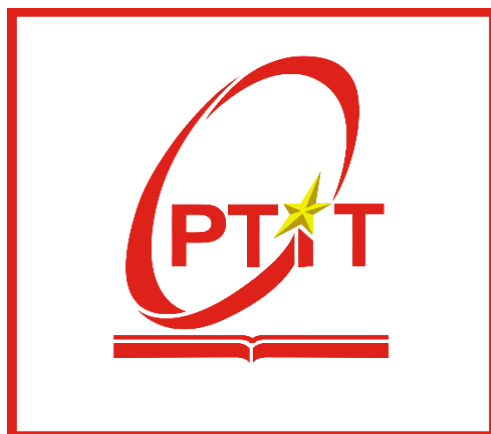


HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN

-----***-----



Đề tài: TÌM HIỂU VỀ GIAO THỨC HTTP VÀ HTTPS

Các thành viên trong nhóm:

1. Lưu Văn Hoàng Hiệp – B20DCAT055
2. Bùi Quang Anh - B20DCAT003
3. Ninh Văn Cường - B20DCAT023
4. **Nguyễn Đăng Tuấn Bảo - B20DCAT015**
5. Đào Văn Chung - B20DCAT027
6. Nguyễn Quang Huy - B20DCAT079

Hà Nội, 2023

Contents

1. Tìm hiểu về giao thức truyền thông siêu văn bản.....	3
1.1. Giao thức truyền thông siêu văn bản là gì?	3
1.2. Giao thức truyền thông siêu văn bản có bảo mật là gì?	5
1.3. Cơ chế bảo mật HSTS	7
1.4. HTTP và HTTPS khác nhau như thế nào?	7
1.3.1 Port trên HTTP và HTTPS	8
1.3.2 Mức độ bảo mật của HTTP và HTTPS.	8
1.3.3 HTTPS bảo mật thông tin người dung.	8
1.3.4 Tránh lừa đảo bằng website giả mạo.	9
1.3.5 Giao thức HTTPS tăng uy tín website đối với người dung.	9
1.3.6 HTTPS chậm hơn HTTP, nhưng không đáng kể.....	10
2. Tìm hiểu về SSL/TLS.	12
2.1. SSL là gì ?	12
2.2. Một số thuật ngữ trong SSL	13
2.3. Trình duyệt làm thế nào để kiểm tra một SSL là có thực hay không?.....	14
2.4. Phương thức hoạt động của SSL	15
• Mã hóa bất đối xứng (Asymmetric Cryptography)	15
• Mã hóa đối xứng (Symmetric Cryptography)	16
2.4.1. Truyền dữ liệu qua SSL	16
2.4.2. SSL Handshake	17
2.5. Tại sao nên sử dụng SSL?	18
3. Tìm hiểu về TLS.....	19
3.1. TLS là gì ?	19
3.2. Phương thức hoạt động của TLS.....	19
3.3. Ứng dụng của TLS	20
3.4. So sánh SSL và TLS	21

1. Tìm hiểu về giao thức truyền thông siêu văn bản.

1.1. Giao thức truyền thông siêu văn bản là gì?

- Giao thức truyền tải siêu văn bản **HTTP (Hypertext Transfer Protocol)** là giao thức tiêu chuẩn cho World Wide Web (www) để truyền tải dữ liệu dưới dạng văn bản, âm thanh, hình ảnh, video từ Web Server tới trình duyệt web của người dùng và ngược lại.

- **HTTP là một giao thức ứng dụng của bộ giao thức TCP/IP** (các giao thức nền tảng cho Internet). Bộ giao thức TCP/IP là một bộ các giao thức truyền thông cài đặt chồng giao thức mà Internet và hầu hết các mạng máy tính thương mại đang chạy trên đó. Bộ giao thức này được đặt theo tên hai giao thức chính là TCP (Transmission Control Protocol – Giao thức điều khiển truyền vận) và IP (Internet Protocol – Giao thức Internet).



- **HTTP hoạt động theo mô hình Client (máy khách) – Server (máy chủ).** Việc truy cập website được tiến hành dựa trên các giao tiếp giữa 2 đối tượng trên. Khi bạn truy cập một trang web qua giao thức HTTP, trình duyệt sẽ thực hiện các phiên kết nối đến server của trang web đó thông qua địa chỉ IP do hệ thống phân giải tên miền DNS cung cấp. Máy chủ sau khi nhận lệnh, sẽ trả về lệnh tương ứng giúp hiển thị website, bao gồm các nội dung như: văn bản, ảnh, video, âm thanh,...

Cấu trúc cơ bản của một yêu cầu HTTP bao gồm:

- Phương thức (method): Xác định hành động mà yêu cầu muốn thực hiện, như GET, POST, PUT, DELETE.
- URL (Uniform Resource Locator): Địa chỉ tài nguyên mà yêu cầu đích đến trên mạng.

- Phiên bản HTTP: Xác định phiên bản của giao thức HTTP được sử dụng, ví dụ: HTTP/1.1.
- Header: Chứa các thông tin bổ sung về yêu cầu, như User-Agent, Accept-Language, Cookie và nhiều hơn nữa.
- Thân (body): Dữ liệu tùy chọn được gửi kèm theo yêu cầu.

-Phản hồi HTTP từ server chứa thông tin về trạng thái của yêu cầu và nội dung tương ứng. Mã trạng thái HTTP phổ biến bao gồm 200 OK (yêu cầu thành công), 404 Not Found (không tìm thấy tài nguyên), và 500 Internal Server Error (lỗi máy chủ).

-Ưu điểm của giao thức HTTP:

1. Đơn giản: HTTP có cấu trúc đơn giản và dễ hiểu, điều này làm cho việc triển khai và phát triển ứng dụng web trở nên dễ dàng hơn.
2. Mở: HTTP là một giao thức mở, tức là thông tin về giao thức và cách thức hoạt động được công khai. Điều này cho phép các nhà phát triển xây dựng ứng dụng và công cụ tương thích với giao thức một cách linh hoạt và sáng tạo.
3. Khả năng mở rộng: HTTP cho phép mở rộng dễ dàng, cho phép thêm các phương thức, tiêu đề và tính năng mới theo nhu cầu. Điều này giúp giao thức thích ứng và phát triển theo xu hướng công nghệ mới.
4. Tương thích ngược: HTTP hỗ trợ tương thích ngược với các phiên bản trước đó. Điều này có nghĩa là các ứng dụng mới có thể tương tác với các máy chủ HTTP cũ và ngược lại.

-Nhược điểm của giao thức HTTP:

1. Không an toàn: HTTP truyền dữ liệu dưới dạng văn bản không mã hóa, điều này có nghĩa là dữ liệu có thể bị đánh cắp hoặc chỉnh sửa bởi bên thứ ba. Việc truyền tải thông tin nhạy cảm như mật khẩu hoặc thông tin tài khoản qua HTTP không an toàn.
2. Không hiệu quả với các ứng dụng đòi hỏi truyền tải dữ liệu lớn: HTTP không được tối ưu cho việc truyền tải dữ liệu lớn hoặc các ứng dụng yêu cầu truyền tải dữ liệu theo thời gian thực. Việc truyền tải các tệp tin lớn có thể gây tốn kém tài nguyên và làm chậm quá trình truyền tải.
3. Không thân thiện với kết nối liên tục: Mỗi yêu cầu HTTP tạo ra một kết nối mới giữa client và server. Điều này gây tốn kém tài nguyên và tăng thời gian phản hồi khi có nhiều yêu cầu được gửi liên tục.
4. Thiếu khả năng xác thực và quyền truy cập: HTTP không cung cấp cơ chế xác thực mạnh mẽ

Giải pháp: Vấn đề tính bảo mật của HTTP chỉ có thể được giải quyết bằng cách chuyển sang sử dụng giao thức truyền thông siêu văn bản có bảo mật (HTTPS) hoặc các giao thức bảo mật khác.

1.2. Giao thức truyền thông siêu văn bản có bảo mật là gì?

- **Giao thức truyền thông siêu văn bản có bảo mật HTTPS (Hypertext Transfer Protocol Secure)** là giao thức HTTP nhưng tích hợp thêm Chứng chỉ bảo mật SSL/TLS nhằm mã hóa các thông điệp giao tiếp để tăng tính bảo mật. Có thể hiểu, HTTPS là phiên bản HTTP an toàn, bảo mật hơn.



- HTTPS hoạt động tương tự như HTTP, tuy nhiên được bổ sung thêm chứng chỉ SSL (Secure Sockets Layer – tầng ổ bảo mật) **hoặc** TLS (Transport Layer Security – bảo mật tầng truyền tải). Hiện tại, đây là các tiêu chuẩn bảo mật hàng đầu cho hàng triệu website trên toàn thế giới.

- Cả SSL và TLS đều sử dụng hệ thống PKI (Public Key Infrastructure - hạ tầng khóa công khai) không đối xứng. Hệ thống này sử dụng hai “khóa” để mã hóa thông tin liên lạc, “khóa công khai” (public key) và “khóa riêng” (private key). Bất cứ thứ gì được mã hóa bằng khóa công khai chỉ có thể được giải mã bởi khóa riêng và ngược lại. Các tiêu chuẩn này đảm bảo các nội dung sẽ được mã hóa trước khi truyền đi, và giải mã khi nhận. Điều này khiến hacker dù có chen ngang lấy được thông tin cũng không thể “hiểu” được thông tin đó.

- Ý tưởng chính của HTTPS là tìm cách tạo ra một kênh truyền tin an toàn trên một mạng không an toàn. Điều này có thể cung cấp những phương thức bảo vệ có hiệu quả chống lại những kẻ “nghe lén” và chống lại sự tấn công của “kẻ đứng giữa” bằng cách dùng một dãy quy tắc mã hóa thích hợp và thiết kế sao cho chứng thư của máy chủ phải được kiểm tra và tin tưởng. “Niềm tin” tạo được trong HTTPS dựa chủ yếu vào cơ sở các cơ quan chứng thực điện tử (CA) được cài đặt trước trên trình duyệt. Do vậy, một sự kết nối HTTPS đến một website có thể được tin cậy khi và chỉ khi các điều kiện sau đây được thực hiện:

1. Người sử dụng tin tưởng rằng trình duyệt của họ thực hiện một cách đúng đắn giao thức HTTPS đã được cài đặt trước với những CA đáng tin cậy.
2. Người sử dụng tin tưởng là CA chỉ chứng thực cho những website hợp pháp, không có quan hệ với những website lừa đảo.
3. Website xuất trình một chứng thư hợp lệ, nghĩa là được ký xác nhận bởi một CA đáng tin cậy.
4. Trong chứng thư chỉ rõ căn cước nhận dạng của website (nghĩa là nếu trình duyệt truy cập đến địa chỉ:
“https://vidu.com” thì chứng thư của website thực sự thuộc về công ty vidu chứ không phải thuộc về tổ chức khác!)
5. Hoặc là mọi can thiệp ngẫu nhiên trên Internet đều đáng tin cậy hoặc là người sử dụng tin tưởng là tầng mạng được mã hóa bởi giao thức bảo mật (TLS hay SSL) là không thể bị nghe lén.

Ưu điểm của việc sử dụng HTTPS:

Bảo mật: HTTPS sử dụng phương thức mã hóa để bảo vệ thông tin truyền tải giữa trình duyệt và máy chủ, ngăn chặn các tấn công gián điệp và đánh cắp thông tin cá nhân của người dùng.

Đáng tin cậy: Chứng chỉ SSL/TLS được cấp phép bởi các tổ chức uy tín, giúp đảm bảo tính toàn vẹn và đáng tin cậy của trang web.

Tăng khả năng tương tác: HTTPS giúp tăng cường sự tương tác giữa người dùng và trang web, cho phép người dùng thực hiện các giao dịch an toàn và tiện lợi hơn.

Tăng khả năng tìm kiếm: Các công cụ tìm kiếm như Google đang ưu tiên các trang web sử dụng HTTPS trong kết quả tìm kiếm, giúp tăng khả năng tìm thấy trang web của bạn.

Nâng cao uy tín: Việc sử dụng HTTPS có thể tạo dấu ấn và tăng uy tín cho trang web của bạn, đặc biệt đối với các trang web chứa thông tin nhạy cảm như thông tin tài khoản ngân hàng, thông tin thẻ tín dụng,...

Tuy nhiên, việc sử dụng HTTPS cũng có một số nhược điểm:

Chi phí: Việc cài đặt và sử dụng HTTPS đòi hỏi một chi phí cho việc mua chứng chỉ SSL/TLS từ các tổ chức uy tín, đặc biệt đối với các tổ chức nhỏ và các trang web cá nhân.

Tăng khối lượng dữ liệu: HTTPS tăng khối lượng dữ liệu được truyền tải giữa máy chủ và trình duyệt, do đó có thể làm chậm hiệu suất của trang web.

Yêu cầu kỹ thuật cao: Cài đặt và cấu hình HTTPS đòi hỏi các kỹ năng kỹ thuật cao và hiểu biết về bảo mật để đảm bảo tính toàn vẹn và bảo mật của trang web.

1.3. Cơ chế bảo mật HSTS

HSTS (HTTP Strict Transport Security) là một giao thức bảo mật mới yêu cầu tất cả kết nối tới một website phải được mã hóa bằng giao thức HTTPS.

HSTS là một hệ thống dựa trên thời gian, nghĩa là trong khoảng thời gian bạn thiết lập trong max-age (tính bằng giây) sẽ đảm bảo rằng trang web của bạn được phục vụ qua giao thức HTTPS.

Khi trình duyệt tương tác với máy chủ web đã bật HSTS, cơ chế tải trước sẽ tìm một header đặc biệt nói rằng trình duyệt chỉ nên sử dụng giao thức HTTPS để kết nối với server.

Ngay cả khi người dùng nhập vào một địa chỉ HTTP thì HSTS cũng sẽ tự động chuyển trang sang HTTPS trước khi tải. Thiết lập này được hỗ trợ trên Chrome, Firefox, Safari, Internet Explorer, Edge và Opera. Ben McIlwain, kỹ sư phần mềm của Google Registry, cho rằng rằng “việc sử dụng HSTS sẽ giúp đảm bảo an toàn mặc định cho mọi kết nối”.

1.4. HTTP và HTTPS khác nhau như thế nào?

- Mặc dù cùng là giao thức truyền tải thông tin trên mạng internet, nhưng HTTP và HTTPS có những điểm khác nhau cốt lõi khiến cho HTTPS được ưa chuộng hơn trên toàn thế giới.



- Sự khác biệt lớn nhất giữa HTTP và HTTPS là chứng chỉ SSL. Về cơ bản, HTTPS là một giao thức HTTP với bảo mật bổ sung. Tuy nhiên, trong thời đại mà mọi thông tin đều được số hóa, thì giao thức HTTPS lại trở nên cực kỳ cần

thiết cho bảo mật website. Dù bạn sử dụng máy tính cá nhân hay công cộng, các tiêu chuẩn SSL sẽ luôn đảm bảo liên lạc giữa máy khách và máy chủ được an toàn, chống bị dòm ngó.

1.3.1 Port trên HTTP và HTTPS

Port là cổng xác định thông tin trên máy khách, sau đó phân loại để gửi đến máy chủ. Mỗi Port mang một số hiệu riêng với chức năng riêng biệt. Giao thức HTTP sử dụng Port 80, trong khi đó HTTPS sử dụng Port 443 – đây chính là cổng hỗ trợ mã hóa kết nối từ máy tính client đến server, nhằm bảo vệ gói dữ liệu đang được truyền đi.

1.3.2 Mức độ bảo mật của HTTP và HTTPS.

- Khi máy khách truy cập một website, giao thức HTTPS sẽ hỗ trợ xác thực tính đích danh của website đó thông qua việc kiểm tra xác thực bảo mật (Security Certificate).

- Các xác thực bảo mật này được cung cấp và xác minh bởi Certificate Authority (CA) – các tổ chức phát hành các chứng thực các loại chứng thư số cho người dùng, doanh nghiệp, máy chủ, mã nguồn, phần mềm. Các tổ chức này đóng vai trò là bên thứ ba, được cả hai bên tin tưởng để hỗ trợ quá trình trao đổi thông tin an toàn.

- Đối với HTTP, vì dữ liệu không được xác thực bảo mật nên sẽ không có gì đảm bảo được phiên kết nối của bạn có đang bị “nghe lén” hay không, hoặc bạn đang cung cấp thông tin cho website thật hay một website giả mạo.

So sánh ưu nhược điểm:

1.3.3 HTTPS bảo mật thông tin người dùng.

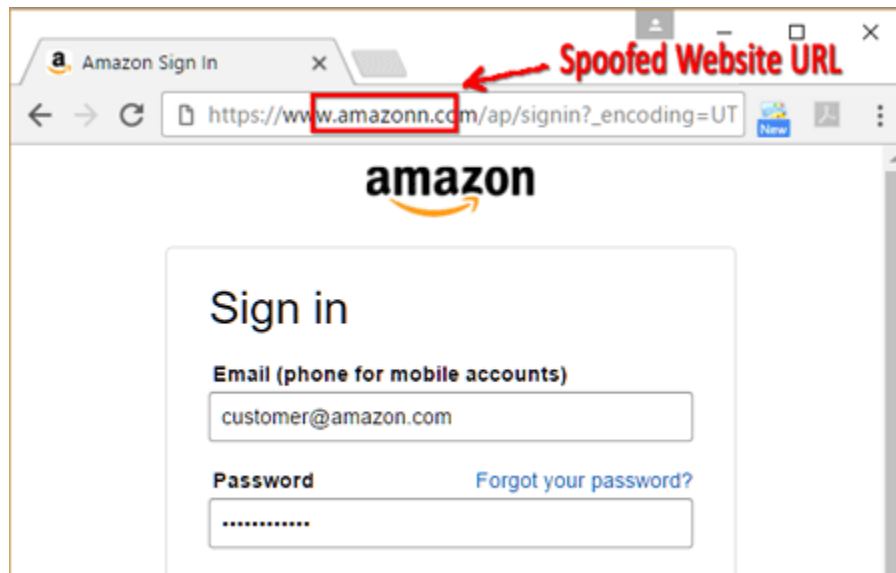
- Giao thức HTTPS sử dụng phương thức mã hóa (encryption) để đảm bảo các thông điệp trao đổi giữa máy khách và máy chủ không bị kẻ thứ ba (hackers) đọc được.

- Nếu truy cập một website không được cài đặt giao thức HTTPS, người dùng sẽ đối diện với nguy cơ bị tấn công sniffing. Hacker có thể “chen ngang” vào kết nối giữa máy khách và máy chủ, đánh cắp các dữ liệu mà người dùng

gửi đi (password, thông tin thẻ tín dụng, văn bản email,...) và các thông tin sẵn có từ website. Thậm chí, mọi thao tác của người dùng trên website đều có thể bị quan sát, ghi lại mà họ không hề hay biết.

- Với giao thức HTTPS, người dùng và máy chủ hoàn toàn có thể tin tưởng rằng các thông điệp chuyển giao qua luôn trong trạng thái nguyên vẹn, không qua bất kì chỉnh sửa, sai lệch nào so với dữ liệu đầu vào.

1.3.4 Tránh lừa đảo bằng website giả mạo.

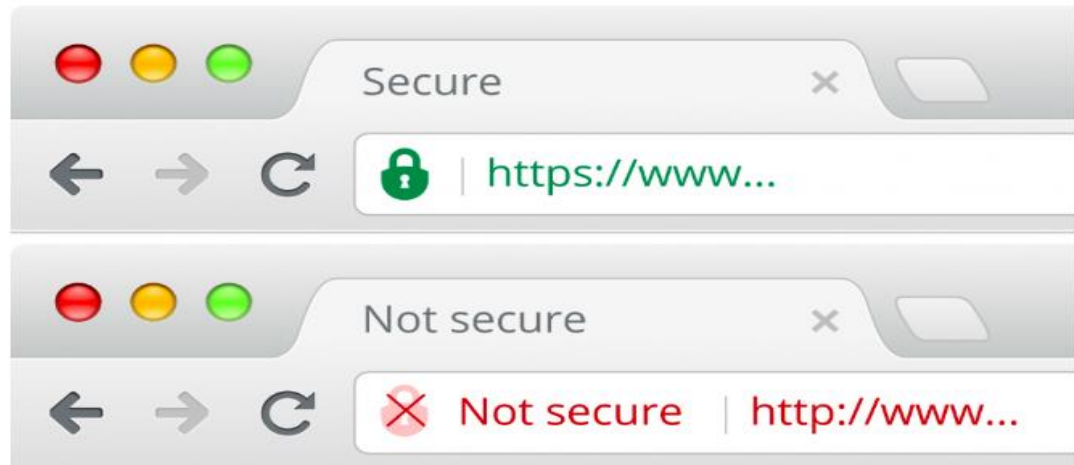


- Trên thực tế, bất kì server nào cũng có thể giả dạng là server của bạn để lấy thông tin từ người dùng, lừa đảo dưới hình thức Phishing. Với giao thức HTTPS, trước khi dữ liệu giữa máy khách và máy chủ được mã hóa để tiếp tục trao đổi, trình duyệt trên máy khách sẽ yêu cầu kiểm tra chứng chỉ SSL từ máy chủ, đảm bảo rằng người dùng đang giao tiếp với đúng đối tượng mà họ muốn. Chứng chỉ SSL/TSL của HTTPS sẽ giúp xác minh đó là website chính thức của doanh nghiệp chứ không phải là website giả mạo.

1.3.5 Giao thức HTTPS tăng uy tín website đối với người dùng.

- Một số trình duyệt web phổ biến như Google Chrome, Mozilla Firefox, Microsoft Edge, hay Apple Safari đều có những cảnh báo người dùng về những website “không bảo mật” sử dụng HTTP. Động thái này giúp bảo vệ thông tin

của người dùng khi lướt web, bao gồm thông tin cá nhân, thẻ ngân hàng và dữ liệu nhạy cảm khác.



=> Trước đây, HTTPS thường được sử dụng cho các website tài chính, ngân hàng, thương mại điện tử để bảo mật thông tin thanh toán online. Tuy nhiên trong thời điểm hiện tại, **HTTPS đã trở thành tiêu chuẩn bảo mật** tối thiểu mà tất cả các website doanh nghiệp cần phải đáp ứng.

- Nhược điểm

1.3.6 HTTPS chậm hơn HTTP, nhưng không đáng kể.

- Nhược điểm duy nhất của HTTPS so với HTTP là sử dụng HTTPS khiến tốc độ giao tiếp (duyệt web, tải trang đích) giữa Client và Server chậm hơn HTTP. Tuy nhiên nhờ công nghệ phát triển, sự khác biệt đã đạt tới giới hạn tiệm cận bằng 0.

Qua phân tích các ưu điểm và nhược điểm của HTTPS, có thể thấy rằng giao thức HTTPS vượt trội hơn hẳn so với HTTP về nhiều mặt, đồng thời còn làm tăng uy tín doanh nghiệp. Đó là lý do tất cả các website đều nên sử dụng HTTPS.

1. 4. Các cách tấn công phổ biến trên HTTP và HTTPS:

- **Tấn công kiểu đệm tràn (Buffer overflow attacks):** Tấn công kiểu đệm tràn là một kỹ thuật khai thác lỗ hổng bảo mật trong các ứng dụng máy tính. Trong khi thực hiện một chức năng, các ứng dụng cần phải lưu trữ dữ liệu trong

bộ nhớ tạm thời (hay còn gọi là bộ nhớ đệm). Kẻ tấn công sẽ gửi đến ứng dụng một lượng dữ liệu lớn hơn dung lượng bộ nhớ tạm thời, gây ra tràn bộ nhớ và khiến hệ thống bị chết đứng hoặc trở nên không ổn định. Tấn công kiểu đệm tràn thường được sử dụng để khai thác và kiểm soát các ứng dụng máy tính.

- **Tấn công DDoS (Distributed Denial of Service attacks):** Tấn công DDoS là một trong những cách tấn công phổ biến nhất trên mạng Internet. Tấn công này được thực hiện bằng cách gửi một lượng lớn yêu cầu truy cập tới một trang web hoặc một máy chủ, từ đó làm cho máy chủ không thể xử lý hết các yêu cầu này và trở nên không khả dụng. Tấn công DDoS thường được thực hiện bởi các botnet, là một mạng lưới các máy tính bị nhiễm virus và bị kiểm soát từ xa bởi kẻ tấn công.

- **Tấn công giả mạo (Spoofing attacks):** Tấn công giả mạo là một kỹ thuật khai thác lỗ hổng bảo mật để giả mạo địa chỉ IP hoặc tên miền của một trang web. Kẻ tấn công sử dụng các phần mềm giả mạo để gửi yêu cầu truy cập tới máy chủ bằng địa chỉ IP hoặc tên miền giả mạo. Nếu thành công, kẻ tấn công có thể lừa người dùng truy cập vào trang web giả mạo và lấy cắp thông tin nhạy cảm hoặc đánh cắp thông tin đăng nhập.

- **Tấn công người đứng giữa (Man-in-the-Middle attacks):** Kẻ tấn công can thiệp vào quá trình truyền tải dữ liệu giữa máy chủ và người dùng, thu thập thông tin cá nhân hoặc đánh cắp thông tin đăng nhập. Ví dụ: kẻ tấn công sử dụng các phương tiện như wifi public không an toàn để theo dõi quá trình truyền tải dữ liệu giữa máy chủ và người dùng, thu thập thông tin cá nhân hoặc đánh cắp thông tin đăng nhập.

- **Tấn công injection (SQL injection attacks, XSS attacks):**

- **SQL injection attacks:** Đây là một cách tấn công nhắm vào hệ thống quản lý cơ sở dữ liệu (DBMS) của một ứng dụng web. Kẻ tấn công sử dụng các trường đầu vào của ứng dụng để chèn vào các câu lệnh SQL có hại, nhằm thu thập thông tin quan trọng, thay đổi hoặc xóa dữ liệu trong cơ sở dữ liệu. Ví dụ, nếu một ứng dụng web cho phép người dùng tìm kiếm sản phẩm theo tên, kẻ tấn công có thể chèn vào câu lệnh SQL như

"SELECT * FROM products WHERE name = 'abc' OR 1=1", để trả về toàn bộ sản phẩm trong cơ sở dữ liệu.

- **XSS attacks:** Đây là một cách tấn công nhắm vào người dùng của một ứng dụng web. Kẻ tấn công sử dụng các trường đầu vào của ứng dụng để chèn vào các mã script có hại, nhằm lấy thông tin cá nhân của người dùng, thay đổi nội dung của trang web hoặc thực hiện các hành động không mong muốn trên trình duyệt của người dùng. Ví dụ, nếu một ứng dụng web cho phép người dùng đăng bài viết, kẻ tấn công có thể chèn vào mã script để đánh cắp cookie của người dùng hoặc thực hiện các hành động khác trên trình duyệt của người dùng.




2. Tìm hiểu về SSL/TLS.

2.1.SSL là gì ?

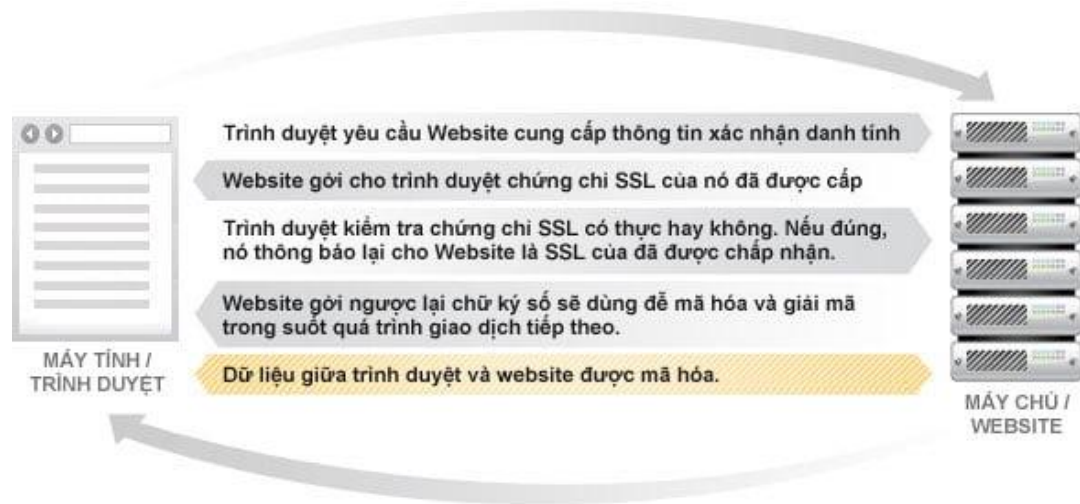
- SSL là viết tắt của từ Secure Sockets Layer. SSL là tiêu chuẩn của công nghệ bảo mật, truyền thông mã hoá giữa máy chủ Web server và trình duyệt. Tiêu chuẩn này hoạt động và đảm bảo rằng các dữ liệu truyền tải giữa máy chủ và trình duyệt của người dùng đều riêng tư và toàn vẹn.

- Chứng thư số SSL cài trên website của doanh nghiệp cho phép khách hàng khi truy cập có thể xác minh được tính xác thực, tin cậy của website, đảm bảo mọi dữ liệu, thông tin trao đổi giữa website và khách hàng được mã hóa, tránh nguy cơ bị can thiệp.

Công nghệ SSL bảo vệ những giao dịch trực tuyến và giúp nâng cao mức độ tin cậy của Website đối với khách hàng chỉ trong 3 bước cơ bản:

		
1. SSL mã hóa các thông tin nhạy cảm trong quá trình giao dịch trực tuyến	2. Mỗi chứng chỉ SSL được tạo ra cho một Website duy nhất.	3. Một cơ quan uy tín đã xác thực danh tính chủ nhân Website trước khi cấp cấp chứng chỉ SSL

SSL là một chuẩn công nghệ được sử dụng bởi hàng triệu trang web trong việc bảo vệ các giao dịch trực tuyến với khách hàng của họ.



Giao thức bảo mật – SSL

2.2. Một số thuật ngữ trong SSL

- **Domain Validation (DV SSL):** Chứng thư số SSL chứng thực cho Domain Name – Website. Khi 1 Website sử dụng DV SSL thì sẽ được xác thực tên domain, website đã được mã hoá an toàn khi trao đổi dữ liệu.
- **Organization Validation (OV SSL):** Chứng thư số SSL chứng thực cho Website và xác thực doanh nghiệp đang sở hữu website đó .
- **Extended Validation (EV SSL):** Cho khách hàng của bạn thấy Website đang sử dụng chứng thư SSL có độ bảo mật cao nhất và được rà soát pháp lý kỹ càng.
- **Subject Alternative Names (SANs SSL):**
Nhiều tên miền hợp nhất trong 1 chứng thư số:

❖ Một chứng thư số SSL tiêu chuẩn chỉ bảo mật cho duy nhất một tên miền đã được kiểm định. Lựa chọn thêm SANs chỉ với chứng thư duy nhất bảo đảm cho nhiều tên miền con. SANs mang lại sự linh hoạt cho người sử dụng, dễ dàng hơn trong việc cài đặt, sử dụng và quản lý chứng thư số SSL. Ngoài ra, SANs có tính bảo mật cao hơn Wildcard SSL, đáp ứng chính xác yêu cầu an toàn đối với máy chủ và làm giảm tổng chi phí triển khai SSL tới tất cả các tên miền và máy chủ cần thiết.

❖ Chứng thư số SSL SANs có thể tích hợp với tất cả các loại chứng thư số SSL của GlobalSign bao gồm: *Chứng thực tên miền (DV SSL)*, *Chứng thực tổ chức doanh nghiệp (OV SSL)* và *Chứng thực mở rộng cao cấp (EV SSL)*.

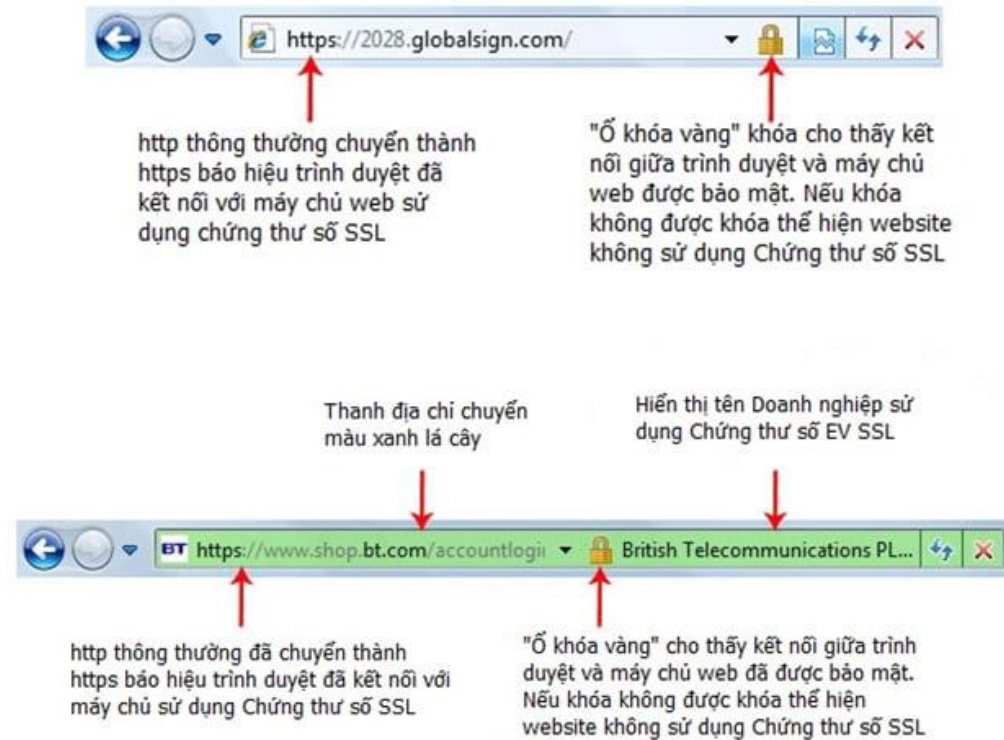
- **Wildcard SSL Certificate (Wildcard SSL):** Sản phẩm lý tưởng dành cho các công thương mại điện tử. Mỗi e-store là một sub-domain và được chia sẻ trên một hoặc nhiều địa chỉ IP. Khi đó, để triển khai giải pháp bảo mật giao dịch trực tuyến (đặt hàng, thanh toán, đăng ký & đăng nhập tài khoản,...) bằng SSL, chúng ta có thể dùng duy nhất một chứng chỉ số Wildcard cho tên miền chính của website và tất cả sub-domain.

2.3.Trình duyệt làm thế nào để kiểm tra một SSL là có thực hay không?

- Khi Website gửi cho trình duyệt một chứng chỉ SSL, Trình duyệt sẽ gửi chứng chỉ này đến một máy chủ lưu trữ các chứng chỉ số đã được phê duyệt. Các máy chủ này được thành lập bởi những công ty uy tín như GlobalSign, VeriSign.

- Về mặt kỹ thuật, SSL sử dụng mã hóa công khai. Kỹ thuật này giúp cho Website và Trình duyệt tự thỏa thuận (bước 4 ở hình trên) một bộ khóa sẽ dùng trong suốt quá trình trao đổi thông tin sau đó.

- Bộ khóa sẽ thay đổi theo mỗi trong lần giao dịch kế tiếp, một người khác sẽ không thể giải mã ngay cả khi có được dữ liệu của máy chủ lưu trữ chứng chỉ số nói trên.



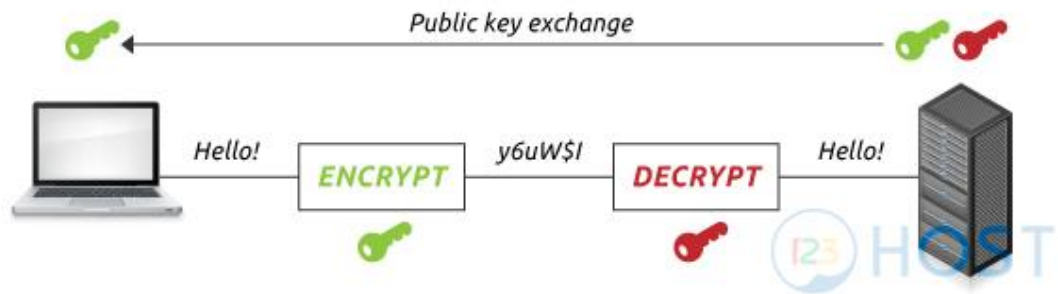
Khi Website gửi cho trình duyệt một chứng chỉ SSL, Trình duyệt sẽ gửi chứng chỉ này đến một máy chủ lưu trữ các chứng chỉ đã được phê duyệt.

2.4. Phương thức hoạt động của SSL

- HTTPS sử dụng giao thức SSL để bảo mật thông tin liên lạc bằng cách truyền dữ liệu được mã hóa. Về cơ bản, SSL hoạt động với các khái niệm sau:

- **Mã hóa bất đối xứng (Asymmetric Cryptography)**

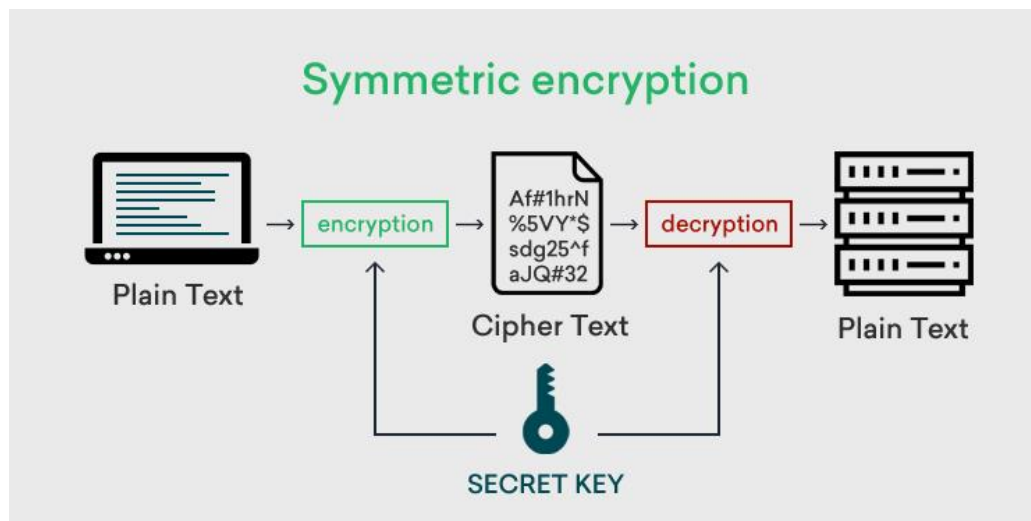
- Mật mã bất đối xứng (còn được gọi là Mã hóa không đối xứng hoặc Mật mã khóa công khai) sử dụng một cặp khóa liên quan đến toán học để mã hóa và giải mã dữ liệu. Trong một cặp khóa, một khóa được chia sẻ với bất kỳ ai quan tâm đến giao tiếp. Nó gọi là Public Key và khóa khác trong cặp khóa được giữ bí mật và được gọi là Private Key. Ở đây, các khóa đề cập đến một giá trị toán học và được tạo ra bằng cách sử dụng một thuật toán toán học để mã hóa hoặc giải mã dữ liệu. Trong mật mã không đối xứng, dữ liệu có thể được ký bằng khóa riêng, chỉ có thể được giải mã bằng khóa công khai liên quan trong một cặp.



- SSL sử dụng mật mã không đối xứng để bắt đầu giao tiếp được gọi là SSL handshake. Các thuật toán mã hóa khóa bất đối xứng được sử dụng phổ biến nhất bao gồm ElGamal, RSA, DSA, kỹ thuật đường cong Elliptic và PKCS.

- **Mã hóa đối xứng (Symmetric Cryptography)**

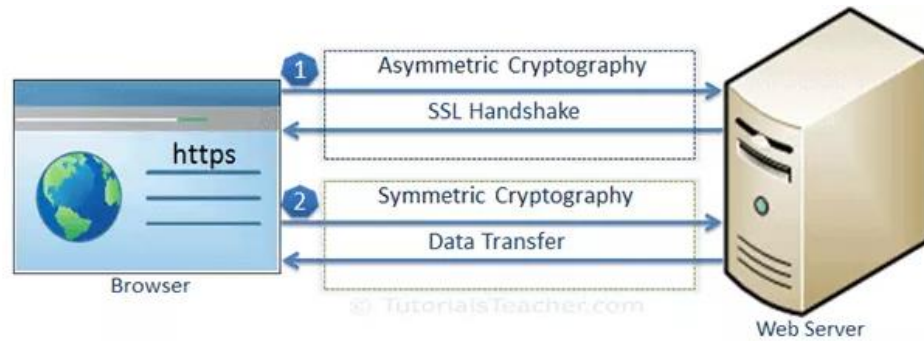
- Trong mật mã đối xứng, chỉ có một khóa mã hóa và giải mã dữ liệu. Cả người gửi và người nhận đều phải có khóa này mà chỉ họ mới biết.



- SSL sử dụng mật mã đối xứng bằng cách sử dụng khóa phiên sau khi quá trình bắt tay ban đầu được thực hiện. Các thuật toán đối xứng được sử dụng rộng rãi nhất là AES-128, AES-192 và AES-256.

2.4.1. Truyền dữ liệu qua SSL

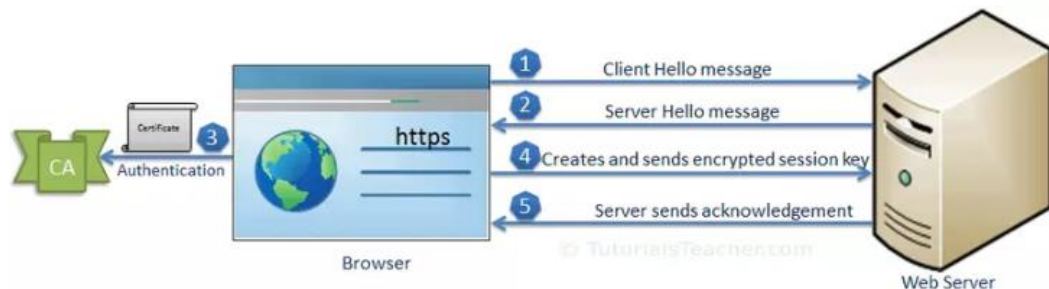
- Giao thức SSL sử dụng mật mã không đối xứng và đối xứng để truyền dữ liệu một cách an toàn. Hình sau minh họa các bước của giao tiếp SSL



- Như bạn có thể thấy trong hình trên, giao tiếp SSL giữa trình duyệt và máy chủ web (hoặc bất kỳ hai hệ thống nào khác) chủ yếu được chia thành hai bước: SSL handshake và truyền dữ liệu thực tế.

2.4.2. SSL Handshake

- Giao tiếp qua SSL luôn bắt đầu bằng SSL Handshake. Handshake SSL là một mật mã không đối xứng cho phép trình duyệt xác minh máy chủ web, lấy khóa công khai và thiết lập kết nối an toàn trước khi bắt đầu truyền dữ liệu thực tế. Hình sau minh họa các bước liên quan đến quá trình SSL Handshake.



1. Máy khách gửi thông báo "client hello". Điều này bao gồm số phiên bản SSL của máy khách, cài đặt mật mã, dữ liệu theo phiên cụ thể và thông tin khác mà máy chủ cần giao tiếp với máy khách bằng SSL.
2. Máy chủ phản hồi bằng một thông báo "server hello". Điều này bao gồm số phiên bản SSL của máy chủ, cài đặt mật mã, dữ liệu theo phiên cụ thể, chứng chỉ SSL có khóa công khai và thông tin khác mà máy khách cần để giao tiếp với máy chủ qua SSL.
3. Máy khách xác minh chứng chỉ SSL của máy chủ từ CA (TCertificate Authority) và xác thực máy chủ. Nếu xác thực không thành công, thì máy khách từ chối kết nối SSL và ném một ngoại lệ. Nếu xác thực thành công, hãy chuyển sang bước 4.

4. Máy khách tạo một session key, mã hóa nó bằng khóa công khai của máy chủ và gửi đến máy chủ. Nếu máy chủ đã yêu cầu xác thực máy khách (chủ yếu là trong giao tiếp máy chủ với máy chủ), thì máy khách sẽ gửi chứng chỉ của chính mình đến máy chủ.

5. Máy chủ giải mã khóa phiên bằng khóa riêng của nó và gửi xác nhận đến máy khách được mã hóa bằng khóa phiên. Do đó, khi kết thúc quá trình bắt tay SSL, cả máy khách và máy chủ đều có khóa phiên hợp lệ mà họ sẽ sử dụng để mã hóa hoặc giải mã dữ liệu thực tế. Khóa công khai và khóa cá nhân sẽ không được sử dụng nữa sau đó.

2.5. Tại sao nên sử dụng SSL?

- Khi bạn đăng ký tên miền để sử dụng các dịch vụ website, email v.v... luôn có những lỗ hổng bảo mật cho hacker tấn công, **SSL** bảo vệ website và khách hàng của bạn.

- **An toàn dữ liệu:** dữ liệu không bị thay đổi bởi hacker.
- **Bảo mật dữ liệu:** dữ liệu được mã hóa và chỉ người nhận đích thực mới có thể giải mã.
- **Chống chối bỏ:** đối tượng thực hiện gửi dữ liệu không thể phủ nhận dữ liệu của mình.

- Tiêu chuẩn xác thực – SSL chỉ được cung cấp bởi các đơn vị cấp phát chứng thư (CA) có uy tín trên toàn thế giới sau khi đã thực hiện xác minh thông tin về chủ thể đăng ký rất kỹ càng mang lại mức độ tin cậy cao cho người dùng Internet và tạo nên giá trị cho các website, doanh nghiệp cung cấp dịch vụ.

- Lợi ích khi sử dụng SSL là gì?

- Xác thực website, giao dịch.
- Nâng cao hình ảnh, thương hiệu và uy tín doanh nghiệp.
- Bảo mật các giao dịch giữa khách hàng và doanh nghiệp, các dịch vụ truy nhập hệ thống.
- Bảo mật webmail và các ứng dụng như Outlook Web Access, Exchange, và Office Communication Server.
- Bảo mật các ứng dụng ảo hóa như Citrix Delivery Platform hoặc các ứng dụng điện toán đám mây.
- Bảo mật dịch vụ FTP.
- Bảo mật truy cập control panel.

- Bảo mật các dịch vụ truyền dữ liệu trong mạng nội bộ, file sharing, extranet.
- Bảo mật VPN Access Servers, Citrix Access Gateway ...
- Website không được xác thực và bảo mật sẽ luôn ẩn chứa nguy cơ bị xâm nhập dữ liệu, dẫn đến hậu quả khách hàng không tin tưởng sử dụng dịch vụ.

Tuy nhiên, công nghệ này đang lỗi thời và được thay thế hoàn toàn bởi TLS.

3. Tìm hiểu về TLS.

3.1. TLS là gì ?

- **TLS** (Transport Layer Security) – Bảo mật tầng vận chuyển. Đây là một giao thức mật mã cung cấp bảo mật đầu cuối cho dữ liệu được gửi giữa các ứng dụng qua Internet. TLS được biết đến chủ yếu thông qua việc sử dụng trong duyệt web an toàn với chuẩn HTTPS. Tuy nhiên, nó cũng có thể và được khuyến khích sử dụng cho các ứng dụng khác như e-mail, truyền tệp, hội nghị truyền hình / âm thanh, nhắn tin tức thì và VoIP, cũng như các dịch vụ Internet như DNS và NTP.

- **TLS** phát triển từ Lớp cổng bảo mật (**SSL** – Secure Sockets Layer) bởi Netscape Communications Corporation vào năm 1994 để bảo mật các phiên duyệt web. TLS không bảo mật dữ liệu trên các hệ thống đầu cuối. Nó chỉ đảm bảo dữ liệu an toàn trên đường truyền qua Internet, tránh khả năng bị nghe trộm hoặc thay đổi nội dung.

- **TLS** thường được triển khai trên TCP để mã hóa các giao thức Lớp ứng dụng như HTTP, FTP, SMTP và IMAP, mặc dù nó cũng có thể được triển khai trên UDP, DCCP và SCTP (ví dụ: đối với ứng dụng dựa trên VPN và SIP).

3.2. Phương thức hoạt động của TLS.

- TLS được cấu thành bởi 3 yếu tố: Khả năng mã hóa, xác thực và tính toàn vẹn. Cụ thể:

- **Khả năng mã hóa:** TLS tiến hành mã hóa các thông tin được truyền tải và giải mã hóa các thông tin nhận về, tránh việc bị mất thông tin bởi nguồn bên ngoài. Điều này giúp dữ liệu được an toàn khỏi hacker.
- **Xác thực:** TLS được tích hợp sẵn khả năng xác thực, giúp đảm bảo độ chân thật của thông tin, dữ liệu.
- **Tính toàn vẹn:** TLS đảm bảo tính toàn vẹn dữ liệu trong các cuộc trao đổi trên internet

- Giao thức TLS hoạt động bằng cách sử dụng mã hóa bất đối xứng và cơ chế xác thực để bảo vệ dữ liệu truyền đi trên mạng. Để hiểu rõ hơn cơ chế hoạt động của TLS, ta có thể phân tích quá trình bảo mật kết nối mạng theo các bước sau:

1. **Khởi tạo kết nối:** Trong quá trình khởi tạo kết nối, hai bên kết nối (ví dụ máy tính của người dùng và máy chủ web) sẽ trao đổi các thông tin cơ bản như phiên bản của TLS, các thuật toán mã hóa và các chứng chỉ bảo mật.
2. **Xác thực chứng chỉ:** Sau khi hai bên đã trao đổi các thông tin cơ bản, máy chủ sẽ gửi một chứng chỉ bảo mật đến máy tính của người dùng. Máy tính của người dùng sẽ kiểm tra chứng chỉ này bằng cách sử dụng các chứng chỉ của tổ chức xác thực đã được lưu trữ trong hệ thống của mình. Nếu chứng chỉ được xác thực, kết nối sẽ được tiếp tục. Nếu không, kết nối sẽ bị ngắt.
3. **Mã hóa dữ liệu:** Sau khi chứng chỉ đã được xác thực, hai bên sẽ sử dụng các private key để mã hóa dữ liệu trước khi gửi đi. Những khóa này được tạo ra từ quá trình trao đổi thông tin cơ bản ban đầu và sẽ khác nhau giữa hai bên. Khi dữ liệu được gửi đi, nó sẽ được mã hóa bằng các khóa này và chỉ người nhận dữ liệu mới có thể giải mã nó bằng các khóa bí mật của họ. Nhờ việc mã hóa dữ liệu, người dùng có thể yên tâm rằng dữ liệu của họ sẽ không bị truy cập bởi bất kỳ ai khác ngoài người nhận dữ liệu.
4. **Gửi dữ liệu:** Sau khi dữ liệu đã được mã hóa, nó có thể được gửi đi an toàn qua mạng. Nếu có bất kỳ ai khác cố gắng truy cập dữ liệu này, họ sẽ không thể đọc được nội dung của nó vì nó đã được mã hóa bằng các khóa bí mật khác.
5. **Giải mã dữ liệu:** Khi dữ liệu đến tại người nhận, nó sẽ được giải mã bằng các khóa bí mật của người nhận để trở thành dữ liệu đọc được. Sau đó, người nhận có thể sử dụng dữ liệu này để thực hiện các thao tác cần thiết.

3.3. Ứng dụng của TLS

- **Truyền dữ liệu trên mạng:** Giao thức TLS được sử dụng để bảo vệ các kết nối mạng trong quá trình truyền thông dữ liệu giữa hai máy tính hoặc hệ thống mạng khác nhau.
- **Truy cập các trang web an toàn:** Giao thức TLS được sử dụng để bảo vệ các kết nối truy cập trang web qua giao thức HTTPS (Hypertext Transfer Protocol Secure). Khi người dùng truy cập vào một trang web qua HTTPS, dữ liệu của họ sẽ được mã hóa.
- **Gửi và nhận email:** Giao thức TLS cũng được sử dụng để bảo vệ các kết nối gửi và nhận email qua giao thức SMTP (Simple Mail Transfer Protocol) và IMAP (Internet Mail Access Protocol). Khi người dùng gửi

hoặc nhận email qua một máy chủ email an toàn, dữ liệu được mã hóa bằng TLS để bảo vệ khỏi các tấn công mạng.

- **Truy cập các dịch vụ trực tuyến:** Giao thức TLS cũng được sử dụng để bảo vệ các kết nối truy cập các dịch vụ trực tuyến, như ngân hàng trực tuyến, bảo hiểm trực tuyến và y tế trực tuyến.

3.4. So sánh SSL và TLS

- TLS là giao thức kế nhiệm trực tiếp của SSL và tất cả các phiên bản SSL hiện không còn được đề xuất sử dụng.

	Chung	Riêng	
		SSL	TLS
Bắt tay SSL/TLS		- Bắt tay trong SSL là 1 kết nối rõ ràng. Nhiều bước	- Bắt tay trong TLS là 1 kết nối ngầm. Ít bước hơn SSL bằng cách loại bỏ các bước bổ sung và giảm số bộ mã hóa vì thế nhanh hơn so với SSL
Thông báo động	Có hai loại thông báo báo động: cảnh báo và nghiêm trọng. Báo động cảnh báo cho biết có lỗi đã xảy ra, nhưng vẫn có thể duy trì kết nối. Báo động nghiêm trọng cho biết phải chấm dứt kết nối ngay lập tức.	- Thông báo báo động SSL không được mã hóa.	- TLS có thêm một loại thông báo báo động được gọi là <i>thông báo đóng phiên</i> . Thông báo đóng phiên báo hiệu phiên kết thúc. - Thông báo báo động TLS cũng được mã hóa để tăng cường bảo mật.
Xác thực thông báo	Đều sử dụng mã xác thực thông báo (MAC), một kỹ thuật mã	- SSL sử dụng thuật toán MD5 – hiện đã lỗi thời – để tạo ra MAC	- TLS sử dụng Mã xác thực thông báo dựa trên băm (HMAC) cho quá

	<p>hóa để xác minh tính xác thực và tính toàn vẹn của thông báo.</p> <p>Bằng cách sử dụng khóa bí mật, giao thức bản ghi tạo ra MAC dưới dạng mã có độ dài cố định và đính kèm vào thông báo gốc.</p>		trình mã hóa và bảo mật phức tạp hơn.
Bộ mã hóa	<p>Bộ mã hóa là một tập hợp các thuật toán tạo khóa để mã hóa thông tin giữa trình duyệt và máy chủ. Thông thường, bộ mã hóa bao gồm thuật toán trao đổi khóa, thuật toán xác thực, thuật toán mã hóa hàng loạt và thuật toán MAC.</p>	<p>- SSL hỗ trợ các thuật toán cũ hơn với các lỗ hổng bảo mật đã biết rõ.</p>	<p>- Một số thuật toán trong TLS đã được nâng cấp từ SSL do lo ngại về bảo mật.</p>
Chứng chỉ		Không còn được sử dụng nữa	Vẫn đang tiếp tục phát triển và sử dụng