MẬT MÃ ỨNG DỤNG TRONG AN TOÀN THÔNG TIN

(Applied Cryptography In Secure Information System)

ThS. Bùi Hữu Đông
buihuudong19@gmail.com
0903.82.36.46

Học viện Kỹ thuật Mật mã, khoa An toàn thông tin

Ngày 7 tháng 12 năm 2020

Buổi 08

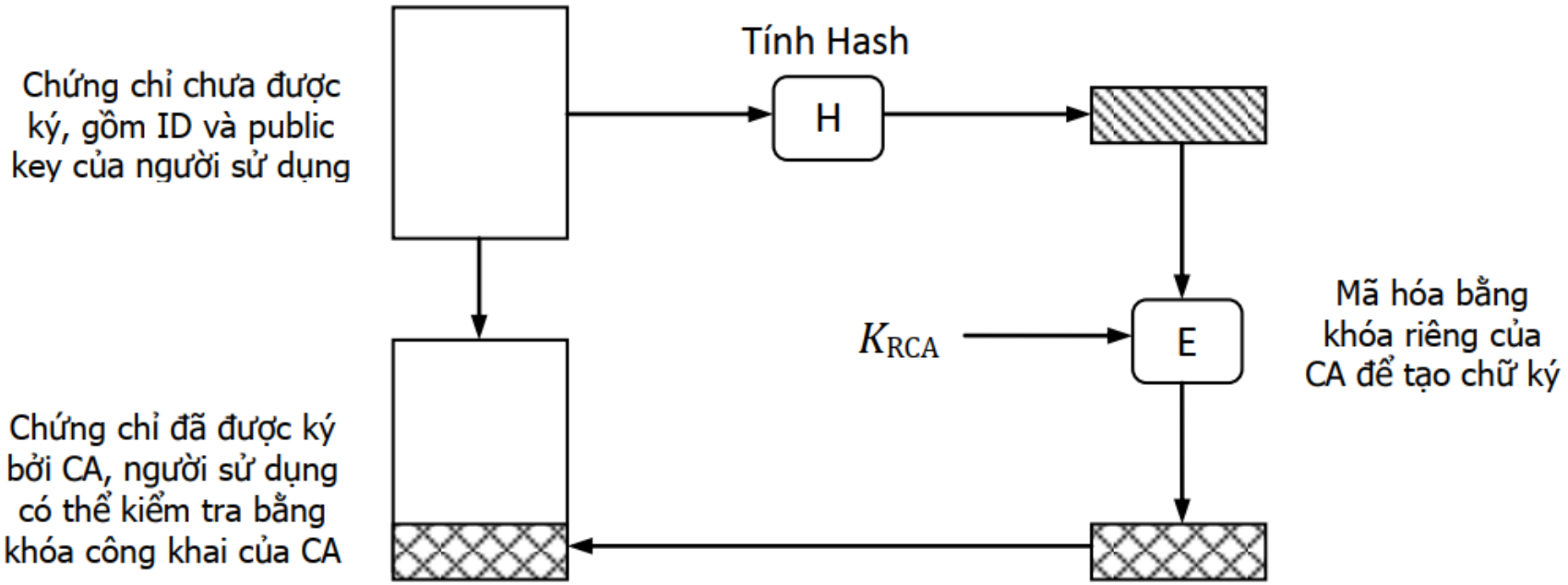
GIAO THỨC VÀ ỨNG DỤNG THỰC TIỄN TRONG ATTT

Giới thiệu

- Học phần này, ta sẽ tìm hiểu việc áp dụng mô hình ở các bài học trước và một số giao thức thực tế
- Trước hết là chuẩn chứng thực $X.509$ (chuẩn thực tiễn áp dụng trong vấn đề trao đổi khóa công khai)
- Tìm hiểu về giao thức bảo mật web Secure Socker Layer (SSL)
- Giao thức bảo mật mạng cục bộ Keberos

Chứng thực X.509

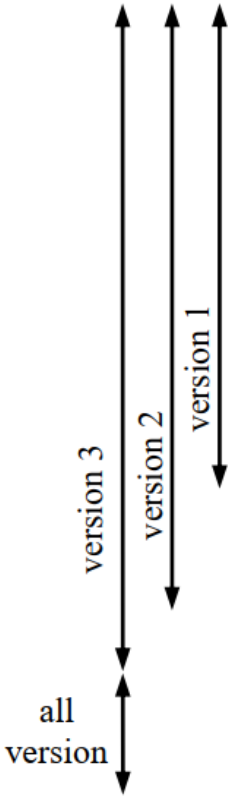
- *Cấu trúc chứng thực X.509*: là một áp dụng dựa trên lý thuyết về chữ ký điện tử. Sơ đồ nguyên tắc để sinh ra chứng thực X.509 như sau:



$$\text{Certificate} = \text{ID}||\text{K}_\text{U}||\text{E}(\text{H}(\text{ID}, \text{K}_\text{U}), \text{K}_\text{RCA})$$

Chứng thực X.509

- *Cấu trúc một chứng chỉ X.509 gồm có các thành phần sau:*



Version	Version 3
Serial Number	05:A0:4C
Certificate Signature Algorithm	PKCS #1 SHA-1 With RSA Encryption
Issuer Name	OU = Equifax Secure Certificate Authority; O = Equifax
Validity (Not Before, Not After)	04/01/2006 17:09:06 PM GMT - 04/01/2011 17:09:06 PM GMT
Subject	CN= login.yahoo.com; OU= Yahoo; O= Yahoo! Inc.
Subject Public Key Algorithm	PKCS #1 RSA Encryption
Subject Public Key	30 81 89 02 81 81 00 b5 6c 4f ee ef 1b 04 5d be…
Issuer Unique Identifie	
Subject Unique Identifier	
Extension for version 3	
Certificate Signature Algorithm	PKCS #1 SHA-1 With RSA Encryption
Certificate Signature Value	50 25 65 10 43 e1 74 83 2f 8f 9c 9e dc 74 64 4e…

Các thành phần của Chứng thực X.509

- Version: phiên bản X.509 của chứng chỉ này, có 3 phiên bản là 1, 2 và 3.
- Serial Number: số serial của chứng chỉ này do trung tâm chứng thực CA ban hành.
- Certificate Signature Algorithm: thuật toán ký chứng chỉ, gồm loại hàm Hash và phương pháp mã hóa khóa công khai.
- Issuer name: Tên của trung tâm chứng thực CA (CN: common name, O: organization, OU: organization unit).
- Validity: thời gian hiệu lực của chứng chỉ
- Subject: tên chủ sở hữu chứng chỉ, cũng gồm có CN, O, OU...
- Subject Public Key Algorithm: thuật toán mã hóa khóa công khai mà tương ứng với khóa công khai trong chứng chỉ.

Các thành phần của Chứng thực X.509

- Subject Public Key: khóa công khai trong chứng chỉ, tức khóa công khai của chủ sở hữu. Đối với RSA thì thuộc tính này lưu giữ giá trị Modulus và Exponent nối tiếp nhau (N và e)
- Issuer Unique Identifier, Subject Unique Identifier: dành cho version 2, ít được sử dụng
- Extension: dành cho version 3
- Certificate Signature Algorithm: thuật toán ký chứng chỉ, giống mục thứ 3.
- Certificate Signature Value: giá trị của chữ ký

Chứng chỉ thường được lưu trên một file có phần mở rộng là .cer

Các tổ chức cung cấp chứng thực X.509

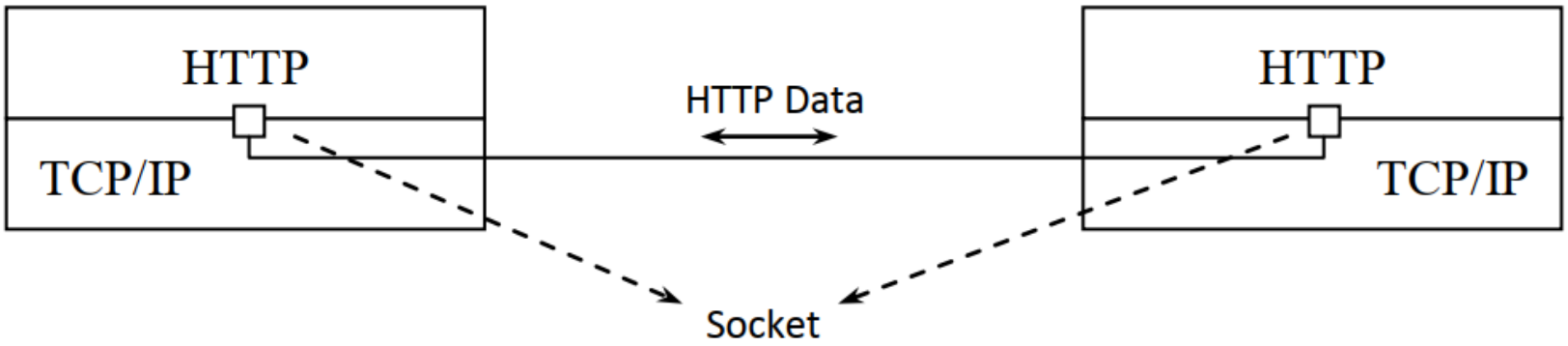
- Trên thế giới hiện nay có nhiều tổ chức cung cấp chứng thực X509 như VeriSign, Equifax, Thawte, SecureNet... VeriSign hiện là tổ chức lớn nhất
- Verisign cung cấp chứng chỉ X509 theo ba mức độ (class):
  - Class 1: ID của một đối tượng là email của đối tượng đó. Sau khi đối tượng đăng ký email và public key qua mạng Internet, Verisign gửi email để kiểm tra địa chỉ email hợp lệ và cấp chứng thực
  - Class 2: ID là địa chỉ nơi ở của đối tượng, Verisign sẽ gửi confirm qua đường bưu điện để kiểm tra địa chỉ hợp lệ.
  - Class 3: đối tượng cần có giấy tờ pháp lý để chứng minh tư cách pháp nhân.

Các định dạng file phổ biến của chứng chỉ X.509

- Dạng DER (.cer): nội dung của chứng chỉ X.509 được lưu dưới format DER, một định dạng dữ liệu binary chuẩn cho các môi trường máy tính
- Dạng PEM (.pem): là dạng DER và được mã hóa dưới dạng text theo chuẩn Base64. Một file text PEM bắt đầu bằng dòng $-----BEGIN CERTIFICATE-----$ và kết thúc bằng dòng $-----END CERTIFICATE-----$
- Dạng PKCS#7 (.p7c hay .p7b): là một định dạng dữ liệu được mã hóa hay ký. Do đó có đi kèm cả chứng chỉ
- Dạng PKCS#10 (.p10 hay .p10): là một định dạng dùng để gửi yêu cầu cấp chứng chỉ X509 đến trung tâm chứng thực. Định dạng này có ID và public key của người yêu cầu.

Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

- Dữ liệu Web được trao đổi giữa trình duyệt và web server được thực hiện qua giao thức HTTP. Client kết nối với server qua socket của giao thức TCP/IP.



Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

- Giao thức HTTP khi thực hiện tìm kiếm từ "Nha Trang" trong website vn.search.yahoo.com:

```
GET /search?p=Nha+Trang&fcss=on&fr=yfp-t-101&toggle=1&cop=&ei=UTF-8 HTTP/1.1
Host: vn.search.yahoo.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.13) Gecko/2009073022
Firefox/3.0.13 (.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Referer: http://vn.yahoo.com/?p=us
```

Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

- Mô hình dữ liệu phản hồi của server yahoo. Dữ liệu này gồm hai phần, phần đầu theo quy định của giao thức HTTP, phần sau là dữ liệu HTML:

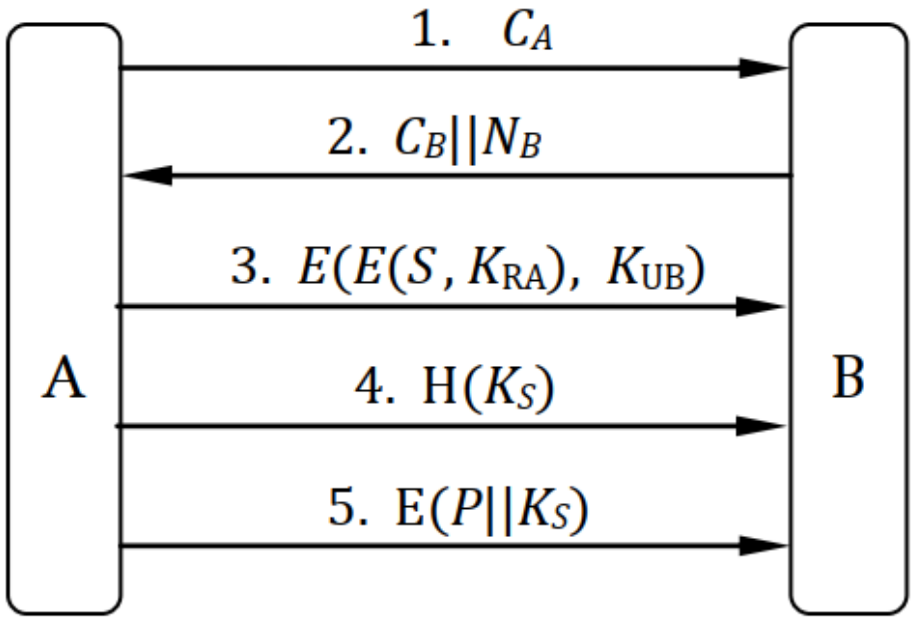
```
HTTP/1.1 200 OK
Date: Fri, 14 Aug 2009 10:25:49 GMT
Cache-Control: private
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: Keep-Alive
Content-Encoding: gzip

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html lang="vi"><head> ... </head>
....
</html>
```

Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

- SSL bảo mật dữ liệu trao đổi qua socket. nên có tên gọi là Secure Socket Layer (URL bắt đầu bằng https://). Đây là giao thức bảo mật kết hợp mã hóa khóa công khai và khóa đối xứng như đã trình bày trong phần trước trong đó mã hóa RSA được dùng để trao đổi khóa phiên của mã hóa đối xứng.

Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

- Xem lại mô hình trao đổi khóa phiên

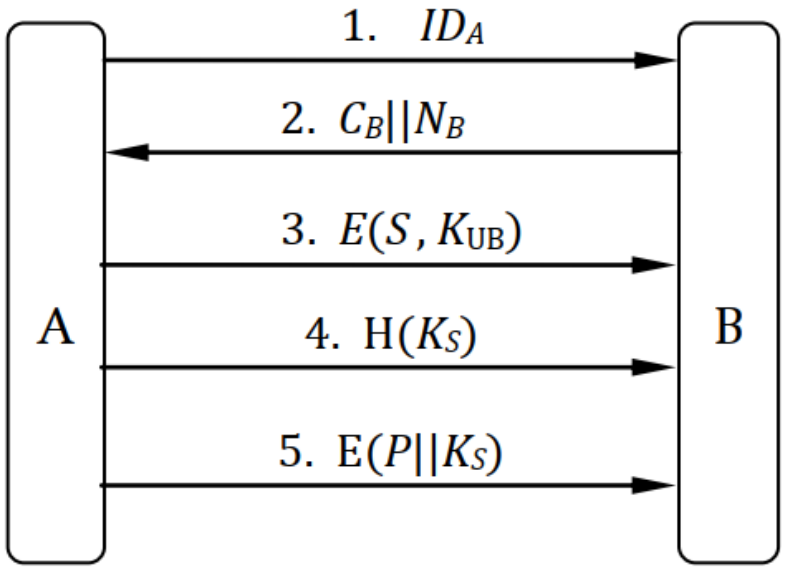


Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

- Mô hình này yêu cầu mỗi người duyệt web (A) và mỗi website (B) đều phải có cặp khóa riêng và khóa công khai. Hay nói cách khác website và người duyệt phải có chứng thực => *gây khó khăn cho người duyệt web vì phải có chứng chỉ*
- Tuy nhiên trong thực tế không phải lúc nào cũng cần chứng thực từ phía người sử dụng.
- Tí dụ, khi bạn mua hàng tại cửa hàng sách Amazon. Amazon không cần biết bạn là ai, chỉ cần bạn có tài khoản để mua hàng (việc bảo mật tài khoản người mua là trách nhiệm của mã hóa đối xứng) => *Vì vậy trong trường hợp này, người duyệt không cần có chứng chỉ*

Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

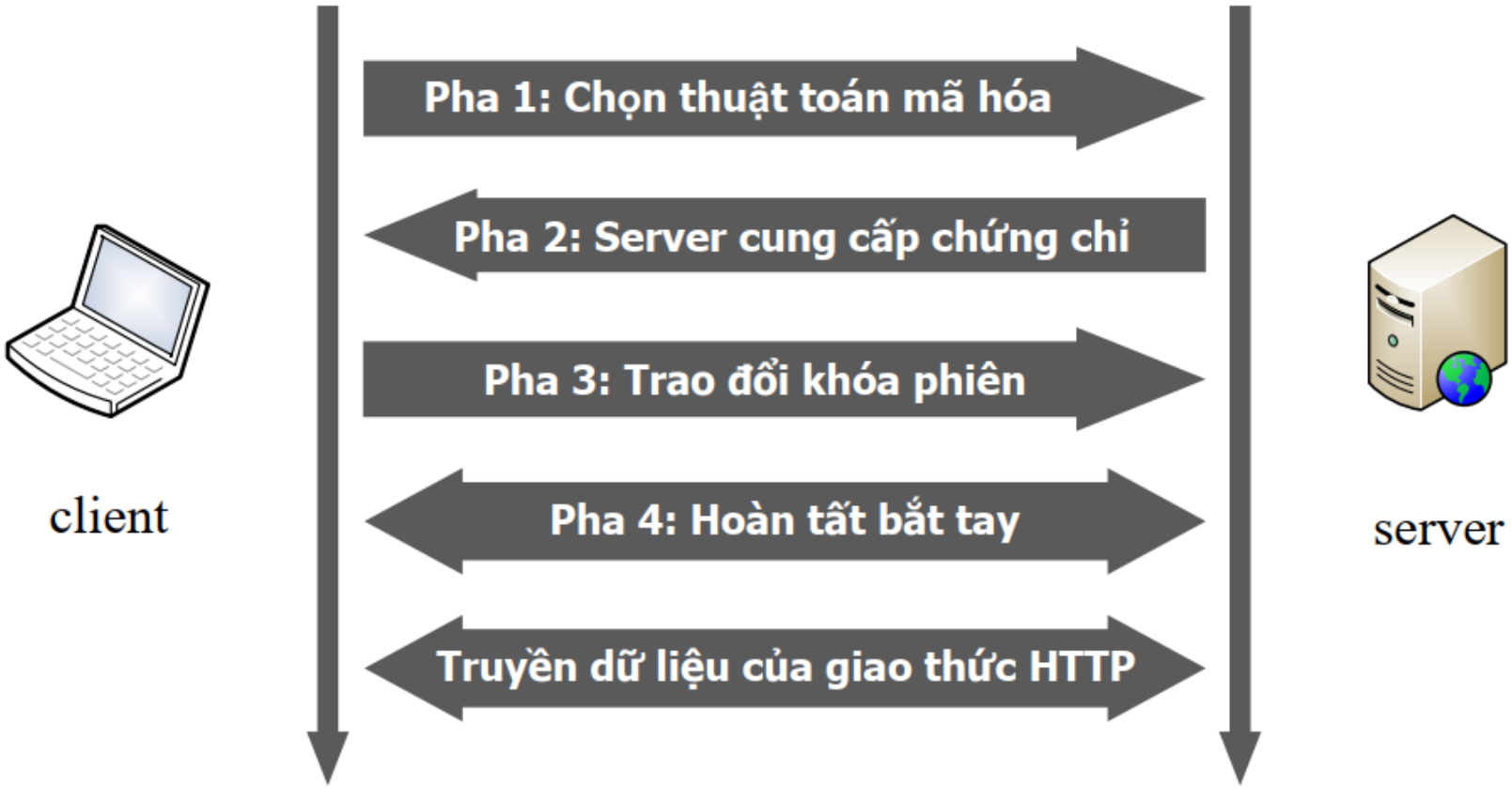
- *Mô hình trao đổi khóa phiên cần chứng thực 1 phía:*



- Mô hình trên đảm bảo ngoài người duyệt A chỉ có website B là biết được khóa phiên K_S , còn A là ai thì website không cần biết

Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

- Các phương pháp mã hóa mà SSL sử dụng: RC4, RC2, DES, 3DES, IDEA, AES



Giao thức bảo mật web Secure Socket Layer - SSL (version 3)

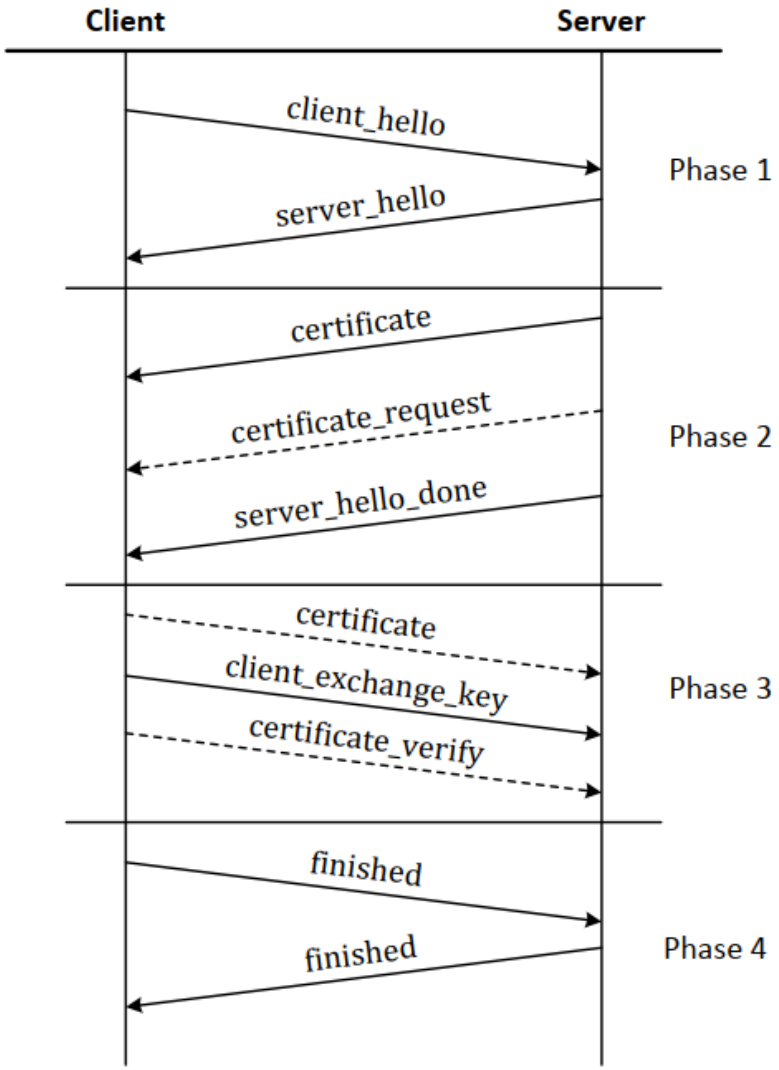
SSL gồm có hai phần cơ bản là:

- Giao thức bắt tay
- Giao thức truyền dữ liệu

Giao thức bắt tay - SSL Handshaking Protocol

Trước khi tiến hành truyền số liệu, SSL thực hiện giao thức bắt tay để chứng thực website và chứng thực người duyệt web, trao đổi khóa phiên và thống nhất các thuật toán mã hóa được sử dụng.

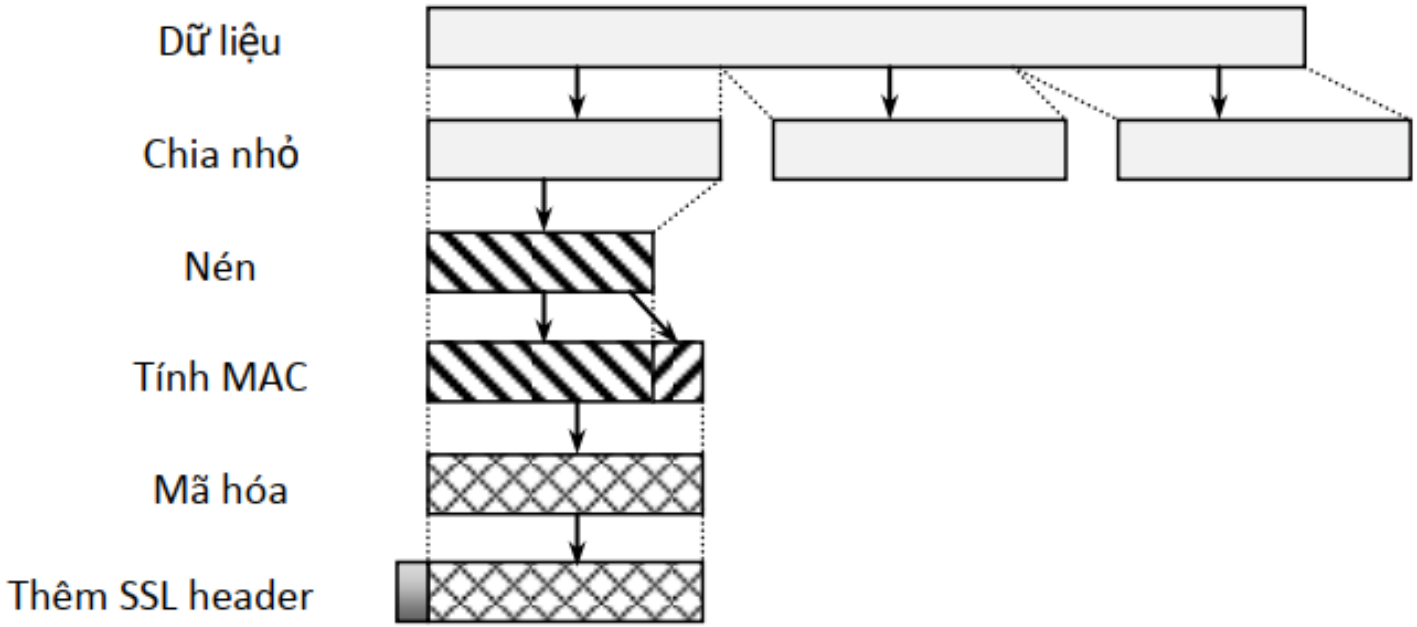
Mô hình giao thức bắt tay



(đường nét đứt là các thông điệp không bắt buộc, chỉ sử dụng khi cần chứng thực từ phía client)

Giao thức truyền số liệu - SSL Record Protocol

- Hình minh họa các bước thực hiện trong quá trình truyền số liệu:



Giao thức truyền số liệu - SSL Record Protocol

- Trong giao thức truyền số liệu, dữ liệu được chia thành các khối có kích thước là 2^{14}. Sau đó, dữ liệu này được nén lại
- Bước tiếp theo giá trị MAC của khối dữ liệu nén được tính theo công thức sau:

$$hash(MAC\_key||pad\_2||hash(MAC\_key||pad\_1||seq\_num||type||length||data))$$

Giao thức truyền số liệu - SSL Record Protocol

Trong đó:

- Hàm hash là hàm MD5 hay SHA-1
- MAC_key: khóa tính MAC đã được client và server thống nhất trong phần bắt tay
- pad_1: byte 0x36 (00110110) được lặp lại 48 lần (384 bít) đối với hàm hash MD5 và 40 lần (320 bít) đối với hàm hash SHA-1
- pad_2: byte 0x5C (10101100) được lặp lại 48 lần đối với MD5 và 40 lần với SHA-1
- seq_num: số thứ tự của khối dữ liệu
- type: loại khối dữ liệu
- length: kích thước khối dữ liệu
- data: khối dữ liệu

Bài tập

Sinh viên thực hiện tìm hiểu về:

- Giao thức bảo mật mạng cục bộ Keberos
- Giao thức truyền tệp có bảo mật (FTPS)
- Giao thức vỏ sò bảo mật (SSH)