**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN 1**

****

**BÁO CÁO BÀI TẬP LỚN**

**MÔN: AN TOÀN BẢO MẬT HỆ THỐNG THÔNG TIN**

**Đề tài: Tìm hiểu về giao thức bảo mật SSL/TLS**

**Giảng viên: Quản Trọng Thế**

|  |  |
| --- | --- |
| Sinh viên thực hiện: | Vũ Trọng Khôi |
| Mã sinh viên: | B22DCCN468 |
| Nhóm lớp: | 11 |

***Hà Nội – 2025***

MụC LỤC

[Danh mục từ viết tắt 2](#_Toc197295253)

[Danh mục hình ảnh 3](#_Toc197295254)

[LỜI NÓI ĐẦU 4](#_Toc197295255)

[I. Tổng quan về giao thức SSL/TLS 5](#_Toc197295256)

[1. Khái niệm 5](#_Toc197295257)

[2. Lịch sử phát triển 5](#_Toc197295258)

[3. Vị trí của SSL/TLS trong mô hình bảo mật Internet 6](#_Toc197295259)

[4. Lợi ích khi sử dụng giao thức SSL/TLS 6](#_Toc197295260)

[II. Kiến trúc giao thức SS L/TLS 7](#_Toc197295261)

[1. Handshake Protocol 7](#_Toc197295262)

[2. Change Cipher Spec Protocol 8](#_Toc197295263)

[3. Alert Protocol 9](#_Toc197295264)

[4. Record Protocol 11](#_Toc197295265)

[5. Kết luận 12](#_Toc197295266)

[III. Cách hoạt động của giao thức SSL/TLS 12](#_Toc197295267)

[1. Thiết lập kết nối. 12](#_Toc197295268)

[2. Trao đổi dữ liệu 13](#_Toc197295269)

[3. Đóng kết nối 14](#_Toc197295270)

[4. Kết luận 14](#_Toc197295271)

[IV. Ứng dụng của giao thức SSL/TLS 14](#_Toc197295272)

[1. Bảo mật truy cập Web - HTTPS 14](#_Toc197295273)

[2. Bảo mật Email 15](#_Toc197295274)

[5. Điện toán đám mây 16](#_Toc197295275)

[6. Kết luận 16](#_Toc197295276)

[BÀI THỰC HÀNH 16](#_Toc197295277)

[TÀI LIỆU THAM KHẢO 19](#_Toc197295278)

## Danh mục từ viết tắt

|  |  |
| --- | --- |
| Từ viết tắt | Từ gốc |
| SSL | Secure Sockets Layer |
| TLS | Transport Layer Security |
| IETF | Internet Engineering Task Force |
| OSI | Open Systems Interconnection |
| TCP | Transmission Control Protocol |
| IP | Internet Protocol |
| HTTP | Hypertext Transfer Protocol |
| SMTP | Simple Mail Transfer Protocol |
| IMAP | Internet Message Access Protocol |
| POP3 | Post Office Protocol version 3 |
| VPN | Virtual Private Network |
| MAC | Message Authentication Code |
| AES | Advanced Encryption Standard |
| DES | Data Encryption Standard |
| RSA | Rivest–Shamir–Adleman |
| DH | Diffie–Hellman |
| API | Application Programming Interface |
| IoT | Internet of Things |

## Danh mục hình ảnh

Hình 1. Lịch sử phát triển của SSL/TLS 4

Hình 2. Mô hình OSI và TCP/IP 5

Hình 3. Kiến trúc giao thức SSL/TLS 6

Hình 4. Change Cipher Spec Protocol gửi thông điệp giữa hai máy 7

Hình 5. Thông điệp của Alert Protocol. 9

Hình 6. Record Protocol 10

Hình 7. Hanshake Protocol 11

Hình 8. Biểu tượng ổ khóa xanh trên thanh địa chỉ trình duyệt 14

## LỜI NÓI ĐẦU

Trong thời đại số hóa mạnh mẽ, các hoạt động trao đổi thông tin qua mạng Internet ngày càng phổ biến và đóng vai trò thiết yếu trong mọi lĩnh vực. Tuy nhiên, đi kèm với sự tiện lợi là nguy cơ mất an toàn thông tin như rò rỉ dữ liệu, tấn công mạng, đánh cắp danh tính hoặc giả mạo thông tin. Do đó, các biện pháp bảo mật trong truyền thông mạng, đặc biệt là bảo mật ở tầng giao vận, trở nên vô cùng quan trọng. Giao thức SSL/TLS là một trong những cơ chế bảo mật phổ biến nhất hiện nay, giúp đảm bảo tính bí mật, toàn vẹn và xác thực dữ liệu trong quá trình truyền tải. SSL/TLS được triển khai rộng rãi trong các trình duyệt web (HTTPS), dịch vụ email, VPN và nhiều hệ thống ứng dụng khác.

Trong bối cảnh các cuộc tấn công mạng ngày càng tinh vi và phức tạp, việc hiểu rõ cơ chế hoạt động, kiến trúc và ứng dụng của giao thức SSL/TLS là điều cần thiết đối với sinh viên ngành công nghệ thông tin, cũng như những người làm việc trong lĩnh vực bảo mật và mạng máy tính. Nắm vững giao thức này không chỉ giúp bảo vệ hệ thống thông tin một cách hiệu quả mà còn là nền tảng để tiếp cận các kỹ thuật bảo mật hiện đại hơn.

Báo cáo là kết quả của quá trình tìm hiểu về giao thức SSL/TLS và được chia thành bốn mục cụ thể như sau:

I. Tổng quan về giao thức SSL/TLS

II. Kiến trúc giao thức SSL/TLS

III. Cách hoạt động giao thức SSL/TLS

IV. Ứng dụng của giao thức SSL/TLS

Hy vọng báo cáo sẽ giúp người đọc có cái nhìn rõ ràng và toàn diện hơn về vai trò, cơ chế hoạt động cũng như tầm quan trọng của giao thức SSL/TLS trong hệ thống mạng hiện đại.

## I. Tổng quan về giao thức SSL/TLS

### 1. Khái niệm

SSL (Secure Sockets Layer) và TLS (Transport Layer Security) là hai giao thức mã hóa được thiết kế để đảm bảo tính bảo mật cho dữ liệu truyền qua mạng Internet. Chúng hoạt động bằng cách thiết lập một kết nối an toàn giữa hai thiết bị, thường là giữa trình duyệt và máy chủ, nhằm bảo vệ dữ liệu khỏi các nguy cơ như nghe lén, giả mạo và thay đổi dữ liệu trong quá trình truyền tải.

TLS là phiên bản nâng cấp và an toàn hơn của SSL. Mặc dù hai thuật ngữ này thường được sử dụng thay thế cho nhau, nhưng hiện nay TLS là giao thức được khuyến nghị sử dụng do SSL đã không còn được phát triển và duy trì nữa. TLS đã trở thành tiêu chuẩn bảo mật không thể thiếu cho các website và ứng dụng trực tuyến.

### 2. Lịch sử phát triển

A diagram of a timeline

AI-generated content may be incorrect.

Hình 1. Lịch sử phát triển của SSL/TLS

SSL 1.0: Được phát triển bởi Netscape vào đầu những năm 1990 và được giới thiệu lần đầu tiên vào năm 1994.

SSL 2.0: Phát hành vào năm 1995, là phiên bản đầu tiên được công bố rộng rãi nhưng nhanh chóng bị thay thế do các vấn đề bảo mật.

SSL 3.0: Ra mắt vào năm 1996 với nhiều cải tiến về bảo mật, tuy nhiên vẫn tồn tại các lỗ hổng nghiêm trọng như Heartbleed và POODLE.

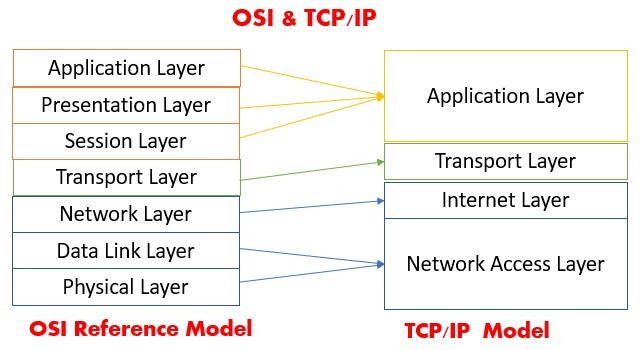
TLS 1.0: Được IETF giới thiệu vào năm 1999 như một bản nâng cấp của SSL 3.0, với mục tiêu khắc phục các lỗ hổng bảo mật và cải thiện hiệu suất.

TLS 1.1 và 1.2: Phát hành lần lượt vào các năm 2006 và 2008, bổ sung các thuật toán mã hóa mạnh hơn và cải thiện khả năng bảo mật.

TLS 1.3: Ra mắt vào năm 2018, loại bỏ nhiều tính năng không an toàn và cải thiện hiệu suất kết nối .

### 3. Vị trí của SSL/TLS trong mô hình bảo mật Internet

Để hiểu rõ vai trò và vị trí của SSL/TLS trong hệ thống mạng, cần đặt giao thức này vào bối cảnh của hai mô hình tham chiếu phổ biến: mô hình và mô hình TCP/IP.



Hình 2. Mô hình OSI và TCP/IP

Mô hình OSI gồm 7 lớp, trong đó SSL/TLS hoạt động chủ yếu ở lớp thứ 6 - Lớp trình bày (Presentation Layer) và một phần ở lớp 5 - Lớp phiên (Session Layer). Vai trò chính ở lớp trình bày là đảm bảo rằng dữ liệu truyền giữa các hệ thống được định dạng và mã hóa theo cách thống nhất. Tuy nhiên, trên thực tế, SSL/TLS thường được triển khai ngay trên lớp giao vận vì nó hoạt động trực tiếp bên trên TCP. Do đó, SSL/TLS đôi khi còn được xem như một lớp trung gian giữa lớp trình bày và giao vận.

Trong mô hình TCP/IP 4 lớp, SSL/TLS nằm giữa lớp giao vận (Transport) và lớp ứng dụng (Application). Giao thức này không thay thế TCP, mà hoạt động song song và bổ sung tính năng bảo mật mà TCP không có. Ở cả hai mô hình SSL/TLS cung cấp các dịch vụ bảo mật như mã hóa dữ liệu, xác thực danh tính và đảm bảo tính toàn vẹn của dữ liệu

Trong thực tế, SSL/TLS được sử dụng để bảo vệ các giao thức ứng dụng như HTTP , SMTP, IMAP và POP3, giúp đảm bảo rằng dữ liệu truyền qua mạng không bị truy cập trái phép hoặc thay đổi bởi các bên thứ ba. Do đó, SSL/TLS không chỉ là giao thức phụ trợ mà còn là nền tảng quan trọng trong hạ tầng bảo mật Internet hiện đại.

### 4. Lợi ích khi sử dụng giao thức SSL/TLS

Giao thức SSL/TLS mang lại nhiều lợi ích quan trọng trong việc bảo mật thông tin khi truyền tải dữ liệu qua mạng:

* Bảo mật dữ liệu: Mã hóa toàn bộ dữ liệu trao đổi giữa hai bên, giúp ngăn chặn nghe lén, đánh cắp thông tin.
* Đảm bảo toàn vẹn: Phát hiện nếu dữ liệu bị thay đổi trong quá trình truyền, ngăn chặn các cuộc tấn công giả mạo nội dung.
* Xác thực danh tính: Giúp người dùng xác minh máy chủ họ đang kết nối có phải là chính chủ.
* Ngăn chặn tấn công mạng: SSL/TLS chống lại các hình thức tấn công như Man-in-the-Middle, packet sniffing hoặc DNS spoofing.

Nhờ những lợi ích trên, SSL/TLS trở thành tiêu chuẩn bảo mật không thể thiếu trong mọi hệ thống Internet hiện đại, từ trang web cá nhân đến các nền tảng ngân hàng, thương mại điện tử và dịch vụ doanh nghiệp.

## II. Kiến trúc giao thức SS L/TLS

Kiến trúc của SSL/TLS được xây dựng dựa trên mô hình phân lớp và bao gồm bốn giao thức con chính, phối hợp với nhau để thiết lập và duy trì một kết nối an toàn: Record Protocol, Handshake Protocol, Alert Protocol và Change Cipher Spec Protocol.

A diagram of a computer application

AI-generated content may be incorrect.

Hình 3. Kiến trúc giao thức SSL/TLS

Kiến trúc này cho phép SSL/TLS vận hành linh hoạt, tương thích với nhiều ứng dụng và có thể mở rộng hoặc nâng cấp mà không làm ảnh hưởng đến các tầng khác.

### 1. Handshake Protocol

Đây là giao thức phức tạp và quan trọng nhất trong việc thiết lập một kết nối SSL an toàn. Giao thức này diễn ra trong giai đoạn đầu của phiên làm việc, cho phép máy khách và máy chủ thực hiện các chức năng:

* Thỏa thuận giao thức, bộ mật mã: máy khách và máy chủ trao đổi thông tin về phiên bản SSL/TLS mà chúng hỗ trợ và các bộ mật mã (tập hợp thuật toán mã hóa, trao đổi khóa, hàm băm). Hai bên sẽ thống nhất một phiên bản giao thức và một bộ mật mã duy nhất để sử dụng cho phiên kết nối.
* Trao đổi khóa: Giao thức cho phép máy khách và máy chủ trao đổi thông tin cần thiết để thiết lập các khóa bao gồm khóa mã hóa phiên và khóa MAC sẽ được sử dụng bởi Record Protocol. Phương pháp trao đổi khóa phụ thuộc vào thuật toán trao đổi khóa đã được thống nhất trong giai đoạn trước đó (Vd: RSA, DH,...)
* Xác thực Server: Mục đích để máy khách có thể tin tưởng rằng đang giao tiếp với đúng máy chủ mong muốn.
* Xác thực Client: Trong một số ứng dụng (Vd: truy cập tài nguyên nhạy cảm, ứng dụng doanh nghiệp,...) máy chủ có thể yêu cầu máy khách xác thực danh tính. Điều này làm tăng mức độ bảo mật bằng cách xác minh cả hai phía. Nếu xác thực không thành công, máy chủ có thể từ chối kết nối hoặc hạn chế quyền truy cập.

Tóm lại, Handshake Protocol là quy trình phức tạp nhưng thiết yếu, là nền tảng cho mọi giao tiếp an toàn tiếp theo trong phiên SSL/TLS. Cụ thể các bước hoạt động chi tiết sẽ được trình bày tại phần mục 1 của phần III.

### 2. Change Cipher Spec Protocol

Change Cipher Spec Protocol là giao thức đơn giản nhất trong bốn giao thức đặc trưng của SSL. Giao thức này chỉ bao gồm một thông điệp một byte duy nhất có giá trị là 1, được gửi từ máy chủ đến máy khách hoặc ngược lại.

Cấu trúc một thông điệp Change Cipher Spec:

struct {

enum { change\_cipher\_spec(1), (255) } type;

} ChangeCipherSpec;

Thông điệp này không phải một dữ liệu thực tế mà chỉ mang tính thông báo cho bên còn lại rằng bộ mật mã đã được thiết lập và các bản ghi tiếp theo sẽ được bảo vệ bằng bộ mật mã, khóa đã được thống nhất trong giai đoạn Handshake Protocol.

A diagram of a computer server

AI-generated content may be incorrect.

Hình 4. Change Cipher Spec Protocol gửi thông điệp giữa hai máy

Thông điệp của Change Cipher Spec Protocol cũng đánh dấu sự kết thúc thành công quá trình thỏa thuận, xác thực giữa máy khách và máy chủ, chuyển sang chế độ giao tiếp được mã hóa.

### 3. Alert Protocol

Alert Protocol có vai trò thông báo về các sự kiện bất thường, lỗi hoặc cảnh báo liên quan đến phiên kết nối SSL/TLS. Mục đích chính của giao thức này là đảm bảo cả hai bên đều có thể nhận biết và xử lý các vấn đề xảy ra một cách thích hợp, duy trình tính ổn định và bảo mật của kết nối.

Cấu trúc thông điệp của Alert Protocol:

enum { warning(1), fatal(2), (255) } AlertLevel;

enum {

close\_notify(0),

unexpected\_message(10),

bad\_record\_mac(20),

...

bad\_certificate(42),

unsupported\_certificate(43),

certificate\_revoked(44),

certificate\_expired(45),

access\_denied(49),

decode\_error(50),

...

protocol\_version(70),

bad\_certificate\_status\_response(113),

unknown\_psk\_identity(115),

certificate\_required(116),

no\_application\_protocol(120),

(255)

} AlertDescription;

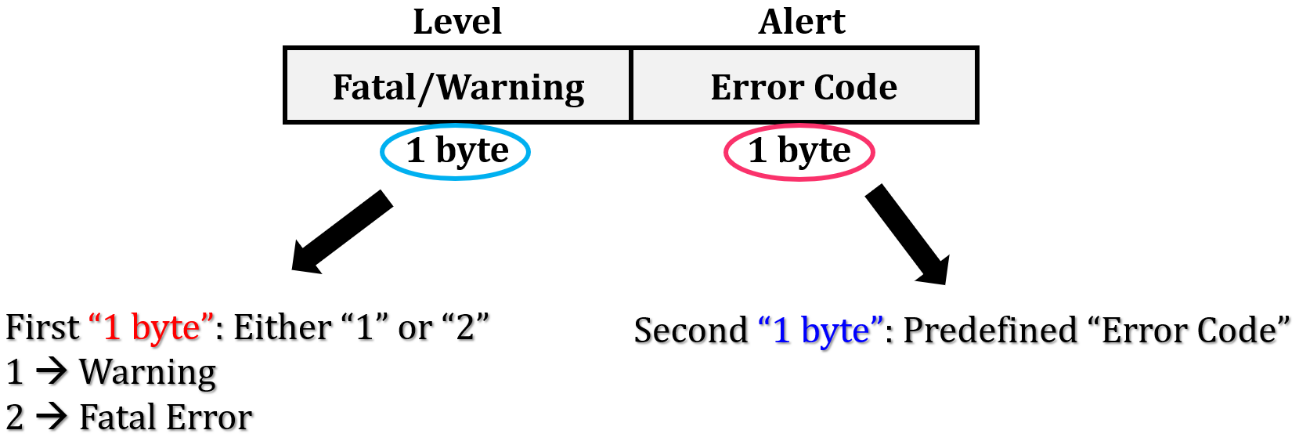
struct {

AlertLevel level;

AlertDescription description;

} Alert;

Một thông điệp cảnh báo trong Alert Protocol bao gồm hai Byte: Byte mức độ cảnh báo và Byte mô tả cảnh báo (Hình 5).



Hình 5. Thông điệp của Alert Protocol.

Byte đầu tiên của thông điệp là Byte Mức độ (Level), chỉ định mức độ nghiêm trọng của cảnh báo:

* ***warning*** (1): cho biết một cảnh báo thông thường. Thường thì kết nối vẫn có thể tiếp tục sau khi cảnh báo được gửi và nhận từ 2 bên,
* ***fatal*** (2): cho biết một lỗi nghiêm trọng. Khi một cảnh báo ở mức độ fatal được gửi, bên gửi sẽ ngay lập tức đóng kết nối SSL/TLS. Sau khi nhận thông báo thì bên nhận cũng sẽ đóng kết nối.

Byte thứ hai là Byte Mô tả cảnh báo (Alert), chứa một mã số (trong khoảng từ 0 - 255) chỉ định một loại cảnh báo cụ thể. Các mã số này đã được chuẩn hóa và định nghĩa trong các đặc tả của giao thức SSL/TLS. Một vài loại mô tả cảnh báo phổ biến có thể kể đến như:

* ***close\_notify*** (0): Báo hiệu việc đóng kết nối một cách an toàn.
* ***unexpected\_message*** (10): Nhận được một thông điệp không phù hợp như là “Thông điệp handshake sai” hay “Dữ liệu ứng dụng quá sớm”.
* ***bad\_record\_mad*** (20): Nhận được một bản ghi có mã xác thực tin nhắn (MAC) không chính xác.
* ***bad\_certificate*** (42): Chứng chỉ bị hỏng, chứa thông tin không hợp lệ.
* ***unsupported\_certificate*** (43): Loại chứng chỉ nhận được không được hỗ trợ.
* ***certificate\_revoked*** (44): Chứng chỉ nhận được đã bị thu hồi.
* ***certificate\_expired*** (45): Chứng chỉ nhận được đã hết hạn.
* ***access\_denied*** (49): Quyền truy cập bị từ chối dựa trên chứng chỉ của máy khách.
* ***decode\_error*** (50): Lỗi trong quá trình giải mã một thông điệp.
* **protocol\_version** (70): Máy khách và máy chủ không thể thống nhất phiên bản giao thức chung.

### 4. Record Protocol

Record Protocol hoạt động ở tầng thấp nhất của kiến trúc SSL, chịu trách nhiệm trực tiếp trong việc xử lý dữ liệu trước khi nó được truyền đi qua mạng. Giao thức này nhận dữ liệu từ các giao thức con lớp cao hơn và xử lý công việc phân đoạn, nén, xác thực và mã hóa dữ liệu (Hình 6).

A diagram of a diagram of a data flow

AI-generated content may be incorrect.

Hình 6. Record Protocol

**Phân mảnh (Fragmentation):** Dữ liệu từ tầng ứng dụng có thể có kích thước lớn, gây khó khăn cho việc truyền tải dữ liệu hiệu quả. Record Protocol chia dữ liệu thành các khối nhỏ hơn, gọi là bản ghi, với kích thước tối đa thường là 16KB. Việc phân mảnh như vậy sẽ giúp việc quản lý và truyền tải dữ liệu linh hoạt hơn, đặc biệt khi đối mặt với các giới hạn về kích thước gói tin trên đường truyền.

**Nén (Compression):** Sau khi phân mảnh, Record Protocol có thể áp dụng các thuật toán nén để giảm kích thước dữ liệu nhằm tối ưu hóa băng thông. Nếu nén được sử dụng, thông tin về thuật toán nén sẽ được chỉ định trong tiêu đề bản ghi.

**Tính giá trị MAC - Mã xác thực thông điệp:** Để đảm bảo tính toàn vẹn của dữ liệu và phát hiện các hành vi giả mạo trong quá trình truyền tải, Record Protocol tính toán một giá trị MAC cho mỗi bản ghi. MAC được tạo ra dựa trên nội dung của bản ghi và một khóa bí mật được chia sẻ. Giá trị MAC này được gắn kèm vào bản ghi trước khi mã hóa. Ở phía nhận, MAC sẽ được tính toán lại và so sánh với giá trị nhận được để xác minh tính toàn vẹn.

**Mã hóa dữ liệu (Encryption):** Để bảo vệ tính bảo mật của dữ liệu, giao thức sẽ mã hóa nội dung bản ghi bằng một thuật toán mã hóa đối xứng (Vd: AES, DES,...) sử dụng một khóa mã hóa chung giữa hai bên. Thuật toán mã hóa và khóa mã hóa được thỏa thuận trong Handshake Protocol. Việc mã hóa đảm bảo rằng chỉ máy khách và máy chủ có khóa giải mã phù hợp mới có thể đọc được nội dung của dữ liệu.

Record Protocol đóng vai trò như một lớp bảo vệ dữ liệu toàn diện, đảm bảo rằng dữ liệu ứng dụng được truyền tải một cách an toàn, toàn vẹn và hiệu quả giữa máy khách và máy chủ sau khi một kết nối an toàn đã được thiết lập thông qua Handshake Protocol.

### 5. Kết luận

Như vậy, phần II đã cung cấp một cái nhìn tổng quan về kiến trúc của giao thức SSL/TLS, làm nổi bật sự phối hợp nhịp nhàng giữa các giao thức con để đảm bảo một phiên giao tiếp an toàn. Sự phân lớp rõ ràng và chức năng riêng biệt của từng giao thức con cho phép SSL/TLS cung cấp một cơ chế bảo mật mạnh mẽ và linh hoạt cho vô số ứng dụng trực tuyến. Kiến thức về kiến trúc giao thức SSL/TLS sẽ giúp tiếp cận các phần sau một cách dễ dàng và đa chiều hơn.

## III. Cách hoạt động của giao thức SSL/TLS

Quá trình hoạt động của giao thức SSL/TLS có thể được chia thành ba giai đoạn: Thiết lập kết nối, Trao đổi dữ liệu, Đóng kết nối.

### 1. Thiết lập kết nối.

Giai đoạn này chính là nhiệm vụ của Handshake Protocol. Một chuỗi các thông điệp sẽ được trao đổi giữa máy khách và máy chủ để thông nhất các tham số bảo mật, bộ mật mã cho phiên làm việc.

A diagram of a server

AI-generated content may be incorrect.

Hình 7. Hanshake Protocol

Quy trình này có thể được mô tả qua các bước (HÌnh 7) như sau:

* **Bước 1 -** Client Hello:Máy khách khởi tạo kết nối bằng cách gửi thông điệp “client hello” đến máy chủ. Thông điệp này chứa các thông tin quan trọng như: Phiên bản SSL/TLS, danh sách các bộ mật mã (thuật toán mã hóa, băm), thuật toán nén dữ liệu mà máy khách hỗ trợ.
* **Bước 2 -** Sever Hello:Máy chủ phản hồi bằng thông điệp “sever hello”. Thông điệp bao gồm phiên bản SSL/TLS, bộ mật mã, phương pháp nén phù hợp mà máy chủ nhận được từ thông điệp “client hello” cùng với chứng chỉ của máy chủ để xác thực danh tính. Máy chủ cũng có thể yêu cầu máy khách gửi lại chứng chỉ nếu được thiết lập xác thực.
* **Bước 3 -** Xác minh chứng chỉ máy chủ:Máy khách kiểm tra tính hợp lệ của chứng chỉ số của máy chủ đồng thời kiểm tra bộ mật mã và các tham số mật mã khác mà máy chủ chọn có phù hợp hay không.
* **Bước 4 -** Trao đổi khóa:Máy khách gửi thông điệp “client key exchange” chứa thông tin khóa bí mật (pre-master secret) đã được mã hóa bằng khóa công khai lấy từ chứng chỉ của máy chủ với thuật toán mã hóa đã được thỏa thuận trước đó.
* **Bước 5 -** Gửi chứng chỉ máy khách: Nếu máy chủ yêu cầu ở bước 2 thì máy khách sẽ gửi chứng chỉ của mình trong thông điệp này để máy chủ xác thực danh tính.
* **Bước 6 -** Xác minh chứng chỉ máy khách:Máy chủ xác minh tính hợp lệ của chứng chỉ máy khách.
* **Bước 7 -** Hoàn tất từ máy khách: Máy khách gửi thông điệp “finished” đến máy chủ. Thông điệp được mã hóa bằng khóa phiên (session key - đã được tạo từ pre-master secret) và chứa một hàm băm của tất cả các thông điệp trao đổi trước đó. Đây là bước xác nhận handshake từ phía máy khách thành công và không bị can thiệp.
* **Bước 8 -** Hoàn tất từ máy chủ: Máy chủ gửi thông điệp “finished” đến máy khách. Tương tự thông điệp của máy khách, thông điệp được mã hóa và chứa hàm băm của các thông điệp, xác nhận phía máy chủ hoàn tất thành công
* **Bước 9:** Hoàn tất giao thức Handshake, chuyển sang giai đoạn tiếp theo.

### 2. Trao đổi dữ liệu

Sau khi Handshake Protocol kết thúc, kết nối bảo mật SSL/TLS được thiết lập. Tất cả dữ liệu ứng dụng được trao đổi giữa hai máy từ thời điểm này sẽ được mã hóa bằng khóa bí mật chung (shared secret key) đã được tạo ra trong giai đoạn trước. Record Layer sẽ đảm nhiệm việc phân mảnh dữ liệu, nén, tính toán mã xác thực tin nhắn - MAC và mã hóa dữ liệu trước khi truyền đi (chi tiết về Record Layer đã được trình bày tại phần 2.4).

Ở phía nhận, quá trình này sẽ diễn ra ngược lại: dữ liệu nhận được giải mã cũng bằng shared secret key thông qua thuật toán mã hóa đối xứng đã được thỏa thuận từ trước, giá trị MAC cũng được xác minh để đảm bảo tính toàn vẹn trước khi được chuyển lên tầng ứng dụng.

### 3. Đóng kết nối

Khi phiên giao tiếp kết thúc, một trong hai bên có thể đóng kết nối an toàn bằng cách gửi một thông điệp *close\_notify* (một thông điệp cảnh bảo ở mức độ *warning*) thông qua Alert Protocol. Bên còn lại cũng sẽ phản hồi bằng một thông điệp *close\_notify* khác. Việc trao đổi thông điệp như vậy đảm bảo rằng cả hai bên đều biết kết nối sắp kết thúc và có thể hoàn tất mọi hoạt động đang dang dở trước khi các kết nối khác được đóng hoàn toàn.

### 4. Kết luận

Giao thức SSL/TLS đảm bảo an toàn cho truyền thông bằng cách trải qua một quy trình bắt tay phức tạp để thiết lập các tham số bảo mật và khóa mã hóa. Sau đó, dữ liệu ứng dụng được bảo vệ thông qua mã hóa và xác thực tính toàn vẹn. Cuối cùng, việc đóng kết nối được thực hiện một cách trật tự. Quy trình này cung cấp một nền tảng vững chắc cho các giao dịch và trao đổi thông tin an toàn trên internet.

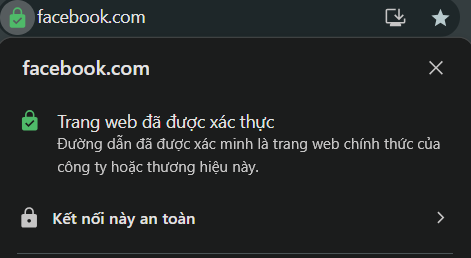
## IV. Ứng dụng của giao thức SSL/TLS

Giao thức SSL/TLS đã trở thành nền tảng không thể thiếu cho việc bảo mật thông tin trên internet.

### 1. Bảo mật truy cập Web - HTTPS

HTTPS là phiên bản bảo mật của giao thức HTTP, nhờ có sự hỗ trợ của SSL/TLS, được sử dụng để truyền tải dữ liệu giữa trình duyệt web và máy chủ web. HTTPS đã trở thành một tiêu chuẩn bảo mật không thể thiếu cho các website. Khi một trang web sử dụng giao thức HTTPS, toàn bộ giao tiếp của người dùng và máy chủ web sẽ được mã hóa bằng SSL/TLS. Điều này đảm bảo:

* Tính bảo mật: Ngăn chặn kẻ xấu đọc thông tin như mật khẩu, thông tin thẻ tín dụng, dữ liệu cá nhân.
* Tính toàn vẹn: Đảm bảo dữ liệu không bị giả mạo hoặc thay đổi trong quá trình truyền tải.
* Xác thực máy chủ: Cho phép trình duyệt xác minh danh tính của máy chủ web, đảm bảo người dùng đang kết nối với đúng trang web mong muốn, tránh web giả mạo.



Hình 8. Biểu tượng ổ khóa xanh trên thanh địa chỉ trình duyệt

Hầu hết các trang web hiện đại, đặc biệt là những trang xử lý thông tin cá nhân hoặc tài chính, đều sử dụng HTTPS để bảo vệ người dùng. Biểu tượng ổ khóa xanh lá cây hoặc tương tự trên thanh địa chỉ trình duyệt (Hình 8) là dấu hiệu cho thấy kết nối đang được bảo vệ bởi giao thức SSL/TLS.

### 2. Bảo mật Email

Các giao thức truyền tải email tiêu chuẩn như SMTP, IMAP, POP3 đều có các phiên bản bảo mật sử dụng giao thức SSL/TLS. Điều này giúp cho:

* Bảo vệ nội dung email: Mã hóa nội dung email khi nó được truyền tải giữa máy chủ email và ứng dụng email.
* Bảo vệ thông tin đăng nhập, xác thực người dùng.
* Ngăn chặn nghe lén: Đảm bảo kẻ xấu không thể dễ dàng theo dõi và đọc được nội dụng email hoặc thông tin tài khoản.

**3. Ứng dụng trong thanh toán, ngân hàng trực tuyến**

Giao thức SSL/TLS được sử dụng rộng rãi trong các ứng dụng thanh toán trực tuyến cũng như các cổng Internet Banking hay Mobile Banking. Một số điểm quan trọng về việc áp dụng SSL/TLS trong lĩnh vực thanh toán, ngân hàng trực tuyến có thể kể đến:

* Mã hóa thông tin thanh toán: Dữ liệu như số thẻ tín dụng, mã CVV, địa chỉ giao hàng được mã hóa để ngăn chặn đánh cắp.
* Mã hóa giao dịch: Mọi thao tác từ đăng nhập, truy vấn số dư đến thực hiện giao dịch đều được bảo vệ bằng TLS.
* Xác thực hai chiều (mutual TLS): Một số hệ thống ngân hàng yêu cầu cả máy chủ và ứng dụng khách xác thực lẫn nhau để tăng tính an toàn.
* Chống giả mạo phiên làm việc: TLS đảm bảo kết nối là duy nhất và không thể sao chép bằng cách tạo các khóa phiên.

Trong thực tế, Các đơn vị cung cấp dịch vụ thanh toán như Stripe, PayPal, VNPAY... đều bắt buộc sử dụng TLS 1.2 trở lên theo yêu cầu của tiêu chuẩn PCI DSS.

**4. Trong ứng dụng trò chuyện và nhắn tin trực tuyến**

Các ứng dụng trò chuyện, nhắn tin trực tuyến như Zalo, Messenger, Telegram, WhatsApp... đều sử dụng SSL/TLS để đảm bảo:

* Tin nhắn được mã hóa khi truyền tải, tăng độ bảo mật, ngăn chặn kẻ tấn công đọc nội dung nếu bắt được gói tin.
* Bảo vệ danh tính người dùng, tránh bị giả mạo hoặc đánh cắp tài khoản.
* Kết hợp với mã hóa đầu cuối (End-toEnd Encryption) để tăng thêm lớp bảo mật.

### 5. Điện toán đám mây

Trong các mô hình cloud, việc truyền dữ liệu giữa người dùng và hệ thống từ xa cần đảm bảo an toàn tuyệt đối. Do đó, việc sử dụng giao thức SSL/TLS sẽ hỗ trợ:

* Bảo vệ thông tin cá nhân, dữ liệu doanh nghiệp khi trao đổi với máy chủ cloud.
* Xác thực danh tính máy chủ cloud, giảm nguy cơ truy cập vào dịch vụ giả mạo.
* Bảo vệ dữ liệu trong mạng công cộng: Mã hóa dữ liệu và xác thực danh tính đảm bảo dữ liệu không thể bị đánh cắp, thay đổi, đọc trộm trong qua trình truyền cũng như lưu trữ trên môi trường mạng công cộng,
* Hỗ trợ truyền tải an toàn qua API, dịch vụ web, điều rất phổ biến trong cloud.

### 6. Kết luận

Trên đây chỉ là một vài những ứng dụng tiêu biểu của giao thức SSL/TLS. Giao thức SSL/TLS đóng vai trò quan trọng trong việc đảm bảo an toàn cho mọi hoạt động truyền dữ liệu trên Internet. Từ các trang web thương mại điện tử, hệ thống thanh toán trực tuyến, dịch vụ đám mây, thiết bị IoT, VPN cho đến các nền tảng nhắn tin tức thời, SSL/TLS đều được sử dụng như một lớp bảo mật tiêu chuẩn để mã hóa dữ liệu, xác thực danh tính và đảm bảo tính toàn vẹn thông tin.

Trong bối cảnh số hóa ngày càng sâu rộng và các mối đe dọa mạng ngày càng tinh vi, việc triển khai SSL/TLS không chỉ là lựa chọn mà là yêu cầu tất yếu để bảo vệ người dùng và hệ thống. Nhờ khả năng thích ứng cao và hỗ trợ đa dạng nền tảng, SSL/TLS tiếp tục là nền tảng bảo mật thiết yếu trong hạ tầng Internet hiện đại và các công nghệ tương lai.

# BÀI THỰC HÀNH

**I, Thuật toán RC4, RC5**

**1. RC4 (Rivest Cipher 4)**

RC4 là một thuật toán mã hóa dòng (stream cipher) được sử dụng phổ biến trong các giao thức bảo mật như SSL/TLS và WEP (Wireless Equivalent Privacy). Thuật toán này hoạt động bằng cách sinh ra một chuỗi khóa giả ngẫu nhiên, sau đó sử dụng nó để mã hóa dữ liệu bằng cách thực hiện phép XOR giữa chuỗi khóa và dữ liệu gốc.

**Cấu trúc và Nguyên lý hoạt động:**

RC4 sử dụng một khóa duy nhất có độ dài từ 1 đến 256 bit. Quá trình mã hóa của RC4 bao gồm các bước sau:

1. Khởi tạo S: RC4 bắt đầu bằng việc khởi tạo một bảng trạng thái S gồm 256 phần tử, được sắp xếp theo một dãy số từ 0 đến 255. Bảng này sau đó được trộn với khóa ban đầu để tạo ra một dãy khóa.
2. Sinh dãy khóa (Keystream Generation): Sau khi khởi tạo bảng S, thuật toán tiếp tục tạo ra một dãy khóa ngẫu nhiên bằng cách thực hiện quá trình "swap" trên bảng S. Dãy khóa này sau đó được XOR với dữ liệu gốc để mã hóa thông tin.
3. Mã hóa và Giải mã: RC4 thực hiện quá trình mã hóa và giải mã gần như giống nhau, chỉ khác nhau ở cách sử dụng dãy khóa để XOR với dữ liệu.

**Ứng dụng:**

RC4 được sử dụng rộng rãi trong các giao thức bảo mật như SSL/TLS, WEP (mặc dù đã bị phá vỡ và thay thế), và các ứng dụng VPN. Tuy nhiên, RC4 hiện nay đã không còn an toàn vì một số lỗ hổng bảo mật đã được phát hiện, đặc biệt là khi sử dụng RC4 với khóa ngắn hoặc trong các cài đặt không an toàn.

**Nhược điểm của RC4**

* Dễ bị tấn công phân tích mã hóa (Ciphertext-only attacks) do việc sử dụng một dãy khóa dài và không ngẫu nhiên hoàn toàn.
* Một số nghiên cứu đã chỉ ra rằng RC4 có thể bị tấn công trong các hệ thống sử dụng số liệu mã hóa dài hoặc không đủ ngẫu nhiên.
* Vì vậy, RC4 hiện không còn được khuyến nghị sử dụng trong các ứng dụng bảo mật.

**2. RC5 (Rivest Cipher 5)**

**Giới thiệu về RC5**

RC5 là một thuật toán mã hóa đối xứng block cipher, được thiết kế để có thể dễ dàng điều chỉnh với các tham số như độ dài khóa, số vòng lặp và kích thước khối mã hóa. Khác với RC4, RC5 hoạt động theo cơ chế mã hóa các khối dữ liệu, thường có kích thước 32 bit, 64 bit, hoặc 128 bit.

**Cấu trúc và Nguyên lý hoạt động:**

RC5 là một block cipher đơn giản nhưng mạnh mẽ, dựa trên các phép toán cơ bản như XOR, cộng modulo và dịch vòng. Các tham số của RC5 bao gồm:

* Kích thước khối: RC5 có thể sử dụng các kích thước khối như 32-bit, 64-bit hoặc 128-bit.
* Độ dài khóa: Khoá có thể có độ dài từ 0 đến 2040 bit.
* Số vòng lặp: RC5 cho phép người dùng cấu hình số vòng lặp (r), thông thường là từ 0 đến 255 vòng lặp.

**Các bước hoạt động của RC5:**

1. Khởi tạo và tạo khóa con: RC5 bắt đầu bằng việc xử lý khóa ban đầu để tạo ra một dãy khóa con (subkeys), sau đó các dãy khóa này sẽ được sử dụng trong quá trình mã hóa.
2. Vòng mã hóa (Rounds): Mỗi vòng mã hóa của RC5 bao gồm các phép toán XOR và cộng modulo, giúp tạo ra một dãy mã hóa mạnh mẽ. Số vòng lặp được quyết định bởi tham số r, và thường từ 12 đến 20 vòng trong thực tế.
3. Mã hóa và Giải mã: RC5 sử dụng một bảng trạng thái và tiến hành mã hóa qua nhiều vòng để tạo ra kết quả cuối cùng. Quá trình giải mã sử dụng cùng các bước như mã hóa nhưng ngược lại với các khóa con.

**Ứng dụng:**

RC5 đã được sử dụng trong nhiều ứng dụng bảo mật như hệ thống VPN, giao thức SSL/TLS, và các phần mềm mã hóa dữ liệu. Tuy nhiên, sự phát triển của các thuật toán mã hóa mạnh mẽ hơn như AES đã làm cho RC5 ít được sử dụng hiện nay.

**Ưu điểm và Nhược điểm của RC5:**

* Ưu điểm: RC5 có thể tùy chỉnh linh hoạt với các tham số như số vòng lặp và độ dài khóa. Nó cung cấp một mức độ bảo mật cao và có hiệu suất tốt trên nhiều hệ thống phần cứng khác nhau.
* Nhược điểm: Một số người cho rằng RC5 có thể kém an toàn khi so với các thuật toán mã hóa hiện đại hơn như AES. Thêm vào đó, việc sử dụng số vòng quá thấp hoặc khóa quá ngắn có thể dẫn đến các vấn đề bảo mật.

**Tổng kết:**

RC4 là một thuật toán mã hóa dòng đã được sử dụng rộng rãi, nhưng hiện nay đã không còn an toàn và không được khuyến cáo trong các ứng dụng bảo mật.RC5 là một thuật toán block cipher mạnh mẽ và linh hoạt, nhưng đã bị thay thế dần bởi các thuật toán mã hóa hiện đại hơn như AES.

Cả hai thuật toán đều đóng góp quan trọng trong việc phát triển các hệ thống bảo mật và mã hóa dữ liệu, nhưng do sự tiến bộ trong nghiên cứu và các cuộc tấn công bảo mật, RC4 và RC5 hiện nay không còn được ưa chuộng.

# TÀI LIỆU THAM KHẢO

[1] Eric Rescola, Tim Dierks, “The Transport Layer Security (TLS) Protocol Version 1.2”, 2008 Available at: <https://datatracker.ietf.org/doc/html/rfc5246>

[2] Eric Rescola, “The Transport Layer Security (TLS) Protocol Version 1.3”, 2018

<https://datatracker.ietf.org/doc/rfc8446/>

[3] Tran Thi Tinh, “SSL/TLS là gì**” ,**  2018, Available at: <https://viblo.asia/p/ssltls-la-gi-Do754wnBlM6>

[4] “Tổng quan về giao thức SSL, TLS”, 2022, Available at: <https://vnpro.vn/thu-vien/tong-quan-ve-giao-thuc-ssl-tls-4667.html>

[5] CloudFlare, “What is HTTPS?”, Available at: <https://www.cloudflare.com/learning/ssl/what-is-https/>

[6] IETF, (2015). Prohibiting RC4 Cipher Suites in TLS. RFC 7465. Available at: https://datatracker.ietf.org/doc/html/rfc7465.

[7] Rivest, R. L. (1994). The RC5 Encryption Algorithm. In: Fast Software Encryption (FSE 1994). Springer. Available at: <https://people.csail.mit.edu/rivest/pubs/Riv94a.pdf>.