# Xây dựng ứng dụng mã hóa sử dụng thư viện OpenSSL

## Giới thiệu thư viện OpenSSL

### Giới thiệu

OpenSSL là một bộ công cụ mạnh mẽ, cấp thương mại và đầy đủ tính năng cho các giao thức Bảo mật lớp truyền tải (TLS) và Lớp cổng bảo mật (SSL). Nó cũng là một thư viện mật mã có mục đích chung. Để biết thêm thông tin về nhóm và cộng đồng xung quanh dự án hoặc để bắt đầu đóng góp của riêng bạn, hãy bắt đầu với trang cộng đồng. Để nhận tin tức mới nhất, hãy tải xuống nguồn, v.v., vui lòng xem thanh bên hoặc các nút ở đầu mỗi trang.

Tài liệu về API OpenSSL hơi mơ hồ. Không có nhiều hướng dẫn về cách sử dụng OpenSSL, vì vậy việc làm cho nó hoạt động trong các ứng dụng có thể hơi rắc rối đối với người mới bắt đầu. Vậy làm cách nào để bạn có thể triển khai một kết nối an toàn cơ bản bằng OpenSSL? Hướng dẫn này sẽ giúp giải quyết vấn đề đó.

Một phần của vấn đề khi học cách triển khai OpenSSL là thực tế là tài liệu không đầy đủ. Tài liệu API không đầy đủ thường ngăn các nhà phát triển sử dụng API, điều này thường đánh dấu sự diệt vong cho nó. Tuy nhiên, OpenSSL vẫn tồn tại và phát triển mạnh mẽ. Tại sao?

OpenSSL là thư viện mở nổi tiếng nhất về giao tiếp an toàn. Tìm kiếm trên Google cho “thư viện SSL” trả về OpenSSL ở đầu danh sách. Nó bắt đầu ra đời vào năm 1998, bắt nguồn từ thư viện SSLeay do Eric Young và Tim Hudson phát triển. Các bộ công cụ SSL khác bao gồm GNU TLS, được phân phối theo Giấy phép Công cộng GNU và Dịch vụ An ninh Mạng Mozilla (NSS).

Vậy điều gì làm cho OpenSSL tốt hơn GNU TLS, Mozilla NSS hoặc bất kỳ thư viện nào khác? Cấp phép là một vấn đề. Ngoài ra, GNS TLS (cho đến nay) chỉ hỗ trợ các giao thức TLS v1.0 và SSL v3.0, và không nhiều hơn thế.

Mozilla NSS được phân phối theo cả Giấy phép Công cộng Mozilla và GNU GPL, cho phép nhà phát triển lựa chọn. Nhưng Mozilla NSS lớn hơn OpenSSL và yêu cầu các thư viện bên ngoài khác xây dựng thư viện, trong khi OpenSSL hoàn toàn khép kín. Và giống như OpenSSL, phần lớn NSS API không được ghi lại. Mozilla NSS có hỗ trợ PKCS # 11, được sử dụng cho các mã thông báo mật mã, chẳng hạn như Thẻ thông minh. OpenSSL thiếu hỗ trợ này.

### Nền tảng

Để tận dụng tối đa bài viết này, bạn nên:

* Thành thạo lập trình C.
* Làm quen với giao tiếp Internet và viết các ứng dụng hỗ trợ Internet.

Không nhất thiết phải làm quen với SSL, vì phần giải thích ngắn gọn về SSL sẽ được đưa ra sau. Kiến thức về mật mã cũng là một lợi thế, nhưng không bắt buộc.

### Khái niệm SSL

SSL là từ viết tắt của Secure Sockets Layer. Nó là tiêu chuẩn đằng sau giao tiếp an toàn trên Internet, tích hợp mật mã dữ liệu vào giao thức. Dữ liệu được mã hóa trước khi nó rời khỏi máy tính của bạn và chỉ được giải mã khi nó đến đích dự kiến. Chứng chỉ và thuật toán mật mã đứng đằng sau cách tất cả hoạt động và với OpenSSL, bạn có cơ hội để chơi với cả hai.

Về lý thuyết, nếu dữ liệu mã hóa bị chặn hoặc bị nghe trộm trước khi đến đích thì không có hy vọng bẻ khóa dữ liệu đó. Nhưng khi các máy tính ngày càng nhanh hơn khi mỗi năm trôi qua và những tiến bộ mới trong phân tích mật mã được tạo ra, thì cơ hội bẻ khóa các giao thức mật mã được sử dụng trong SSL đang bắt đầu tăng lên.

SSL và các kết nối an toàn có thể được sử dụng cho bất kỳ loại giao thức nào trên Internet, cho dù đó là HTTP, POP3 hay FTP. SSL cũng có thể được sử dụng để bảo mật các phiên Telnet. Mặc dù mọi kết nối đều có thể được bảo mật bằng SSL, nhưng không nhất thiết phải sử dụng SSL trên mọi loại kết nối. Nó nên được sử dụng nếu kết nối sẽ mang thông tin nhạy cảm.

### Khái niệm OpenSSL

OpenSSL không chỉ là SSL. Nó có khả năng đào tin nhắn, mã hóa và giải mã các tệp, chứng chỉ số, chữ ký số và số ngẫu nhiên. Thư viện OpenSSL có khá nhiều thứ, nhiều thứ có thể được đưa vào một bài báo.

OpenSSL không chỉ là API, nó còn là một công cụ dòng lệnh. Công cụ dòng lệnh có thể làm những điều tương tự như API, nhưng tiến xa hơn một bước, cho phép khả năng kiểm tra máy chủ và máy khách SSL. Nó cũng cung cấp cho nhà phát triển ý tưởng về các khả năng của OpenSSL.

### Sử dụng thư viện OpenSSL

#### Cơ chế phát hiện lỗi