TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Học phần : Nhập môn An toàn thông tin

Đề tài : Bảo mật cho Web Service

Giảng viên hướng dẫn : PGS.TS Nguyễn Linh Giang

Sinh viên thực hiện:

Thân Minh Duy - 20170063

Hà Hữu Linh - 20173230

Trần Xuân Đức - 20173034

Nguyễn Bá Quân - 20173315

Hà nội, tháng 6 năm 2020

Lời nói đầu

Ngày nay công nghệ thông tin đang là nền công nghệ mũi nhọn trong chiến lược phát triển kinh tế, xây dựng đất nước của hầu hết các quốc gia. Các sản phẩm công nghệ thông tin đã và đang được ứng dụng rộng rãi trong mọi lĩnh vực của đời sống kinh tế, xã hội và hầu hết đều đem đến những giá trị thiết thực. Đối tượng phục vụ chủ yếu của ngành công nghệ thông tin hiện nay chính là các tổ chức, các cơ sở doanh nghiệp

Bảo mật luôn luôn là một vấn đề hàng đầu cho tất cả các loại ứng dụng, đặc biệt là các ứng dụng web. Từ những ngày đầu của Internet người ta đã quan tâm đến tính an toàn trong trao đổi thông tin. Tuy không có sự an toàn tuyệt đối nhưng những phát triển trong lĩnh vực này thì rất nhanh và mang lại nhiều thành quả vì đây là vấn đề cấp bách của nhiều doanh nghiệp. Không có một mức an toàn thích hợp, sự khai thác thương mại của Internet thì không hoàn toàn an toàn. Do đó những giải thuật để kiểm chứng, sự mã hóa khóa thông tin, và chữ ký số hóa có thể là những giải pháp cung cấp một mức đủ an toàn.

Nhiều tổ chức sử dụng nhiều hệ thống phần mềm để quản lý. Các hệ thống phần mềm khác nhau thường cần trao đổi dữ liệu với nhau và dịch vụ Web là phương thức giao tiếp cho phép hai hệ thống phần mềm trao đổi dữ liệu này qua Internet. Do đó, ngoài việc nghiên cứu làm sao để tạo ra một dịch vụ web tốt mang lại nhiều lợi ích thì việc nghiên cứu để làm sao mang lại sự an toàn cho dịch vụ web đó cũng là một trong những vấn đề quan trọng nhất.

Nội dung

Chương 1: Tổng quan về Web Service.

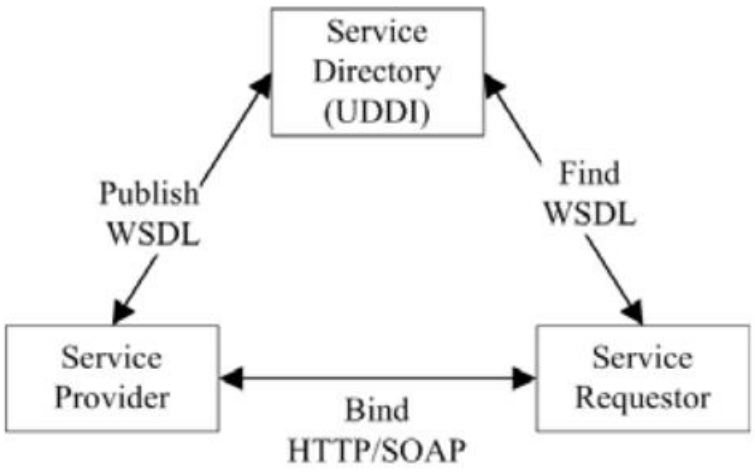
1.1 Giới thiệu

Thuật ngữ Dịch vụ Web (WS) là:

- *Một dịch vụ được cung cấp bởi một thiết bị điện tử tới một thiết bị điện tử khác, trao đổi thông điệp với nhau qua World Wide Web, hoặc*
- *Một máy chủ chạy trên một thiết bị máy tính, lắng nghe yêu cầu tại một cổng cụ thể qua một mạng, phục vụ các tài liệu web (HTML, JSON, XML, images), và tạo các dịch vụ ứng dụng web, phục vụ cho giải quyết các miền vấn đề cụ thể thông qua Web (WWW, Internet, HTTP)*

Các dịch vụ web là hiện thực hóa phổ biến nhất của kiến trúc hướng dịch vụ (SOA). Dịch vụ web là một thành phần phần mềm bất khả xâm phạm trên Web thông qua thông điệp XML (XML, 2005) tuân theo tiêu chuẩn SOAP (SOAP, 2003). Thành phần này cung cấp một hoặc nhiều thao tác để thực hiện các hành động hữu ích thay mặt cho cuộc gọi từ máy khách. Các định dạng và thao tác của thông điệp đầu vào và đầu ra được mô tả bằng WSDL (Christensen & Curbera, 2001). Dựa trên các tiêu chuẩn web, các dịch vụ Web trở nên độc lập ngôn ngữ và nền tảng. Sự mô tả của các dịch vụ theo cách tập trung ngôn ngữ là rất quan trọng đối với việc sử dụng rộng rãi các dịch vụ Web. Để sử dụng chung, một dịch vụ phải được mô tả và quảng cáo. WSDL chăm sóc mô tả bằng cách cung cấp một ngôn ngữ để mô tả một dịch vụ đủ chi tiết để gọi bất kỳ hoạt động nào của nó. Các nhà cung cấp dịch vụ mô tả các dịch vụ Web của họ và quảng cáo chúng trong một sổ đăng ký chung được gọi là UDDI (UDDI, 2002). Điều này cho phép người yêu cầu dịch vụ tìm kiếm sổ đăng ký và tìm dịch vụ phù hợp với yêu cầu của họ. UDDI cho phép tạo ra các đăng ký có thể truy cập trên Web. Một sổ đăng ký chứa nội dung từ các mô tả WSDL cũng như thông tin bổ sung như dữ liệu về nhà cung cấp. Khách hàng có thể sử dụng một hoặc nhiều đăng ký để khám phá các dịch vụ liên quan.

Để mô tả các dịch vụ web hơn nữa, chúng ta hãy xem xét một kịch bản ví dụ. Một công ty tên là X là nhà phân phối sản phẩm. Họ theo dõi khách hàng, hàng hóa và đơn đặt hàng của họ thông qua một hệ thống mà họ có trong nhà. Họ không muốn cung cấp quyền truy cập không giới hạn vào hệ thống này cho khách hàng của mình, nhưng họ muốn khách hàng của họ có thể đặt hàng dễ dàng hơn. Sử dụng các dịch vụ Web, X có thể tạo giao diện cho hệ thống bên trong của họ để khách hàng có thể tra cứu và xác thực, đặt hàng sản phẩm. Với các dịch vụ này, X chỉ cần cung cấp định nghĩa WSDL về dịch vụ cho khách hàng của họ và khách hàng sẽ có thể soạn bất kỳ hệ thống nào về phía họ để xử lý việc đặt hàng theo bất kỳ cách nào họ thấy phù hợp.



Hình 1-1: Web Service [2]

Sự tiến bộ của các công nghệ dịch vụ Web hứa hẹn sẽ có tác dụng sâu rộng trên Internet và mạng doanh nghiệp. Các dịch vụ web dựa trên Ngôn ngữ đánh dấu có thể mở rộng (XML), SOAP và các tiêu chuẩn mở có liên quan và được triển khai trong Kiến trúc hướng dịch vụ (SOA) cho phép dữ liệu và ứng dụng tương tác mà không cần sự can thiệp của con người thông qua các kết nối động và quảng cáo. Công nghệ dịch vụ web có thể được triển khai trong nhiều kiến trúc khác nhau, có thể cùng tồn tại với các công nghệ và phần mềm khác phương pháp thiết kế, và có thể được áp dụng theo cách thức tiến hóa mà không yêu cầu chuyển đổi lớn đối với các ứng dụng và cơ sở dữ liệu cũ.

Hiện nay, WS trở thành dịch vụ mãnh mẽ, cung cấp lợi ích cho cả doanh nghiệp, khách hàng, cá nhân, trong nhiều lĩnh vực thực tế như: thương mại, du lịch, chứng khoán, …

1.2 Các tiêu chuẩn dịch vụ Web cơ bản

XML, SOAP, WSDL và UDDI (Graham & Simenov, 2002) là các yếu tố cơ bản để triển khai cơ sở hạ tầng SOA dựa trên các dịch vụ Web (xem Hình 1-1). XML là tiêu chuẩn để biểu diễn dữ liệu; SOAP chỉ định lớp vận chuyển để gửi tin nhắn giữa người tiêu dùng và nhà cung cấp; WSDL mô tả các dịch vụ Web; và UDDI được sử dụng để đăng ký và tra cứu các dịch vụ Web.

XML

XML là môt tiêu chuẩn để biểu diễn dữ liệu, đã được chọn làm ngôn ngữ để mô tả các dịch vụ Web. XML được chấp nhận làm tiêu chuẩn để trao đổi dữ liệu trên Web cho phép cấu trúc dữ liệu trên Web. Đây là ngôn ngữ cho dữ liệu bán cấu trúc và đã được đề xuất như một giải pháp cho các vấn đề tích hợp dữ liệu, vì nó cho phép mã hóa và hiển thị dữ liệu linh hoạt, bằng cách sử dụng siêu dữ liệu để mô tả cấu trúc dữ liệu (sử dụng DTD hoặc XSD). Một tài liệu XML được định dạng tốt sẽ tạo ra một cây cân bằng gồm các bộ thẻ mở và đóng, mỗi thẻ có thể bao gồm một số cặp giá trị thuộc tính.

WSDL

Ngôn ngữ mô tả dịch vụ web cung cấp ngôn ngữ dựa trên XML tiêu chuẩn được sử dụng để mô tả các dịch vụ web. Bạn có thể sử dụng tài liệu WSDL để ghi lại các mô tả dịch vụ xác định các hoạt động mà dịch vụ Web cung cấp và để thực hiện chúng. WSDL sử dụng định dạng XML để mô tả các dịch vụ mạng dưới dạng tập hợp các điểm cuối hoạt động trên các thông điệp chứa thông tin hướng

đến tài liệu hoặc hướng thủ tục. Các hoạt động và các thông điệp được mô tả trừu tượng, và sau đó được liên kết với một giao thức mạng và định dạng thông báo cụ thể để xác định điểm cuối.

SOAP

Giao thức truy cập đối tượng đơn giản là một tiêu chuẩn để trao đổi thông tin dựa trên XML giữa các ứng dụng phân tán, truyền dữ liệu qua các giao thức truyền tải tiêu chuẩn như HTTP. SOAP là một giao thức độc lập, nhẹ, bao gồm các phần sau:

- Một phong bì (envelope) xác định một khung (framework) để mô tả những gì trong một thông điệp và cách xử lý nó.
- Một tập hợp các quy tắc mã hóa để biểu diễn các thể hiện của các loại dữ liệu do ứng dụng xác định.
- Một quy ước để đại diện cho các thủ tục gọi và trả lời từ xa.
- Một quy ước ràng buộc để trao đổi thông điệp bằng một giao thức cơ bản.

UDDI

Sổ đăng ký UDDI là một thư mục có thể tìm kiếm các dịch vụ Web mà Người yêu cầu Dịch vụ có thể sử dụng để tìm kiếm Dịch vụ Web và truy cập các tài liệu WSDL. UDDI cung cấp sổ đăng ký phân tán cho các doanh nghiệp và các mô tả dịch vụ của họ được triển khai theo định dạng XML phổ biến và hoạt động như một dịch vụ "Yellow Pages" xác định cách xuất bản và khám phá thông tin về Dịch vụ web.

→ Tóm lại, UDDI cho phép người dùng xuất bản và / hoặc tìm các dịch vụ web được chỉ định; WSDL cho phép người dùng mô tả các dịch vụ web theo định dạng chuẩn, có thể truy cập được; và SOAP cho phép người dùng liên kết và sử dụng Dịch vụ web từ một ứng dụng. Các tiêu chuẩn dành riêng cho Dịch vụ Web này cung cấp cơ sở hạ tầng để xuất bản (sử dụng WSDL và UDDI), tìm (sử dụng WSDL và UDDI) và liên kết (sử dụng WSDL và SOAP) theo cách tương thích.

1.3 Các dịch vụ REST và SOAP

Các dịch vụ Web đang ngày càng được quan tâm và gia tăng nhanh chóng trong vài năm trở lại đây. Như chúng ta đã biết, một dịch vụ web được mô tả như một phương thức cho giao tiếp, trao đổi thông tin giữa các thiết bị qua một mạng.

Trong thời đại phát triển công nghệ hiện đại như hiện nay, có rất nhiều cách có thể được sử dụng để tạo ra các ứng dụng doanh nghiệp. Việc lựa chọn cái này hơn cái kia chỉ nên dựa trên các đối số kỹ thuật và khả năng của chúng được phân phối bởi mỗi phương án. Do đó, các dịch vụ web đã đạt được sự phổ biến to lớn trong cách các thiết bị giao tiếp với nhau.

Có rất nhiều công nghệ có thể thực hiện giao tiếp này, chẳng hạn như RMI (Remote Method Invocation), CORBA hoặc DCOM. Nhưng, khi nói đến bảo mật hoặc khả năng tương thích, các công nghệ này dường như gây ra nhiều rắc rối. Thay vào đó, công nghệ hiện đại thường dựa trên hai mô hình mới: SOAP và REST. (Tihomirovs & Grabis, 2016).

Cả hai, SOAP và REST đều dựa trên kiến trúc hướng dịch vụ (SOA). Quá trình phát triển của họ bao gồm một thứ gọi là API Web, đại diện cho giao diện tiêu thụ dịch vụ của họ.

Các ứng dụng khác nhau như hội thảo, web hoặc ứng dụng xã hội có thể được phát triển bằng các dịch vụ web này, vì không cần có bất kỳ kiến thức nào trước khi sử dụng, điều mà làm cho chúng trở nên độc lập nền tảng và kết nối lỏng lẻo. (Adamopoulos, 2014)

SOAP được thiết kế để trở thành một giao thức độc lập nền tảng, nhẹ cho môi trường phân tán, phân tán sử dụng Internet và XML để trao đổi thông tin giữa các nút. Nó đại diện cho một giao thức thông điệp, sử dụng XML để xác định giao tiếp và HTTP để truyền các thông điệp này. Đó là giao tiếp thông điệp một chiều, không trạng thái giữa các nút hoặc thiết bị, từ người gửi đến người nhận. (Halili et al., 2012)

Trong khi đó, REST đại diện cho kiến trúc máy khách-máy chủ nơi máy khách gửi các yêu cầu, trong khi máy chủ xử lý chúng và trả về các phản hồi. Nó được giới thiệu vào năm 2000, bởi Roy Fielding. Không giống như SOAP, các dịch vụ REST không tự giới hạn ở XML, thay vào đó, nó cũng hỗ trợ JSON (Ký hiệu đối tượng JavaScript), văn bản thuần túy, v.v. (Halili & Kasa, 2011).

1.4 Các dịch vụ web RESTFUL

REST - (Representational state transfer) như tên ngụ ý, nó phải liên quan đến mối quan hệ giữa máy khách và máy chủ và cách lưu trữ trạng thái. Kiến trúc REST dựa trên kiểu kiến trúc máy khách / máy chủ. Do đó, các yêu cầu và phản hồi được xây dựng dựa trên quá trình chuyển giao tài nguyên. Tất cả các tài nguyên được xác định bởi Mã định danh tài nguyên thống nhất (URI) duy nhất, thường đại diện cho một tài liệu nắm bắt trạng thái của tài nguyên. Nói chung, kiến trúc kiểu REST nhẹ hơn nhiều so với SOAP. Nó không yêu cầu các định dạng như tiêu đề được bao gồm trong thông điệp, giống như nó được yêu cầu trong kiến trúc SOAP. Mặt khác, nó phân tích cú pháp JSON, một ngôn ngữ có thể đọc được bởi con người được thiết kế để cho phép trao đổi dữ liệu và giúp máy tính phân tích và sử dụng dễ dàng hơn. Nó được ước tính là nhanh hơn khoảng một trăm lần so với XML.

Một tài liệu JSON đơn giản:

{

“first_name” : “Linh”,

“last_name”: “Ha”

}

Có một số nguyên lý thiết kế các yêu cầu Restful Web Service. Địa chỉ là một nguyên tắc REST trong đó các tập dữ liệu được mô hình hóa để hoạt động như các tài nguyên được đánh dấu URI. Tính không trạng thái là một nguyên tắc khác mà người thiết kế dịch vụ REST sẽ phải tuân theo. Điều này có nghĩa là mọi giao dịch phải độc lập và không được liên quan đến bất kỳ giao dịch nào trước đó, vì tất cả dữ liệu cần thiết để thực hiện và xử lý yêu cầu được chứa trong yêu cầu đó, do đó, máy chủ sẽ không phải duy trì dữ liệu phiên của khách hàng. Giao diện thống nhất yêu cầu một giao diện đồng nhất và tiêu chuẩn được sử dụng để truy cập tài nguyên, tức là sử dụng bộ phương thức HTTP cố định. Nếu người thiết kế dịch vụ tuân thủ các nguyên tắc này, thì gần như đảm bảo rằng ứng dụng REST sẽ đơn giản và gọn nhẹ.

Có 4 phương thức phổ quát được mô tả trong REST là : GET, PUT, POST, DELETE.

Ứng dụng web theo kiến trúc REST chúng tôi gọi là RESTful web service. Các RESTful web services sử dụng các phương thức GET, PUT, POST và DELETE để truy xuất, tạo, cập nhật và xóa tài nguyên. (Sinha et al., 2014)

REST đang trở thành hướng đi cho tương tác hệ thống, bao gồm việc sử dụng các RESTFul web service chủ yếu theo cách các nhà cung cấp *cloud* trưng bày dịch vụ của họ. Trong thời đại ngày nay, chúng ta có thể dễ dàng kết luận rằng hầu hết các dự án mới đều dựa trên kiến trúc RESTFul, để tạo và cung cấp các dịch vụ chuyên nghiệp. Không chỉ những gã khổng lồ công nghệ như Facebook, Google hay Twitter sử dụng REST trong những ngày này. Điều này, nhờ vào kiến trúc REST, mọi ứng dụng có thể mở rộng theo chiều ngang theo cách dễ nhất có thể. (Festim Halili & Erenis Ramadani, 2017).

Các lợi ích của REST.

- REST sử dụng định dạng thông điệp nhỏ hơn và cung cấp hiệu quả chi phí theo thời gian và hiệu suất tốt hơn do các thông điệp JSON tạo ra giao tiếp và không cần xử lý chuyên sâu.
- Hỗ trợ giao tiếp không trạng thái.
- Đơn giản để học và triển khai.
- Sử dụng hiệu quả các phương thức HTTP.
- Băng thông nhẹ vì thông điệp truyền qua của nó là JSON
- Có thể sử dụng nhiều format khác nhau.
- Sử dụng HTTP cho vấn đề bảo mật.
- Nó làm cho dữ liệu có sẵn như là tài nguyên. (Kumari, 2015)

Các nhược điểm của REST

- So sánh với SOAP thì nó không bao gồm tất cả các loại tiêu chuẩn dịch vụ Web như là Bảo mật, Giao dịch, …
- REST không đáng tin cậy.
- Các yêu cầu REST (đặc biệt là GET) không phù hợp với lượng dữ liệu lớn.
- Độ trễ trong thời gian xử lý yêu cầu và sử dụng băng thông

Chương 2: Tổng quan về bảo mật cho Web Services.

Những thách thức bảo mật được đưa ra bởi cách tiếp cận dịch vụ Web là ghê gớm và không thể tránh khỏi. Nhiều tính năng làm cho dịch vụ Web trở nên hấp dẫn, bao gồm khả năng truy cập dữ liệu lớn hơn, kết nối ứng dụng với ứng dụng động và tự chủ tương đối (thiếu sự can thiệp của con người) là mâu thuẫn với các mô hình và điều khiển bảo mật truyền thống.

Trong bảo mật cho web services, các vấn đề khó khăn và các vấn đề chưa được giải quyết tồn tại, chẳng hạn như bảo vệ những điều sau đây:

- Tính bí mật và tính toàn vẹn của dữ liệu được truyền qua các giao thức dịch vụ Web trong các giao service-to-service, bao gồm cả dữ liệu đi qua các dịch vụ trung gian.
- Tính toàn vẹn về chức năng của các dịch vụ Web yêu cầu thiết lập sự tin cậy giữa các dịch vụ trên một transaction-by-transaction cơ bản.
- Sẵn sàng đối mặt với các cuộc tấn công từ chối dịch vụ khai thác lỗ hổng duy nhất cho các công nghệ dịch vụ Web, đặc biệt là nhắm mục tiêu các dịch vụ cốt lõi, như dịch vụ khám phá, trong đó các dịch vụ khác dựa vào.

Các công nghệ bảo mật mạng thông thường (như firewalls) không đủ để bảo vệ các SOAs vì những lý do sau:

- Các SOAs là động và có thể hiếm khi bị ràng buộc hoàn toàn với các ranh giới vật lý của một mạng.
- SOAP và REST được truyền qua HTTP, được phép truyền mà không bị hạn chế qua hầu hết các tường lửa.

Bảo mật cho web service được dựa trên các khái niệm quan trọng, bao gồm:

- **Định danh và Xác thực** (Identification and Authentication) : Xác minh danh tính của người dùng, quy trình hoặc thiết bị, thường là điều kiện tiên quyết để cho phép truy cập vào tài nguyên trong hệ thống thông tin.
- **Ủy quyền** (Authorization). Sự cho phép sử dụng tài nguyên máy tính, được cấp bởi, trực tiếp hoặc gián tiếp, bởi một chủ sở hữu ứng dụng hoặc hệ thống.
- **Toàn vẹn** (Integrity). Thuộc tính mà dữ liệu không bị thay đổi một cách trái phép trong khi lưu trữ, trong quá trình xử lý hoặc vận chuyển.
- **Không bác bỏ** (Non-repudiation). Đảm bảo rằng người gửi thông tin được cung cấp bằng chứng chuyển giao tin và người nhận được cung cấp bằng chứng về danh tính của người gửi → không thể phủ nhận việc đã xử lý thông tin.
- **Bí mật** (Confidentiality): Giữ nguyên các hạn chế được ủy quyền đối với truy cập và tiết lộ thông tin, bao gồm các biện pháp bảo vệ quyền riêng tư và thông tin độc quyền
- **Riêng tư** (Privacy): Hạn chế quyền truy cập.

2.1 Các khía cạnh bảo mật cho web services

Các khía cạnh bảo mật các dịch vụ web đã được xác định là: bảo mật thông điệp (secure messaging), bảo vệ tài nguyên (resource protection), đàm phán hợp đồng (negotiation of contracts), quản lý tin cậy (trust management), và các thuộc tính bảo mật (security properties).

Mỗi khía cạnh là điều cần thiết cho sự phát triển của các ứng dụng bảo mật sử dụng các dịch vụ Web, nhưng mỗi khía cạnh ảnh hưởng đến một tầng khác nhau của dịch vụ Web. Phần này mô tả từng khía cạnh bảo mật và cung cấp tổng quan về những công nghệ có sẵn, và những gì còn phải làm.

2.1.1 Bảo mật thông điệp

Các dịch vụ web dựa trên Internet để liên lạc. Do một số API (như SOAP) không được thiết kế để bảo mật, các thông điệp có thể được xem hoặc sửa đổi bởi những kẻ tấn công khi các thông điệp đi qua Internet. Có một số tùy chọn có sẵn để bảo mật các thông điệp web services:

- Chuyển từ HTTP qua HTTPS: Vì các thông điệp SOAP được truyền bằng HTTP, nên việc sửa đổi một dịch vụ Web để hỗ trợ HTTPS là khá dễ dàng.
- Mã hóa XML và chữ ký XML: Các tiêu chuẩn bảo mật XML được phát triển bởi W3C cho phép nội dung XML được ký và mã hóa. Vì tất cả các thông điệp SOAP được viết bằng XML, các nhà phát triển dịch vụ web có thể ký hoặc mã hóa bất kỳ phần nào của thông điệp SOAP bằng các tiêu chuẩn này, nhưng không có cơ chế tiêu chuẩn nào để thông báo cho người nhận về cách các tiêu chuẩn này được áp dụng cho thông điệp.
- WS-Security được phát triển để cung cấp các phần mở rộng SOAP xác định các cơ chế sử dụng Mã hóa XML và Chữ ký XML để bảo mật các thông điệp SOAP.

2.1.2 Bảo mật tài nguyên

Khi tài nguyên được cung cấp công khai, điều quan trọng là phải đảm bảo rằng chúng được bảo vệ đầy đủ. Thông thường, các dịch vụ web chỉ dành cho những người yêu cầu được ủy quyền, yêu cầu các cơ chế kiểm soát truy cập. Để thực hiện kiểm soát truy cập, các dịch vụ web cần xác định và xác thực lẫn nhau. Một số phương thức khác nhau có sẵn, bao gồm xác thực lớp vận chuyển, xác thực mã thông báo thông qua đặc tả WS-Security bằng cách sử dụng các xác nhận SAML hoặc các mã thông báo khác và tiêu đề xác thực SOAP. Việc ủy quyền cho các dịch vụ Web thường được thực hiện thông qua triển khai tùy chỉnh, nhưng XACML là một tiêu chuẩn OASIS có sẵn để thực hiện các quyết định ủy quyền, loại bỏ thời gian và chi phí liên quan đến việc phát triển và thử nghiệm một giải pháp tùy chỉnh.

Những thách thức phải đối mặt trong việc bảo vệ tài nguyên bên ngoài đơn giản là cung cấp các cơ chế kiểm soát truy cập. Mục tiêu của kẻ tấn công có thể không chỉ đơn giản là truy cập dịch vụ Web. Thay vào đó, các mục tiêu của kẻ tấn công có thể bao gồm phá vỡ dịch vụ, hoạt động như một kẻ trung gian, nghe lén dịch vụ, mạo danh dịch vụ hoặc thậm chí sử dụng các điểm yếu trong triển khai dịch vụ để kiểm soát nền tảng máy chủ.

2.1.3 Đàm phán hợp đồng

Một trong những mục tiêu chính của SOA là tạo thuận lợi cho việc tự động hóa các quy trình kinh doanh bằng cách cho phép các dịch vụ tự động khám phá lẫn nhau và ngay lập tức tận dụng các chức năng được cung cấp. Để tạo thuận lợi cho các giao dịch kinh doanh, các dịch vụ web cần có khả năng tạo, thực thi và tuân thủ các hợp đồng giữa các tổ chức. Ví dụ: dịch vụ tín dụng dựa trên các

dịch vụ Web của tổ chức khác. Hợp đồng giữa hai tổ chức đảm bảo rằng tất cả các dịch vụ Web sẽ hoạt động như mong đợi và thông tin được truyền giữa các tổ chức sẽ được bảo mật đúng cách. Trong nhiều tình huống, các hợp đồng này được các tổ chức đàm phán và thỏa thuận trước khi thực hiện có thể bắt đầu. Lý tưởng nhất là các dịch vụ web sẽ có thể đàm phán và thỏa thuận các hợp đồng đó bằng điện tử, ngay sau khi phát hiện ra trong thời gian chạy để tận dụng chức năng mới ngay lập tức. Đàm phán các hợp đồng như vậy điện tử mở ra một số phân nhánh pháp lý tiềm năng cho các tổ chức liên quan. Do đó, thay vì lý tưởng này, nhiều SOA dựa vào một hợp đồng ngầm được cung cấp bởi giao diện WSDL của một dịch vụ Web và hy vọng nó sẽ hoạt động như quảng cáo.

Bộ tiêu chuẩn ebXML cung cấp các công cụ để đàm phán các quy trình và hợp đồng kinh doanh sử dụng các dịch vụ Web. Tuy nhiên, ebXML đã được phát triển để thay thế cho Trao đổi dữ liệu điện tử (EDI) và do đó, thường được coi là quá phức tạp để sử dụng cho các dịch vụ Web thông thường. Vì các dịch vụ Web ebXML dựa trên SOAP, các phần của tiêu chuẩn ebXML có thể được áp dụng riêng cho các tổ chức nhỏ. Thông thường, giao diện WSDL hoặc mục đăng ký của một dịch vụ Web riêng lẻ có thể được coi là một hợp đồng ngầm giữa các dịch vụ, nhưng không có tiêu chuẩn nào hỗ trợ cho việc thực thi các hợp đồng ngầm.

Các dịch vụ web có thể có các yêu cầu QoS hoặc QoP cụ thể. Ví dụ: dịch vụ tín dụng có thể yêu cầu một số thông tin nhất định được mã hóa và ký bằng WS-Security, trong khi dịch vụ yêu cầu có thể yêu cầu phản hồi được bảo đảm thông qua tin nhắn đáng tin cậy. Bộ tiêu chuẩn ebXML cung cấp hỗ trợ cho các thuộc tính bảo mật trong hợp đồng, nhưng nó không hỗ trợ đầy đủ các thuộc tính bảo mật tự động đàm phán. Tiêu chuẩn WS-Choreography cung cấp một số hỗ trợ để đàm phán các yêu cầu bảo mật. Một lĩnh vực nghiên cứu đầy hứa hẹn là dịch vụ Semantic Web. Sử dụng các công nghệ Web Ngữ nghĩa, các dịch vụ Web có thể tìm kiếm thông minh các dịch vụ Web khác với các thuộc tính cụ thể, bao gồm các thuộc tính bảo mật.

2.1.4 Quản lý tin cậy

Các tiêu chuẩn dịch vụ web vốn đã linh hoạt và đã cho phép một số mô hình kiến trúc phát triển: mô hình ủy thác được môi giới, mô hình tin cậy theo cặp, mô hình ủy thác liên kết và mô hình phòng thủ vành đai. Mặc dù các mô hình này sử dụng thuật ngữ tin cậy, nhưng chúng bị giới hạn để có thể tin tưởng vào danh tính của dịch vụ. Có thể thiết lập danh tính của dịch vụ web không có nghĩa là bản thân dịch vụ đó đáng tin cậy. Luôn có khả năng một dịch vụ Web đã đi vào trạng thái sai lầm hoặc đã bị xâm phạm.

Trong bài báo năm 1996, McKnight và Chervany định nghĩa độ tin tưởng là "mức độ mà một người tin (và cảm thấy tin tưởng) rằng người kia đáng tin trong tình huống [3]". Dựa trên định nghĩa này, việc xác thực danh tính của dịch vụ Web có thể không đủ khi xác định có tin tưởng dịch vụ Web từ xa hay không. Khi mối quan hệ tin cậy trải rộng trên nhiều tổ chức, các yêu cầu đối với các dịch vụ Web riêng lẻ sẽ khác nhau. Vì lý do này, bất kể nhà cung cấp có phải là một thực thể đáng tin cậy về mặt nhận dạng hay không, người yêu cầu không nên cho rằng họ sẽ không gửi nội dung sai lệch hoặc có khả năng gây hại để đáp ứng yêu cầu của người yêu cầu. Tương tự, vì các nhà cung cấp lắng nghe (như một máy chủ) cho các yêu cầu từ nhiều người yêu cầu khác nhau, họ không nên cho rằng nội dung sai lệch hoặc độc hại sẽ không được gửi thay cho các yêu cầu hợp lệ. Tuy nhiên, xác định và xác thực các dịch vụ Web là một bước thiết yếu để thiết lập lòng tin. Mỗi mô hình ủy thác cung cấp các lợi ích và nhược điểm khác nhau, cho phép niềm tin được hỗ trợ trong nhiều môi trường khác nhau.

Mô hình tin cậy theo cặp là đơn giản nhất trong tất cả các kiến trúc tin cậy, nhưng có khả năng mở rộng ít nhất. Trong kiến trúc cặp đôi, mỗi dịch vụ Web được cung cấp trong cấu hình, thông tin bảo mật của tất cả các dịch vụ web khác sẽ được tương tác để các giao dịch và dịch vụ web đó có thể được tin cậy. Cách tiếp cận này giúp loại bỏ sự cần thiết của các nhà phát triển để phối hợp với các thực thể khác, nhưng nó tạo ra một kiến trúc bảo mật không đồng nhất và không đồng nhất vì việc thêm một dịch vụ Web mới sẽ yêu cầu thêm thông tin mới vào tất cả các dịch vụ hiện có mà nó có thể tương tác. Khi SOA trở nên rộng lớn và năng động, việc thêm một dịch vụ mới có thể trở nên tốn thời gian và tài nguyên.

Trong mô hình tin cậy được môi giới, một bên thứ ba độc lập hoạt động như một bên thứ ba đáng tin cậy (TTP) cho dịch vụ Web. Giao diện người yêu cầu và nhà cung cấp với bên thứ ba cho một loạt các dịch vụ bảo mật. Không giống như mô hình tin cậy theo cặp, các dịch vụ web sử dụng mô hình ủy thác được môi giới cần phải được thiết kế với giao diện của nhà môi giới, để thông tin nhận dạng có thể được dịch vụ Web truy xuất chính xác. Cách tiếp cận này giúp giảm bớt sự phân phối thông tin nhận dạng giữa các dịch vụ Web; mỗi dịch vụ Web sẽ chỉ cần xác minh danh tính của nhà môi giới tin cậy thay vì danh tính của tất cả các dịch vụ Web trong SOA.

Một mô hình tin cậy liên kết cho phép các dịch vụ web từ các tổ chức khác nhau tương tác liền mạch với nhau thông qua các cơ chế liên kết khác nhau. Nó xây dựng dựa trên cả hai mô hình tin cậy được môi giới và cặp đôi bằng cách cho phép các tổ chức sử dụng các nhà môi giới ủy thác trung tâm của riêng họ trong khi dựa vào niềm tin theo cặp hoặc tín nhiệm được môi giới giữa các tổ chức. Mỗi tổ chức muốn liên kết phải thực hiện theo các quy trình và giao thức kinh doanh phức tạp, nhưng kết quả cuối cùng cho phép các dịch vụ Web của mỗi tổ chức tương tác với một vài hoặc không có thay đổi nào đối với cấu hình ban đầu của họ.

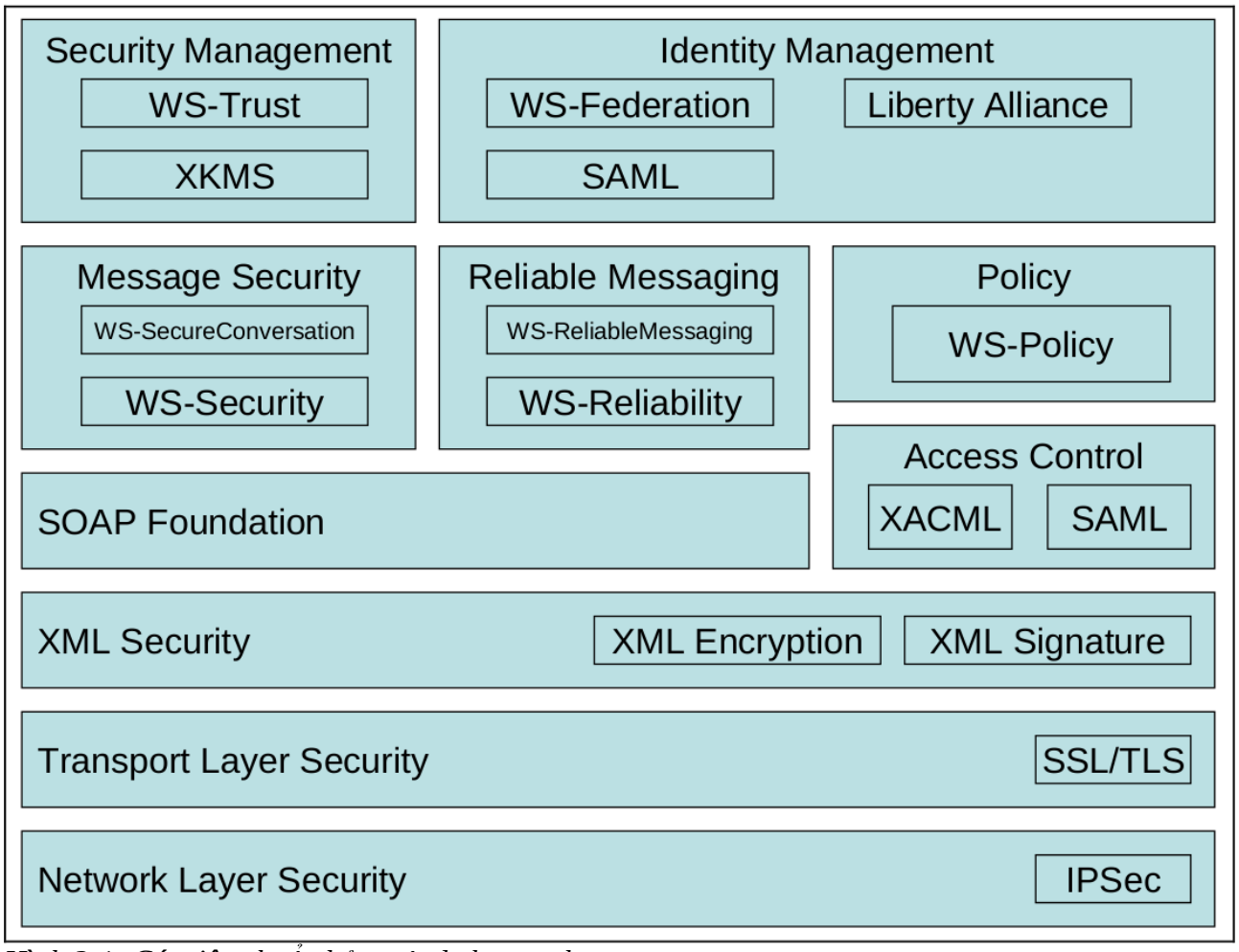
Một kiến trúc dịch vụ Web thường được sử dụng là chiến lược phòng thủ vành đai. Các thiết bị được gọi là cổng XML được đặt giữa nhà cung cấp và người yêu cầu. Cổng XML hoạt động như một proxy cho dịch vụ Web bằng cách thực hiện chức năng liên quan đến bảo mật tại vị trí của nó. Mặc dù các cổng XML là các công cụ hữu ích trong chiến lược bảo mật của tổ chức, nhưng chúng không phải là thuốc chữa bách bệnh. Nếu kẻ tấn công bỏ qua cổng XML, tất cả các dịch vụ Web nội bộ sẽ dễ bị tấn công. Các dịch vụ Web nội bộ phải được thiết kế, phát triển và định cấu hình an toàn.

2.1.4 Các yêu cầu đối với bảo mật phần mềm

Tất cả các phần mềm, bao gồm các dịch vụ Web, cần đáp ứng các yêu cầu về hiệu suất, chi phí, khả năng sử dụng và bảo mật. Ví dụ về các yêu cầu có thể có đối với phần mềm bảo mật là khả năng dự đoán, tính chính xác và tính khả dụng.

2.2 Ngăn xếp các tiêu chuẩn bảo mật web service

Các cộng đồng tiêu chuẩn mở đã tạo ra các dịch vụ Web đã phát triển một số tiêu chuẩn bảo mật cho các dịch vụ Web. Hình 2-1 minh họa một mô hình tham chiếu nổi tiếng cho các tiêu chuẩn bảo mật dịch vụ Web. Mô hình tham chiếu này ánh xạ các tiêu chuẩn khác nhau đến các lớp chức



Hình 2-1: Các tiêu chuẩn bảo mật dịch vụ web

năng khác nhau của việc triển khai dịch vụ Web điển hình. Các lớp này được mô hình hóa theo Mô hình tham chiếu OSI nhưng không được hiểu là phân cấp chặt chẽ.

Các tiêu chuẩn tại các lớp bảo mật mạng, truyền tải và XML được sử dụng để bảo mật các thông điệp khi chúng được truyền qua mạng. Các tiêu chuẩn bảo mật IPsec, SSL / TLS (Secure Sockets Layer/Transport Layer Security), Mã hóa XML và Chữ ký XML mỗi hoạt động trên các thông điệp SOAP ở một cấp độ khác nhau.

Trên lớp Bảo mật XML, có hai loại tiêu chuẩn: tiêu chuẩn được xây dựng dựa trên tiêu chuẩn SOAP và tiêu chuẩn độc lập. Các tiêu chuẩn bảo mật thư WS-Security và WS-SecureConversation xác định cách sử dụng Chữ ký XML, Mã hóa XML và thông tin đăng nhập để bảo mật SOAP ở lớp thông điệp trong khi các tiêu chuẩn truyền tin đáng tin cậy xác định các giao thức và cấu trúc cần thiết để đảm bảo rằng các thông điệp sẽ được nhận. Các tiêu chuẩn kiểm soát truy cập không phải là duy nhất cho các dịch vụ Web; XACML có thể xác định chính sách truy cập cho bất kỳ hệ thống nào và SAML có thể được sử dụng để xác định các xác nhận trong bất kỳ môi trường nào. WS-Policy của lớp chính sách xác định ngữ pháp để truyền đạt các yêu cầu chính sách của dịch vụ Web.

Thông số kỹ thuật quản lý bảo mật xác định các dịch vụ Web khác để quản lý thông tin xác thực, chẳng hạn như chứng chỉ PKI trong SOA. Các tiêu chuẩn quản lý danh tính tận dụng các tiêu chuẩn kiểm soát truy cập, tiêu chuẩn chính sách và tiêu chuẩn SOAP để cung cấp các dịch vụ phân phối và quản lý danh tính và thông tin người dùng trong SOA.

Chương 3: Thực nghiệm bảo mật cho web service

REST (Representational State Transfer) là phong cách kiến trúc đang thúc đẩy phát triển web và ứng dụng di động hiện đại. Trên thực tế, phát triển và tương tác với RESTful Web Services là một kỹ năng cần thiết trong bất kỳ công việc phát triển phần mềm hiện đại nào. Đôi khi, bạn phải tương tác với một API hiện có và trong các trường hợp khác, bạn phải thiết kế API RESTful từ đầu và làm cho nó hoạt động với JSON (JavaScript Object Notation).

Khi thiết kế một RESTful Web service, có một số điều cần tuân thủ như:

- Validation: Thẩm định tất cả các đầu vào trên server. Bảo vệ server của bạn chống lại các tấn công SQL hay NoSQL injection
- Session Based Authentication (xác thực dựa trên phiên): Sử dụng xác thực dựa trên phiên để xác thực người dùng bất cứ khi nào yêu cầu được thực hiện đối với phương thức Dịch vụ web.
- Không có dữ liệu nhạy cảm trong URL: Không có dữ liệu nhạy cảm trong URL - Không bao giờ sử dụng tên người dùng, mật khẩu hoặc mã thông báo phiên trong một URL, các giá trị này sẽ được chuyển đến Dịch vụ web thông qua phương thức POST.
- Hạn chế trong thực thi phương thức: Cho phép hạn chế sử dụng các phương thức như phương thức GET, POST, DELETE. Phương thức GET sẽ không thể xóa dữ liệu.
- Xác thực XML / JSON không đúng định dạng - Kiểm tra đầu vào được định dạng tốt được truyền cho phương thức dịch vụ web.
- Ném các thông báo lỗi chung - Phương thức dịch vụ web nên sử dụng các thông báo lỗi HTTP như 403 để hiển thị cấm truy cập, v.v.

Python là một trong những ngôn ngữ lập trình phổ biến nhất. Nó là mã nguồn mở, đa nền tảng và bạn có thể sử dụng nó để phát triển bất kỳ loại ứng dụng nào, từ các trang web đến các ứng dụng tính toán khoa học cực kỳ phức tạp. Các nhà cung cấp điện toán đám mây quan trọng và phổ biến nhất giúp bạn dễ dàng làm việc với Python và các framework Web liên quan của nó. Do đó, Python là một lựa chọn lý tưởng để phát triển Dịch vụ Web RESTful.

Có một số framework cho phép phát triển các RESTful Web service, với mục đích thực nghiệm một kịch bản bảo mật cho web service đơn giản, nhóm chúng em chọn flask như là một framework chính cho việc thử nghiệm.

Môi trường cài đặt thực hành:

Hệ điều hành – Ubuntu 18.04 LTS

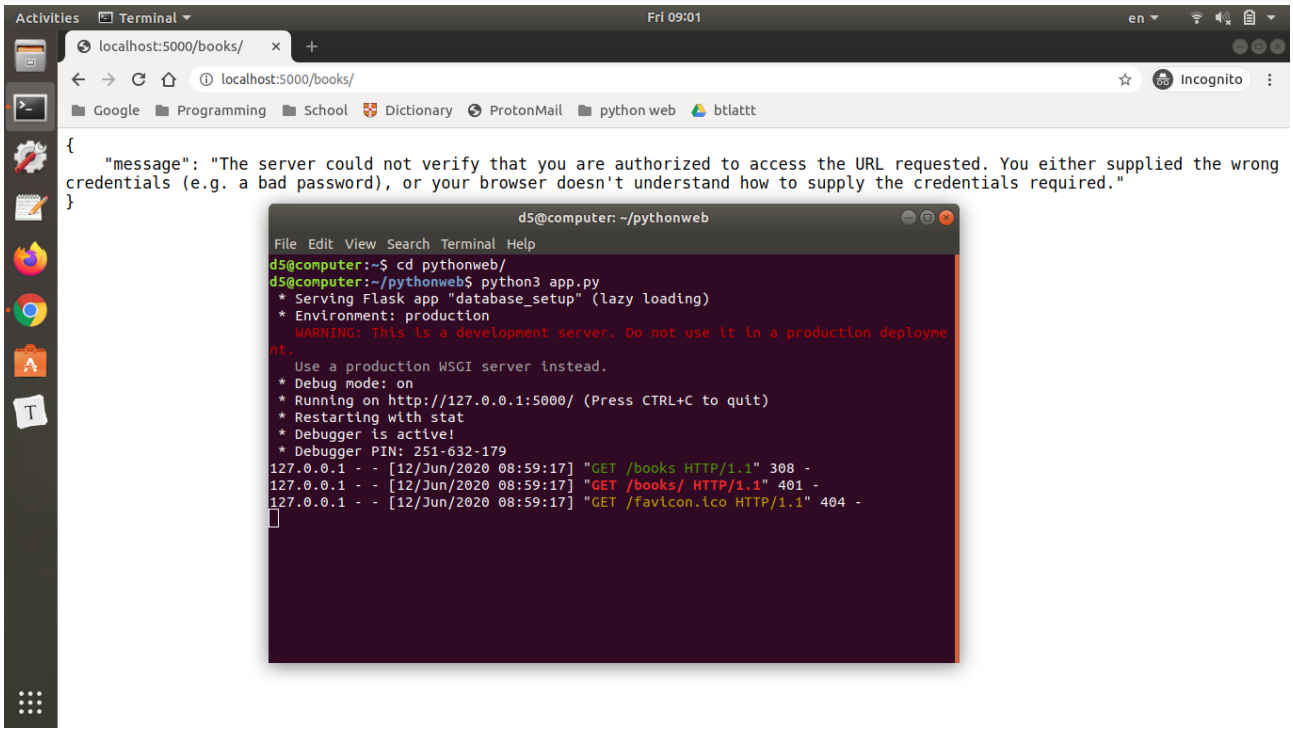
Các framework, thư viện sử dụng : flask, flask-restful, SQLAlchemy

Trình duyệt web : Google Chrome

Ứng dụng : Quản lý cơ sở dữ liệu sách, mỗi sách gồm có 4 thông tin : định danh (bid), tiêu đề (title), tác giả (author), thể loại (genre)

Các thao tác có thể thực hiện : Đăng nhập, truy cập CSDL và thêm một sách mới (tương ứng GET và POST của HTTP method)

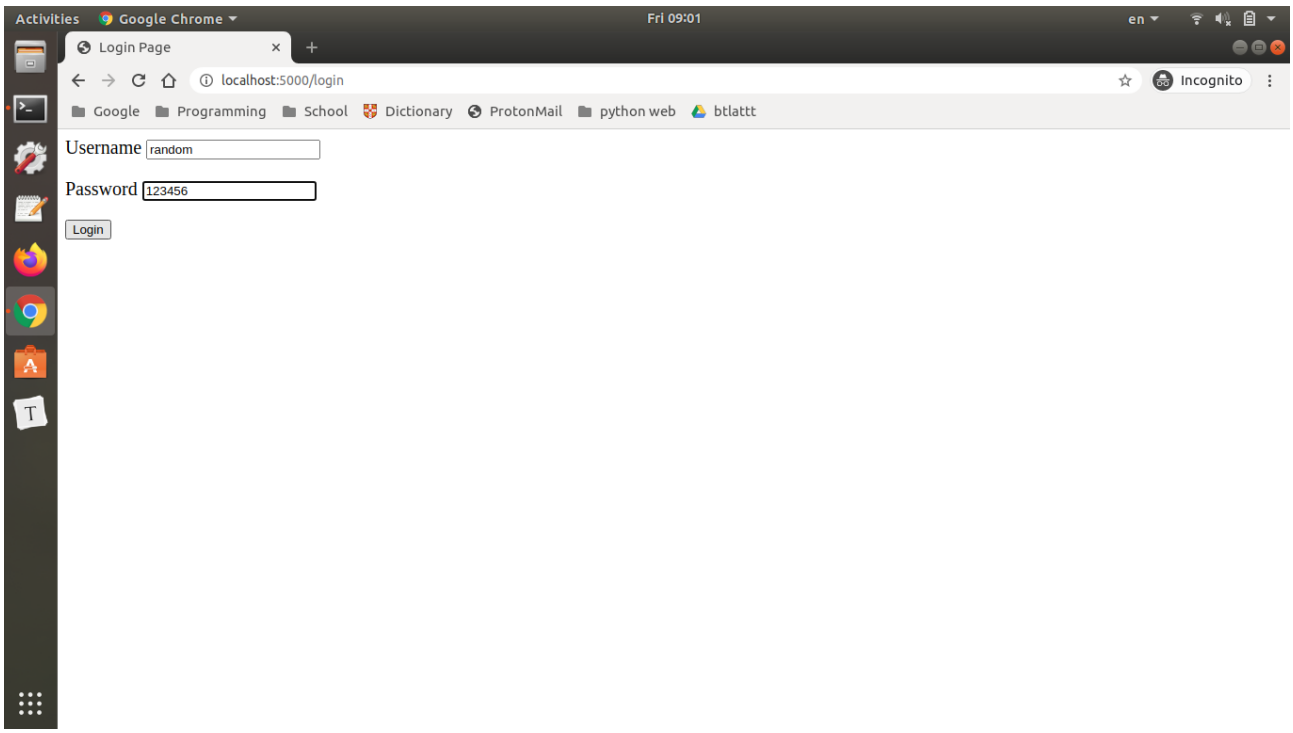
- Truy cập luôn vào database

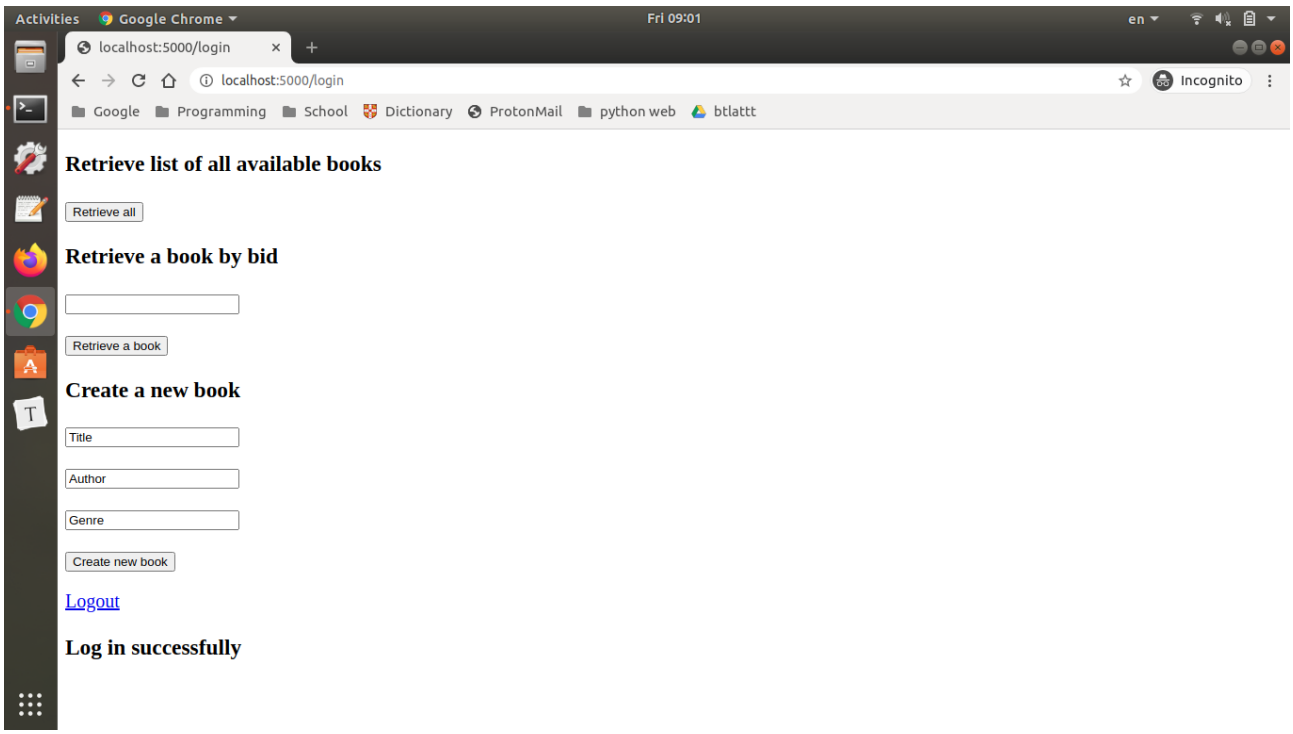


Server yêu cầu phải đăng nhập, nên hiển nhiên không thể truy cập được.

Từ màn hình ta thấy server đã trả về HTTP status code 401, tức là lỗi Unauthorized

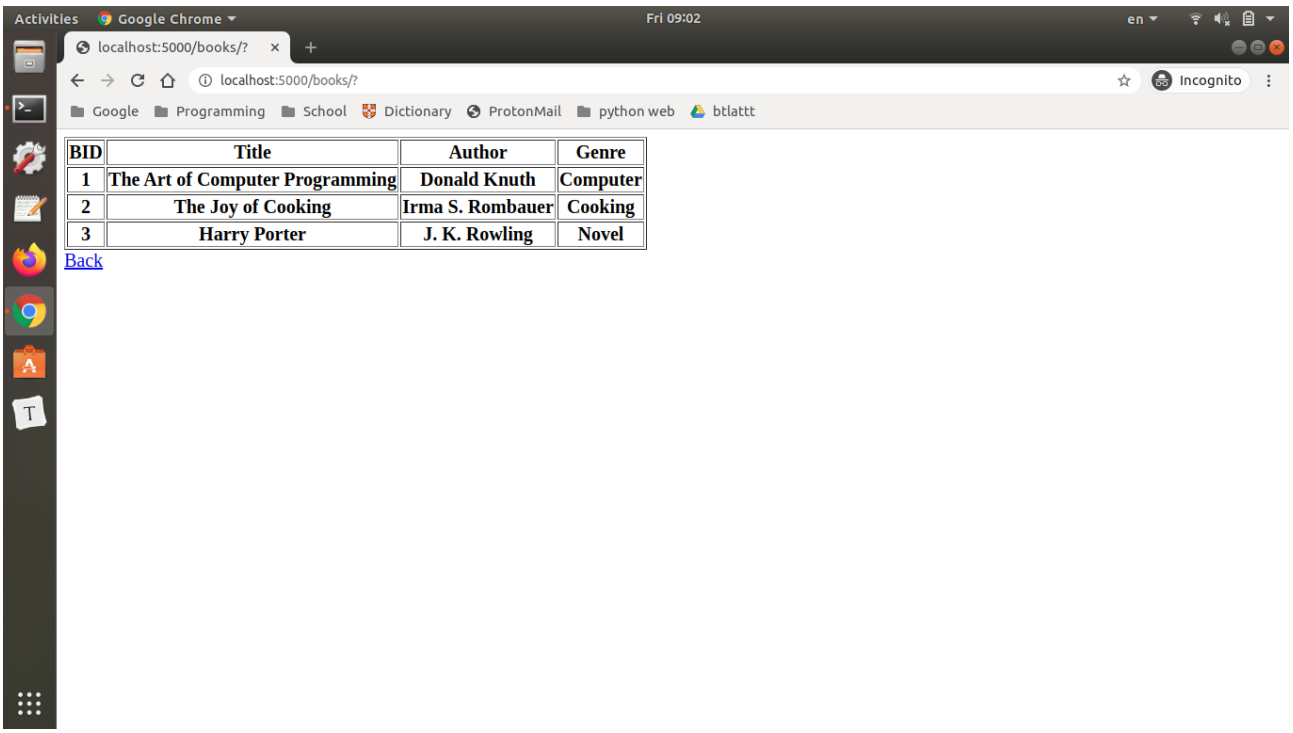
- Đăng nhập



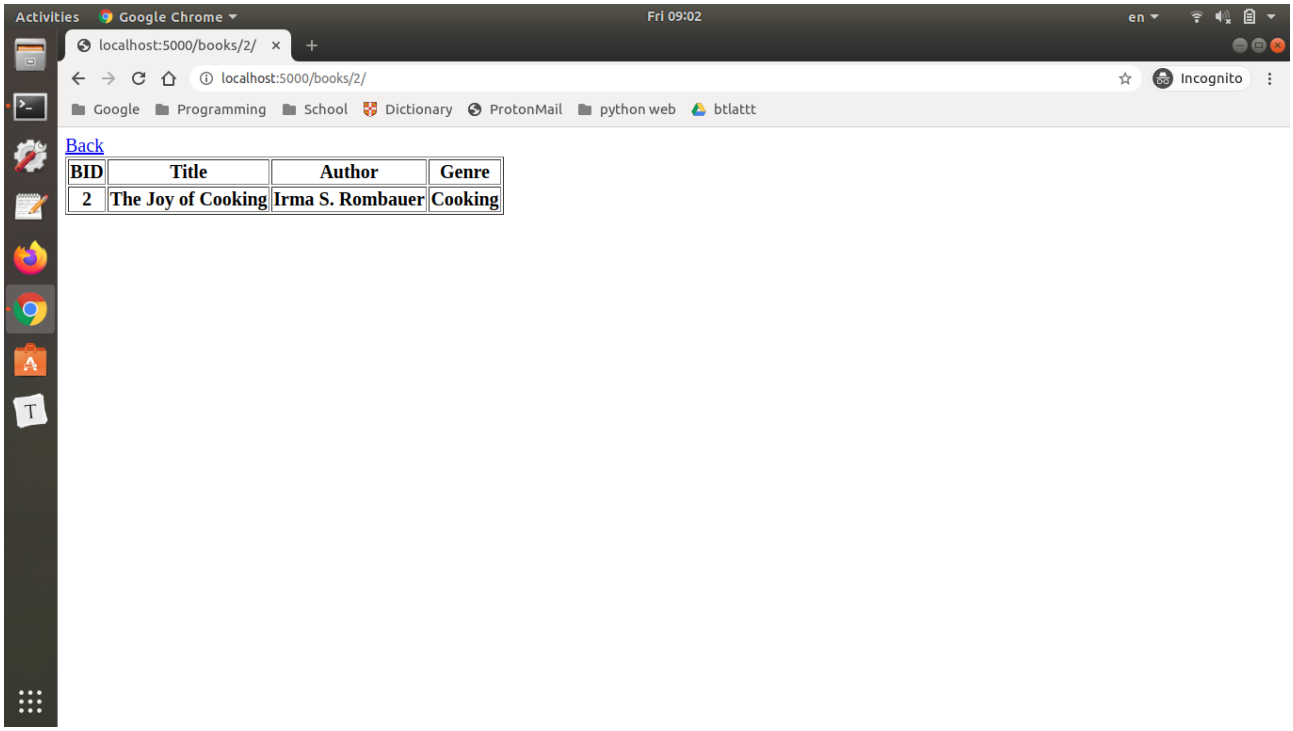


Màn hình hiển thị các chức năng hiện có.

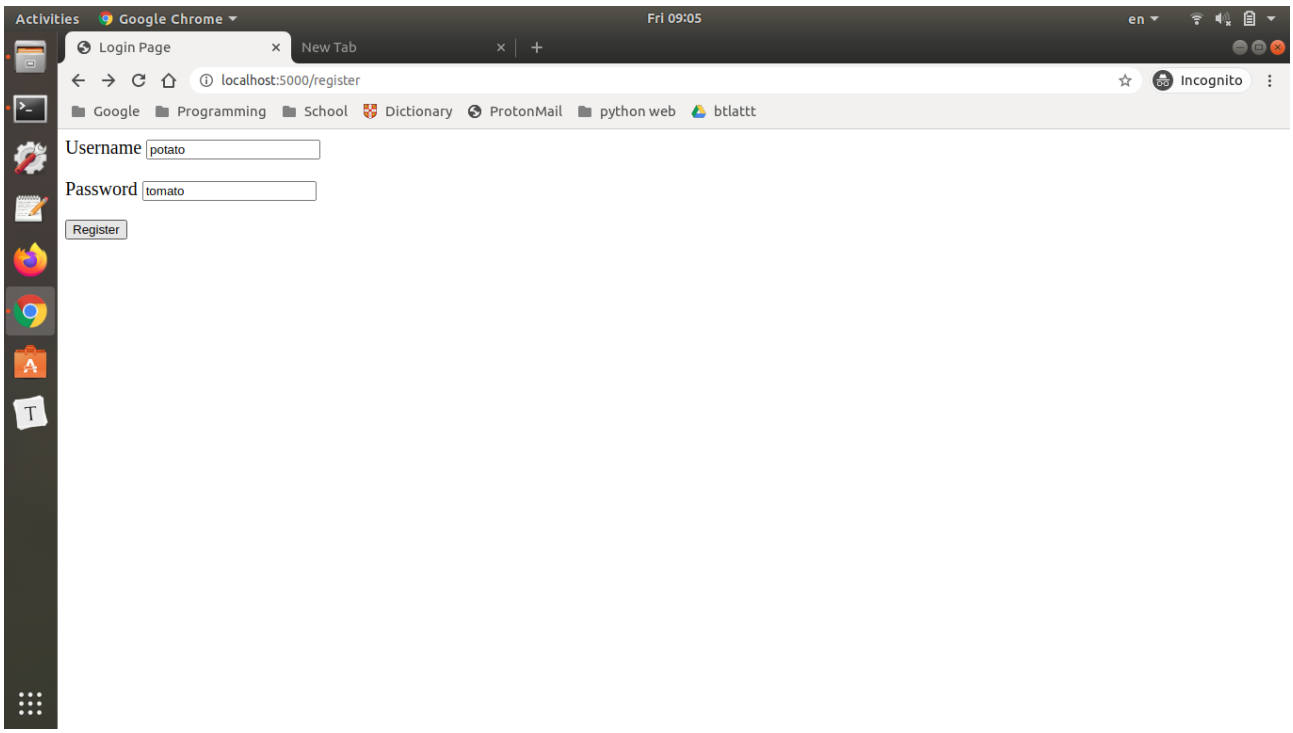
- Truy cập lại vào đường dẫn cũ để lấy danh sách toàn bộ các sách có trong database



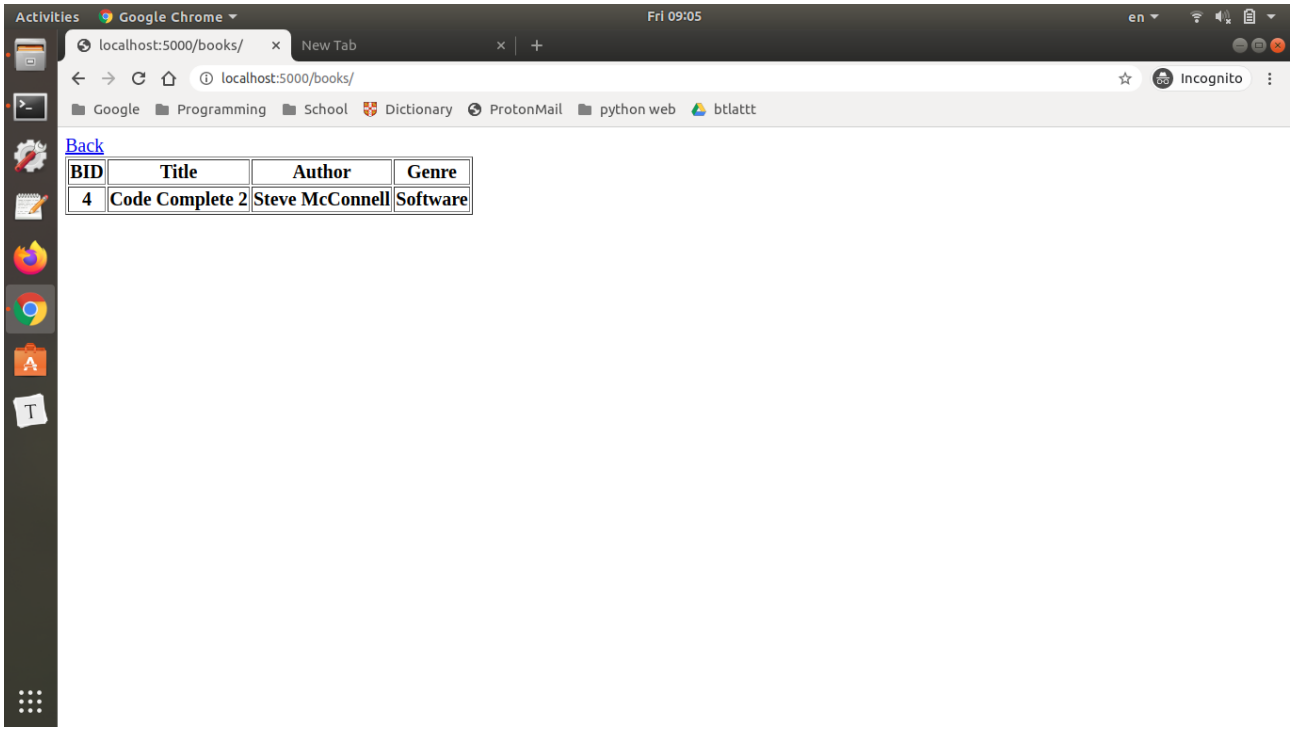
- Có thể lấy dữ liệu từng sách một, bằng cách truyền định danh vào



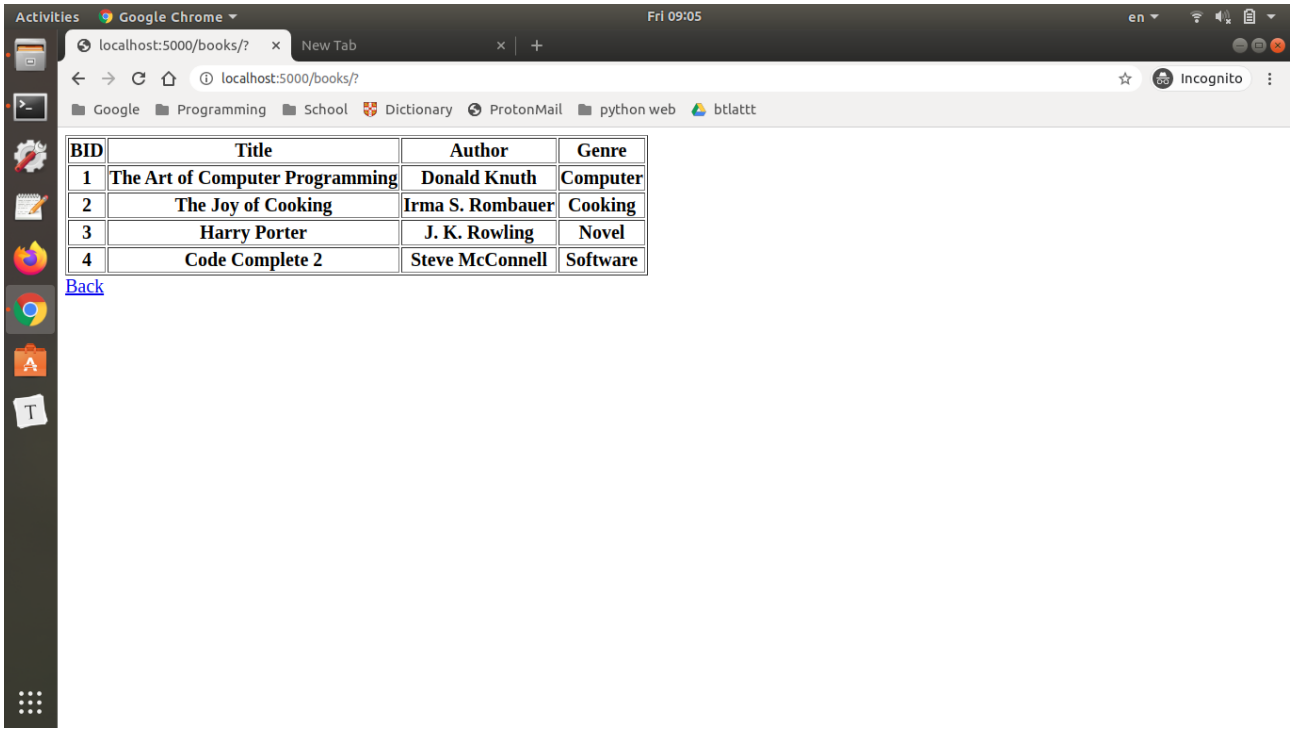
- Tạo một tài khoản mới



- Sau đó thêm một cuốn sách mới vào database

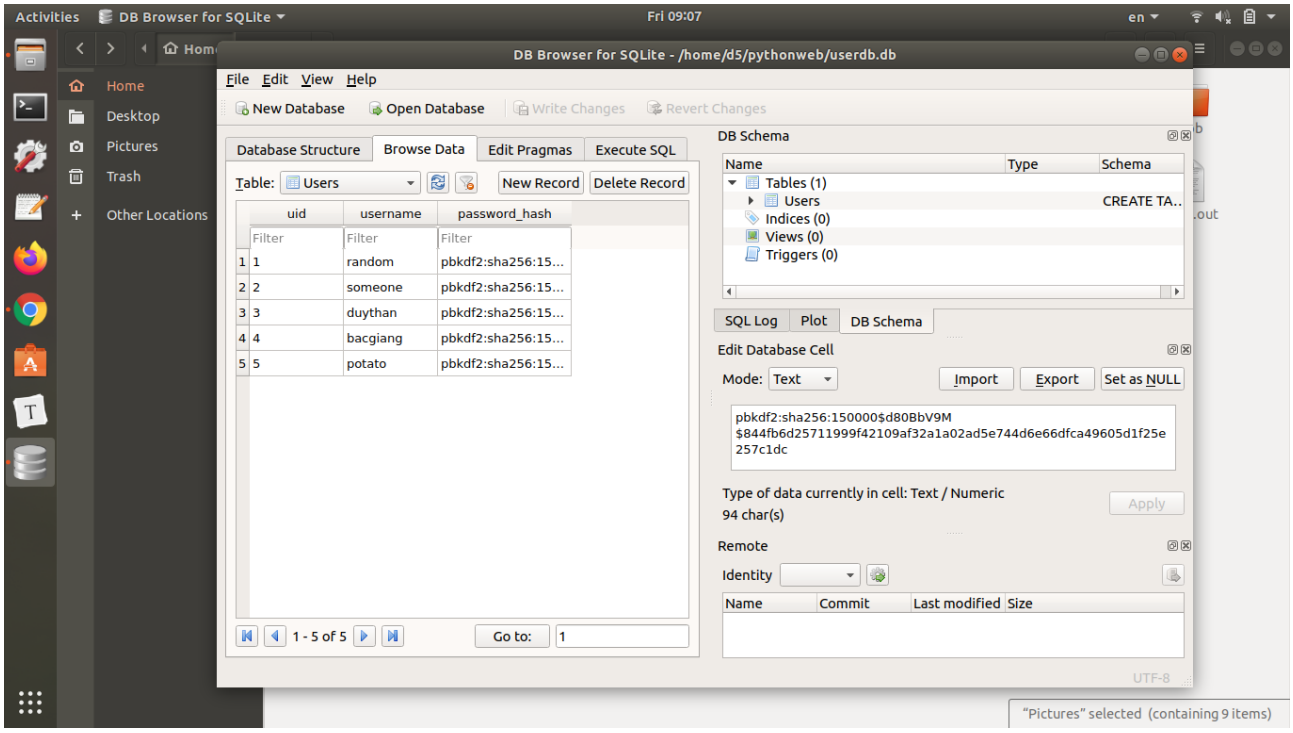


- Truy cập lại đường dẫn lấy toàn bộ sách



XÁC THỰC NGƯỜI DÙNG :

- Cơ sở dữ liệu của người dùng



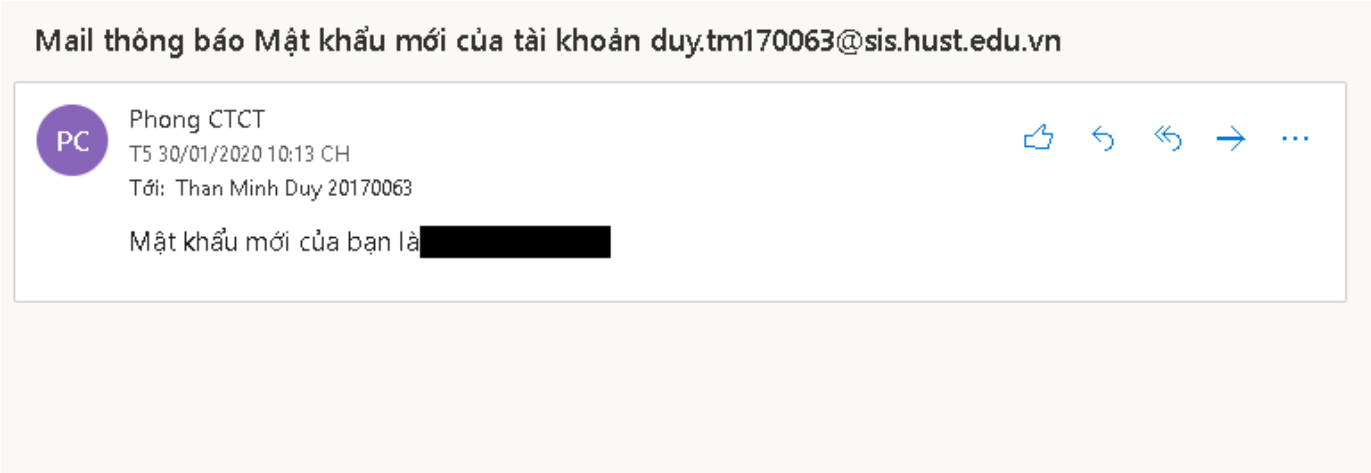
Từ màn hình này, ta có thể thấy:

1. Tất cả mật khẩu đã được mã hóa
2. Thuật toán được sử dụng là SHA256, với 150000 vòng lặp, và giá trị được lưu gồm có cả salt value (xâu ngẫu nhiên được thêm vào để tăng độ khó giải mã), và giá trị băm

Tuy nhiên, hiện nay vẫn còn một số hệ thống lưu mật khẩu của người dùng dưới dạng bản rõ, không mã hóa, điều này tiềm ẩn rất nhiều nguy cơ bảo mật, với hậu quả chung là dữ liệu cá nhân của người dùng bị lộ:

1. Server bị mất quyền kiểm soát, hay database của người dùng bị đọc
2. Khi admin đang chỉnh sửa database, ai đó có thể "nhòm" qua và nhớ tài khoản người dùng
3. Người quản lý server (admin) lạm dụng quyền của mình và truy cập trái phép vào tài khoản người dùng

Ví dụ về một hệ thống vẫn còn sử dụng cách thức này:



(Từ **ctsv.hust.edu.vn,** thông báo khi sinh viên mới đăng ký tài khoản, hoặc đổi mật khẩu)

Tài liệu tham khảo

[1] Bài giảng nhập môn An toàn Thông tin, PGS.TS Nguyễn Linh Giang, ĐH Bách Khoa Hà Nội.

[2] A Study on the Security Mechanism for Web Services, Kou Hongzhao.