\oplus$ V, K_i=P(W).
6. Chuyển về bước 3
7. Kết thúc

Biến đổi mã

Đây là giai đoạn mã 4 từ con 64 bit của từ H bởi các khoá K_i (i=1, 2, 3, 4). Chúng ta cần các dữ liệu sau: $H=h_4||h_3||h_2||h_1$, $h_i \in V_{64}(2)$ và bộ khoá K_i (i=1, 2, 3, 4). Sau khi mã ta có $s_i = E_{K_i}(h_i), i = 1,2,3,4$ và vecto kết quả là: $S= s_4||s_3||s_2||s_1$.

Biến đổi trộn

Trộn dãy thu được nhờ thanh dịch. Các dữ liệu ban đầu là các từ H, M, S $\in V_{256}(2)$. Biến đổi $\notin$: $V_{256}(2) \to V_{256}(2)$ chuyển từ $\eta_{16}||\eta_{15}||...||\eta_1$, $\eta_i \in V_{16}(2)$, i=1..16 thành từ

$\eta_1 \oplus \eta_2 \oplus \eta_3 \oplus \eta_4 \oplus \eta_{13} \oplus \eta_{16}||\eta_{16}||..||\eta_2$

Khi đó giá trị băm từng bước là từ

$\kappa(M,H) = \notin^{61}(H \oplus \notin(M \oplus \notin^{12}(S)))$ với $\notin^i$ là biến đổi $\notin$ làm i lần.

Tính giá trị băm

Giá trị ban đầu để tính giá trị hàm băm là dãy M $\in$ B*. Tham số là vecto khởi điểm H- một từ cố định tuỳ ý từ $V_{256}(2)$.

Thủ tục tính hàm h ở mỗi vòng lặp sử dụng các đại lượng sau:

M $\in$ B* - phần của dãy M, chưa đi qua hàm băm ở những vòng lặp trước

H $\in V_{256}(2)$ - giá trị hiện tại của hàm băm

$\Sigma \in V_{256}(2)$ - giá trị hiện tại của tổng kiểm tra

L $\in V_{256}(2)$ - giá trị hiện tại của độ dài phần của dãy M đã được xử lý ở các vòng lặp trước.

Thuật toán tính hàm h được chia làm 3 giai đoạn:

Giai đoạn 1. Gán các giá trị ban đầu cho các đại lượng hiện tại

$M:=M; H:=H; \Sigma:=0^{256}; L:=0^{256}$

Giai đoạn 2. Kiểm tra điều kiện |M| > 256. Nếu đúng thì chuyển sang bước 3. Ngược lại thì thực hiện các phép tính sau:

$L:=<L+|M|>_{256}; M':=0^{256-|M|}||M; \Sigma:=\Sigma [+] M';$

$H:=\kappa(M',H); H:=\kappa(L,H); H:=\kappa(\Sigma,H)$

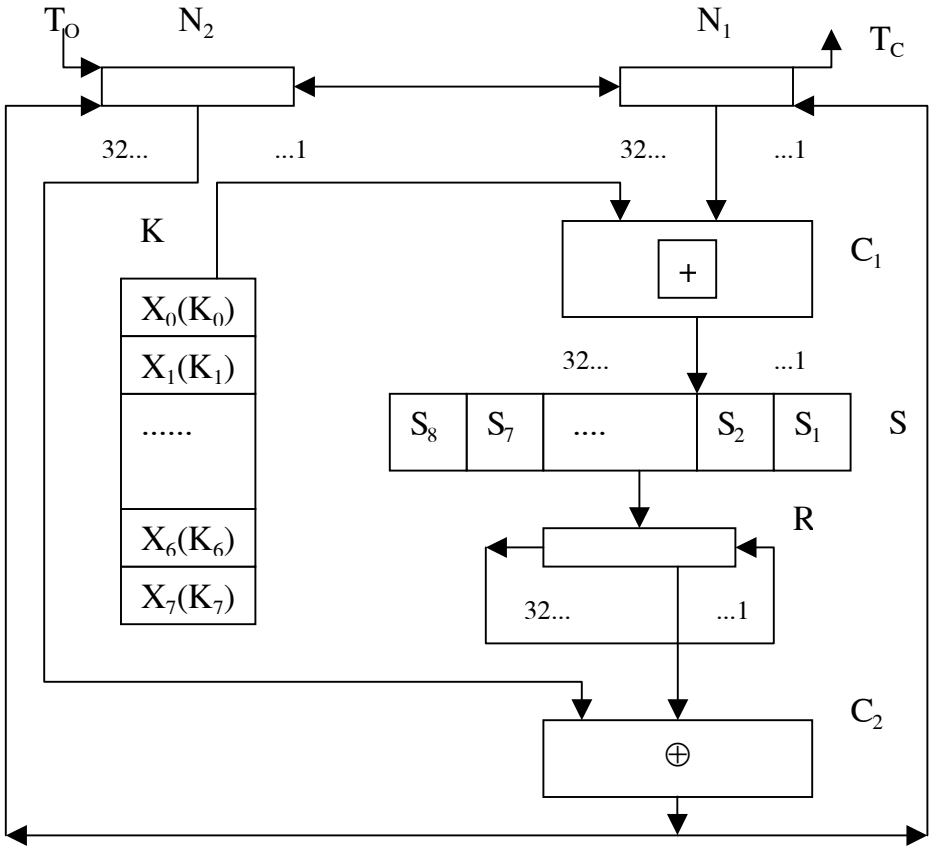
Kết thúc thuật toán. H là giá trị hàm băm.

Giai đoạn 3. Tính từ con $M_S \in V_{256}(2)$ của từ M ($M=M_P||M_S$). Tiếp tục thực hiện dãy phép tính:

H:=κ(M',H); L:=<L+256>$_{256}$; Σ:= Σ [+]M_S; M:=M_P.
Chuyển lên bước 2.

Chuẩn mã dữ liệu GOST 28147-89[6]

Tại Liên bang Nga có một chuẩn mật mã duy nhất cho các hệ thống thông tin. Nó là bắt buộc cho các cơ quan nhà nước, tổ chức, xí nghiệp, ngân hàng và các công sở khác có hoạt động gắn liền với việc đảm bảo an toàn thông tin quốc gia. Đối với các tổ chức khác hay cá nhân thì GOST 28147-89 mang tính khuyến cáo.



Chuẩn này được thiết lập có tính đến kinh nghiệm trên thế giới, nói riêng, đã chú ý đến những điểm yếu và khả năng không thực hiện được của DES, vì vậy việc áp dụng chuẩn có tiện hơn. Thuật toán mã hoá được xây dựng theo cấu trúc Feistel.

Đối với phép nối từ chúng ta dùng ký hiệu như là phép nhân. Ngoài ra, sẽ sử dụng các phép cộng sau:

A ⊕ B - phép cộng từng bit theo modulo 2;
A [+] B - cộng theo modulo 2^{32};

[6] C.U.Mfhbxtd, D.D. Ujyxfhjd, H.T.Cthjd, Jcyjds cjdhtvtyyjq rhbgnjuhfabb, Vjcrdf, Ujhzxfz kbybz-Ntktrjv, 2002, cnh. 37-39.

A {+} B - cộng theo modulo 2^{32}-1;

Thuật toán có một số chế độ làm việc. Trong mọi chế độ đều sử dụng khoá W có 256 bit, đó là 8 số có 32 bit: W=X(7)X(6)X(5)X(4)X(3)X(2)X(1)X(0).
Khi giải mã chúng ta cũng dùng khoá đó.

Chế độ làm việc cơ sở của thuật toán là chế độ thay thế đơn giản. Bây giờ chúng ta sẽ mô tả chế độ đó.

Các ký hiệu trên hình vẽ:

N_1, N_2- các thanh ghi lưu 32 bit;
S_1- tổng theo modulo 2^{32};
S_2- tổng theo modulo 2;
R- thanh ghi dịch vòng 32-bit;
K - thiết bị nhớ khoá gồm 256 bit được chia thành 8 từ 32-bit;
S - khối thay thế gồm 8 hộp thế S_1,..., S_8.

Giả sử bản rõ được chia thành các khối có 64 bit, chúng được ký hiệu là T_O. Thủ tục mã gồm 32 vòng. Đầu tiên, khoá k có 256 bit được đưa vào thiết bị nhớ khoá, tạo thành 8 khoá con k_i: $k=k_7k_6...k_1$.

Dãy các bit của T_O được phân thành hai nửa có 32-bit (trái và phải):

$T_O =(a_1(0),..., a_{32}(0), b_1(0),...,b_{32}(0))$

Các bit trong mỗi nửa được lấy ra theo thứ tự ngược lại để tạo nên hai từ có 32 bit là a(0) và b(0):

$a(0) = (a_{32}(0), a_{31}(0), ...,a_1(0))$,
$b(0) = (b_{32}(0), b_{31}(0), ...,b_1(0))$.

Hai vecto 32 bit a(0) và b(0) được đưa vào các thanh ghi lưu N_1 và N_2 trước vòng mã thứ nhất. a(0) trong N_1 và b(0) trong N_2.

Giả sử $a(j) = (a_{32}(j), a_{31}(j), ...,a_1(j))$, $b(j) = (b_{32}(j), b_{31}(j), ...,b_1(j))$ là nội dung của các thanh ghi lưu N_1 và N_2 sau vòng mã thứ j. Chúng ta ký hiệu f là hàm mã. Thế thì:

với j =1..24

$$a(j) = f(a(j-1) + k_{j-1(\mathrm{mod}\,8)}) \oplus b(j-1)$$

$$b(j) = a(j-1)$$

với j=25..31

$$a(j) = f(a(j-1) + k_{32-j(\mathrm{mod}\,8)}) \oplus b(j-1)$$

$$b(j) = a(j-1)$$

với j=32

$$a(32) = a(31)$$

$$b(32) = f(a(31) + k_0) \oplus b(31)$$

Việc tính hàm f trải qua 2 giai đoạn:

- Ở giai đoạn thứ nhất, tham số x có 32 bit được chia thành 8 vecto có 4 bit. Bộ 4 bit thứ i được ánh xạ thành 4 bit nhờ các phép thế S_i (i=1..8). S_i là các phép hoán vị của tập các số nguyên từ 0 đến 15. Như vậy S(x) là một vecto có 32 bit.
- Giai đoạn thứ hai: nhờ thanh ghi R, S(x) được dịch vòng về bên trái 11 vị trí.

Kết quả của phép mã T_O là T_C được lấy ra từ các thanh ghi lưu N_1 và N_2 sau 32 vòng mã theo thứ tự từ trái qua phải.

TC=($a_1(32)$, $a_2(32)$,..., $a_{32}(32)$, $b_1(32)$, $b_2(32)$,...,$b_{32}(32)$).

Chú ý: các S-box có thể được sử dụng làm khoá thời gian dài.

Bộ luật Liên bang Nga về chữ ký điện tử số

Chương I. Các điều khoản chung

***Điều 1*. Mục đích và phạm vi áp dụng bộ luật Liên bang này**

1. Đảm bảo điều kiện luật pháp cho việc sử dụng chữ ký điện tử số là mục đích của bộ luật liên bang này, theo nó thì chữ ký điện tử số trong các văn bản điện tử được công nhận có giá trị như chữ ký bằng tay trên các văn bản giấy tờ.
2. Hiệu lực của bộ luật liên bang này được áp dụng cho các quan hệ được xuất hiện khi tiến hành các hợp đồng pháp lý-dân sự và trong các trường hợp khác được xem xét trước bằng luật pháp của Liên bang Nga.

Hiệu lực của bộ luật Liên bang này không áp dụng cho các quan hệ được xuất hiện khi sử dụng các tương tự khác của chữ ký tay.

***Điều 2*. Điều chỉnh pháp quyền các quan hệ trong lĩnh vực sử dụng chữ ký điện tử số.**

Điều chỉnh pháp quyền các quan hệ trong lĩnh vực sử dụng chữ ký điện tử số được thực hiện theo bộ luật Liên bang này, bộ luật dân sự của Liên bang Nga, luật Liên bang "Về thông tin, thông tin hoá và bảo vệ thông tin", luật Liên bang "Về thông tin", và các luật liên bang khác và các văn bản luật pháp chuẩn của Liên bang Nga đã được áp dụng theo các bộ luật đã nêu, và cũng được thực hiện theo thoả thuận của các bên.

***Điều 3*. Các khái niệm cơ bản được sử dụng trong bộ luật Liên bang này**

Vì các mục đích của bộ luật Liên bang này, các khái niệm cơ bản sau được sử dụng:

văn bản điện tử - đó là văn bản, trong đó thông tin được biểu diễn ở dạng điện tử-số;

chữ ký điện tử số- phụ lục của một văn bản điện tử, dùng để bảo vệ văn bản điện đó khỏi giả mạo, nhận được như kết quả của phép biến đổi mật mã đối với thông tin với việc sử dụng khoá bí mật của chữ ký điện tử số và cho phép nhận dạng người chủ của chứng nhận khoá chữ ký, đồng thời cũng khẳng định việc thông tin trong văn bản điện tử không bị xuyên tạc;

người chủ của (giấy) chứng nhận khoá chữ ký- người được trung tâm chứng thực cấp cho giấy chứng nhận khoá ký và nắm giữ khoá bí mật tương ứng của chữ ký điện tử số, khoá bí mật này cho phép với sự giúp đỡ của các phương tiện chữ ký điện tử số tạo ra chữ ký điện tử số của người đó trong các văn bản điện tử (ký các văn bản điện tử);

các phương tiện chữ ký điện tử số- các công cụ máy móc hay phần mềm, đảm bảo việc thực hiện một trong các chức năng sau: tạo ra chữ ký điện tử số trong

các văn bản điện tử với việc sử dụng khoá bí mật của chữ ký điện tử số, khẳng định tính chân thực của chữ ký điện tử số trong các văn bản điện tử bằng việc sử dụng khoá bí mật của chữ ký điện tử số, tạo ra các khoá bí mật và công khai của chữ ký điện tử số;

giấy chứng nhận của các phương tiện chữ ký điện tử số- văn bản trên giấy, được cấp theo các quy định của hệ thống chứng thực để khẳng định tính tuân thủ các yêu cầu đã được thiết lập của các phương tiện chữ ký điện tử số;

khoá bí mật của chữ ký điện tử số- dãy duy nhất các ký hiệu, được biết bởi người chủ của chứng nhận khoá chữ ký và dùng để tạo ra chữ ký điện tử số trong các văn bản điện tử bằng việc dùng các phương tiện chữ ký điện tử số;

khoá công khai chữ ký điện tử số- dãy duy nhất các ký hiệu, tương ứng với khoá bí mật chữ ký điện tử số, được biết bởi một người sử dụng bất kỳ của hệ thống thông tin và được dùng để khẳng định tính chân thực của chữ ký điện tử số trong văn bản điện tử bằng cách sử dụng các phương tiện chữ ký điện tử số;

giấy chứng nhận khoá chữ ký- văn bản trên giấy hoặc văn bản điện tử cùng với chữ ký điện tử số của người có trách nhiệm thuộc trung tâm chứng thực, trong đó có khoá công khai của chữ ký điện tử số, và được trung tâm chứng thực cấp cho người tham gia hệ thống thông tin để khẳng định tính chân thực của chữ ký điện tử số và nhận dạng người chủ của chứng nhận khoá chữ ký;

khẳng định tính chân thực của chữ ký điện tử số trong văn bản điện tử- kết quả tán thành của việc kiểm tra bằng các phương tiện chứng thực tương ứng chữ ký điện tử số cùng với việc sử dụng chứng nhận khoá chữ ký tính phụ thuộc của chữ ký điện tử số vào người chủ của chứng nhận khoá chữ ký và tính thiếu vắng sự thay đổi trong văn bản điện tử đã được ký bởi chữ ký điện tử số đã cho;

người sử dụng chứng nhận khoá chữ ký- người sử dụng các thông tin nhận được tại trung tâm chứng thực về giấy chứng nhận khoá chữ ký để kiểm tra tính phụ thuộc của chữ ký điện tử số với người chủ của chứng nhận khoá chữ ký;

hệ thống thông tin sử dụng chung- hệ thống thông tin mở để sử dụng cho bất kỳ người hay cơ quan pháp luật nào, những người này không bị từ chối trong các dịch vụ của nó;

hệ thống thông tin doanh nghiệp- hệ thống thông tin mà những người tham gia là một nhóm người giới hạn, được xác định bởi người chủ của hệ thống hoặc bằng thoả thuận của những người tham gia hệ thống thông tin này.

Chương II. Các điều kiện sử dụng chữ ký điện tử số

Điều 4. Các điều kiện để công nhận giá trị như nhau của chữ ký điện tử số và chữ ký viết tay.

1. Chữ ký điện tử số trong văn bản điện tử có giá trị như chữ ký viết tay trong văn bản trên giấy nếu như đồng thời tuân thủ các điều kiện sau:

chứng nhận khoá chữ ký thuộc về chữ ký điện tử số này không mất hiệu lực (còn tác dụng) tại thời điểm kiểm tra hoặc tại thời điểm ký văn bản điện tử nếu như có bằng chứng xác định thời điểm ký;

tính chân thực của chữ ký số trong văn bản điện tử được khẳng định;

chữ ký điện tử số được sử dụng theo như những qui định được chỉ ra trong giấy chứng nhận khoá chữ ký.

2. Người tham gia hệ thống thông tin có thể đồng thời là chủ của một số lượng bất kỳ giấy chứng nhận khoá chữ ký. Khi đó văn bản điện tử cùng với chữ ký điện tử số có giá trị pháp lý khi thực hiện các quan hệ được chỉ ra trong giấy chứng nhận khoá chữ ký.

***Điều 5*. Sử dụng các phương tiện chữ ký điện tử số**

1. Việc sinh các khoá chữ ký điện tử số được thực hiện để sử dụng trong:

hệ thống thông tin sử dụng chung bởi người tham gia hệ thống hay bởi trung tâm chứng thực theo yêu cầu của người tham gia;

hệ thống thông tin doanh nghiệp theo quy cách được thiết lập trong hệ thống đó.

2. Khi sinh khoá chữ ký điện tử số để sử dụng trong hệ thống thông tin sử dụng chung cần phải chỉ áp dụng các phương tiện đã được cấp phép chữ ký điện tử số. Việc đền bù thiệt hại gây ra do sinh khoá chữ ký điện tử số bởi các phương tiện chữ ký điện tử số không được cấp phép có thể được quy trách nhiệm cho những người tạo ra và những người phân phối các phương tiện này theo như pháp luật Liên bang Nga.
3. Việc sử dụng các phương tiện chữ ký điện tử số không được cấp phép và các chữ ký điện tử số được sinh ra bởi chúng trong các hệ thống thông tin doanh nghiệp của các cơ quan liên bang thuộc chính quyền quốc gia, các cơ quan chính quyền quốc gia của các chủ thể Liên bang Nga và các cơ quan điều hành địa phương là không được phép.
4. Việc cấp phép các phương tiện chữ ký điện tử số được thực hiện theo luật pháp Liên bang Nga về cấp phép sản phẩm và dịch vụ.

***Điều 6*. Chứng nhận khoá chữ ký**

1. Chứng nhận khoá chữ ký cần chứa các thông tin sau:

số đăng ký duy nhất của chứng nhận khoá chữ ký, ngày bắt đầu và kết thúc thời hạn hiệu lực của chứng nhận chữ ký số nằm ở danh sách của trung tâm chứng thực;

họ, tên và tên đệm của người chủ chứng nhận khoá chữ ký hay bí danh của người chủ. Trong trường hợp sử dụng bí danh, trung tâm chứng thực cần ghi điều đó vào chứng nhận khoá chữ ký;

khoá công khai của chữ ký điện tử số;

tên của các phương tiện chữ ký điện tử số, mà khoá công khai của chữ ký điện tử số này được sử dụng với;

tên và địa điểm của trung tâm chứng thực cấp ra chứng nhận khoá chữ ký;

các chỉ dẫn về các quan hệ mà khi thực hiện nó thì văn bản điện tử với chữ ký điện tử số sẽ có giá trị pháp lý.

2. Trong trường hợp cần thiết, trong chứng nhận khoá chữ ký, trên cơ sở các văn bản đã được xác nhận còn chỉ ra chức vụ (cùng với tên và địa điểm của cơ quan

đã lập ra chức danh đó) và nghề nghiệp của người chủ chứng nhận khoá chữ ký, còn theo sự xuất trình ở dạng viết - những tư liệu khác đã được xác nhận bởi các giấy tờ khác.

3. Chứng nhận khoá chữ ký cần phải được trung tâm chứng thực đưa vào danh sách của các chứng nhận khoá chữ ký không muộn hơn ngày bắt đầu có hiệu lực của chứng nhận khoá chữ ký.
4. Để kiểm tra tính sở hữu chữ ký điện tử số với người chủ của chứng nhận khoá chữ ký, những người sử dụng được trao thông tin về ngày và thời gian cấp chứng nhận, tư liệu về hiệu lực của chứng nhận khoá chữ ký (có hiệu lực, dừng có hiệu lực, thời hạn dừng có hiệu lực, đã bị huỷ bỏ, ngày và thời gian huỷ bỏ chứng nhận khoá chữ ký) và tư liệu về danh sách của các chứng nhận khoá chữ ký. Trong trường hợp cấp chứng nhận khoá chữ ký ở dạng văn bản trên giấy, chứng nhận này được thể hiện theo khuôn mẫu của trung tâm chứng thực và được làm tin bởi chữ ký tay của người có chức trách và dấu của trung tâm chứng thực. Trong trường hợp cấp chứng nhận khoá chữ ký và các dữ liệu bổ sung đã nêu ở dạng văn bản điện tử, chứng nhận đó cần phải được ký bởi chữ ký điện tử số của người có trách nhiệm thuộc trung tâm chứng thực.

Điều 7. Thời hạn và cách thức lưu giữ chứng nhận khoá chữ ký tại trung tâm chứng thực

1. Thời hạn lưu giữ chứng nhận khoá chữ ký ở dạng văn bản điện tử tại trung tâm chứng thực được xác định bằng thoả thuận giữa trung tâm chứng thực và người chủ của chứng nhận khoá chữ ký. Khi đó đảm bảo quyền truy nhập của những người tham gia hệ thống thông tin vào trung tâm chứng thực để nhận được chứng nhận khoá chữ ký.
2. Thời hạn lưu giữ chứng nhận khoá chữ ký ở dạng văn bản điện tử tại trung tâm chứng thực sau khi huỷ bỏ chứng nhận khoá chữ ký cần không được ít hơn thời hạn được thiết lập bởi luật liên bang về thời hạn kiện tụng đối với các quan hệ được chỉ ra trong chứng nhận khoá chữ ký.
   Khi hết thời hạn lưu giữ đã chỉ ra, chứng nhận khoá chữ ký được loại bỏ khỏi danh sách của các chứng nhận khoá chữ ký và được đưa vào chế độ lưu trữ.
   Thời hạn lưu trữ không ít hơn 5 năm. Quy tắc đưa bản sao của chứng nhận khoá chữ ký trong giai đoạn này được thiết lập theo như luật pháp Liên bang Nga.
3. Chứng nhận khoá chữ ký ở dạng văn bản trên giấy được lưu trữ theo quy định được thiết lập theo luật pháp Liên bang Nga về văn khố và lưu trữ.

Chương III. Các trung tâm chứng thực

Điều 8. Vị trí của trung tâm chứng thực

1. Cơ quan luật pháp thực hiện các chức năng được xem xét bởi luật Liên bang này là trung tâm chứng thực cấp các chứng nhận khoá chữ ký để sử dụng trong các hệ thống thông tin dùng chung. Khi đó, trung tâm chứng thực cần phải có các khả năng tài chính và vật chất cần thiết, cho phép nó thi hành trách nhiệm

dân sự trước những người sử dụng chứng nhận khoá chữ ký đối với những thiệt hại mà có thể gây ra do tính không tin cậy của những tư liệu chứa ở trong các chứng nhận khoá chữ ký.

Các yêu cầu đối với các điều kiện vật chất và tài chính của trung tâm chứng thực được xác định bởi chính phủ liên bang theo trình bày của cơ quan toàn quyền liên bang của chính quyền hành pháp.

Vị trí của trung tâm chứng thực, đảm bảo sự hoạt động của hệ thống thông tin doanh nghiệp, được xác định bởi người chủ của nó hoặc bằng thoả thuận của những người tham gia hệ thống này.

2. Hoạt động của trung tâm chứng thực thuộc về lĩnh vực cấp phép theo như luật pháp Liên bang Nga về cấp phép hoạt động của các dạng khác nhau.

***Điều 9*. Hoạt động của trung tâm chứng thực**

1. Trung tâm chứng thực:

chuẩn bị chứng nhận khoá chữ ký;

sinh ra các khoá của chữ ký điện tử số theo yêu cầu của những người tham gia hệ thống thông tin với đảm bảo giữ kín khoá bí mật chữ ký điện tử số;

dừng và cho khôi phục hiệu lực của các chứng nhận khoá chữ ký, cũng như việc huỷ bỏ nó;

ghi danh sách các chứng nhận khoá chữ ký, đảm bảo tính thời sự và khả năng truy nhập tự do đến nó của mọi thành viên tham gia hệ thống thông tin;

kiểm tra tính duy nhất của khoá công khai chữ ký điện tử số trong danh sách các chứng nhận khoá chữ ký và trong lưu trữ của trung tâm chứng thực.

đưa ra các chứng nhận khoá chữ ký trong dạng văn bản trên giấy và (hoặc) trong dạng văn bản điện tử cùng với thông tin hiệu lực của nó;

thực hiện theo yêu cầu của người sử dụng chứng nhận khoá chữ ký việc khẳng định tính chân thực của chữ ký điện tử số trong văn bản điện tử trong quan hệ của các chứng nhận khoá chữ ký đã được trung tâm đưa ra;

có thể cung cấp cho những người tham gia hệ thống thông tin những dịch vụ khác có liên quan tới việc sử dụng chữ ký điện tử số.

2. Việc chuẩn bị các chứng nhận khoá chữ ký được thực hiện trên cơ sở đơn xin của người tham gia hệ thống thông tin có chứa các thông tin được chỉ ra ở điều 6 bộ luật Liên bang này và các tư liệu cần thiết để nhận dạng người chủ khoá chữ ký. Đơn được ký bằng tay bởi người chủ của chứng nhận khoá chữ ký. Các thông tin chứa trong đơn cần phải được khẳng định bằng cách trình ra các giấy tờ tương ứng.
3. Khi chuẩn bị các chứng nhận khoá chữ ký bởi trung tâm chứng thực, hai bản chứng nhận khoá chữ ký được thể hiện ở dạng văn bản trên giấy, nó phải được ký bằng tay bởi người chủ của chứng nhận khoá chữ ký và người đại diện của trung tâm chứng thực, và cả dấu của trung tâm chứng thực. Một bản chứng nhận khoá chữ ký được trao cho người chủ chứng nhận khoá chữ ký, bản thứ hai còn lại ở trung tâm chứng thực.

4. Các dịch vụ để cấp cho những người tham gia hệ thống thông tin những chứng nhận khoá chữ ký, được đăng ký bởi trung tâm chứng thực, đồng thời cùng với thông tin về hiệu lực của nó ở dạng văn bản điện tử là không phải trả tiền.

Điều 10. Quan hệ giữa các trung tâm chứng thực và các cơ quan toàn quyền liên bang của chính quyền hành pháp

1. Trung tâm chứng thực cho đến khi bắt đầu sử dụng chữ ký điện tử số của người đại diện trung tâm chứng thực để cam đoan thay mặt trung tâm chứng thực các chứng nhận khoá chữ ký nhất định phải trình tại cơ quan đại diện liên bang của chính quyền hành pháp chứng thực của người đại diện trung tâm chứng thực ở dạng văn bản điện tử cũng như chứng nhận này ở dạng văn bản trên giấy cùngvới chữ ký tay của người đại diện đã nêu, được cam đoan bởi chữ ký của người lãnh đạo và dấu của trung tâm chứng thực.
2. Cơ quan đại diện liên bang của chính quyền hành pháp thực hiện một danh sách quốc gia duy nhất các chứng nhận khoá chữ ký được đưa ra bởi các trung tâm chứng thực làm việc với những người tham gia các hệ thống thông tin dùng chung (các trung tâm này cam đoan về những chứng nhận khoá chữ ký do mình đưa ra), đảm bảo khả năng truy nhập tự do tới danh sách này và cấp các chứng nhận khoá chữ ký cho những đại diện tương ứng của các trung tâm chứng thực.
3. Các chữ ký điện tử số của những người đại diện các trung tâm chứng thực có thể được sử dụng chỉ sau khi đưa nó vào một danh sách hợp nhất toàn quốc gia gồm các chứng nhận khoá chữ ký. Việc sử dụng các chữ ký điện tử này vào các mục đích không liên quan đến việc cam đoan các chứng nhận khoá chữ ký và các tư liệu về hiệu lực của nó là không cho phép.
4. Cơ quan đại diện liên bang của chính quyền hành pháp:

thực hiện theo yêu cầu của mọi người, tổ chức, cơ quan liên bang của chính quyền hành pháp, các cơ quan chính quyền quốc gia của các chủ thể thuộc Liên bang Nga và các cơ quan tự điều hành địa phương việc khẳng định tính đúng đắn của chữ ký điện tử số của các đại diện trung tâm chứng thực trong các chứng nhận khoá chữ ký do họ cấp phát;

thực hiện theo các điều khoản về cơ quan đại diện liên bang của chính quyền lập pháp các uỷ quyền khác để đảm bảo hiệu lực của bộ luật Liên bang này.

Điều 11. Các trách nhiệm của trung tâm chứng thực trong quan hệ với người chủ chứng nhận khoá chữ ký

Trung tâm chứng thực khi chuẩn bị chứng nhận khoá chữ ký nhận về mình các trách nhiệm sau theo quan hệ với người chủ của chứng nhận khoá chữ ký:

đưa chứng nhận khoá chữ ký vào danh sách chứng nhận khoá chữ ký;

đảm bảo việc cung cấp chứng nhận khoá chữ ký cho những người tham gia hệ thống thông tin yêu cầu;

dừng hiệu lực của chứng nhận khoá chữ ký theo yêu cầu người chủ của nó;

thông báo cho người chủ chứng nhận khoá chữ ký về các sự kiện được biết bởi trung tâm chứng thực và bằng một cách nào đó có thể ảnh hưởng đến khả năng sử dụng tiếp theo của chứng nhận khoá chữ ký;

các tránh nhiệm khác được thiết lập bằng các điều khoản luật chuẩn mực hoặc thoả thuận của các bên.

***Điều 12*. Các trách nhiệm của người chủ chứng nhận khoá chữ ký**

1. Người chủ của chứng nhận khoá chữ ký phải:

không sử dụng cho chữ ký điện tử số các khóa công khai và bí mật của chữ ký điện tử số nếu như biết rằng các khoá đó đang được sử dụng hoặc đã được sử dụng;

giữ bí mật khoá mật của chữ ký điện tử số;

ngay lập tức yêu cầu dừng hiệu lực của chứng nhận khoá chữ ký khi có cơ sở cho rằng bí mật đối với khoá mật chữ ký điện tử số bị vi phạm.

2. Khi không tuân thủ những yêu cầu được nêu ra trong điều này, việc đền bù những thiệt hại gây ra do nó được gán cho trách nhiệm của người chủ chứng nhận khoá chữ ký.

***Điều 13*. Chấm dứt hiệu lực của chứng nhận khoá chữ ký**

1. Hiệu lực của chứng nhận khoá chữ ký có thể bị dừng bởi trung tâm chứng thực trên cơ sở chỉ ra người hay cơ quan có quyền như vậy theo như luật hay thoả ước, còn trong hệ thống thông tin doanh nghiệp còn theo như các quy tắc đã được thiết lập dể sử dụng nó.
2. Thời gian từ lúc nhận được tại trung tâm chứng thực lệnh về việc dừng hiệu lực của chứng nhận khoá chữ ký cho đến khi đưa thông tin tương ứng vào danh sách chứng nhận khoá chữ ký cần được thiết lập theo như quy tắc chung đối với tất cả những người chủ chứng nhận khoá chữ ký.
3. Hiệu lực của chứng nhận khoá chữ ký theo lệnh của người (cơ quan) đại diện toàn quyền được dừng lại trong một thời hạn được tính theo ngày nếu hiệu lực đó không được thiết lập bởi các quy định luật chuẩn tắc hoặc thoả ước. Trung tâm chứng thực khôi phục hiệu lực của chứng nhận khoá chữ ký theo lệnh của người (cơ quan) đại diện toàn quyền. Trong trường hợp, nếu đã hết hạn chỉ ra mà không có lệnh về việc khôi phục hiệu lực của chứng nhận khoá chữ ký thì chứng nhận đó bị huỷ bỏ.
4. Theo lệnh của người (cơ quan) đại diện về dừng hiệu lực của chứng nhận khoá chữ ký, trung tâm chứng thực loan báo điều này cho mọi người sử dụng chứng nhận khoá chữ ký bằng cách đưa vào danh sách chứng nhận khoá chữ ký thông tin tương ứng chỉ ra ngày, thời gian và thời hạn dừng hiệu lực của chứng nhận khoá chữ ký, đồng thời thông báo cho người chủ của chứng nhận khoá chữ ký và người (cơ quan) đại diện đã ra lệnh dừng hiệu lực của chứng nhận khoá chữ ký.

***Điều 14*. Huỷ bỏ chứng nhận khoá chữ ký**

1. Trung tâm chứng thực, nơi đã cấp chứng nhận khoá chữ ký, bắt buộc phải huỷ bỏ nó:

khi hết thời hạn có hiệu lực;

khi bị mất sức mạnh pháp lý về chứng nhận của các phương tiện tương ứng của chữ ký điện tử số được sử dụng trong các hệ thống dùng chung;

trong trường hợp nếu trung tâm chứng thực biết chắc chắn việc dừng hiệu lực của văn bản trên cơ sở đó lập ra chứng nhận khoá chữ ký;

trong các trường hợp khác được thiết lập bởi các điều luật chuẩn hoặc thoả thuận giữa các bên.

2. Trong trường hợp huỷ bỏ chứng nhận khoá chữ ký, trung tâm chứng thực thông báo về việc đó đến mọi người sử dụng chứng nhận khoá chữ ký bằng cách đưa vào danh sách các chứng nhận khoá chữ ký thông tin tương ứng cùng với việc chỉ ra ngày, giờ huỷ bỏ chứng nhận khoá chữ ký, trừ trường hợp huỷ bỏ chứng nhận khoá chữ ký khi hết thời hạn hiệu lực của nó, đồng thời cũng thông báo điều này cho người chủ chứng nhận khoá chữ ký và người (cơ quan) đại diện đã ra lệnh huỷ bỏ chứng nhận khoá chữ ký.

Điều 15. Chấm dứt hoạt động của trung tâm chứng thực

1. Hoạt động của trung tâm chứng thực nơi cấp chứng nhận khoá chữ ký để sử dụng trong các hệ thống thông tin dùng chung, có thể được dừng lại theo quy tắc được thiết lập bởi luật dân sự.
2. Trong trường hợp dừng hoạt động của trung tâm chứng thực đã được chỉ ra ở điểm 1 của điều này, các chứng nhận khoá chữ ký được cấp bởi trung tâm chứng thực đó, có thể được chuyển giao cho một trung tâm chứng thực khác theo thoả thuận với những người chủ chứng nhận khoá chữ ký.
   Các chứng nhận khoá chữ ký không được chuyển cho một trung tâm chứng thực khác, được huỷ bỏ và chuyển cơ quan toàn quyền liên bang để đưa vào lưu trữ theo điều 7 của luật Liên bang này.
3. Hoạt động của trung tâm chứng thực đảm bảo hoạt động của hệ thống thông tin doanh nghiệp được dừng theo quyết định của người chủ hệ thống đó, đồng thời cũng theo thoả thuận của những người tham gia hệ thống nếu có sự chuyển giao trách nhiệm của trung tâm này cho một trung tâm khác hoặc khi xoá bỏ hệ thống thông tin doanh nghiệp.

Chương IV. Các đặc điểm sử dụng chữ ký điện tử số

Điều 16. Sử dụng chữ ký điện tử số trong lĩnh vực điều hành nhà nước

1. Các cơ quan liên bang của chính quyền hành pháp, các cơ quan chính quyền nhà quốc gia của các chủ thể trong Liên bang Nga, các cơ quan tự điều hành địa phương và các tổ chức tham gia vào việc trao đổi văn bản với các cơ quan đã nêu trên sử dụng các chữ ký điện tử số của những người lãnh đạo các cơ quan, tổ chức đó để ký các văn bản điện tử của mình.
2. Các chứng thực khoá chữ ký của những người đại diện toàn quyền các cơ quan liên bang của chính quyền quốc gia được đưa vào danh sách các chứng nhận khoá chữ ký được quản lý bởi cơ quan đại diện liên bang của chính quyền hành pháp, và được cấp phát cho những người sử dụng chứng nhận khoá chữ ký từ

danh sách này theo quy định được thiết lập bởi bộ luật Liên bang này cho các trung tâm chứng thực.

3. Quy tắc cấp chứng nhận khoá chữ ký của những người đại diện các cơ quan chính quyền quốc gia của các chủ thể trong Liên bang Nga và những người đại diện các cơ quan tự điều hành địa phương được thiết lập bởi các điều luật chuẩn tắc của các cơ quan tương ứng.

Điều 17. Sử dụng chữ ký điện tử số trong hệ thống thông tin doanh nghiệp

1. Hệ thống thông tin doanh nghiệp cung cấp cho những người tham gia hệ thống thông tin sử dụng chung các dịch vụ của trung tâm chứng thực của hệ thống thông tin doanh nghiệp cần phải đáp ứng được các yêu cầu được quy định bởi bộ luật Liên bang này cho các hệ thống thông tin dùng chung.
2. Quy tắc sử dụng chữ ký điện tử số trong hệ thống thông tin doanh nghiệp được thiết lập bởi quyết định của người chủ hệ thống đó hay bởi thoả thuận của những người tham gia hệ thống.
3. Nội dung thông tin trong các chứng nhận khoá chữ ký, quy tắc đưa vào danh sách các chứng nhận khoá chữ ký, quy định lưu giữ các chứng nhận khoá chữ ký đã được huỷ bỏ, các trường hợp mất hiệu lực pháp lý của các chứng nhận đã nêu ra trong hệ thống thông tin doanh nghiệp được thể chế hoá bằng quyết định của người chủ hệ thống hay bởi thoả thuận giữa những người tham gia hệ thống thông tin doanh nghiệp.

Điều 18. Công nhận chứng nhận khóa chữ ký của nước ngoài

Chứng nhận khoá chữ ký của nước ngoài, được chứng thực theo như luật pháp của nước mà chứng nhận khoá chữ ký ấy đã đăng ký, được công nhận trên lãnh thổ Liên bang Nga trong trường hợp thực hiện các qui trình được thiết lập bởi luật pháp Liên bang Nga về việc công nhận giá trị pháp lý của các văn bản nước ngoài.

Điều 19. Các trường hợp thay thế con dấu

1. Nội dung văn bản trên giấy, được đảm bảo bởi con dấu và được chuyển thành văn bản điện tử, theo như các điều luật chuẩn hoặc thoả thuận giữa các bên có thể được đảm bảo bằng chữ ký điện tử số của người đại diện cơ quan cơ con dấu.
2. Trong các trường hợp được quy định bởi luật hoặc các điều luật chuẩn khác của Liên bang Nga hoặc thoả thuận các bên, chữ ký điện tử số trong văn bản điện tử, chứng nhận của nó chứa các tư liệu cần thiết để thực hiện các quan hệ đang xét đến về quyền lực của người chủ, được công nhận có giá trị như chữ ký tay của người trên văn bản bằng giấy được đảm bảo bằng con dấu.

Chương V. Các điều khoản thi hành và chuyển giao

Điều 20. Thi hành các văn bản pháp lý chuẩn mực theo như luật Liên bang này.

1. Các điều luật chuẩn của Liên bang Nga được đưa vào hoạt động theo như luật Liên bang này trong vòng 3 tháng kể từ ngày luật Liên bang này có hiệu lực.
2. Các văn bản hành chính của các trung tâm chứng thực, nơi cấp các chứng nhận khoá chữ ký để sử dụng trong các hệ thống thông tin dùng chung, được đưa vào hoạt động theo luật Liên bang này trong vòng 6 tháng kể từ ngày nó có hiệu lực.

Điều 21. Các điều khoản chuyển giao

Các trung tâm chứng thực được thành lập sau ngày luật Liên bang này có hiệu lực, trước khi được cơ quan đại diện liên bang của chính quyền hành pháp đưa vào danh sách các chứng nhận khoá chữ ký cần phải đáp ứng các yêu cầu của luật Liên bang này, ngoại trừ yêu cầu xuất trình trước đó các chứng nhận khoá chữ ký của các người đại diện của mình trước cơ quan đại diện liên bang của chính quyền hành pháp. Các chứng nhận tương ứng cần phải trình cho các cơ quan tương ứng không chậm hơn 3 tháng sau kể từ ngày có hiệu lực của luật Liên bang này.

Tổng thống Liên bang Nga **V. Putin**

Phụ lục 1. Mô tả DSS- chuẩn chữ ký số của Mỹ[7]

Ngày 7 tháng 1 năm 2000, nước Mỹ đã ra chuẩn chữ ký số mới, đó là chuẩn DSS (Digital Signature Standard) (FIPS PUB 186-2). Theo chuẩn này, chữ ký điện tử số có thể được tạo ra theo một trong 3 thuật toán sau:

- DSA (Digital Signature Algorithm) dựa vào bài toán logarithm rời rạc
- RSA DSA (ANSI X9.31)
- EC DSA (ANSI X9.63) dựa vào bài toán logarithm trong nhóm các điểm của đường cong eliptic trên trường hữu hạn.

Bây giờ chúng ta sẽ lần lượt mô tả 3 thuật toán trên.

Thuật toán DSA dựa vào bài toán logarithm rời rạc

1. Giai đoạn chuẩn bị, chọn các tham số

Chọn các số p, q và g sao cho p là số nguyên tố, $2^{l-1} < p < 2^l$ với l là bội của 64 và $512 \le l \le 1024$; q là ước nguyên tố của p-1 dài 160 bit ($2^{159} < q < 2^{160}$); g là phần tử bậc q trong Z_p. g được chọn ở dạng $g= h^{(p-1)/q}$ với $1 < h < p-1$ và $g= h^{(p-1)/q} > 1$. Ba số này được công bố công khai và chung cho tất cả mọi người trong hệ thống.

2. Tạo chữ ký

Tính giá trị hàm băm của m là h(m). Hàm băm được dùng là SHA-1 được công nhận là chuẩn của Mỹ từ năm 1992 (FIPS PUB 180-1). Giá trị của hàm băm h(m) có độ dài 160 bit.

Tiếp theo, người ký chọn ngẫu nhiên giá trị k, $0 < k < q$, tính k^{-1} (mod q) và tính cặp số

$$r = gk \pmod p \pmod q$$
$$s = k^{-1} (h(m)+xr) \pmod q$$

Cặp giá trị (r,s) là chữ ký điện tử của thông báo m. Sau khi sinh chữ ký số, giá trị k được huỷ bỏ.

3. Kiểm tra chữ ký

Giả sử nhận được thông báo m. Phương trình kiểm tra là $r \equiv g^{h(m).s^{-1}} .y^{r.s^{-1}} \pmod p \pmod q$. Thực vậy:

$$g^{h(m).s^{-1}} .y^{r.s^{-1}} \pmod p \pmod q = g^{h(m).s^{-1}} .g^{x.r.s^{-1}} \pmod p \pmod q =$$
$$= g^{s^{-1}(h(m)+xr)} \pmod p \pmod q = g^{(k^{-1}(h(m)+xr))^{-1}(h(m)+xr)} \pmod p \pmod q =$$
$$= g^k \pmod p \pmod q \equiv r$$

[7] C.U.Mfhbxtd, D.D. Ujyxfhjd, H.T.Cthjd, Jcyjds cjdhtvtyyjq rhbgnjuhfabb, Vjcrdf, Ujhzxfz kbybz-Ntktrjv, 2002, cnh. 92-96.

Thuật toán RSA DSA

Thuật toán RSA đã khá quen thuộc nên không được mô tả. Cái mà chúng ta quan tâm đến là độ dài số cũng không được đề cập đến.

Thuật toán dựa trên đường cong eliptic

1. Chọn tham số

Sử dụng các trường hữu hạn nguyên tố Galois và trường Galois có đặc số 2. Trên các trường này lại xét các đường cong eliptic. Việc chọn các trường số trong chuẩn được làm để tăng hiệu quả tính toán của các phép nhân trong trường. Vì mục đích đó, các modulo nguyên tố được chọn là các số Mercen tổng quát (ở bảng 1).

Chuẩn chọn cố định các đường cong được sử dụng trong các thuật toán và các điểm cơ sở ví dụ trên các đường cong đó. Người sử dụng có thể sử dụng các điểm cơ sở đã được nêu trong chuẩn, hoặc có thể tự sinh ra (nếu như muốn đảm bảo phân biệt mật mã các mạng máy tính). Nói riêng, các đường cong P-192, P-224, P-256, P-384, P-521 là các đường cong eliptic có bậc nguyên tố dạng $y^2 = x^3 - 3x + b$ trên trường GF(p).

Bảng 1

Đường cong	p
P-192	$2^{192} - 2^{64} - 1$
P-224	$2^{224} - 2^{96} + 1$
P-256	$2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$
P-384	$2^{384} - 2^{128} - 2^{96} + 2^{32} - 1$
P-521	$2^{521} - 1$

Để xây dựng các đường cong trên trường đặc trưng 2 chọn các đa thức sinh như ở bảng 2. Có thể sử dụng biểu diễn trường như một đa thức hoặc trong một cơ sở chuẩn.

Bảng 2

Đường cong	Đa thức sinh p(t)	Dạng của cơ sở chuẩn
K-163, B-163	$t^{163} + t^7 + t^6 + t^3 + 1$	4
K-233, B-233	$t^{233} + t^{74} + 1$	2
K-283, B-283	$t^{283} + t^{12} + t^7 + t^5 + 1$	6
K-409, B-409	$t^{409} + t^{87} + 1$	4
K-571, B-571	$t^{571} + t^{10} + t^5 + t^2 + 1$	10

Các hằng số b được chọn theo kích thước của từng trường. Ví dụ, đường cong P-521 được cho bởi hệ số ở dạng cơ số 16:

b = 051 953eb961 8e1c9a1f 929a21a0 b68540ee a2da725b 99b315f3
b8b48991 8ef109e1 56193951 ec7e937b 1652c0bd 3bb1bf07
3573df88 3d2c34f1 ef451fd4 6b503f00

và nó có bậc là (theo hệ cơ số 10):

r = 68647976601306097149819007990813932172694353001430540939446345918554318339765539424505774633321719753296399637136332111386476861244038034037280889270700544 9

Vì các đường cong định nghĩa trong chuẩn có bậc nguyên tố, các nhóm điểm trong chúng là nhóm vòng (phần phụ đại số (cofactor) của các đường cong bằng 1) và bậc của điểm cơ sở n đúng bằng r.

Với mỗi trường GF(2^m) trong chuẩn chỉ ra 2 đường cong: đường cong giả ngẫu nhiên dạng $y^2+xy = x^3+x^2+b$ có phần phụ đại số bằng 2 và đường cong đặc biệt Kobliz, hay đường nhị phân dị thường dạng $y^2+xy=x^3+ax^2+1$ với a=0 hay 1. Phần phụ đại số của đường Kobliz bằng 2 nếu a=1 và bằng 4 nếu a=0. Ví dụ, đối với trường GF(2^{163}) thì đường cong giả ngẫu nhiên được cho bởi hằng số
b= 2 0a601907 b8c953ca 1481eb10 512f7874 4a3205fd, còn đường Kobliz được cho bởi hệ số a=1.

2. Sinh cặp chìa khoá

Người sử dụng chọn khoá bí mật là số nguyên d, $0 < d < n$, với n là bậc của điểm cơ sở G trên đường cong eliptic. Tính Q = dG và công bố Q như là khoá công khai

3. Sinh chữ ký số

Để ký thông báo m, người sử dụng làm như sau;

1) chọn số ngẫu nhiên k, $0 < k < n-1$;
2) tính $kG=(x_1, y_1)$, $r=x_1 \bmod n$. Nếu r=0 thì quay trở về bước 1;
3) Tính $k^{-1} \bmod n$;
4) Tính e=SHA(m). Chọn phiên bản của hàm băm phụ thuộc vào trường số được dùng.
5) tính $s=k^{-1}(e+dr) \bmod n$. Nếu s=0 thì quay lại bước 1.

Chữ ký của m là cặp (r,s).

4. Kiểm tra chữ ký

Để kiểm tra chữ ký người nhận thực hiện các thao tác sau:

1) kiểm tra rằng r và s nằm trong khoảng (0, n);
2) tính e = SHA(m);
3) xác định $w=s^{-1} \bmod n$;
4) tìm $u_1=ew \bmod n$ và $u_2 = rw \bmod n$;
5) tính $X= u_1G +u_2Q$. Nếu $X= \infty$ thì chữ ký bị loại bỏ, ngược lại, tính $v=x_1 \bmod n$, với $X = (x_1, y_1)$;

Chữ ký chỉ được chấp nhận nếu như v=r.

Tính đúng đắn của lược đồ trên chứng minh không khó. Nếu m thực sự do người gửi chuyển đến thì $s = k^{-1}(e+dr) \bmod n$. Từ đó

$$k \equiv s^{-1}(e+dr) \equiv s^{-1}e+s^{-1}rd \equiv we +wrd \equiv u_1 +u_2d \pmod n.$$

Như vậy, $u_1G+u_2Q = (u_1+u_2d)G = kG$, vì vậy v=r.

Chú ý: trong tài liệu cũng không đề cập đến hàm băm đi cùng với các thuật toán RSA DSA và EC DSA, nhưng có lẽ là SHA-1 (có giá trị băm gồm 160-bit).

Phụ lục 2. Họ các hàm băm SHA[8]

SHA-1

Hàm băm SHA-1 (Secure Hash Algorithm) được chấp nhận như chuẩn ở Mỹ năm 1992 và được dùng cùng với thuật toán chữ ký số DSS lúc bấy giờ (chính là phương án thứ nhất trong chuẩn DSS năm 2000). Với bản tin M, thuật toán sinh ra một thông báo có 160-bit được gọi là Message Digest và cái đó được dùng khi sinh ra chữ ký số. Bây giờ ta sẽ xem xét thuật toán chi tiết hơn.

Trước hết, thông báo ban đầu được thêm vào sao cho độ dài của nó là bội của 512 bit. Ngay cả khi độ dài của thông báo đã là bội của 512 bit thì việc thêm vào vẫn được thực hiện. Việc thêm vào được thực hiện như sau: thêm số 1, sau đó là các số 0 sao cho độ dài có được là 64 nhỏ hơn bội của 512 bit, sau đó là 64 bit biểu diễn độ dài của thông báo ban đầu.

Sau đó đặt giá trị ban đầu cho 5 biến có 32 bit bằng các hằng số cơ số 16:

A=67452301
B=EFCDAB89
C=98BADCFE
D=10325476
E=C3D2E1F0

Tiếp theo, 5 biến này được copy vào các biến mới a, b, c, d, e.

Chu trình chính có thể mô tả đơn giản bằng mã lệnh như sau:

```
for (t=0; t<80;t++) {
        temp = (a<<<5) + f_t(b,c,d)+ e + W_t + K_t;
        e=d;d=c;c=b<<<30; b=a;a=temp;
}
```

với <<< là phép toán dịch vòng về bên trái; K_t là các hằng số cơ số 16 được xác định theo công thức sau:

$$K_t = \begin{cases} 5A827999, t = 0..19 \\ 6ED9EBA1, t = 20..39 \\ 8F1BBCDC, t = 40..59 \\ CA62C1D6, t = 60..79 \end{cases}$$

các hàm $f_t(x,y,z)$ được chi như sau;

[8] C.U.Mfhbxtd, D.D. Ujyxfhjd, H.T.Cthjd, Jcyjds cjdhtvtyyjq rhbgnjuhfabb, Vjcrdf, Ujhzxfz kbybz-Ntktrjv, 2002, cnh. 114-121.

$$f(x,y,z)=\begin{cases} x\wedge y\vee\neg x\wedge z, t=0..19 \\ x\oplus y\oplus z, t=20..39,60..79 \\ x\wedge y\vee x\wedge z\vee y\wedge z, t=40..59 \end{cases}$$

các giá trị W_t nhận được từ các khối 32-bit của khối 512 bit đang được xử lý của thông báo đã được mở rộng theo quy tắc sau:

$$W_t=\begin{cases} M_t, t=0..19 \\ (W_{t-3}\oplus W_{t-8}\oplus W_{t-14}\oplus W_{t-16})<<<1, t=20..79 \end{cases}$$

Sau khi kết thúc chu trình chính các giá trị a, b, c, d và e được cộng tương ứng với A, B, C, D và E; sau đó chuyển sang xử lý khối 512 bit tiếp theo của thông báo mở rộng. Giá trị đầu ra của hàm băm là các giá trị A, B, C, D và E được ghép lại.

Các hàm băm SHA-256, SHA-512 và SHA-384

Tính bền vững của hàm băm trong việc tìm một đầu vào có cùng giá trị băm gần bằng $2^{n/2}$ với n là độ dài của giá trị đầu ra. Do ở Mỹ đã nghiên cứu ra chuẩn mã dữ liệu mới với độ dài khoá bằng 128, 192 và 256 bit nên có nhu cầu lập ra các thuật toán băm đồng hành cũng có độ bền vững như vậy. Bây giờ chúng tôi sẽ mô tả các thuật toán tính hàm băm có độ dài đầu ra là 256, 512 và 384 bit, chúng được đề nghị công nhận là chuẩn mới ở Mỹ.

Trước hết xem xét thuật toán SHA-256. Việc mô tả nó có thể chia làm 2 phần: hàm nén và thuật toán xử lý thông báo. Hàm nén về thực chất chính là thuật toán mã khối có kích thước khối bằng 256 của giá trị hàm băm trung gian với khoá là khối hiện hành của thông báo[9] . Bên cạnh những ký hiệu thông thường còn sử dụng các ký hiệu sau: R^n - phép dịch từ về bên phải n vị trí; S^n -phép dịch vòng từ về bên phải n vị trí. Kích thước từ là 32 bit. Phép cộng được tiến hành theo modulo 2^{32}. Vecto khởi điểm của phép băm $H^{(0)}$ là một bộ gồm 8 từ 32 bit, đó chính là phần thập phân của căn bậc 2 tính từ 8 số nguyên tố đầu tiên:

H(0)= {6a09e667, bb67ae85, 3c6ef372, a54ff53a, 510e527f, 9b05688c,
1f83d9ab, 5be0cd19}.

Các tính toán tiếp theo như sau:

1. Tiền xử lý. Thông báo đem băm được kéo dài sao cho độ dài là bội của 512 bit. Cách thêm vào giống như đối với SHA-1.
2. Phân chia. Thông báo đã được kéo dài được chia thành các khối 512 bit $M^{(1)}$,..., $M^{(N)}$.
3. Chu trình chính:

for i=1 to N {
// khởi đầu các thanh ghi a, b, c, d, e, f, g, h của giá trị trung gian thứ (i-1)

[9] Dùng mã khối để kiến thiết hàm băm, điều này giống với GOST P 34.11-94.

$a = H_1^{(i-1)}; b = H_2^{(i-1)}; c = H_3^{(i-1)}; d = H_4^{(i-1)};$

$e = H_5^{(i-1)}; f = H_6^{(i-1)}; g = H_7^{(i-1)}; h = H_8^{(i-1)};$

// áp dụng hàm nén SHA-256 cho các thanh ghi a, b,..., h

for j=0 to 63 {

Tính Ch(e,f,g), Maj(a,b,c), $\Sigma_0(a)$, $\Sigma_1(e)$, và W_j // công thức ở sau

T_1=h+ $\Sigma_1(e)$ +Ch(e,f,g) +K_j +W_j;

T_2=$\Sigma_0(a)$ +Maj(a,b,c) ;

h=g; g=f; f=e; e=d+T_1;

d=c; c=b; b=a; a=T_1+T_2;

}

// tính $H^{(i)}$ là giá trị trung gian thứ i

$H_1^{(i)} = a + H_1^{(i-1)}; H_2^{(i)} = b + H_2^{(i-1)}; ...; H_8^{(i)} = h + H_8^{(i-1)};$

}

Giá trị cuối cùng của hàm băm là $H^{(N)} = (H_1^{(N)}, H_2^{(N)}, ..., H_8^{(N)})$.

Trong SHA-256 sử dụng 6 hàm logic, các tham số của chúng là các từ 32-bit.

Ch(x,y,z) = (x$\wedge$y)$\oplus$ ($\neg$x$\wedge$z)

Maj(x,y,z) = (x$\wedge$y)$\oplus$ (x$\wedge$z) $\oplus$(y$\wedge$z)

$\Sigma_0(x)$=$S^2(x)\oplus$ $S^{13}(x)\oplus$ $S^{22}(x)$

$\Sigma_1(x)$=$S^6(x)\oplus$ $S^{11}(x)\oplus$ $S^{25}(x)$

$\sigma_0(x)$=$S^7(x)\oplus$ $S^{18}(x)\oplus$ $R^3(x)$

$\sigma_1(x)$=$S^{17}(x)\oplus$ $S^{19}(x)\oplus$ $R^{10}(x)$

Các khối thông báo được mở rộng W_0,..., W_{63} được tính như sau:

$W_j = M_j^{(i)}, j = 0,...,15;$

for $j = 16$ *to* 63 {

$W_j = \sigma_1(W_{j-2}) + W_{j-7} + \sigma_0(W_{j-15}) + W_{j-16}$

}

Các hằng số từ K_0,..., K_{63} được lấy là 32 bit đầu của căn bậc 3 từ 64 số nguyên tố đầu, ở dạng hex là

428a2f98	71374491	b5c0fbcf	e9b5dba5	3956c25b
59f111f1	923f82a4	ab1c5ed5	d807aa98	12835b01
243185be	550c7dc3	72be5d74	80deb1fe	9ddc06a7
c19bf174	e49b69c1	efbe4786	0fc19dc6	240ca1cc
2de92c6f	4a7484aa	5cb0a9dc	76f988da	983e5152
a831c66d	b00327c8	bf597fc7	c6e00bf3	d5a79147

06ca6351	14292967	27b70a85	2e1b2138	4d2c6dfc
53380d13	650a7354	766a0bb	81c2c92e	92722c85
a2bfe8a1	a81a664b	c24b8b70	c76c51a3	d192e819
d6990624	f40e3585	106aa070	19a4c116	1e376c08
2748774c	34b0bcb5	391c0cb3	4ed8aa4a	5b9cca4f
682e6ff3	748f82ee	78a5636f	84c87814	8cc70208
90befffa	a4506ceb	bef9a3f7	c67178f2	

Hàm SHA-512 có cấu trúc giống như SHA-256, nhưng làm việc với các từ 64 bit. Thoạt đầu văn bản được thêm vào sao cho có độ dài là bội của 1024. Cách thêm vào như sau: đầu tiên là 1, sau đó là một số số 0 sao cho độ dài văn bản là 128 nhỏ hơn bội của 1024, sau đó là 128 bit chỉ độ dài của văn bản.

Vecto khởi điểm của hàm băm cũng tương tự: lấy 64 bit đầu tiên là phần phân số của căn bậc 2 từ 8 số nguyên tố đầu tiên:

Sau đó văn bản đem băm được chia ra làm các khối 1024 bit là $M^{(1)}$,..., $M^{(N)}$. Sau đó xử lý từng khối một. Vòng lặp chính của SHA-512 cũng giống như của SHA-256, nhưng mọi cái được tiến hành với từ 64 bit thay cho từ 32 bit (nói riêng, phép cộng theo modulo 2^{64}). Hàm nén chỉ khác số vòng lặp:

for j=0 to 79 {
Tính Ch(e,f,g), Maj(a,b,c), $\Sigma_0(a)$, $\Sigma_1(e)$, và W_j // công thức ở sau
T_1=h+ $\Sigma_1(e)$ +Ch(e,f,g) +K_j +W_j;
T_2=$\Sigma_0(a)$ +Maj(a,b,c) ;
h=g; g=f; f=e; e=d+T_1;
d=c; c=b; b=a; a=T_1+T_2;
}

Giá trị của hàm băm cũng như vậy: $H^{(N)} = (H_1^{(N)}, H_2^{(N)}, ..., H_8^{(N)})$. Cái khác ở đây là định nghĩa của các hàm logic:

$\Sigma_0(x)=S^{28}(x)\oplus S^{34}(x)\oplus S^{39}(x)$
$\Sigma_1(x)=S^{14}(x)\oplus S^{18}(x)\oplus S^{41}(x)$
$\sigma_0(x)=S^{1}(x)\oplus S^{8}(x)\oplus R^{7}(x)$
$\sigma_1(x)=S^{19}(x)\oplus S^{61}(x)\oplus R^{6}(x)$

Các khối của thông báo mở rộng W_0,..., W_{79} cũng được tính tương tự như của SHA-256:

$W_j = M_j^{(i)}, j = 0,...,15;$
for $j = 16$ *to* 79 {
$W_j = \sigma_1(W_{j-2}) + W_{j-7} + \sigma_0(W_{j-15}) + W_{j-16}$
}

Các hằng số từ K_0,..., K_{79} được lấy là 64 bit đầu của căn bậc 3 từ 80 số nguyên tố đầu, ở dạng hex là

Hàm SHA-384 được định nghĩa giống như SHA-512 nhưng vecto khởi điểm lấy từ 64 bit thập phân đầu tiên của căn bậc 2 của các số nguyên tố từ thứ 9 đến thứ 16.

Sau đó lấy 384 bit bên trái ở đầu ra, đó là giá trị của hàm băm SHA-384.

Phụ lục 3. Chuẩn AES của Mỹ[10]

Vào cuối năm 1996, Viện chuẩn quốc gia Mỹ (NIST) đã công bố cuộc thi để tạo ra chuẩn mã dữ liệu mới của nước Mỹ để thay cho DES. Chuẩn này được gán một cái tên là AES - Advanced Encryption Standard. Quá trình lựa chọn trải qua 2 giai đoạn, sau giai đoạn 1 còn lại 15 ứng cử viên, sau giai đoạn 2 còn lại 5 ứng cử viên.

Ngày 2 tháng 10 năm 2000, quyết định cuối cùng đã được đưa ra. Thuật toán Rijndael được chọn làm chuẩn mới. Thuật toán này được nghiên cứu bởi Vincent Rijmen và Joan Daemen (hai người Bỉ), đó là một thuật toán không dùng cấu trúc Feistel.

Để mô tả thuật toán cần sử dụng trường Galois GF(2^8) được xây dựng bằng cách mở rộng trường GF(2) theo các nghiệm của đã thức bất khả qui $m(x)=x^8+x^4+x^3+x+1$. Đa thức này được chọn với mục đích vì tính hiệu quả của việc biểu diễn các phần tử trong trường. Các phép toán cơ bản dùng trong thuật toán được thực hiện trong trường này.

Thuật toán Rijndael là một mã khối có độ dài khối thay đổi và độ dài khoá thay đổi. Độ dài khối và độ dài khoá có thể chọn không phụ thuộc vào nhau và bằng 128, 192 hay 256 bit. Mã pháp là một dãy các phép lặp được thực hiện trên một cấu trúc trung gian được gọi là trạng thái (thuật ngữ này được lấy từ lý thuyết otomat hữu hạn). Trạng thái có thể biểu diễn ở dạng một hình chữ nhật các byte. Hình chữ nhật có 4 dòng, còn số cột (được ký hiệu là Nb) bằng độ dài khối chia cho 32. Khoá của phép mã cũng được biểu diễn bằng một khối hình chữ nhật tương tự gồm các byte (có 4 dòng) còn số các cột (được ký hiệu là Nk) bằng độ dài khoá chia cho 32. Đầu vào và đầu ra của thuật toán được biểu diễn ở dạng dãy các byte có độ dài tương ứng. Bảng trạng thái và bảng khoá được điền đầu tiên theo cột, sau đó theo dòng. Số các vòng lặp được ký hiệu qua Nr và phụ thuộc vào Nb và Nk theo như bảng sau:

	Nb=4	Nb=6	Nb=8

[10] C.U.Mfhbxtd, D.D. Ujyxfhjd, H.T.Cthjd, Jcyjds cjdhtvtyyjq rhbgnjuhfabb, Vjcrdf, Ujhzxfz kbybz-Ntktrjv, 2002, cnh. 28-34.

Nk=4	10	12	14
Nk=6	12	12	14
Nk=8	14	14	14

Biên đổi của vòng lặp được cấu thành từ 4 biến đổi khác nhau. Viết theo kiểu ngôn ngữ C như sau:

```
Round (State, RoundKey) {
        ByteSub(State);
        ShiftRow(State);
        MixColumn(State);
        AddRoundKey(State,RoundKey);
}
```

Vòng lặp cuối cùng có khác chút ít so với các vòng khác:

```
FinalRound(State, RounKey) {
        ByteSub(State);
        ShiftRow(State);
        AddRoundKey(State,RoundKey);
}
```

Sau đây chúng tôi sẽ mô tả từng biến đổi một.

ByteSub

Đây là khối thay thế byte phi tuyến có ngược (S-box), gồm 2 phép toán:

1. Mỗi byte được thay bằng một phần tử ngược của nó trong trường GF(2^8). Byte 0 được thay bằng chính nó.
2. Với mỗi bayte thực hiện ánh xạ affine trong trường GF(2) như sau:

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Biến đổi affine này có thể viết ở dạng đa thức như sau:

$$b(x)=(x^7+x^6+x^2+x)+ a(x)(x^7+x^6+x^5+x^4+1) \bmod(x^8+1).$$

Đa thức được nhân với được chọn nguyên tố cùng nhau với đa thức lấy modulo, cho nên phép nhân có ngược.

Phép biến đổi ngược với ByteSub sẽ gồm: biến đổi affine ngược với ánh xạ trên và phép lấy phần tử ngược trong GF(2^8).

ShiftRow

Đây là phép dịch vòng về phía trái các dòng của ma trận trạng thái với các bước khác nhau. Dòng thứ nhất không dịch, dòng 1 dịch đi C1 vị trí, dòng 2- C2 vị trí và dòng 3 - C3 vị trí. Số vị trí dịch đi được cho trong bảng sau:

Nb	C1	C2	C3
4	1	2	3
6	1	2	3
8	1	3	4

Phép biến đổi ngược là phép dịch vòng các dòng về phía phải cũng với số vị trí như trên.

MixColumn

Trong biến đổi này, các cột của ma trận trạng thái được xem như các đa thức trên trường GF(2^8). Phép biến đổi bao gồm việc nhân từng cột theo modulo(x^4+1) với một đa thức cố định là

c(x) = ‘03h’x^3 + ‘01h’x^2 + ‘01h’x +’02h’.

Đây là đa thức nguyên tố cùng nhau với x^4+1 vì vậy phép nhân có ngược. ở dạng ma trận biến đổi này có dạng

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

Biến đổi ngược là phép nhân với đa thức là phần tử ngược của c(x) theo modulo(x^4+1). Đó là

d(x)= ‘0Bh’x^3 + ‘0Dh’x^2 + ‘09h’x +’0Eh’.

AddRoundKey

Đây chính là phép cộng từng bit theo modulo 2 từng byte của ma trận trạng thái với từng byte tương ứng của ma trận khoá. Phép biến đổi này trùng với biến đổi ngược.

Thuật toán tạo khoá

Các khoá cho các vòng lặp được tạo từ khoá mã bằng thuật toán tạo khoá gồm có hai thành phần: mở rộng khoá và chọn khóa cho vòng lặp. Các nguyên tắc chính để xây dựng nó là:

- Tổng số các bit khoá cho các vòng lặp bằng độ dài khối mã nhân với số vòng lặp cộng 1 (ví dụ, cho khối 128 bit và 10 vòng lặp cần 1408 bit khoá)
- Khoá mã được mở rộng đến khoá mở rộng.
- Khoá của vòng lặp được lấy từ khoá mở rộng bằng cách sau: khoá đầu tiên gồm Nb từ thứ hai, khoá thứ hai- Nb từ tiếp theo,...

Thuật toán mở rộng khoá

Khoá mở rộng là một mảng tuyến tính các từ gồm 4-byte và được ký hiệu là W[Nb*(Nr+1)]. Hàm mở rộng khoá phụ thuộc vào Nk. Tồn tại hai phương án: một cho Nk ≤ 6 và một cho Nk > 6.

Với Nk ≤ 6 thuật toán như sau:

```
KeyExpansion(byte Key[4*Nk], word W[Nb*(Nr+1)] {
        for (i=0; i<Nk;i++)
                W[i]=(Key[4*i], Key[4*i+1], Key[4*i+2], Key[4*i+3]);
        for (i=Nk; i<Nb*(Nr+1); i++) {
                tem=W[i-1];
                if (i% Nk==0)
                        temp=SubByte(RotByte(temp)) ^ Rcon[i/Nk];
                W[i]=W[i-Nk] ^ temp;
        }
}
```

Hàm SubByte(W) ở đây trả về từ mà mỗi byte của nó là kết quả của việc áp dụng mã thay khối đến byte nằm ở vị trí tương ứng của từ đầu vào. Hàm RotBye(W)- phép dịch vòng các byte trong từ, tức là từ (a,b,c,d) được biến thành (b,c,d,a).

Với Nk>6 thuật toán như sau:

```
KeyExpansion(byte Key[4*Nk], word W[Nb*(Nr+1)] {
        for (i=0; i<Nk;i++)
                W[i]=(Key[4*i], Key[4*i+1], Key[4*i+2], Key[4*i+3]);
        for (i=Nk; i<Nb*(Nr+1); i++) {
                tem=W[i-1];
                if (i% Nk==0)
                        temp=SubByte(RotByte(temp)) ^ Rcon[i/Nk];
                else if (i%Nk==4)
                        temp=SubByte(temp);
                W[i]=W[i-Nk] ^ temp;
        }
}
```

Các hằng số Rcon không phụ thuộc vào Nk và bằng

Rcon[i]=(RC[i], ‘00’, ‘00’, ‘00’),

với RC[i] là biểu diễn của các phần tử x^{i-1} của trường GF(2^8), tức là RC[1]=1 (tức ‘01’) và RC[i]=x.Rc[i-1], còn x =’02’).

Chọn khoá cho vòng lặp
Khoá cho vòng lặp thứ i được lấy từ bộ đêm của khoá mở rộng, bắt đầu từ W[Nb*i] đến W[Nb*(i+1)].

Như vậy, quá trình mã gồm 3 giai đoạn:

1. Cộng khoá lần đầu
2. Nr-1 vòng lặp
3. vòng lặp cuối

Về mặt lập trình có thể hình dung như sau:

```
Rijndael (State, CipherKey) {
        KeyExpansion(CipherKey, ExpandedKey);
        AddRoundKey(State, ExpandedKey);
        For (i=1; i <Nr;i++) Round(State, ExpandedKey+NB*i);
        FinalRound(State, ExpandedKey+Nb*Nr);
}
```

Chú ý: Giá trị ban đầu cho State?

Phụ lục 4
So sánh thuật toán chuẩn mã GOST 28147-89 của Nga và thuật toán Rijndael được chọn là chuẩn mã hoá mới của Mỹ

E. A. Primenko và A. A. Vinokurov

Ngày 2 tháng 10 năm 2000, bộ thương mại Mỹ đã tổng kết cuộc thi tuyển chọn thuật toán mã hoá mới của nước Mỹ. Người chiến thắng là thuật toán Rijndael (thuật toán này do 2 người Bỉ là Joan Daemen và Vincent Rijmen đệ trình lên NIST, mô tả nó có thể xem ở địa chỉ http://csrc.nist.gov/encryption/aes/round1/docs.htm). Cuộc thi đã được công bố 2 năm trước đó bởi Viện tiêu chuẩn và công nghệ quốc gia Mỹ với sự uỷ quyền của bộ thương mại. Đầu tiên, 15 thuật toán đã được đệ trình lên, sau một cuộc đánh giá trung gian còn lại 5 thuật toán, từ đó người ta chọn ra ứng cử viên để phê duyệt làm chuẩn. Thuật toán mã hoá mới được thay cho DES, đó là chuẩn mã hoá của Mỹ từ năm 1977.

Đây là bài báo của hai tác giả người Nga, E. A. Primenko (Bộ môn an toàn thông tin, trường Đại học Tổng hợp Matxcơva) và A. A. Vinokurov (Bộ môn toán và lập trình bảo vệ thông tin, trường HUUE), được đăng trong tạp chí "CBCNTVS MTPJGFCYJCNB", số 37 (tháng 2-3, 2001) và số 39 (tháng 6-7, 2001).

DES được thiết kế tại phòng nghiên cứu của hãng IBM vào nửa đầu những năm 70 của thế kỷ 20 và thuộc về họ các mã pháp khởi nguồn từ thuật toán Lucifer cũng được nghiên cứu tại nơi đó một số năm trước. Kiến trúc này, có tên gọi là mạng Feistel có vị trí quan trọng trong mật mã học cho đến ngày hôm nay: phần lớn các mã pháp hiện đại đều có dạng này, trong đó có cả chuẩn mã của Nga GOST 28147-89. Trong bài viết này sẽ phân tích so sánh GOST 28147-89 và Rijndael, trên cơ sở đó tiến hành việc so sánh phương pháp cổ điển và hiện đại trong việc xây dựng mã khối [1].

Trước khi so sánh trực tiếp các mã pháp, cần phải chú ý đến giải pháp kiến trúc đặc trưng cho các mã pháp của những năm 70 đã nhắc tới ở trên. Vào thời gian đó, công nghệ vi xử lý mới có những bước đi ban đầu trên con đường phát triển như hiện nay, khi đó mới có sản xuất công nghiệp những vi mạch có độ tích hợp nhỏ và vừa, vì vậy tiêu chuẩn chủ yếu trong thiết kế là đảm bảo độ bền vững chấp nhận được với các yêu cầu khắc nghiệt về độ phức tạp khi thể hiện. Vào thời điểm này, khả năng kỹ thuật cơ sở để thể hiện thuật toán đã được tăng lên nhiều lần, và khả năng mã thám chúng cũng tăng lên với tỷ lệ đó. Cho nên, các yêu cầu ban đầu về tính tiết kiệm khi thể hiện thuật toán không còn quan trọng nữa, trong khi đó yêu cầu về tính bền vững lại tăng lên. Chính trong ngữ cảnh đó xảy ra các thay đổi hiện đại trong các phương cách xây dựng mã khối có chìa khoá mật.

Bảng 1. Các đặc tính so sánh được của các thuật toán GOST 28147-89 và Rijndael

Chỉ tiêu	GOST 28147-89	Rijndael
Kích thước khối, bit	64	128, 192, 256
Kích thước khoá, bit	256	128, 192, 256
Kiến trúc	Mạng cân bằng Feistel	Hình vuông
Số vòng	32	10, 12, 14 (số vòng phụ thuộc vào kích thước của khoá và khối mã, lấy số lớn nhất trong chúng, nếu bằng 128 thì có 10 vòng, nếu bằng 192 thì có 12 vòng, nếu bằng 256 thì có 14 vòng)
Phần của khối rõ được mã sau mỗi vòng	nửa khối (32 bit)	cả khối (128, 192 hay 256)
Kích thước của khoá vòng, bit	nửa độ dài khối (32 bit)	bằng độ dài khối (128, 192 hay 256)
Cấu trúc vòng	Đơn giản	Tương đối phức tạp
Các phép toán được sử dụng	Chỉ có phép cộng, thay thế và phép dịch	Sử dụng rộng rãi các phép toán trên trường hữu hạn
Tính tương đương của biến đổi thuận và nghịch	Chính xác đến thứ tự của các khoá vòng	Chính xác đến vecto của các phần tử khoá, bảng các thay thế và hằng số của thuật toán

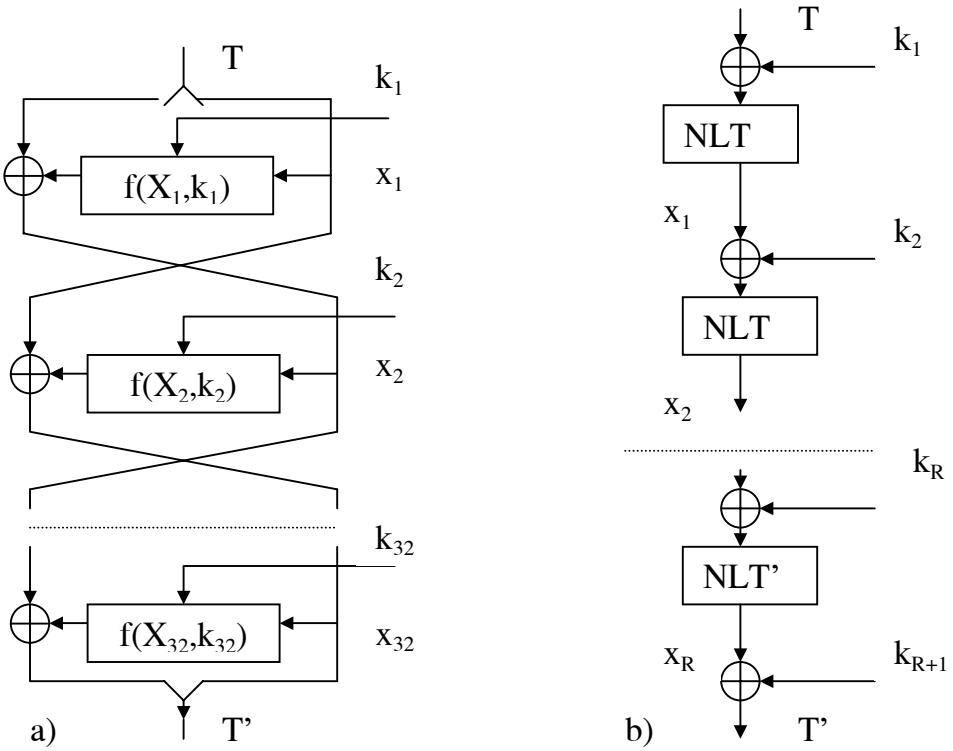
So sánh các đặc tính chung của 2 thuật toán

Các đặc tính so sánh được của thuật toán GOST 28147-89 và Rijndael đã được dẫn ra ở bảng 1. Khác với thuật toán của Nga, kích thước khối mã và kích thước khoá trong thuật toán Rijndael có thể thay đổi, điều này là do trong nó có sử dụng cấu trúc "hình vuông". Tính chất này cho phép thay đổi độ bền vững cũng như tốc độ thực hiện thuật toán theo sự phụ thuộc vào các yếu tố bên ngoài khi cài đặt trong một giới hạn nhất định, tuy nhiên không rộng lắm, đó là số các vòng, và cùng với nó là tốc độ trong các trường hợp khác biệt nhau nhất vào khoảng 1,4 lần.

So sánh các nguyên tắc kiến trúc chung

Việc phân tích thuật toán GOST 28147-89 cũng như phần lớn các mã pháp thuộc thế hệ đầu tiên được thiết kế vào những năm 70 và nửa đầu những năm 80, được dựa trên kiến trúc mạng Feistel cân bằng [2]. Nguyên tắc chính của kiến trúc này là cả quá trình mã gồm một loạt các vòng có kiểu giống nhau. Tại mỗi vòng, khối được mã T được chia thành hai nửa (T_0, T_1), một trong chúng được thay đổi bằng phép cộng modulo 2 theo từng bit với giá trị được làm ra từ phần còn lại và bộ phận khoá vòng với sự giúp đỡ của hàm mã. Giữa các vòng, hai phần của khối đổi

chỗ cho nhau, như vậy, tại vòng sau, phần của khối thay đổi ở vòng trước sẽ không thay đổi và ngược lại. Lược đồ thuật toán GOST 28147-89 ở hình 1(a). Kiến trúc như vậy cho phép dễ dàng nhận được phép giải mã từ một hàm mã phức tạp, và có thể là không có ngược. Đặc tính quan trọng của phương pháp này là tại mỗi vòng chỉ mã đúng một nửa khối.



T, T' - khối rõ và khối mã
k_i - thành phần khoá vòng
Xi - trạng thái của quá trình mã sau vòng thứ i
f(X,k)- hàm mã của thuật toán GOST
NLT, NLT'- biến đổi phi tuyến thông thường và biến đổi phi tuyến cho vòng cuối cùng của thuật toán Rijndael
R - số vòng của Rijndael (10, 12 hay 14)

Hình 1. Lược đồ biến đổi dữ liệu khi mã
theo thuật toán GOST (a) và Rijndael (b)

Mã Rijndael có một kiến trúc khác về mặt nguyên tắc, nó được gọi là "hình vuông" theo tên của mã pháp đầu tiên có cấu trúc kiểu này cũng do chính các tác giả của Rijndael thiết kế ra một số năm trước đây. Kiến trúc này dựa trên các biến đổi trực tiếp khối được mã khi được biểu diễn ở dạng ma trận của các byte. Việc mã cũng gồm một loạt các bước có kiểu giống nhau, đó là các vòng, nhưng tại mỗi vòng cả khối đều được biến đổi chứ không có phần nào của khối được giữ nguyên. Như vậy, sau một vòng cả khối được mã, cho nên, để đảm bảo độ phức tạp tương ứng và tính phi tuyến của biến đổi, số các bước yêu cầu như vậy sẽ ít hơn 2 lần so với cấu

trúc Feistel. Mỗi vòng bao gồm từng bit theo modulo 2 giữa trạng thái hiện tại của khối được mã và thành phần khoá vòng, sau đó là một phép biến đổi phi tuyến phức tạp của cả khối, biến đổi phi tuyến này được kiến thiết từ 3 biến đổi đơn giản hơn sẽ được xem xét chi tiết ở phần sau. Lược đồ của mã Rijndael được đưa ra ở hình 1(b).

So sánh các vòng mã

Trong thuật toán GOST sử dụng hàm mã tương đối không phức tạp, gồm có phép cộng nửa khối vào với thành phần khoá vòng tương ứng theo modulo 2^{32}, tám phép thay thế thực hiện một cách độc lập trong các nhóm 4-bit và phép hoán vị bit (quay 11 bit về phía hàng cao). Lược đồ phép mã này ở hình 2(a).

Trong thuật toán Rijndael, khối được mã và các trạng thái trung gian của nó trong quá trình biến đổi được biểu diễn ở dạng ma trận kích thước 4 x n byte, với n = 4, 6, 8 tuỳ thuộc vào kích thước khối. Hàm biến đổi phi tuyến trong thuật toán Rijndael bao gồm 3 phép biến đổi đơn giản sau thực hiện lần lượt:

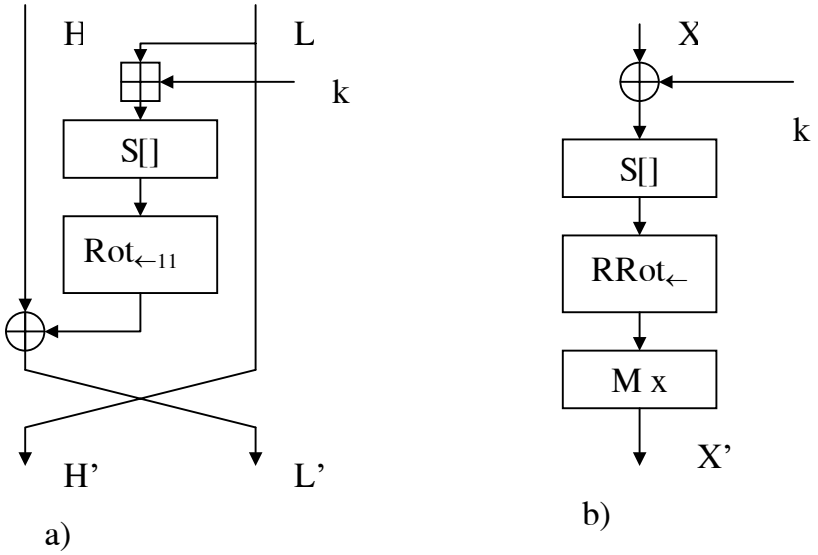
- thay thế byte- mỗi byte của khối được biến đổi được thay bằng giá trị mới, lấy từ một vecto thay thế chung cho tất cả các byte của ma trận
- phép dịch vòng theo byte trong các dòng của ma trận: dòng đầu tiên không đổi, dòng thứ hai dịch vòng về phía trái một byte, dòng thứ ba và thứ tư dịch vòng về bên trái tương ứng 2 hay 3 byte ứng với n=4 hay 6; còn với n=8 là 3 hay 4 byte.
- nhân ma trận- ma trận nhận được ở bước trên nhân trái với ma trận hồi chuyển kích cỡ 4 x 4 :

$$M = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

Các phần tử của cả hai ma trận được xem như là các phần tử của trường hữu hạn $GF(2^8)$, tức là các đa thức có bậc không quá 7, hệ số của chúng là các bit, các phép cộng và nhân theo modulo 2. Trong trường hữu hạn này, phép cộng các byte như là phép cộng từng bit theo modulo 2, còn phép nhân được lấy theo modulo của đa thức bất khả qui $x^8+x^4+x^3+x+1$ với các hệ số từ GF(2). Lược đồ vòng của thuật toán Rijndael biểu diễn ở hình 2(b).

Nếu trong thuật toán GOST phép hoán vị 2 nửa khối được đưa vào các vòng mã như chỉ ra ở hình 2(a), thì có thể nhận thấy rằng trong cả hai thuật toán, các vòng mã luôn giống nhau, trừ vòng cuối cùng, tại đó thiếu một phần phép toán. Cách như vậy cho phép nhận được một cách thể hiện gọn ghẽ hơn, cả trong thiết bị lẫn lập trình. Tại vòng cuối cùng của GOST không có phép hoán vị 2 nửa khối đã được mã, còn đối với Rijndael là phép nhân bên trái với ma trận M. Trong cả hai thuật

toán được bàn đến, điều này đảm bảo tính tương đương cấu trúc của biến đổi mã và giải mã.



X, X' - khối được biến đổi ở đầu vào và ra của vòng
(H, L), (H', L') - phần cao và phần thấp ở đầu vào và ra của vòng
k - khoá vòng
S[], hàm thay thế, được nhóm theo 4 bit cho GOST và của các byte cho Rijdael
Rot←11 – phép quay từ 32-bit về phía bit cao 11 lần
RRot- phép toán quay ma trận theo dòng của thuật toán Rijndael
Mx - nhân ma trận M trong thuật toán Rijndael với ma trận dữ liệu ở bên trái

Tính tương đương của biến đổi xuôi và ngược.

Trong thuật toán GOST tính tương đương cấu trúc của biến đổi xuôi và ngược không được đảm bảo một cách đặc biệt mà là hệ quả đơn giản của việc áp dụng giải pháp kiến trúc. Trong một mạng cân bằng Feistel bất kỳ, hai biến đổi này tương đương và chỉ khác nhau thứ tự sử dụng của các khoá thành phần: khi giải mã, các khoá thành phần được sử dụng với thứ tự ngược lại so với thứ tự được sử dụng lúc mã.

Thuật toán Rijndael được xây dựng trên cơ sở các biến đổi trực tiếp. Cũng như đối với tất cả các thuật toán tương tự, biến đổi ngược được xây dựng từ việc đảo ngược các bước của biến đổi xuôi theo thứ tự ngược lại. Do đó, việc đảm bảo tính đồng nhất của biến đổi xuôi và ngược như trong cấu trúc Feistel là không thể đạt được. Tuy vậy, bằng một giải pháp kiến trúc đặc biệt cũng đạt được một mức độ gần tương ứng: biến đổi xuôi và ngược là đồng nhất với sự chính xác đến các hằng số được sử dụng trong chúng. Trong bảng 2 dẫn ra 2 vòng cuối của thuật toán Rijndael và phép đảo ngược hình thức của nó.

Bảng 2. Hai vòng của thuật toán Rijndael
và phép đảo ngược hình thức của nó

Biến đổi xuôi	Biến đổi ngược
$X = X \oplus k_{R-1}$	$X = X \oplus k_{R+1}$
$X = S(X)$	$X = RRot_{\rightarrow}(X)$
$X = RRot_{\leftarrow}(X)$	$X = S^{-1}(X)$
$X = M \times X$	$X = X \oplus k_R$
$X = X \oplus k_R$	$X = M^{-1} \times X$
$X = S(X)$	$X = RRot_{\rightarrow}(X)$
$X = RRot_{\leftarrow}(X)$	$X = S^{-1}(X)$
$X = X \oplus k_{R+1}$	$X = X \oplus k_{R-1}$

Trước hết cần chú ý rằng phép toán thay thế theo từng byte (S) có tính giao hoán với phép dịch theo byte các dòng của ma trận:

$$S^{-1}(RRot_{\rightarrow}(X)) = RRot_{\rightarrow}(S^{-1}(X)).$$

Ngoài ra, theo các quy tắc của đại số ma trận theo định luật kết hợp cũng có thể thay đổi thứ tự cộng từng bit theo moduko 2 của khoá và phép nhân ma trận:

$$M^{-1} \times (X \oplus k_R) = (M^{-1} \times X) \oplus (M^{-1} \times k_R)$$

Áp dụng các thay đổi đã chỉ ra vào cột hai của bảng 2, chúng ta nhận được dãy phép toán sau trong hai vòng của biến đổi ngược (bảng 3).

Bảng 3. Hai vòng của thuật toán
Rijndael và ngược của nó

Biến đổi xuôi	Biến đổi ngược
$X = X \oplus k_{R-1}$	$X = X \oplus k_{R+1}$
$X = S(X)$	$X = S^{-1}(X)$
$X = RRot_{\leftarrow}(X)$	$X = RRot_{\rightarrow}(X)$
$X = M \times X$	$X = M^{-1} \times X$
$X = X \oplus k_R$	$X = X(M^{-1} \oplus k_R)$
$X = S(X)$	$X = S^{-1}(X)$
$X = RRot_{\leftarrow}(X)$	$X = RRot_{\rightarrow}(X)$
$X = X \oplus k_{R+1}$	$X = X \oplus k_{R-1}$

Từ việc so sánh các cột của bảng 3 ta dễ thấy rằng, cấu trúc hoạt động của biến đổi xuôi và ngược giống nhau. Kết quả dễ dàng tổng quát hoá cho một số vòng bất kỳ. Như vậy, trong thuật toán Rijndael thủ tục mã và giải mã hoạt động như nhau và chỉ khác nhau ở các chi tiết sau:

- Trong biến đổi ngược sử dụng phép thế vecto, ngược về hoạt động với vecto thay thế ở biến đổi xuôi.
- Trong biến đổi ngược, số byte mà theo nó mỗi dòng của ma trận dữ liệu dịch đi trong phép toán dịch từng dòng theo byte khác đi so với biến đổi xuôi.
- Trong biến đổi ngược, tại bước nhân ma trận khối dữ liệu được nhân về bên trái với ma trận là ngược với cái được sử dụng trong biến đổi xuôi, đó là

$$M = \begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix}$$

- Trong biến đổi ngược, các phần tử khoá được sử dụng theo thứ tự ngược lại, ngoài ra, tất cả các phần tử trừ phần từ đầu tiên và cuối cùng cần phải nhân phía bên trái với ma trận M^{-1}.

Như vậy, tương tự như GOST, trong thuật toán Rijndael có thể trùng hợp việc thực hiện thủ tục mã và giải mã khi thể hiện bằng chương trình cũng như bằng thiết bị.

Chuẩn bị khoá

Trong chuẩn mã của nước Nga, để tạo ra các phần tử khoá 32-bit từ khoá 256-bit một phương pháp đơn giản được áp dụng. Khoá được hiểu như một mảng gồm 8 phần tử khoá :

$$K = (k_1, k_2, k_3, k_4, k_5, k_6, k_7, k_8) .$$

Các phần tử này được sử dụng trong các vòng mã - mỗi khoá được "xem đến" 3 lần theo thứ tự xuôi và một lần theo chiều ngược lại, cuối cùng là mỗi phần tử khoá được sử dụng đúng 4 lần. Trong bảng 4 chỉ ra thứ tự của vòng và phần tử khoá được sử dụng cho vòng đó.

Vòng	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Phần tử khoá	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8
Vòng	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Phần tử khoá	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_8	k_7	k_6	k_5	k_4	k_3	k_2	k_1

Bảng 4- Thứ tự sử dụng các phần tử khoá trong vòng mã của GOST 28147-89

Trong thuật toán Rijndael sử dụng lược đồ phức tạp hơn một chút, có tính đến khả năng khác nhau về kích thước của khối mã và khoá. Tồn tại hai thuật toán để sinh ra dãy các phần tử khoá - cho khoá kích thước 128/192 bit và cho khoá kích thước 256 bit, chúng tương đối giống nhau và chỉ khác nhau chút ít. Khoá và dãy khóa được biểu diễn ở dạng vecto của các từ gồm 4 byte, đoạn đầu tiên của dãy được điền chính bằng các từ của khoá- cũng như trong GOST. Các từ sau của dãy khoá được chọn theo quan hệ đồng dư từng nhóm một, là bội của kích cỡ khoá. Từ 4-

byte đầu tiên của nhóm như vậy được tạo bởi việc sử dụng một biến đổi phi tuyến đủ phức tạp, những từ còn lại theo một quan hệ tuyến tính đơn giản:

$$\omega_i = \begin{cases} \omega_{i-Nk} \oplus G(\omega_{i-1}), i \bmod N_k = 0 \\ \omega_{i-Nk} \oplus \omega_{i-1}, i \bmod N_k \neq 0 \end{cases}$$

Ở đây N_k là số các từ có 4 byte trong khoá (bằng 4 hay 6, nếu bằng 8 thì dùng công thức khác); G() – là biến đổi phi tuyến phức tạp các phần tử có 4 byte, gồm có phép dịch theo byte, thay thế byte theo vecto thay thế và cộng theo bit modulo 2 với vecto phụ thuộc vào thứ tự của nhóm các phần tử đang được tạo ra:

$$G(x0 = S(R_{\leftarrow}8(x)) \oplus P(i/N_k),$$

trong đó S -là hàm thay thế byte được mô tả trên đây; $R_{\leftarrow}8$ - phép toán dịch vòng từ 4 byte đi 8 bit về bên trái; còn $P(i/N_k)$- là từ 4 byte, được kiến thiết một cách đặc biệt và không phụ thuộc vào khoá. Việc sử dụng biến đổi G(c) mang tính phức tạp và tính phi tuyến vào lược đồ chuẩn bị các phần tử khoá, làm cho việc phân tích mã trở nên khó khăn.

Các từ 4 byte nhận được trong dòng được mô tả trên đây được nhóm thành các phần tử khoá có kích thước cần thiết, bằng với kích thước của khối mã và được sử dụng trong các vòng mã như đã mô tả ở các mục trước.

Như đã trình bày trên đây, thuật toán tạo dãy khoá trong mã Rijndael là phức tạp hơn so với trong GOST. Tuy vậy, nó cũng đủ đơn giản và hiệu quả và không đóng góp đáng kể vào khối lượng tiêu tốn tính toán chung khi mã- tốc độ mã cùng với việc tính các phần tử khoá chút ít nhỏ hơn tốc độ mã cùng với khoá đã được chuẩn bị từ trước.

Chọn các nút thay thế và các hằng số khác

Các phần tử khoá dùng trong thời gian dài (các nút thay thế) là những hằng số quan trọng nhất của GOST 28147-89. Chúng không được chỉ ra trong chuẩn mà được cung cấp bởi các tổ chức chuyên ngành đặc biệt, chuyển cho người sử dụng khoá mã này. Vì thế không đưa ra một tiêu chuẩn thiết kế nào cho các nút thay thế này. Từ những suy luận chung có thể nhận thấy rằng, trước hết, các nút thay thế được chọn bằng việc sử dụng một trong những phương pháp thiết kế các nút, ví dụ như việc sử dụng hàm bent (A. Varpholomev và các tác giả khác "Các hệ mã khối. Các tính chất cơ bản và phương pháp phân tích độ bền vững" , Matxcơva, MIFI, 1998), sau đó được đánh giá theo một số tiêu chuẩn, khi đó những nút không đủ tiêu chuẩn sẽ bị bỏ đi. Trong số các tiêu chuẩn đánh giá, có lẽ, có mặt các tiêu chuẩn sau:

- độ phức tạp và tính phi tuyến của hàm bool mô tả các nút
- đặc tính vi phân của các nút thay thế
- đặc tính tuyến tính của các nút thay thế

Khác với những người tạo ra GOST 28147-89, các tác giả của mã pháp Rijndael không giấu các tiêu chuẩn thiết kế vecto thay thế. Khi thiết kế chúng, bên cạnh các

yêu cầu đơn giản như tính có ngược và tính đơn giản khi mô tả còn có những suy tính sau được để ý đến:

- cực tiểu hoá đặc tính tương quan lớn nhất theo giá trị giữa các tổ hợp tuyến tính của các bit vào và các bit ra (xác định tính bền vững đối với tấn công tuyến tính)
- cực tiểu hoá giá trị không tầm thường lớn nhất trong bảng EXOR (xác định tính bền vững đối với tấn công vi sai)
- độ phức tạp của biểu thức đại số mô tả nút trong GF(2).

Phép toán thay thế byte trong thuật toán Rijndael được cho bởi phương trình:

$$S(X)= (x^7+ x^6 + x^2 +x) + X^{-1} (x^7 + x^6 +x^5 +x^4 +1) \bmod (x^8+1)$$

Biến đổi này bắt đầu bởi phép toán lấy ngược trong trường $GF(2^8)$ của byte được thay thế – giá trị 00 khi đó được giữa nguyên- sau đó giá trị thu được đi qua một phép biến đổi affine. Các đa thức của phép biến đổi được chọn sao cho trong ánh xạ kết quả không có điểm bất động ($S(X) = X$) hay những điểm phản bất động ($S(X) = \sim X$).

Các tác giả của thuật toán Rijndael đã nhận thấy rằng nếu nút thay thế nào đó gây ra mối nghi ngờ vào chất lượng của nó và vào việc không có "con đường ngầm" thì có thể thay nó bằng một nút khác. Hơn nữa, cấu trúc của mã pháp và số các vòng được chọn sao cho ngay cả trong trường hợp *vecto thay thế được chọn ngẫu nhiên* thì mã pháp vẫn bền vững đối với tấn công vi sai và tấn công tuyến tính.

Trong số các hằng số quan trọng khác của thuật toán Rijndael cần phải chú ý đến ma trận M, trong phép biến đổi của từng vòng ta có sử dụng nó vào phép nhân. Mục đích của bước này là để khuyếch tán những thay đổi trong một byte ra tất cả cột của ma trận. Khi chọn ma trận M cùng với các yêu cầu truyền thống như tính có ngược và tính đơn giản cần chú ý đến các đặc trưng mong muốn sau của biến đổi:

- tính tuyến tính trong trường GF(2);
- đủ độ khuyếch tán;
- tốc độ thực hiện trong các bộ vi xử lý 8 bit.

Bước này của phép biến đổi cũng có thể biểu diễn như là phép nhân các cột của ma trận cải biến dữ liệu (còn dữ liệu được xem như những đa thức bậc 3 có hệ số từ trường $GF(2^8)$) với một ma trận có các hệ số cũng từ trường $GF(2^8)$ theo modulo của đa thức (x^4+1). Theo như yêu cầu thứ 3 được liệt kê ở trên, các hệ số của thừa số nhân cần nhỏ nhất trong các khả năng có thể . Các tác giả của mã pháp đã chọn đa thức sau:

$$c(x) = 03.x^3 + 01.x^2 + 01.x + 02.$$

Cũng thủ tục này, khi viết ở dạng ma trận tương đương với phép nhân bên trái với ma trận luân lưu M.

Như vậy, khác với GOST 28147-89, các vòng của nó được lập từ các phép toán truyền thống cho các thuật toán mật mã thuộc "làn sóng thứ nhất" - phép hoán vị trong các nhóm bít nhot, phép hoán vị (dịch) theo bit và phép cộng tổ hợp các thành phần dữ liệu- khi kiến thiết mã pháp Rijndael đã sử dụng rộng rãi phương pháp đại số. Điều đó liên quan chủ yếu đến hai biến đổi chính của mã pháp- phép thay thế theo byte và phép toán xáo trộn các cột của ma trận dữ liệu bằng cách nhân bên trái với ma trận M.

Tính năng suất và độ tiện lợi khi thể hiện

Khi đánh giá tính hiệu quả đạt được đối với cài đặt thiết bị của mã pháp thì tiêu chuẩn chủ yếu là số lượng và độ phức tạp của các phép tính cơ sở cần phải thực hiện trong một vòng lặp, và cả khả năng song song hoá thuật toán. Khi đánh giá tính hiệu quả của các cài đặt chương trình có thể thì mối quan tâm chính là việc cài đặt trên những nền tảng 32 bit, vì các máy 32 bit tại thời điểm hiện tại chiếm chủ yếu cộng đồng máy tính của loài người. Việc cài đặt trên các bộ vi xử lý 8 bit cũng cần chú ý, vì đó là công nghệ chủ yếu của thẻ thông minh. Các thiết bị tương tự có thể được sử dụng trong các hệ thống khác nhau thanh toán chuyển khoản, chúng ngày một trở nên phổ biến trên thế giới – số người sử dụng các hệ thống như vậy trong thời gian cuối tăng với tốc độ lớn.

Chuẩn mã GOST 28147-89 của nước Nga thuận tiện thể hiện trong thiết bị cũng như phần mềm. Với kích thước khối dữ liệu bằng 64, công việc chủ yếu được tiến hành với một nửa của khối này, đó là các từ 32 bit, điều này cho phép thể hiện hiệu quả chuẩn mã của nước Nga trên phần lớn các máy tính hiện đại.

Khi thể hiện trên các máy 32 bit thì phép toán khó khăn nhất là phép thế. Các phép thay thế được xem xét bởi GOST trong các nhóm 4 bit khi lập trình cho khả năng hợp nhất từng cặp và thực hiện phép thế trong các nhóm 8 bit, như vậy hiệu quả hơn đáng kể. Tổ chức thích đáng của phép thế cũng cho phép tránh khỏi việc thực hiện phép quay từ ở đầu ra của hàm mã - nếu lưu giữ các nút thay thế như là các mảng từ có 4 byte mà phép quay đã được thực hiện trong nó. Bảng thay thế mở rộng như vậy yêu cầu cho việc lưu trữ nó 4 x 28 x 4 = 212 byte hay 4 K bộ nhớ vận hành. Các bước tối ưu hoá được chỉ ra thể hiện vòng mã theo GOST trong 10 lệnh máy, gồm cả việc phân chia và tải vào các thanh ghi từng byte riêng biệt của các từ có 4 byte. Tính đến khả năng song song hoá của bộ xử lý Pentium đối với việc thực hiện song song hoá các lệnh, một vòng của GOST có thể được thực hiện trong 6 nhịp hoạt động của bộ vi xử lý, toàn bộ quá trình mã trong 32 x 6 = 192 nhịp. Thêm vào đó là 8 nhịp cho các phép lưu giữ khác nhau trong bộ xử lý, chúng ta nhận được đánh giá chi phí thời gian của bộ xử lý là 200 nhịp khi thực hiện chu trình mã theo thuật toán GOST 28147-89. Trên bộ xử lý Pentium Pro-200 điều đó cho phép đạt được giới hạn tốc độ một triệu khối mã trong một giây hay 8 M Byte/s (trong thực tế đại lượng đó sẽ nhỏ hơn).

GOST cũng có khả năng thực hiện hiệu quả trên các bộ vi xử lý 8 bit, bởi vì các phép tính cơ sở tạo nên nó có trong bộ lệnh của phần lớn các bộ điều khiển phổ dụng nhất. Khi đó phép cộng theo modulo 232 buộc phải chia ra một phép cộng không nhớ và 3 phép cộng có nhớ, được thực hiện tuần tự. Tất cả các phép toán còn lại cũng có thể biểu diễn trong thuật ngữ của các toán hạng 8 bit.

Khi thực hiện GOST bằng thiết bị, một vòng chính là việc thực hiện liên tiếp 3 phép toán của các tham số 32 bit: cộng, phép thế (được thực hiện đồng thời tại cả 8 nhóm có 4 bit) và phép cộng từng bit theo modulo 2. Phép dịch vòng không là một phép toán riêng biệt, vì được đảm bảo bằng chuyển mạch đơn giản của các dây dẫn. Như vậy, với thực hiện bằng thiết bị, một vòng mã yêu cầu thực hiện 106 phép toán cơ sở, và công việc này không thể song song hoá được.

Đặc điểm thể hiện của thuật toán Rijndael

Bây giờ chúng ta xét đến các đặc điểm của việc thực hành thuật toán Rijndael. Đây là một thuật toán định hướng theo byte, có nghĩa là hoàn toàn có thể phát biểu theo thuật ngữ của các phép tính theo byte. Trong thuật toán sử dụng rộng rãi các phép toán đại số trên trường hữu hạn, trong đó phép nhân trong $GF(2^8)$ là khó thể hiện nhất. Việc thực hiện trực tiếp các phép tính đó dẫn đến một thể hiện rất không hiệu quả của thuật toán. Tuy vậy, cấu trúc byte của mã pháp mở ra các khả năng rộng lớn cho việc lập trình. Phép thế byte theo bảng cùng với phép nhân sau đó với hằng số trong trường $GF(2^8)$ có thể biểu diễn như là một phép thế theo bảng. Trong biến đổi xuôi có 3 hằng số được sử dụng (01, 02, 03) và vì thế cần có 3 bảng như vậy, còn trong biến đổi ngược –có 4 hằng số (0E, 0D, 0B, 09). Khi tổ chức khéo quá trình mã thì phép dịch byte theo từng dòng của ma trận dữ liệu có thể không cần thực hiện. Khi viết cho các máy 32 bit có thể cài đặt phép thế theo byte và phép nhân phần tử của ma trận dữ liệu với cột của ma trận M như là một phép thay 8 bit bằng 32 bit. Như vậy, tất cả chương trình cho một vòng mã của phương án khối dữ liệu 128 bit sẽ dẫn đến: 4 lệnh tải các phần tử khoá vào thanh ghi, 16 lệnh tải byte vào thanh ghi và lấy từ bộ nhớ ra giá trị đã được đánh chỉ số. Giá trị này được sử dụng trong phép tính theo byte. Đối với các bộ xử lý Intel Pentium không có đủ số thanh ghi còn cần thêm 4 lệnh tải nội dung các thanh ghi vào bộ nhớ, như vậy trên những bộ xử lý đã chỉ ra một vòng mã theo thuật toán Rijndael có thể thực hiện sau 40 lệnh hoặc sau 20 nhịp của bộ xử lý có tính đến khả năng thực hiện song song các lệnh bởi bộ xử lý. Cho 14 vòng mã của một chu trình mã sẽ cần 280 nhịp, cộng thêm một số nhịp thêm vào để cộng thêm khoá. Thêm vào một số nhịp cho phép giữ chậm bên trong bộ vi xử lý, chúng ta nhận được đánh giá 300 nhịp cho một chu trình mã. Trên bộ xử lý Pentium Pro-200 về mặt lý thuyết cho phép đạt đến tốc độ khoảng 0,67 triệu khối trong một giây hay khoảng 8,5 MByte/s (mỗi khối có 128 bit). Đối với các phương án có số vòng ít hơn thì tốc độ tăng lên theo tỷ lệ.

Phép tối ưu chỉ ra trên đây, tuy vậy, yêu cầu tiêu tốn một lượng xác định bộ nhớ. Cho mỗi cột của ma trận M xây dựng vecto thay thế của mình từ 1 byte sang từ có 4 byte. Hơn nữa, cho vòng cuối cùng, trong đó không có phép nhân với ma trận M, cần có một vecto thay thế riêng cùng kích cỡ. Điều này yêu cầu sử dụng $5 \times 2^8 \times 4$

= 5 KB bộ nhớ để lưu trữ các nút thay thế khi mã và cũng một lượng như thế cho khi dịch, tất cả là 10 KB. Đối với các máy tính hiện đại trên cơ sở Intel Pentium dưới sự điều khiển của hệ điều hành Windows 9x/NT/2000 thì không là một yêu cầu gì lớn cả.

Kiến trúc định hướng theo byte của thuật toán Rijndael hoàn toàn cho phép thể hiện hiệu quả nó trên các bộ vi xử lý 8 bit, chỉ sử dụng các phép tải vào/ra thanh ghi, lấy các byte đã được đánh chỉ số trong bộ nhớ và phép cộng bit theo modulo 2. Đặc điểm đã chỉ ra cũng cho phép thực hành lập trình hiệu quả thuật toán. Một vòng mã cần thực hiện 16 phép thê theo byte cộng thêm "loại trừ hoặc" theo bit trên các khối 128 bit, chúng có thể thực hiện trong 3 giai đoạn. Tổng lại là 4 thao tác cho một vòng hoặc 57 thao tác cho một chu trình mã 14 vòng có tính đến một số thao tác thêm cho phép cộng khoá theo modulo 2 – tức là vào khoảng 2 lần ít hơn GOST. Vì thuật toán Rijndael có độ dài khối 2 lần lớn hơn, nên điều này dẫn đến ưu thế gấp 4 lần về tốc độ với điều kiện thực hiện máy trên cơ sở cùng một công nghệ. Chú ý rằng đánh giá trên chỉ là thô.

Khi đánh giá các đặc trưng thực tế tốc độ bằng chương trình của hai thuật toán trên nền Intel Pentium, với thuật toán Rijndael chúng ta xem xét phương án có 14 vòng. Cho mỗi thuật toán, bằng ngôn ngữ C đã viết một hàm tương đương để mã một khối, trong đó dãy các vòng được trải ra ở dạng mã tuyến tính - điều này cho phép đạt được tốc độ tối đa. Trong những hàm tương đương đó sử dụng các thông tin khóa ngẫu nhiên và các nút thay thế ngẫu nhiên, nhưng điều đó không ảnh hưởng gì đến tốc độ thi hành bởi vì tốc độ thực hiện của các lệnh được sử dụng không phụ thuộc vào các toán hạng của nó. Các hàm thực hiện việc mã được gọi hàng chục triệu lần và đo thời gian nó chạy, con số này được dùng làm chỉ số về tốc độ. Để biên dịch và xây dựng module thực thi đã sử dụng trình dịch Intel C++ v 4.5, vì nó cho phép nhận được mã lệnh với tốc độ cao nhất. Cũng đã thử nghiệm với các trình biên dịch MS Visual C++ v 6.0, Borland C ++ v 5.5 và gnu C++ v 2.95.2, nhưng mã nhận được khi sử dụng chúng cho tốc độ kém hơn. Mã được tối ưu hoá đối với các bộ xử lý Intel Pentium và Intel Pentium Pro/II/III. Với sự trợ giúp của các bài toán thử nghiệm, tốc độ thực hiện các mã pháp đã được đo trên các bộ xử lý Intel Pentium 166 MHz và Intel Pentium III 433 MHz. Kết quả đo ở trong bảng sau:

Các chỉ số về tốc độ thực hiện của các thuật toán được so sánh

Bộ xử lý	GOST 28147-89	Rijndael 14 vòng
Pentium 166	2,04 Mbyte/s	2,46 MByte/s
Pentium III 433	8,30 MByte/s	9,36 MByte/s

Như vậy, các thuật toán được xem xét có tốc độ so sánh được với nhau khi thực hiện trên nền 32 bit. Trên các nền 8 bit, bức tranh có lẽ cũng như vậy. Còn đối với việc cài đặt trên phần cứng, khác với thuật toán mã GOST, Rijndael cho phép đạt được một mức độ song song hoá cao khi thực hiện thuật toánvì thao tác với các khối có kích thước nhỏ hơn và số vòng ít hơn, nên về mặt lý thuyết việc cài đặt

cứng nó sẽ đạt được tốc độ nhanh hơn so với GOST trên cùng một nền công nghệ- theo các đánh giá thô là vào 4 lần.

Việc so sánh được tiến hành trên đây cho các tham số của 2 thuật toán mã hoá GOST 28147-89 và Rijndael đã chỉ ra rằng, mặc dù có sự khác biệt đáng kể trong nguyên tắc kiến thiết mà các mã pháp dựa vào, các thông số làm việc chính là gần như nhau. Điểm ngoại trừ là, gần như chắc chắn, Rijndael có ưu thế hơn về tốc độ so với GOST khi cài đặt máy trên cùng một công nghệ. Theo các tham số quan trọng về độ bền vững cho những thuật toán dạng đó, không thuật toán nào có được ưu thế đáng kể, ngay cả tốc độ của một chương trình tối ưu trên bộ xử lý Intel Pentium là cũng như nhau, điều này có thể ngoại suy ra mọi bộ xử lý 32 bit hiện đại khác.

Như vậy, có thể rút ra kết luận là chuẩn mã dữ liệu của nước Nga đáp ứng được các yêu cầu của các mã pháp hiện đại và có thể là chuẩn trong một thời gian dài nữa. Bước dễ thấy tiếp theo trong việc tối ưu hoá nó có thể là việc chuyển phép thế trong các nhóm 4 bit sang phép thế theo byte, điều này sẽ làm tăng hơn nữa tính bền vững của thuật toán đối với các dạng phân tích mã đã biết.

Phụ lục 5

Nhận xét về thuật toán mã hóa GOST[11]

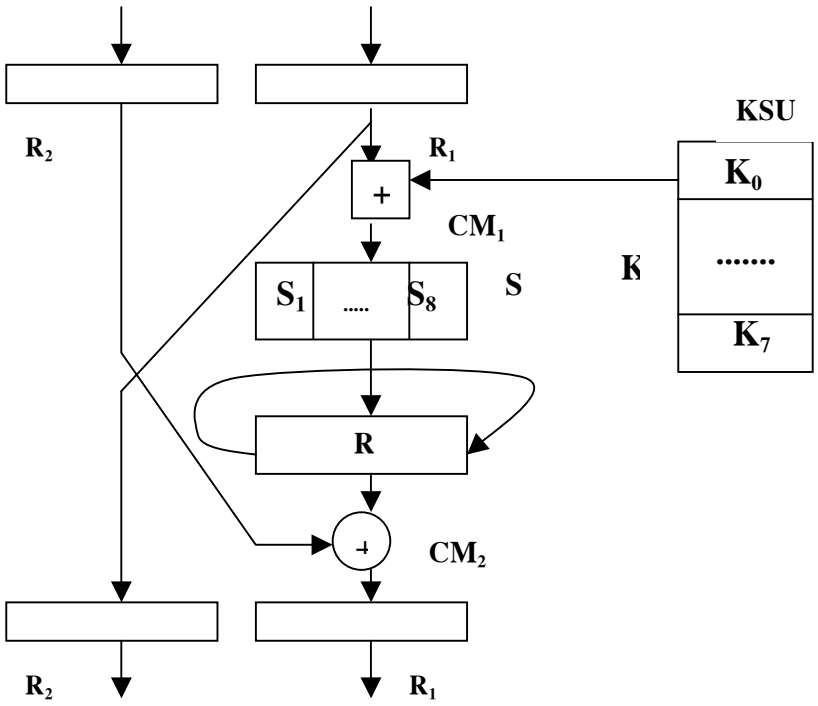
C. Charnes, L. O'Connor, J. Pieprzyk,
R. Safavi-Naini, Y. Zheng

1. Mở đầu

Mô tả chi tiết của thuật toán mã hóa Liên xô đã được công bố trong GOST 28147-89 [2]. Mục đích của những người thiết kế là cung cấp một thuật toán mã hóa có độ mật mềm dẻo. Thuật toán là một ví dụ của hệ mật kiểu DES cùng với lịch trình khóa được đơn giản hóa tối đa. Nó mã thông báo 64-bit thành bản mã có 64-bit bằng khóa 256-bit. Tài liệu về GOST [2] đã khuyến cáo 4 dạng sử dụng sau: thay thế đơn giản (electronic code-book mode), kiểu dòng (stream mode, trong [2] được gọi là Γ-mode), kiểu dòng có phản hồi (stream mode with feedback) và kiểu xác thực (authentication mode).

2. Mô tả thuật toán GOST

Thuật toán GOST gồm 32 vòng lặp (hai lần nhiều hơn DES). Mỗi vòng lặp được chỉ ra ở hình 1. Có hai phần tử là bí mật trong thuật toán này: khóa mã K 256-bit và định nghĩa các S-box S_1, ..., S_8.



Hình 1. Đường đi của dữ liệu trong một vòng mã/dịch của GOST

[11] **Advances in Cryptology- EUROCRYPT-94, p.434-438**

Khóa mật mã K = (K_0,..., K_7) được lưu trữ trong thiết bị lưu trữ khóa (key storage unit-KSU) như một dãy của 8 từ 32-bit (K_0,..., K_7). Mỗi từ khóa 32-bit K_i được gọi là khóa thành phần (i=0,...,7). Để mã một văn bản 64-bit, trước hết nó được chia thành 2 nửa 32-bit và được đặt vào thanh ghi 32-bit R_1 và R_2. Nội dung của thanh ghi R_1 được cộng theo modulo 2^{32} vào khóa thành phần K_0 (bộ cộng CM_1), tức là

$$R_1 + K_0 (\mathrm{mod}\, 2^{32})$$

Dãy thu được 32-bit được chia thành 8 khối 4-bit. 8 khối 4-bit này là đầu vào của 8 S-box tương ứng S_1,..., S_8. Mỗi S_i, i=1,...,8 là một phép hoán vị. 8 đầu ra 4-bit của các S-box được lưu vào thanh ghi dịch R, nội dung của thanh này được dịch trái 11 bit (về phía bit bậc cao). Nội dung của thanh R bây giờ được cộng modulo 2 (Exclusive-Or hoặc XOR) với nội dung của thanh ghi R_2 bằng bộ cộng CM_2. Nội dung từ sẽ được lưu trong R_1 và giá trị cũ của R_1 được lưu trong R_2. Đến đây kết thúc vòng lặp thứ nhất.

Các vòng lặp khác tương tự như vòng lặp thứ nhất. Trong vòng lặp thứ hai, chúng ta sử dụng khóa K_1 từ KSU. Các vòng lặp thứ 3, 4, 5, 6, 7, 8 sử dụng tương ứng các khóa thành phần K_2, K_3,..., K_7. Các vòng lặp từ 9 đến 16 và từ 17 đến 24 cũng sử dụng các khóa thành phần này. Các vòng lặp từ 25 đến 32 sử dụng các khóa thành phần theo thứ tự ngược lại, tức là vòng lặp thứ 25 sử dụng khóa K_7, vòng lặp thứ 26 sử dụng khóa K_6 và cứ tiếp tục như vậy. Vòng lặp cuối cùng dùng khóa K_0. Cho nên thứ tự của các khóa thành phần trong 32 vòng lặp là:

K_0,...., K_7, K_0,...., K_7, K_0,...., K_7, K_7,...., K_0.

Sau 32 vòng lặp, đầu ra từ bộ cộng CM_2 được đặt trong R_2, còn R_1 giữ nguyên giá trị cũ. Nội dung của các thanh ghi R_1 và R_2 là bản mã 64-bit cho bản rõ có 64-bit.

3. Các tính chất tổng quát của GOST

Thuật toán GOST lặp lại cấu trúc tổng thể của DES. Rõ ràng là những người thiết kế nó đã cố gắng để đạt được sự cân bằng giữa tính hiệu quả của thuật toán và độ mật của nó. Nó sử dụng các khối được xây dựng thường lệ và đơn giản. Đặc biệt, GOST khác DES ở những điểm sau:

1. Lịch trình khóa phức tạp được bỏ qua và thay bằng dãy có qui tắc của các khóa thành phần.
2. Khóa mật mã có độ dài đến 256 bit so với 56 bit của DES. Hơn nữa, lượng thực tế của thông tin mật trong hệ thống, bao gồm cả các S-box, gộp lại xấp xỉ 610 bit thông tin.
3. 8 hộp S-box S_1, ..., S_8 là các hoán vị S_i: $GF(2^4) \rightarrow GF(2^4)$, về tổng thể chỉ đòi hỏi không gian lưu trữ tương đương với 2 S-box của DES.
4. Khóa con cho mỗi vòng lặp là 32-bit cho một phép cộng có nhớ chứ không phải là phép XOR 48-bit như trong DES.

5. Phép hoán vị khối bất qui tắc P trong DES được thay bởi thanh ghi dịch đơn giản R, nó quay nội dung đi 11 bit về bên trái sau mỗi vòng.
6. Số vòng được tăng từ 16 lên 32.

Do độ mật của thuật toán phụ thuộc cả vào khóa mật mã và 8 phép thế S_i, i=1,..,8, nên người sử dụng cần phải biết nên chọn 2 thành phần bí mật này như thế nào? Khóa mật mã có thể chọn ngẫu nhiên, nhưng việc chọn các hóan vị S_i được dành cho người có chức trách, người biết chọn những hoán vị tốt. Từ quan điểm của người sử dụng, độ mật được liên quan tới tính bảo mật của khóa K. Chú ý rằng người có trách nhiệm có thể chọn các S-box sao cho họ có thể phá được hệ mật (ví dụ, bằng cách chọn các hoán vị tuyến tính hay affine).

Chúng ta hãy xem xét cấu trúc của thuật toán GOST, bạn có thể hỏi xem phải chăng việc sử dụng các hoán vị thay cho một lớp lớn hơn nhiều tất cả các hàm số có thể làm giảm độ an toàn? Even và Goldreich [1] đã chứng minh rằng một phép mã kiểu DES bất kỳ với một phép lặp

$$L'=R$$
$$R'=L\oplus f(R)$$

sinh ra một nhóm luân phiên (alternating group) với $f:GF(2^{32}) \rightarrow GF(2^{32})$ là hàm Boolean, đầu vào là (L, R), còn đầu ra là (L', R'). Sau đó Pieprzyk và Zhang [3] đã chỉ ra rằng nếu f là hoán vị thì hàm mã kiểu DES vẫn sinh ra nhóm luân phiên. Như vậy, việc sử dụng các hoán vị thay cho các hàm không làm suy giảm độ mật của thuật toán khi xem xét với một số vòng lớn.

Việc kết nối của CM_1, các S-box và phép dịch vòng R có thể xem như hàm vòng F. Hàm F ánh xạ chuỗi đầu vào 32-bit vào chuỗi đầu ra có cùng độ dài, đối tượng bị kiểm soát bởi khóa 32-bit. Phần trung tâm của hàm F là 8 S-box kích thước 4*4. Trước hết, hàm F thực hiện việc chia chuỗi đầu vào 32-bit thành 8 khối, mỗi khối 4 bit, và sau đó thay mỗi khối bằng 4 bit được định ra bởi S-box tương ứng.

Bạn có thể thấy các bit ra của hàm F bị ảnh hưởng bởi các tổ hợp khác nhau của đầu vào phụ thuộc vào vị trí của nó. Điều này được giải thích bởi tính chất của việc kết nối phép cộng modulo 2^{32} và các S-box. Phép cộng modulo 2^{32} tạo đầu ra là phi tuyến tất cả (trừ một trường hợp- bit ra có nghĩa nhỏ nhất). Điều này dẫn chúng ta tới bổ đề.

Bổ đề 1. Các đầu ra của S_i bị ảnh hưởng bởi 4i bit của bản rõ từ thanh ghi R_1 và 4i bit của khóa thành phần (i=1,...,8).

Vị trí của S_1 làm cho nó đặc biệt dễ bị tổn thương đối với tấn công tuyến tính. Hoán vị S_1 được chọn cẩn thận nhất sao cho đầu ra của S_1 có độ phi tuyến tối

đa. Nếu quan tâm đến hàm F thì GOST hoàn toàn sánh được với DES vì nó có quan hệ giữa đầu vào/đầu ra phức tạp hơn.

Một vấn đề hết sức thú vị là việc chọn các S-box sao cho nó có tính phi tuyến cao. Nói chung, phép cộng làm tăng tính tuyến tính, nhưng cũng có trường hợp nó làm suy giảm độ phi tuyến của hàm F đến mức kém hơn độ phi tuyến của chính các S-box.

4. Các phép dịch vòng R trong GOST

Ảnh hưởng chính của hàm vòng là cung cấp tính khuyếch tán. Để nghiên cứu điều này, chúng ta giả thiết rằng KSU=0 và bất chấp điều này. Trước hết chúng ta chỉ tập trung vào hiệu ứng trộn của phép dịch vòng trong một nhánh của thuật toán. Chúng ta xem xét hai trường hợp của thuật toán trong chế độ thay thế đơn giản:

1. Các S-box là ánh xạ đồng nhất
2. Các S-box là các hàm căng hoàn toàn (complete spread function), có nghĩa là mỗi bit vào ảnh hưởng đến mọi đầu ra của S-box, hay tương đương là mỗi bit ra phụ thuộc vào mọi bit vào.

Ký hiệu R_1^i là đầu vào của nửa bên phải của thuật toán tại vòng thứ i và $a_0^i,.....,a_{31}^i$ là từng bít đầu vào tại vòng này.

4.1 Trường hợp 1: Các S-box là ánh xạ đồng nhất

Chúng ta xem xét các bit bị ảnh hưởng bởi a_0^1, bit đầu vào thứ nhất của vòng 1. Chúng ta có

$$a_0^1 \to a_{11}^2 \to a_{22}^3 \to a_1^4 \to a_{12}^5 \to a_{23}^6 \to a_2^7 \Rightarrow$$
$$a_3^{10} \Rightarrow a_4^{13} \Rightarrow a_5^{16} \Rightarrow a_6^{19} \Rightarrow a_7^{22} \Rightarrow a_8^{25} \Rightarrow a_9^{28} \Rightarrow a_{10}^{31}$$

Ký hiệu $\Rightarrow$ có nghĩa là sau 3 vòng lặp. Như vậy, sau 32 vòng lặp thì a_0^1 ảnh hưởng tới tất cả các bit khác của một nửa R_1 đúng một lần. Dễ dàng thấy rằng mọi phép dịch vòng rot(i) khác (dịch R_1 đi i vị trí) cũng có tính chất này nếu như gcd(i,32)=1, hay có nghĩa là i lẻ.

4.2 Trường hợp 2: Các S-box là hàm căng hoàn toàn

Bây giờ chúng ta xem xét ảnh hưởng của bit a_0^1 trong trường hợp đầu vào của S-box ảnh hưởng tới mọi bit ra. Ảnh hưởng của bit này được lan truyền tới tất cả 32 bit của R_1 chỉ sau 8 vòng. Chúng ta cũng chú ý rằng

$$a_0^1 \Rightarrow a_0^4 \to a_0^5 \Rightarrow a_0^7 \to a_0^8$$

trong đó $\to$ chỉ 1 vòng lặp còn $\Rightarrow$ chỉ nhiều vòng lặp.

Mức độ lan truyền của mỗi vòng xác định các mối phụ thuộc hàm, có nghĩa là nếu tại vòng 1, 16 bit được ảnh hưởng thì một bit bị ảnh hưởng tại vòng 4 phụ thuộc vào 16 bit đầu vào.(??)

Chúng ta chú ý rằng nếu phép dịch vòng là không phải bội của 4 thì sự lan truyền xảy ra và 8 vòng là con số cần thiết để ảnh hưởng tới tất cả mọi bit của R_1. Tuy nhiên, việc lan truyền phụ thuộc vào phép dịch. Ví dụ, nếu phép dịch là rot(1) thì ảnh hưởng của a_0^1 không đến được a_{31}^i với i<8. Chúng ta có thể so sánh các phép quay vòng bằng cách đưa ra độ đo ρ(i)- *đó là số vòng nhỏ nhất cần thiết để các bit bị ảnh hưởng chiếm tất cả các vị trí trong R_1*. Chúng ta đã thấy rằng ρ(1)=8 và ρ(11) = 4.

Vì gcd(i,32)=1 nên hoặc i ≡ 1 (mod 4) hoặc i ≡ 3 (mod 4). Bây giờ, thay đổi 1 hoặc 3 bit đầu vào ảnh hưởng tới cả 4 bit đầu ra của S-box. Cho nên để so sánh ảnh hưởng của các phép quay chúng ta chỉ cần xem xét các phép quay rot(i) với hoặc i=1, 5, ..., 29 hoặc i=3, 7,..., 31. ρ(i) hoàn toàn được xác định bằng thương của i chia cho 4. Từ chú ý này chúng ta kết luận rằng số vòng nhỏ nhất sao cho các bit bị ảnh hưởng bởi a_0^1 bao phủ tất cả các vị trí trong R_1 ít nhất một lần được chỉ ra như trong Bảng 1. Chúng ta chú ý rằng giá trị nhỏ nhất của ρ(i) bằng 4; điều này xảy ra với các phép quay rot(9), rot(11), rot(21) và rot(23). Có thể thấy rằng số vòng nhỏ nhất cần thiết cho khuyếch tán hoàn toàn ít nhất bằng 4 đối với tất cả các phép quay. Với bất kỳ phép quay nào, khối 4-bit khuyếch tán thành khối 8-bit sau 2 vòng. Sau 3 vòng, khối 8 bit khuyếch tán thành khối 12-bit. Mặc cho các khối này được sắp xếp như thế nào chúng vẫn không phủ hết 32 bit (Nhiều nhất nếu chúng không có vùng chung, chúng sẽ phủ 4 +8+12=24 bit). Cho nên ít nhất cần 4 vòng để có khuyếch tán hoàn toàn.

Bảng 1. Sự lan truyền gây ra bởi phép quay

rot(i)	1/3	5/7	9/11	13/15	17/19	21/23	25/27	29/31
ρi)	8	5	4	5	5	4	5	8

Cũng chú ý rằng 11 và 23 không là ước của 2^{32}-1, đó là modulus của bộ cộng CM_4 (bộ cộng CM_4 được sử dụng trong chế độ dòng –xem [2]).Điều này có ảnh hưởng tới quyết định chọn phép quay 11 bit cho thanh ghi dịch vòng.

Trong phân tích trên chúng ta còn chưa tính đến việc đổi 2 nửa. Để nghiên cứu việc này chúng ta có thể bắt đầu thuật toán với R_2=0. Giả sử R_2^i ký hiệu đầu vào bên tay trái của thuật toán tại vòng thứ i, chúng ta cũng ký hiệu rot(i) bởi r_i và S là ánh xạ sinh bởi S-box. (S: $GF(2^{32}) \rightarrow GF(2^{32})$). Cùng với các qui ước này, các phương trình tượng trưng cho 2 vòng của thuật toán là:

$$R_1^2 = R_1 \oplus r_i S r_i S R_1$$

$$R_2^2 = r_i S R_1$$

và sau 3 vòng là

$$R_1^3 = r_1 S R_1 \oplus r_i S(R_1 \oplus r_i S r_i S R_1),$$
$$R_2^3 = R_1 \oplus r_i S r_i S R_1$$

Nếu chúng ta giả thiết rằng

$$r_i S(R_1 \oplus r_i S r_i S R_1) = r_i S R_1 \oplus r_i S r_i S r_i S R_1$$

chúng ta có thể nói một điều gì đó về tính khuyếch tán của R_1 bởi hai nửa.Sử dụng quan hệ này, chúng at thấy rằng sau 5 vòng cả R_1^5 và R_2^5 đều chứa thành phần $r_1 S r_i S r_i S r_i S R_1$. Nhưng cái này có thể viết lại nếu sử dụng dữ kiện là $r_i S = S' r_i$ với S' nào đó (các r_i và các S-box tạo thành nhóm). Cho nên

$$r_1 S r_i S r_i S r_i S R_1 = S^{(4)} r_1^4 R_1$$

($S^{(n)}$ ký hiệu tổ hợp của n S-box, tức là tích của các hoán vị được sinh ta). Theo Bảng 1, chúng ta thấy rằng 5 vòng là yêu cầu cho tính khuyếch tán theo cả hai bên cho thuật toán với các phép quay rot(9), rot(11), rot(21), rot(23).

5.Lựa chọn các S-box

Chúng ta chú ý rằng GOST có độ dài khóa hữu ích xấp xỉ 610 bit, trong đó 256 bit được sử dụng để biểu diễn khóa còn các bit còn lại để biểu diễn các S-box. Mỗi một trong 8 S-box là một phép hoán vị của các số nguyên [0, 1, 2, .., 15] và có cả thảy $16! \cong 2^{44.2}$ ánh xạ như vậy. Từ đó suy ra rằng cần 354≅ 8* 44.2 bit để chỉ ra 8 S-box ngẫu nhiên từ tập tất cả các hoán vị 4-bit, tựu trung lại là 610=256+354 bit khóa. Để giảm độ dài khóa, những người thiết kế có thể làm bằng cách sinh ta một tập các S-box có kích thước tương đối nhỏ, ví dụ như 10,000 và sử dụng khóa để chỉ ra S-box trong tập được chọn cố định ấy.

Như đã nói ở trên, tập tất cả các S-box là rất lớn, và không cho phép vét cạn để tìm ra S-box nhằm tối ưu hóa các tiêu chuẩn. Các thí nghiệm đã được làm bằng cách chọn ngẫu nhiên các S-box, sau đó các S-box đã chọn được kiểm tra tính thích hợp bằng tấn công vi sai và tấn công tuyến tính.

Tài liệu tham khảo

1. S. Even and O. Goldreich. Des-like functions can generate the alternating group. IEEE Transactions on Information Theory, 29(6):863-865, November 1983.
2. National Soviet Bureau of Standards. Information Processing Systems. Cryptographic Protection. Cryptographic Algorithm. GOST 28147-89, 1989.
3. J. P. Pierrzyk and Xian-Mo Zhang. Permutation generators of alternating groups. In Advances in Cryptology- AUSCRYPT'90, J.Sebery, J. Pieprzyk (Eds), Lecture Notes in Computer Science, Vol.453, pages 237-244. Springer Verlag, 1990.