**Báo cáo môn Cơ sở an toàn thông tin nhóm 10**

**1. Giới thiệu chung về bảo mật PGP:**

**1.1 Mã hoá PGP là gì?**

Mã hóa PGP (Pretty Good Privacy) là một hệ thống được sử dụng cho việc mã hoá email và mã hóa các file nhạy cảm. Kể từ khi được Phil Zimmermann phát minh vào năm 1991, PGP đã trở thành thước đo tiêu chuẩn trong việc bảo mật email.

Sự phổ biến của PGP dựa trên hai yếu tố. Đầu tiên là hệ thống này ban đầu được cung cấp dưới dạng phần mềm miễn phí và lan truyền nhanh chóng giữa những user muốn có thêm mức độ bảo mật cho các email của họ. Thứ hai là vì PGP sử dụng cả mã hóa symmetric và mã hóa public-key, nó cho phép những user chưa từng gặp mặt gửi tin nhắn được mã hóa cho nhau mà không cần trao đổi private encryption key.

**1.2 Mục đích sử dụng PGP:**

Mục đích sử dụng PGP là phục vụ cho việc mã hóa thư điện tử. Phần mềm mã nguồn mở PGP hiện nay đã trở thành một giải pháp mã hóa cho các công ty lớn, chính phủ cũng như các cá nhân. Các ứng dụng của PGP được dùng để mã hóa bảo vệ thông tin lưu trữ trên máy tính xách tay, máy tính để bàn, máy chủ và trong quá trình trao đổ email hoặc chuyển file, chữ ký số…

**1.3 PGP hoạt động như thế nào:**

PGP hoạt động bằng cách đặt các lớp bảo mật được mã hóa lên trên phần nội dung dựa trên văn bản của ứng dụng.

Trong trường hợp email client, PGP bảo mật nội dung email bằng thuật toán mã hóa, xáo trộn văn bản theo cách mà nếu bị chặn thì cũng không thể đọc được.

Với nội dung của một email được xáo trộn, key tương ứng cần thiết để “mở khóa” mã đó cũng được mã hóa, thường sử dụng public key (khóa công khai) do RSA hoặc Diffie-Hellman cung cấp. Sau đó, email được mã hóa, cùng với key mã hóa tương ứng được gửi đến người nhận.

Khi gói này đến email client của người nhận, ứng dụng sẽ sử dụng private key (khóa riêng tư) để mở khóa mã hóa email và sau đó giải mã nội dung.

**2. Nội dung:**

**2.1. Giải thuật trong PGP:**

2.1.1. Mã hóa đối xứng

*a) IDEA ( Improved Proposed Encyption Standard )*

IDEA ra đời từ những năm 1991 có tên IPES (Improved Proposed Encyption Standard). Đến năm 1992 được đổi tên thành IDEA được tạo ra bởi 2 tác giả là Xuejia Lai và James Massey. Thiết kế loại mã này dựa trên phép cộng modulo 2(OR), phép cộng modulo 216 và phép nhân modulo 216+1 (số nguyên tố 65537). Loại mã này rất nhanh về phần mềm (mọi chíp xử lý của máy tính cá nhân có thể thực hiẹn phép nhân bằng một lệnh đơn).

IDEA được sử dụng trong PGP v2.0 và được kết hợp sau khi phiên bản v1.0, BassOmatic được phát hiện là không an toàn. Tính đến thời điểm hiện tại vẫn chưa có cuộc tấn công nào cho phép phá hủy được hoàn toàn thuật toán IDEA do đó đây là một thuật toán có độ an toàn cao.

IDEA là loại mã khối sử dụng một Chìa khóa 128 bit để mã hóa dữ liệu trong những khối 64 bít với 8 vòng lặp. Mỗi lần lặp IDEA sử dụng 3 phép toán khác nhau, mỗi phép toán thao tác trên hai đầu vào 16 bít để sản sinh một đầu ra 16 bít đơn. Ba phép toán đó là:

• Phép XOR theo bít

• 2. Phép cộng modulo 216 với đầu vào và đầu ra là những số nguyênkhông dấu 16 bít. Hàm này láy hai số nguyên 16 bit làm dầu vào và sản sinh một tổng 16 bít; nếu bị tràn sang bít thứ 17, thì bit này bị vứt bỏ.

• Phép nhân số nguyên theo modulo 216+1. với đầu vào và đầu ra là những số nguyên 16 bít. Trừ trường hợp cả khối đều là 0 thì được xem như 216

*b) 3DES*

Thuật toán DES (Data Encryption Standard) được chính phủ Mỹ tạo ra năm 1977 (NIST và NSA) dựa trên các công việc mà IBM làm. DES thuộc loại mã khối 64 bits với khoá dài 64 bits. Thuật toán DES đầu tiên đã được nghiên cứu trong thời gian dài nhưng nó vẫn có thể bẻ gãy bằng những máy có mục đích đặc biệtđể tìm ra khóa.

Thuật toán 3DES gồm 3 chìa khóa 64 bit, tức là chiều dài khóa là 192 bit. Thuật toán 3DES cải thiện độ mạnh của thuật toán DES bằng việc sử dụng một quá trình mã hóa và giải mã sử dụng 3 khóa với kích cỡ không gian khóa 168 bit. Các chuyên gia xác định rằng 3DES rất an toàn. Nhược điểm của nó là chậm hơn một cách đáng kể so với các thuật toán khác. Bản thân DES đã chậm do dùng các phép hoán vị bit. Lý do duy nhất để dùng 3DES là nó đó được nghiên cứu rất kỹ lưỡng.

2.1.2. Mã hóa bất đối xứng:

*a) RSA*

Thuật toán RSA được phát minh năm 1978 bởi Rivest, Shamir và Adleman được coi như là một hệ chuẩn đối với các hệ mật mã khoá công khai. Thuật toán RSA có hai khóa: khóa công khai (hay khóa công cộng) và khóa bí mật (hay khóa cá nhân). Mỗi khóa là những số cố định sử dụng trong quá trình mã hóa và giải mã. Khóa công khai được công bố rộng rãi cho mọi người và được dùng để mã hóa. Những thông tin được mã hóa bằng khóa công khai chỉ có thể được giải mã bằng khóa bí mật tương ứng. Nói cách khác, mọi người đều có thể mã hóa nhưng chỉ có người biết khóa cá nhân (bí mật) mới có thể giải mã được.

Thuật toán sử dụng chế độ mã hóa khối P, C là một số nguyên ∈ (0, n)

Nhắc lại: C = EPU (P): mã hóa khóa PU

P = DPR(EPU (P)): giải mã khóa PR (ko cho phép tính được PR từ PU)

Dạng mã hóa/ giải mã:

C = Pe mod n

P = cd mod n = Ped mod n

PU = {e, u} -> Public

PR = {d, n} -> Private

- Người gửi và người nhận biết giá trị của n và e, nhưng chỉ người nhận biết giá trị của d

- Mục đích: tìm các giá trị e, d, n (chọn) để tính P và C

Nhận xét:

-Có thể tìm giá trị của e, d, n sao cho Ped = P mod n với P < n

-Không thể xác định d nếu biết e và n

*b) ElGamal/ Diffie Hellman*

Trong PGP thuật toán Diffie Hellman được gọi là DH và thường được dung để trao đổi khoá và không được dùng để ký. Vì nếu dùng để ký thì chữ ký sẽ khá lớn. Trong lúc đó, ElGamal có thể dùng để ký và bảo mật mặc dù chữ ký sẽ phải dùng hai số cùng kích thước là 1024 bit trong khi RSA chỉ cần một con số có độ dài là 1024 bit. Đối với DSA thì chỉ cần 2 con số có độ dài là 160 bit.

*c) DSA*

DSA là một phiên bản đăc biệt của ElGamal. Đây là phiên bản ElGamal cần một lượng lớn các tính toán đối với con số có độ dài 1024 bit, mặc dù các con số chữ ký được chọn ra là một tập con của 2160 phần tử. Các nhà thiết kế đã thành công khi tạo ra một thủ tục chỉ cần 160 bit để thể hiện nhóm con của các phần tử đó. Điều này đã làm cho các chữ ký được sinh ra có kích thước khá nhỏ nó chỉ cần hai con số có độ lớn là 160 bit thay vì phải dùng hai số lớn có độ dài 1024 bit

2.1.3 Hàm Hash:

Hàm hash được định nghĩa là một ánh xạ

H: X-->{0,1}k

Trong đó X là không gian các bản rõ độ dài tuỳ ý, {0,1}k là tập các dãy số 0,1 có độ dài K cho trước. Hàm Hash được xây dựng sao cho thỏa mãn các tínhchất cơ bản sau:

1. Tính chất một chiều

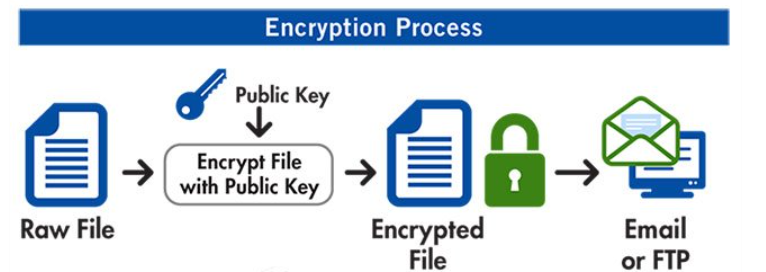
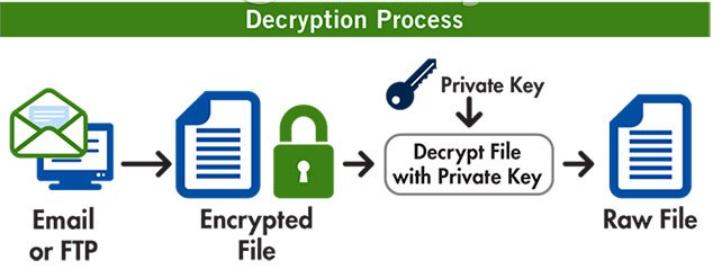
2. Hàm Hash yếu

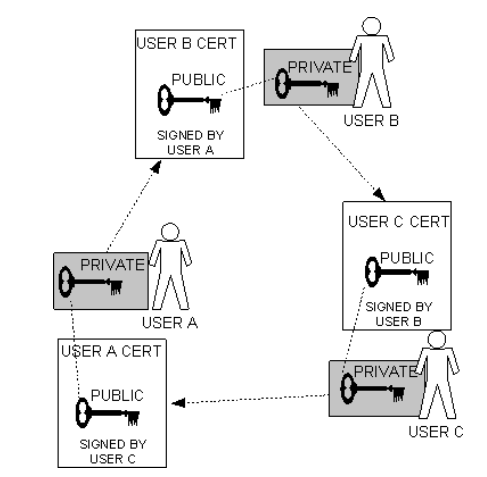
3. Hàm Hash mạnh

Hàm hash được gọi là thoả mãn tính chất một chiều nếu cho trước giá trị hash (giá trị đã được rút gọn) Z thì về mặt tính toán không thể tìm được giá trị thông báo x sao cho Z = h(x). Tuy nhiên cho trước thông báo x thì việc xác định Z = h(x) lại được thực hiện nhanh chóng.Hàm h được gọi là có tính chất yếu nếu cho trước một thông báo x thì về mặt tính toán không thể tìm được một thông báo x’≠x (x,x’∈X) sao cho h(x’)=h(x).Còn hàm hash được gọi là có tính chất mạnh nếu tổng thực hành không thể tìm được 2 thông báo x, x’∈X khác nhau sao cho h(x’) = h(x). Các phiên bản trước PGP sử dụng hàm băm MD5 để băm dữ liệu còn hiện nay MD5 được thay bằng thuật toán SHA

**2.2. Mô hình kiến trúc**

\*Mô hình tổng quan:





Hai dịch vụ chính mà PGP cung cấp cho người dùng là: mã hóa và xác thực thông điệp. Khi thiết kế một ứng dụng bảo mật email, người thiết kế phải đương đầu với hai vấn đề chính, trước hết, phải bảo mật ứng dụng bằng những giải thuật nào?

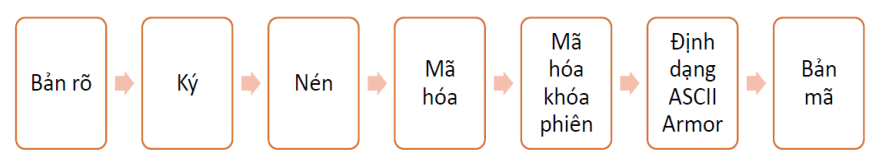
Trong trường hợp của PGP, những dịch vụ của nó dựa vào ba giải thuật: IDEA (mã hóa khóa bí mật), RSA (mã hóa khóa công khai) và MD5 (Hàm băm an toàn).

**2.3. Hoạt động**

2.3.1 Mã hoá

Hoạt động thực tế của PGP để gửi và nhận thông điệp bao gồm năm dịch

vụ: chữ ký số, mã hóa thông điệp trong PGP. Quy trình thực hiện theo các bước sau:



**Hình 1: Quá trình mã hóa 1 thông điệp trong PGP**

*a) Chuẩn bị file*

Mỗi lần thực hiện, PGP chỉ xử lý một file. Những file được xử lý bởi PGP nói chung thường là văn bản. Đây là dạng phổ biến nhất của truyền thông email. Nhưng PGP có thể chấp nhận bất kỳ file nào, kể cả file nhị phân, file PICT... Một trong những dịch vụ tiện lợi do PGP cung cấp cho phép người dùng gửi file theo đường email bình thường.

*b) Chữ ký số*

Khi nhận file đầu vào, bước đầu tiên của PGP là tạo một chữ ký số để gán vào file. Đây chỉ là một dịch vụ lựa chọn. Nếu người gửi yêu cầu chữ ký số, PGP sẽ tạo một mã băm của file và sau đó mã hóa mã băm với RSA sử dụng cho khóa riêng tư người gửi. Kết quả mã hóa mã băm là chữ ký số cho file này. Chữ ký số bảo đảm file này là của người gửi và file đó không bị biến dạng.

*c) Nén*

Việc nén lại sẽ giúp tiết kiệm thời gian truyền, không gian đĩa và quan trọng hơn là giúp tăng cường tính bảo mật của mật mã. Hầu hết các kỹ thuật phân tích mã hóa được tìm thấy trong bản rõ để phá mật mã. Nén làm giảm bớt đi các mô hình này, qua đó giúp tăng cường khả năng chống giải mã. Tuy nhiên người dùng có thể lựa chọn dùng nén hoặc không.

*d) Mã hóa*

Đầu tiên người dùng sẽ sử dụng thuật toán mã hóa đối xứng mã hóa bản rõ bằng một khóa chung (còn gọi là khóa phiên). Tiếp theo người dùng sẽ sử dụng cặp khóa công khai bí mật được tạo bởi thuật toán mã hóa bất đối xứng. Sử dụng khóa công khai trong cặp khóa công khai – bí mật mã hóa khóa phiên được tạo ra sau quá trình mã hóa bản rõ bằng thuật toán mã hóa đối xứng.

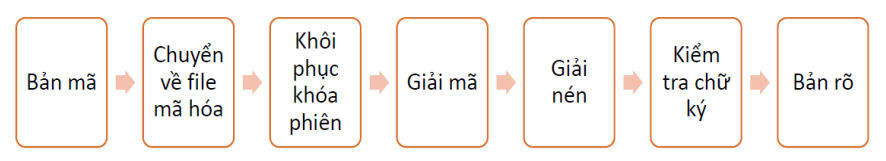
Phần mã hóa thông điệp gửi đi của PGP sử dụng cả hai thuật toán mã hóa đối xứng và mã hóa bất đối xứng để tận dụng ưu thế của cả hai. Thuật toán mã hóa bất đối xứng đảm bảo việc phân phối khóa phiên trong hệ thống với độ bảo mật cao còn thuật toán mã hóa bí mật có ưu thế về tốc độ mã hóa và giải mã (nhanh hơn cỡ 1000 lần).

*e) Tính tương thích Email*

Nếu ký, nén hoặc mã hóa được thực hiện trên file gốc thì khối dữ liệu được sản sinh ra là những dữ liệu nhị phân. Tuy nhiên, nhiều hệ thống email không thể xử lý với dữ liệu nhị phân mà chỉ có thể xử lý những file văn bản. Khắc phục hạn chế này, PGP chuyển đổi dữ liệu nhị phân thành những ký tự có thể in được. PGP sử dụng khuôn dạng ASCII armor để chuyển đổi dữ liệu.

2.3.2 Giải mã:

Hình dưới mô tả quá trình giải mã một thông điệp trong PGP. Về cơ bản, để giải mã, PGP chỉ cần thực hiện đảo ngược các bước của quá trình mã hóa.



**Hình 2: Quá trình giải mã một thông điệp trong PGP**

Đầu tiên PGP sẽ thực hiện việc chuyển file bản mã về lại dạng nhị phân để thực hiện giải mã. Tiếp theo người dùng sẽ sử dụng khóa riêng tư của mình trong cặp khóa công khai – riêng tư để thực hiện việc giải mã khóa phiên. Sau khi có được khóa phiên thực hiện việc quá trình giải mã bản rõ. Việc giải nén sẽ được thực hiện để khôi phục đầy đủ các mô hình trong văn bản. Cuối cùng là việc kiểm tra chữ ký để xem văn bản có bị sửa đổi hay xâm phạm trong quá trình truyền đi hay chưa.

2.3.3 Khóa:

Khóa là một giá trị làm việc với một thuật toán mã hóa để tạo ra một bản mã cụ thể. Về cơ bản khóa là những con số rất lớn. Kích thước của khóa được đo bằng bit. Trong các thuật toán mã hóa, khóa càng lớn thì tính bảo mật càng cao.

Tuy nhiên kích thước của cặp khóa công khai – bí mật so với khóa thông thường là không hề liên quang với nhau. Như một khóa thông thường 80 bit có sức mạnh tương đương với một khóa công khai 1024 bit. Kích thước khóa là quan trọng cho sự an toàn, nhưng các thuật toán được sử dụng cho từng loại là rất khác nhau. Vì thế không thể so sánh chỉ kích thước khóa của các hệ mật mã với nhau.

Nền tảng những thao tác của PGP là yêu cầu mỗi người dùng có một cặp khóa công khai – bí mật cũng như các bản sao chép các khóa công khai của người nhận. Mặc dù một cặp khóa công khai – bí mật về mặt toán học là có liên quan đến nhau, nó rất khó để có thể suy ra được một khóa bí mật nếu như chỉ có khóa công khai. Tuy nhiên, vẫn có thể suy ra được khóa bí mật nếu có đủ thời gian và khả năng tính toán. Điều này dẫn đến một vấn đề rất quan trọng là làm sao để chọn ra được một khóa đúng kích cỡ, tức là đủ lớn để có thể đảm bảo an toàn và đủ nhỏ để có thể áp dụng một cách nhanh chóng. Ngoài ra bạn cũng cần phải xem xét những ai có thể cố gắng đọc các tập tin của bạn, họ có bao nhiêu thời gian và khả năng họ có thể.

Khóa được lưu trữ ở dạng mã hóa. PGP lưu trữ các khóa trong hai tập tin trên đĩa cứng của bạn. Một cho khóa công cộng và một cho khóa bí mật. Những tập tin này được gọi là một vòng khóa.

*a) Khóa công khai*

PGP thường lưu lại những chìa khóa công khai mà người dùng thu được. Các khóa này được tập hợp và lưu lại trên vòng khóa công khai. Mỗi mục vòng gồm các phần:

- Khóa công khai

- User ID chủ nhân của khóa công khai này, tên đặc trưng của chủ nhân

- Một keyID, là định danh cho khóa này

- Thông tin khác liên quan đến độ tin cậy của khóa và chủ nhân của nó.

*b) Khóa bí mật*

Để sử dụng PGP, người dùng cần phải có một khóa bí mật. Nếu muốn người dùng có thể tạo nhiều khóa bí mật. Vòng khóa bí mật chứa đựng thông tin của mỗi khóa.

- Khóa riêng gồm 128 bit được sinh ra nhờ một passphrase và hàm băm

MD5

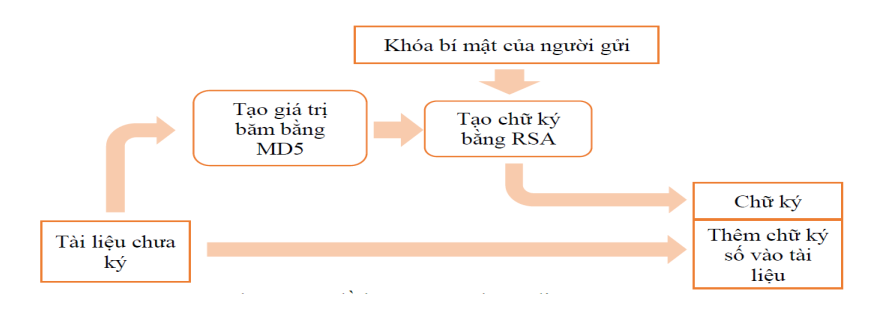
- User ID.

- Key ID của khóa công khai tương ứng.

2.3.4 Chữ ký số

Một chữ ký số phục vụ cùng một mục đích như một chữ ký viết tay. Tuy nhiên một chữ ký viết tay rất dễ dàng bị giả mạo. Một chữ ký số cao cấp hơn một chữ ký viết tay là gần như không thể làm giả, và nó là minh chứng cho nội dung của thông tin cũng như danh tính của người ký.

Chữ ký số cho người nhận thông tin xác minh tính xác thực của nguồn gốc thông tin, và cũng xác nhận rằng thông tin còn nguyên vẹn. Một chữ ký số công khai rất quan trọng trong cung cấp chứng thực và toàn vẹn dữ liệu.

Cách thức làm việc của chữ ký số được mô tả trong hình:

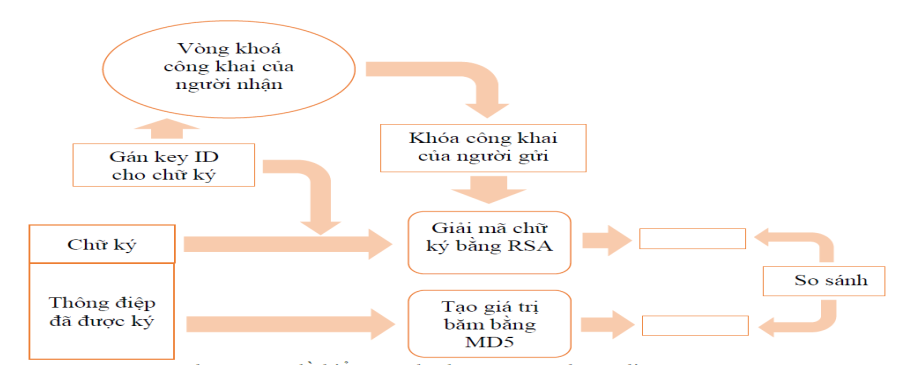
**Hình 3: Lược đồ ký trên 1 thông điệp PGP**

Người gửi tạo ra một thông điệp:

1. PGP sử dụng MD5 băm thông điệp tạo ra một mã băm 128 bit

2. Người gửi lấy khóa bí mật trên vòng khóa để sử dụng

3. PGP mã hóa mã băm bằng RSA sử dụng chìa khóa bí mật của người gửi, và gán kết quả vào thông điệp. Key ID của khóa công khai của người gửi tương ứng gắn liền với chữ ký



**Hình 4: Lược đồ kiểm tra chũ ký trên một thông điệp**

PGP của người nhận:

1. PGP lấy Key ID được gán trong chữ ký và sử dụng nó để lấy khóa công khai đúng từ vòng khóa công khai.

2. PGP sử dụng RSA với khóa công khai của người gửi để giải mã khôi phục mã băm.

3. PGP tạo ra một mã băm mới cho thông điệp và so sánh nó với mã băm

giải mã. Nếu cả hai trùng nhau, thông điệp được xác thực.

Sự kết hợp của MD5 và RSA cung cấp một sơ đồ chữ ký số hiệu quả. Với sức mạnh của RSA, người nhận chắc chắn rằng chỉ người sở hữu riêng với khóa thích hợp mới có thể tạo chữ ký. Với sức mạnh của MD5, người nhận chắc chắn rằng không ai khác có thể tạo ra một thong điệp mới mà mã băm trùng với mã băm của thông điệp gốc và vì vậy không thể trùng với chữ ký của thông điệp gốc.

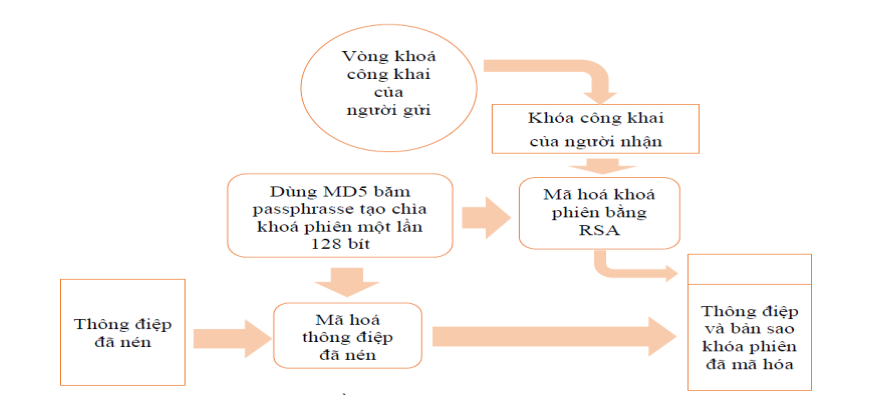
2.3.5 Nén

PGP sẽ mặc định nén thông điệp sau khi ký nhưng trước quá trình mã hóa. Điều này có lợi cho việc cất giữ không gian vừa cho truyền thông email vừa cho lưu trữ trên máy tính. PGP sử dụng giải thuật Zip để nén thông điệp. Thực chất giải thuật Zip tìm kiếm những chuỗi ký tự lặp lại trong dữ liệu vào và thay thế những chuỗi như vậy với những mã gọn hơn.

2.3.6 Mã hóa và giải mã thông điệp

Một dịch vụ cơ bản khác của PGP cung cấp là mã hóa những thông điệp để truyền đi hoặc cất giữ trên máy tính. Trong cả hai trường hợp đều sử dụng giải thuật mã hóa truyền thống IDEA. Những phiên bản mới nhất, PGP sử dụng thuật toán AES thay vì IDEA.

Trong khi các thuật toán mã hóa luôn chú trọng vào vấn đề phân phối khóa. Thì với PGP mỗi khoá truyền thống chỉ được sử dụng một lần; với mỗi thông điệp chỉ có một khóa 128 bít ngẫu nhiên được tạo ra. Vì chỉ được sử dụng một lần, nên khoá phiên được gắn vào thông điệp và truyền cùng với thông điệp. Để bảo vệ khoá phiên, PGP sử dụng RSA với khoá công cộng của người nhận.



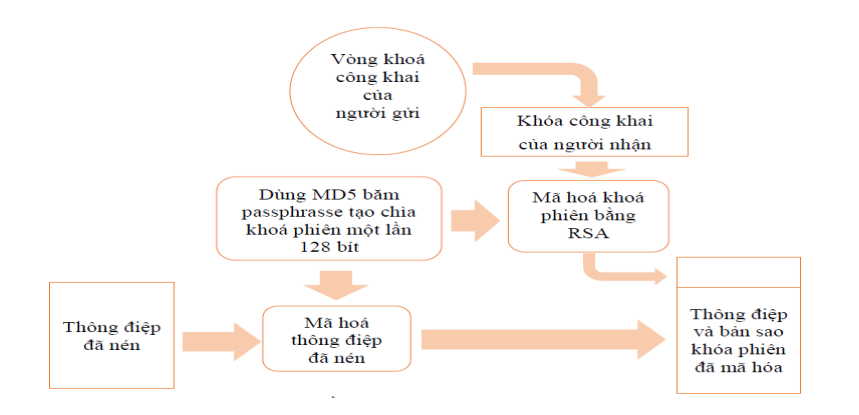
**Hình 5: Lược đồ mã hóa thông điệp trong PGP**

Hình 6 minh họa vấn đề này bao gồm các bước sau:

1. PGP chỉ tạo một số 128 bit ngẫu nhiên nhờ việc băm passphrase của người gửi bằng MD5 và sử dụng nó làm khóa phiên cho thông điệp.

2. PGP mã hóa thông điệp sử dụng khóa phiên.

3. PGP mã hóa khóa phiên với RSA. Sử dụng khóa công khai của người nhận được gắn vào khóa phiên đó mã hóa.



**Hình 6: Lược đồ giải mã thông điệp trong PGP**

Hình 7 mô tả quá trình giải mã thông điệp

1. PGP lấy key ID được gán vào thông điệp và sử dụng nó để lấy khóa bí mật đúng từ vòng khóa bí mật. Một người dùng có thể có hơn một khóa riêng.

2. Người nhận cung cấp một passphrase. Nó cho phép PGP giải mã khóa riêng của người nhận.

3. PGP sử dụng RSA với khóa riêng để giải mã và khôi phục khóa phiên.

4. PGP sử dụng khóa phiên giải mã thông điệp.

**2.4 Ứng dụng của PGP**

Mục tiêu ban đầu của PGP nhằm vào mật mã hóa nội dung các thông điệp thư điện tử và các tệp đính kèm cho người dùng phổ thông. Bắt đầu từ 2002, các sản phẩm PGP đã được đa dạng hóa thành một tập hợp ứng dụng mật mã và có thể được đặt dưới sự quản trị của một máy chủ. Các ứng dụng PGP giờ đây bao gồm: thư điện tử, chữ ký số, mật mã hóa ổ đĩa cứng máy tính xách tay, bảo mật tệp và thư mục, bảo mật các phiên trao đổi IM, mật mã hóa luồng chuyển tệp, bảo vệ các tệp và thư mục lưu trữ trên máy chủ mạng.

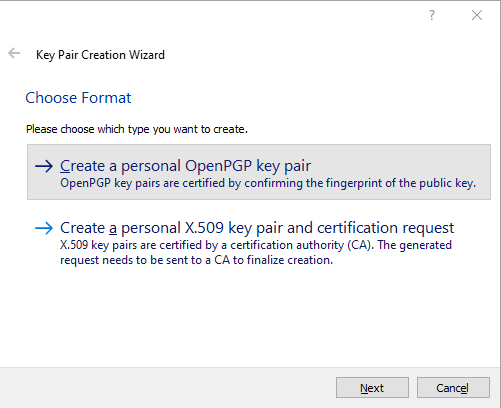
***1. Tạo cặp khóa bất đối xứng*:**

Truy cập <http://www.gpg4win.org/> tải về chương trình GnuPG từ trang chủ và thực hiện cài đặt bình thường.

Sau khi cài đặt xong thực hiện tạo cặp khóa PGP:

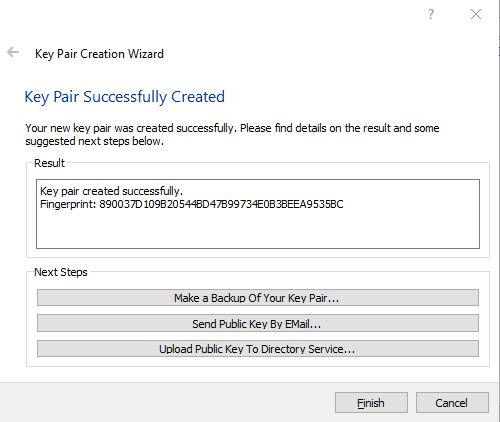
- Mở công cụ Kleopatra (giao diện đồ họa của gpg4win) để tạo một cặp khóa bất đối xứng mới (khóa công khai và bí mật). *File -> New Key Pair*.

- Bảng *Key Pair Creation Wizard* hiện ra, chọn kiểu khóa muốn tạo:



- Để tạo cặp khóa cá nhân, chọn *Create a personal OpenPGP key pair.* Sau đó điền đầy đủ thông tin *họ tên và email*, Trong tùy chọn nâng cao bạn có thể thiết lập dạng mã hóa bằng RSA hay DSA, và thời hạn của Key…trong phần cài đặt nâng cao (*Advanced Settings).* Nếu muốn tạo mật khẩu bảo vệ cặp khóa thì tích chọn *Protect the generated key with a passphrase.*

- Xem lại thông tin một lần nữa, sau đó click *Create,* sẽ có thông báo yêu cầu nhập và xác nhận mật khẩu bảo vệ. Bạn nên chọn một mật khẩu mạnh để chống lại các công cụ dò đoán mật khẩu. Cặp khóa của bạn sẽ được tạo trong vài giây (như hình).



- Bạn nên chọn “*Make a Backup If Your Key Pair*” để lưu khóa vào một nơi an toàn. Click *Finish* để kết thúc quá trình tạo cặp khóa.

- Chọn dòng chứa cặp khóa mới của bạn -> click chuột phải -> click *Export* để lưu khóa công khai trên desktop.

- Bạn sẽ phải trao đổi khóa công khai của bạn cho người nhận. Nhiều người đã chọn cách để khóa công khai trên trang web cá nhân của họ, hoặc cũng có thể gửi đính file đính kèm đến cho mọi người.

***2. Import Keys/Delete Keys:***

- Khi bạn có được Public Key của một ai đó. Bạn cần phải Add nó vào Key Database của bạn để sau này sẽ sử dụng đến nó. Bạn sẽ dùng chính nó để giải mã hoá các dữ liệu đã được chính chủ nhân của nó mã hoá bằng Public Key mà bạn đang có ở các lần sau. Ngược lại bạn cũng có thể xóa 1 key ra khỏi CSDL.

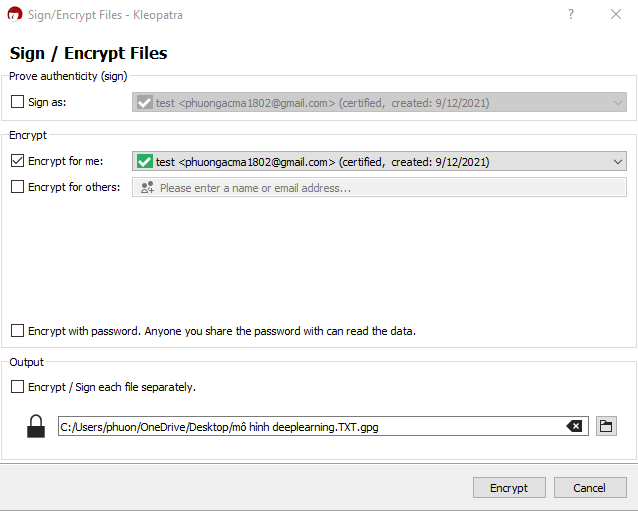
***3. Mã hoá và giải mã hoá (Encrypt And Decrypt):***

- Trong quá trình mã hóa và giải mã hóa không chỉ cẩn public key và secret key của bạn mà còn cần đến Public key của những người mà bạn muốn trao đổi dữ liệu với họ một cách an toàn. Khi mã hoá một đối tượng dữ liệu cho người khác thì bạn sẽ phải chọn chính Public Key của họ để mã hoá nó. Sau đó gửi cho họ, họ sẽ dùng chính Secret Key của mình để giải mã hoá dữ liệu mà bạn đã mã hoá bằng chính Public Key của họ. Chính vì vậy phương pháp mã hoá dữ liệu này tỏ ra rất an toàn.

***+ Mã hóa (Encrypt):***

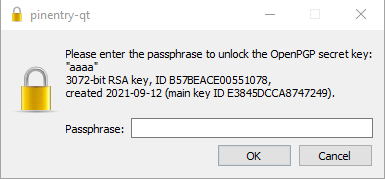
Trước khi muốn mã hoá dữ liệu và trao đổi với họ bạn phải có và đã bổ xung Public Key của họ vào Database Key của bạn. Nói một cách dễ hiểu ta đã dùng chính Public Key của họ để mã hoá dữ liệu rồi gửi lại cho họ.

* Chọn *File -> Sign and Encrypt*. Bảng Encrypt sẽ xuất hiện:
* Chọn *Encrypt for Me* hoặc *Encrypt for others* và điền Public key của người nhận.
* Chọn *Encrypt*. File data.gpg được tạo ra.

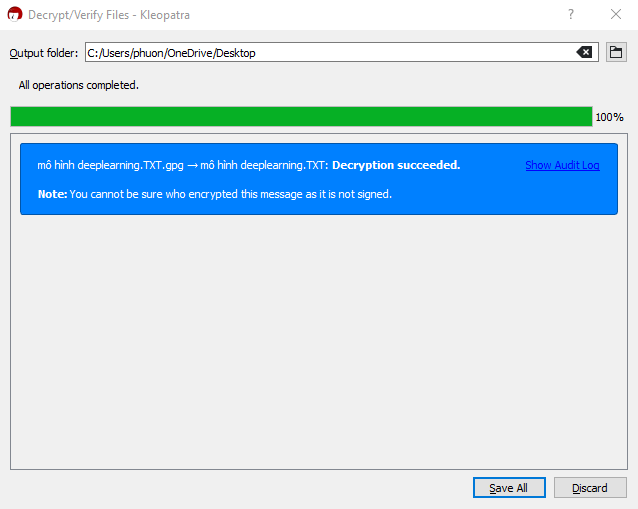
*****+ Giải mã (Decrypt):***

Sau khi nhận file đã được mã hóa, quá trình giải mã diễn ra như sau:

* Chọn *File -> Decrypt and Verify.* Sau đó chọn file mã hóa và nhập *passphrase* của người gửi.



* Bảng *Decrypt/Verify Files* xuất hiện:



* Chọn thư mục lưu file giải mã rồi chọn *Save All.*

**3. Kết luận :**

**3.1 PGP có an toàn không?**

Đã có một số tranh cãi về mức độ bảo mật của PGP. Vào năm 2011, các nhà nghiên cứu đã phát hiện ra rằng các khóa mã hóa ngắn (32-bit hoặc nhỏ hơn) không an toàn đến mức một số người cho rằng chúng chẳng có tác dụng bảo mật nào cả.

Nhưng điều này không có nghĩa là PGP có vấn đề nghiêm trọng về bảo mật, mà chỉ đơn giản là phải sử dụng khóa dài (lớn hơn 32-bit) mà thôi.

Và mặc dù đã có một số bài viết chỉ ra các lỗi bảo mật trong một số triển khai PGP, chẳng hạn như lỗ hổng Efail. Nhưng điều quan trọng là phải nhận ra rằng PGP vẫn rất an toàn.

**3.2 Ưu điểm và nhược điểm của PGP:**

*\* Ưu điểm của PGP:*

Ưu điểm chính của mã hóa PGP là không thể phá vỡ. Đó là lý do tại sao nó vẫn được các nhà báo và nhà hoạt động sử dụng và tại sao nó thường được coi là cách tốt nhất để cải thiện bảo mật cloud. Nói tóm lại, về cơ bản không ai có thể xâm nhập hệ thống. Cho dù là hacker hay thậm chí là NSA cũng không thể phá vỡ mã hóa PGP.

*\* Nhược điểm của PGP:*

Mấu chốt lớn nhất của mã hóa PGP là nó không thân thiện với user. Điều này đang thay đổi nhờ các giải pháp có sẵn mà PGP sẽ sớm áp dụng. Nhưng việc sử dụng PGP có thể tiết kiệm thời gian và công việc đáng kể cho lịch trình hàng ngày của bạn. Ngoài ra, những người sử dụng hệ thống cần phải biết cách thức hoạt động của nó, phòng trường hợp họ đưa ra các lỗ hổng bảo mật do sử dụng nó không đúng cách. Điều này có nghĩa là các doanh nghiệp đang cân nhắc chuyển sang PGP sẽ cần phải được đào tạo.

Vì lý do đó, nhiều doanh nghiệp có thể muốn xem xét các lựa chọn thay thế. Ví dụ, có những ứng dụng nhắn tin được mã hóa như Signal, cung cấp mã hóa dễ sử dụng hơn. Về mặt lưu trữ data, ẩn danh có thể là một giải pháp thay thế tốt cho mã hóa và có thể là cách sử dụng tài nguyên hiệu quả hơn.

Cuối cùng, bạn nên biết rằng PGP mã hóa tin nhắn của bạn, nhưng nó không làm cho bạn ẩn danh. Không giống như các trình duyệt ẩn danh sử dụng proxy server hoặc làm việc thông qua VPN để ẩn vị trí thực của bạn. Email được gửi qua PGP có thể được truy tìm từ người gửi và người nhận. Các dòng tiêu đề của chúng cũng không được mã hóa, vì vậy bạn không nên đưa bất kỳ thông tin nhạy cảm nào vào đó.

**3.3 Kết luận chung:**

Mã hóa PGP có thể là một công cụ mạnh mẽ trong việc bảo vệ data, quyền riêng tư và bảo mật của bạn. Nó cung cấp cho bạn một phương pháp gửi email tương đối dễ dàng, hoàn toàn an toàn và cũng cho phép bạn xác minh danh tính của những người bạn đang giao tiếp. Vì các tiện ích bổ sung PGP cũng có sẵn cho hầu hết các ứng dụng email chính, nên hình thức mã hóa này thường dễ thực hiện.

Tất cả những điều này đã nói, việc bảo mật email chỉ là một khía cạnh của an ninh mạng. Bạn nên đảm bảo rằng, ngoài PGP, bạn cũng sử dụng platform bảo mật data mạnh và phần mềm Data Loss Prevention. Sử dụng càng nhiều công cụ càng tốt là cách tốt nhất để đảm bảo quyền riêng tư và bảo mật của bạn.

**4. Tổng kết báo cáo thực hành:**

Với đề tài tìm hiểu giao thức bảo mật PGP chúng em đã làm sang tỏ được

một số vấn đề như:

• Giới thiệu về giao thức bảo mật PGP, các thuật toán liên quan

• Quy trình thực hiện mã hóa và giải mã của PGP.

• Cài đặt ứng dụng minh họa kỹ thuật mã hóa PGP.

Tuy nhiên vẫn còn một số hạn chế đó là:

• Chưa trình bày được một cách cụ thể và rõ ràng hơn về giao thức PGP và các thuật toán liên quan.

• Chưa thực hành nghiên cứu sâu về ứng dụng của PGP.