TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

-----📖-----

ĐỀ TÀI

BẢO MẬT WEBSERVICES

Giảng viên: Nguyễn Linh Giang

Thực hiện: Nhóm – 11

Sinh viên: Nguyễn Văn Hùng 20173152

Nguyễn Thế Quang 20173324

Nguyễn Quang Linh 20173229

Hà Nội, tháng 06 năm 2020

MỤC LỤC

CHƯƠNG 1: MỞ ĐẦU

1.1. Đặt vấn đề

Ngày nay, cùng với sự phát triển của Internet, Web Service cũng trở thành một kỹ thuật dùng để liên kết và tương tác giữa các ứng dụng trên các máy tính khác nhau thông qua môi trường Internet. Ngày càng có nhiều nhà cung cấp dịch vụ muốn đưa các dịch vụ ra công cộng và vấn đề lớn nhất mà các nhà cung cấp đang phải đối mặt chính là bảo mật cho Web Service. Việc đảm bảo an toàn cho Web Service là một vấn đề đặc biệt quan trọng, nhất là đối với những dịch vụ liên quan tài chính, thị trường chứng khoán và thương mại điện tử. Vấn đề bài toán đặt ra là làm thế nào để những thông tin, dữ liệu được trao đổi một cách an toàn mà không bị tấn công.

Để giải quyết vấn đề bảo mật trên đường truyền, có rất nhiều kĩ thuật bảo mật đang được các doanh nghiệp nhỏ và vừa triển khai trên hệ thống mạng nhằm đảm bảo thật tốt vấn đề an ninh khi giao dịch trên Internet. Trong đề tài này, chúng ta sẽ cùng tìm hiểu về các kĩ thuật đó.

1.2. Nội dung bài toán

Với những yêu cầu mà thực tế đặt ra, trong bài tập lớn này chúng em sẽ tìm hiểu và làm rõ các kỹ thuật bảo mật Web Service hiện có. Cũng trong đề tài này sẽ thực hiện xây dựng một ví dụ demo bảo mật web service đơn giản sử dụng kĩ thuật trên.

CHƯƠNG 2: GIỚI THIỆU KIẾN TRÚC HƯỚNG DỊCH VỤ

2.1.Thực trạng hiện tại

Phần mềm ngày nay đang ngày càng trở nên phức tạp và dường như đang vượt khỏi khả năng kiểm soát của các mô hình phát triển phần mềm hiện có. Hàng chục năm qua,nhiều kiến trúc phần mềm đã được xây dựng và triển khai nhằm giải quyết các vấn đề này.Thế nhưng độ phức tạp phần mềm vẫn cứ tiếp tục tăng và dường như đã trở nên vượt quá khả năng xử lý của các kiến trúc truyền thống. Nguyên nhân khiến cho độ phức tạp của các hệ thống phần mềm không ngừng tăng cao như thế là do sự xuất hiện của nhiều công nghệ mới tạo nên môi trường không đồng nhất, trong khi nhu cầu về trao đổi, chia sẻ, tương tác giữa các hệ thống không thể đáp ứng được trong một môi trường như vậy. Một nguyên nhân khác cũng góp phần dẫn đến tình trạng khó khăn như thế chính là vấn đề lập trình dư thừa và không thể tái sử dụng. Những vấn đề trước chưa giải quyết, mà nay các tổ chức lại phải đối mặt với những thách thức mới: đáp ứng nhanh chóng các sự thay đổi về thiết bị, giảm chi phí phát triển, tăng tính tương thích và khả năng tái sử dụng,... Tất cả đã tạo nên một áp lực nặng nề đối với các nhà phát triển phần mềm.

2.2. Khái niệm SOA

SOA (Service-Oriented Architecture) - kiến trúc hướng dịch vụ. Hiểu một cách cơ bản, SOA là tập hợp các dịch vụ kết nối "mềm dẻo" với nhau, có giao tiếp được định nghĩa rõ ràng và độc lập với nền tảng hệ thống, và có thể tái sử dụng. SOA là cấp độ cao hơn của phát triển ứng dụng, chú trọng đến quy trình nghiệp vụ và dùng giao tiếp chuẩn để giúp che đi sự phức tạp kỹ thuật bên dưới. Nói cách khác, SOA là:

- Một kiểu kiến trúc phần mềm gồm nhiều thành phần độc lập được thể hiện thành những dịch vụ (service), mỗi dịch vụ thực hiện quy trình nghiệp vụ nào đó của doanh nghiệp.

- Các thành phần được nối kết qua cổng giao tiếp, có tính kế thừa các thành phần đang tồn tại, và sự tương tác giữa chúng không cần quan tâm đến việc chúng được phát triển trên nền tảng công nghệ nào. Điều này khiến hệ thống có thể mở rộng và tích hợp một cách dễ dàng.

Bản chất SOA chỉ đơn thuần là sự đáp ứng đối với một thách thức ngày càng lớn: đó là yêu cầu thực tế của doanh nghiệp thay đổi ngày càng nhanh, đến mức những cấu trúc ứng dụng kiểu truyền thống khó giải quyết nổi. SOA sẽ đáp ứng được yêu cầu đó, nó sẽ trợ giúp cho hoạt động doanh nghiệp có thể quản lý được (manageable), linh hoạt hơn và sẵn sàng thay đổi hơn. Một chuyên gia của IBM từng nói: "SOA được xây dựng để thay đổi (built to change), chứ không chỉ để tồn tại (not built to last)". Từ góc độ doanh nghiệp thì có thể coi SOA là một phương pháp để tái cấu trúc hạ tầng thông tin của doanh nghiệp.

Một số ưu điểm của việc phát triển ứng dụng hướng dịch vụ (SOA)

- Thứ nhất, tái sử dụng phần mềm. Nếu một dịch vụ có quy mô và kích thước phù hợp sau đó nó có thể được tái sử dụng cho lần kế tiếp. Điều này đồng nghĩa sẽ làm giảm công sức phát triển và chi phí về mặt tài chính cho cả hai phía: nhà phát triển phần mềm và các khách hàng (doanh nghiệp).
- Thứ hai, linh hoạt khi mở rộng, kết nối và tích hợp. Giả sử rằng các dịch vụ sẽ không được tái sử dụng, thì ta có thể đưa ra nhiều giá trị nếu ta làm cho hệ thống CNTT chỉnh sửa dễ dàng hơn.

CHƯƠNG 3: WEB SERVICE

3.1. Giới thiệu về Service

3.1.1. Khái niệm

Service là một ứng dụng với người dùng, một thao tác được thực hiện một hoặc nhiều lần trong một tiến trình và được thực hiện bởi một hay nhiều người.

Service là một hệ thống có khả năng nhận một hay nhiều yêu cầu xử lý và sau đó đáp ứng lại bằng cách trả về một hay nhiều kết quả. Quá trình nhận yêu cầu và trả kết quả về được thực hiện thông qua các giao diện đã được định nghĩa trước đó. Thông thường việc giao tiếp này được thực hiện trên các giao diện đã được chuẩn hóa và sử dụng rộng rãi.

Một hệ thống được thiết kế theo kiểu hướng Service là một hệ thống trong đó các chức năng của hệ thống được xây dựng dựa trên các service có độ kết dính thấp. Các service trong hệ thống giao tiếp với nhau thông qua việc gởi nhận các thông điệp.

3.1.2. Các đặc điểm chính của Service

Mỗi service được xây dựng dựa trên các giao diện chuẩn hóa đã được sử dụng rộng rãi. Chi tiết hiện thực của mỗi service sẽ không được thể hiện ra bên ngoài. Mỗi service chỉ công bố một số các giao diện của nó cho user có thể dùng để gởi các yêu cầu và nhận kết quả trả về.

Mỗi Service có tính độc lập cao, có thể được xây dựng và đưa vào sử dụng mà không phụ thuộc vào các service khác.

Trao đổi dữ liệu: các Service không truyền các class và type. Thay vào đó, các class và type sẽ được đặc tả hình thức.

3.2. Tổng quan về Web Service

3.2.1. Khái niệm

Web Service Web Service là một giao diện truy cập mạng đến các ứng dụng chức năng, được xây dựng từ việc sử dụng các công nghệ chuẩn Internet.

Thuật ngữ Web Service diễn tả một cách thức tích hợp các ứng dụng trên nền website lại với nhau bằng cách sử dụng các công nghệ XML, SOAP, WSDL,UDDI trên nền tảng các giao thức Internet với mục tiêu tích hợp ứng dụng và truyền thông điệp. XML được sử dụng để đánh dấu dữ liệu, SOAP được dùng để truyền dữ liệu, WSDL được sử dụng để mô tả các dịch vụ có sẵn và UDDI được sử dụng để liệt kê những dịch vụ nào hiện tại đang có sẵn để có thể sử dụng. Web Service cho phép các tổ chức có thể trao đổi dữ liệu với nhau mà không cần phải có kiến thức hiểu biết về hệ thống thông tin đứng sau Firewall kia.

Không giống như mô hình khách/chủ truyền thống,Web Service không cung cấp cho người dùng một giao diện đồ hoạ nào, Web Service đơn thuần chỉ là việc chia sẻ các dữ liệu logic và xử lý các dữ liệu đó thông qua một giao diện chương trình ứng dụng được cài đặt xuyên suốt trên mạng máy tính.

Web Service cho phép các ứng dụng khác nhau từ các nguồn khác nhau có thể giao tiếp với các ứng dụng khác mà không đòi hỏi nhiều thời gian lập trình, do tất cả các quá trình giao tiếp đều tuân theo định dạng XML, cho nên Web Service không bị phụ thuộc vào bất kì hệ điều hành hay ngôn ngữ lập trình nào.

Web Service cung cấp tính trừu tượng cho các giao diện chuẩn, cho nên sẽ không nảy sinh ra bất kì vấn đề gì trong quá trình tương tác. Web Service cho phép giao tiếp giữa các nền tảng khác nhau có thể hoạt động cùng nhau theo nguyên tắc tạo ra một nền tảng trung gian có liên quan.

⇨ Tóm lại: Web Service là:

♣ Làm việc xuyên qua tường lửa và proxy

♣ Sẵn sàng đối với các nền tảng máy trạm khác nhau

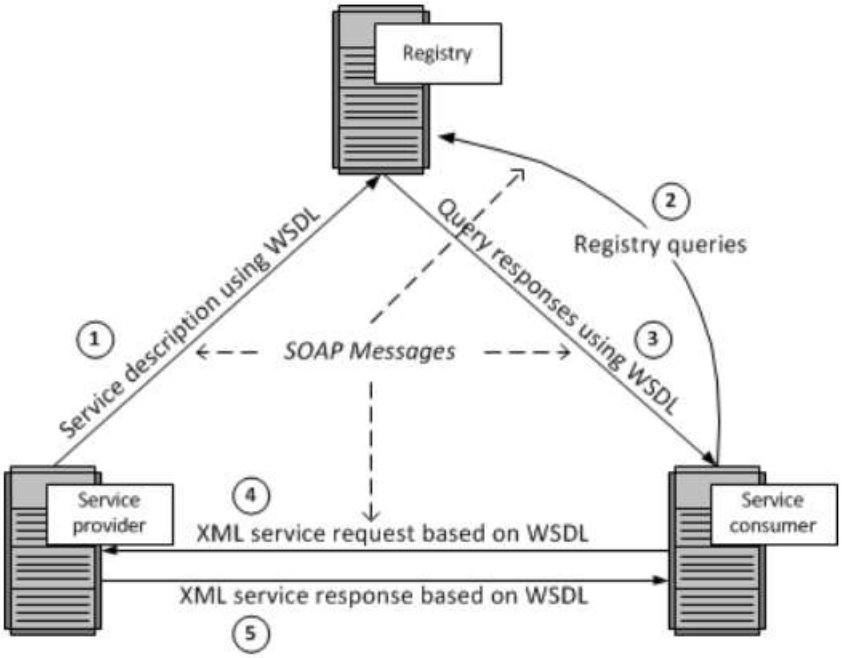
♣ Một dịch vụ phần mềm được trình bày trên web thông qua giao thức SOAP, được mô tả bằng một tệp WSDL và được đăng ký trên UDDI.

3.2.2. Đặc điểm Web Service

Cho phép khách/chủ tương tác với nhau cả trong môi trường khác nhau.

XML và HTTP là nền tảng kỹ thuật chính. Phần lớn kỹ thuật của Web Service được xây dựng là những dự án nguồn mở cho nên độc lập và vận hành được với nhau.

Web Service rất linh động: với UDDI và WSDL thì việc mô tả và phát triển Web Service có thể tự động hóa. Web Service bao gồm nhiều mô đun và có thể công bố trên mạng Internet. Web Service có thể chia sẻ và gọi thực hiện qua mạng và có độ an toàn riêng tư.



Mô hình Web Service

Nhà cung cấp đăng ký Web Service với UDDI.

Người sử dụng tìm kiếm dịch vụ trên UDDI qua một URL thích hợp.

UDDI trả lại một bản mô tả WSDL cho nhà cung cấp.

Người sử dụng triệu gọi dịch vụ bằng một cuộc gọi SOAP tới nhà cung cấp.

Nhà cung cấp trả lại kết quả của cuộc gọi SOAP cho người sử dụng.

3.3.2. Ưu điểm

Cho phép chương trình được viết bằng các ngôn ngữ khác nhau trên các nền tảng khác nhau giao tiếp được với nhau dựa trên một nền tảng tiêu chuẩn.

Thao tác đơn giản (chỉ dùng URL).

Làm việc với các giao thức chuẩn Web như XML, HTTP và TCP/IP.

Sự an toàn của máy chủ cơ sở dữ liệu luôn được bảo mật một cách chắc chắn.

Web Service làm giảm giá thành cho việc tích hợp các hệ thống khác nhau.

3.3.3. Nhược điểm

Phụ thuộc vào tốc độ đường truyền Internet.

Web Service thiếu cơ chế khôi phục đủ tin cậy để đảm bảo giao dịch được khôi phục lại trạng thái ban đầu trong trường hợp xảy ra sự cố.

Số lượng các ứng dụng cộng tác cùng hoạt động sẽ ảnh hưởng tới hiệu suất tối ưu của Web Service.

Tải trọng: ứng dụng Web Service là các ứng dụng sử dụng rất nhiều thông điệp. Khả năng bùng nổ số lượng giao dịch trao đổi sẽ làm hệ thống máy chủ ứng dụng và kiến trúc hạ tầng hệ thống thông tin của doanh nghiệp trở nên ngưng trệ.

Vì Web Service đòi hỏi kết nối thông qua khá nhiều máy chủ trung gian cho nên băng thông/tốc độ của hạ tầng mạng và các yêu tố liên quan tới hệ thống rõ ràng có vai trò quan trọng góp phần cải thiện hiệu năng của toàn bộ các ứng dụng Web Service.

3.4. Các thành phần chính của Web Service

Các thành phần chính của Web Service

XML được sử dụng để định dạng dữ liệu, SOAP được sử dụng trao đổi dữ liệu, WSDL được sử dụng để mô tả dịch vụ hiện có và UDDI được sử dụng để liệt kê các Web Service hiện có.

3.4.1. Giao thức giao vận HTTP

3.4.1.1. Giao thức HTTP

Tầng giao vận liên quan tới cơ chế sử dụng để chuyển yêu cầu dịch vụ và thông tin phản hồi từ phía nhà cung cấp dịch vụ tới người sử dụng dịch vụ. Có rất nhiều tiêu chuẩn sử dụng xung quanh WS, nhưng phổ biến nhất vẫn là giao thức HTTP.

Giao thức HTTP thường được sử dụng đối với yêu cầu dịch vụ và đáp ứng.

3.4.1.2. Ưu điểm

HTTP là nền tảng hạ tầng phổ biến và sẵn sàng nhất.

Giao thức HTTP hoàn toàn mở và khai triển trên rất nhiều loại hệ thống.

Hầu hết mọi tổ chức đều chấp nhận cho phép trao đổi thông tin dựa trên giao thức HTTP vượt qua tường lửa bảo vệ.

3.4.1.3. Nhược điểm

HTTP là một giao thức đơn giản và không có tính trạng thái, không được thiết kế đặc biệt cho mục đích vận chuyển dữ liệu của các ứng dụng.

Giao thức không hỗ trợ lưu trữ trạng thái.

Không phải là một giao thức đáng tin cậy phù hợp với nhu cầu truyền dữ liệu.

3.4.2. Giao thức truyền thông SOAP

3.4.2.1. Khái niệm

SOAP là gì:

♣ SOAP là giao thức truyền thông giữa các ứng dụng.

♣ SOAP được thiết kế để liên lạc qua Internet và làm việc qua tường lửa.

♣ SOAP độc lập nền tảng, độc lập ngôn ngữ.

♣ SOAP dựa trên XML, đơn giản và dễ mở rộng.

SOAP có đặc trưng:

♣ SOAP được thiết kế đơn giản và dễ mở rộng

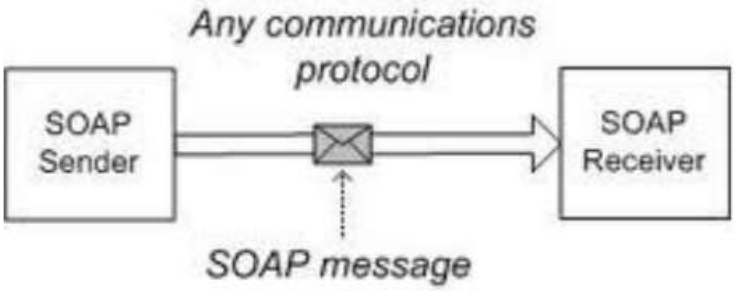
♣ Tất cả các message SOAP đều được mã hóa sử dụng XML.

♣ SOAP sử dụng giao thức truyền dữ liệu riêng.

♣ SOAP không bị ràng buộc bởi ngôn ngữ lập trình hoặc công nghệ nào

♣ SOAP không quan tâm đến công nghệ gì được sử dụng để thực hiện miễn là người dùng sử dụng các message theo định dạng XML.

⇨ Tóm lại: SOAP là giao thức mà định nghĩa cái cách để chuyển một XML message từ A đến B dựa trên giao thức chuẩn web HTTP (hoạt động trên cổng 80) qua giao thức Internet TCP/IP.



Thông điệp SOAP

Tại sao phải có SOAP:

♣ Phát triển các ứng dụng cho phép các chương trình trao đổi qua Internet.

♣ Các ứng dụng liên lạc với nhau bằng cách sử dụng các cuộc gọi thủ tục ở xa giữa các đối tượng.

♣ SOAP cung cấp cách để liên lạc giữa các ứng dụng chạy trên các hệ điều hành khác nhau,với các công nghệ khác nhau và ngôn ngữ khác nhau.

3.4.2.2. Định dạng thông điệp

Một thông điệp SOAP là một văn bản XML được mô tả bởi một thành phần Envelop,chứa một thành phần Body bắt buộc và một thành phần Header không bắt buộc. Thành phần Body có thể chứa một số Body Entries. Thành phần không bắt buộc Fault chỉ có trong thông điệp khi có báo cáo về một quá trình xử lý ngoại lệ.

Phần tử Body mô tả về phương thức dưới dạng XML và chỉ chứa các tham số hay các trường dưới dạng các thẻ.

Với Document người phát triển phải xử lý gần như là toàn bộ, họ phải đưa ra một loạt các tham số dưới dạng các thẻ XML.

3.4.2.3. Mã hóa thông điệp

Dữ liệu được mã hoá và gói vào trong phần tử Body của một thông điệp và được gửi đến Host. Host giải mã dữ liệu được định dạng XML về dạng đối tượng ban đầu.

SOAP Remote Procedure Call (RPC encoding): Là kiểu mã hóa đơn giản nhất cho người phát triển. Bạn gọi tới một đối tượng từ xa, kèm theo là các tham số cần thiết. Các tham số được chuyển lần lượt dưới dạng XML và truyền đến đích sử dụng giao thức giao vận như HTTP hay SMTP. Sau khi nhận được, dữ liệu được chuyển trở lại thành dạng đối tượng và kết quả được trả về cho phương thức gọi. SOAP RPC xử lý tất cả công việc mã hóa và giải mã,thậm chí đối với các kiểu dữ liệu phức tạp.

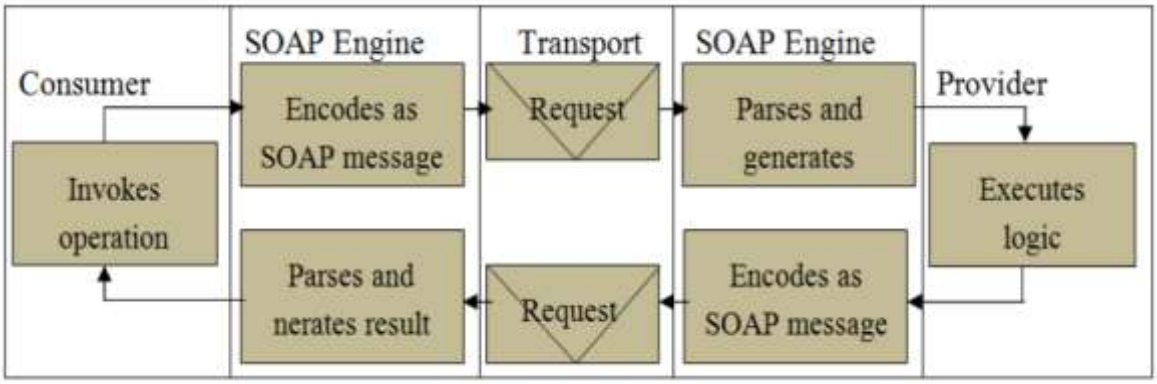
SOAP Remote Procedure Call Literal encoding (SOAP RPC-literal): Sử dụng một dạng thức mã hóa do người sử dụng chỉ định để mã và giải mã dữ liệu dạng XML.

SOAP document-style encoding: Toàn bộ XML được gửi đến máy chủ và người lập trình xác định giao thức giao vận,phân tích dữ liệu dạng XML ở thông điệp yêu cầu và đáp ứng để tìm dữ liệu cần thiết.

3.4.2.4. Quá trình xử lý thông điệp

Một thông điệp SOAP giúp cho khách hàng và nơi cung cấp Web Service hoàn thành những tác vụ mà không lo lắng đến sự phức tạp của việc xử lý thông điệp SOAP.

Một processor của khách hàng chuyển các lời yêu cầu phương thức vào trong một thông điệp SOAP.Thông điệp này được truyền qua tầng giao vận (HTTP và SMTP) tới processor của nơi cung cấp,tại đây thông điệp sẽ được phân tích thành lời yêu cầu phương thức. Sau đó nơi cung cấp sẽ thực hiện những bước logic cần thiết và trả lại kết quả cho processor của nó,processor này sẽ phân tích thông tin trong thông điệp hồi đáp. Thông điệp này được truyền qua tầng giao vận tới khách hàng yêu cầu. Processor của nó phân tích thông điệp hồi đáp thành kết quả dưới dạng một đối tượng.



Quá trình xử lý thông điệp SOAP

3.4.3. Ngôn ngữ đánh dấu,mở rộng XML

3.4.3.1. Khái niệm XML

XML là nền tảng của Web Service và được dùng để trao đổi dữ liệu.

XML là một chuẩn nổi tiếng cho việc tổ chức,lưu trữ và trao đổi dữ liệu.

XML được hỗ trợ bởi hầu hết các ngôn ngữ lập trình hiện đại (DotNet, Java…)

XML được sử dụng rộng rãi trong việc trao đổi dữ liệu trên môi trường Internet.

XML dùng các thẻ để tổ chức và lưu trữ dữ liệu

3.4.3.2. Đặc điểm của XML

XML là tự do và mở rộng được. Trong XML các thẻ không được định nghĩa trước mà do người dùng tự phát minh ra thẻ.

XML rất quan trọng đối với sự phát triển của web trong tương lai.

XML sẽ là công cụ xử lý và truyền dữ liệu phổ biến nhất.

XML là công cụ dùng được trên mọi nền phần cứng, độc lập với phần cứng và phần mềm để truyền (trao đổi,chia sẻ) thông tin.

3.4.3.3. XML được sử dụng như thế nào

XML được thiết kế để lưu trữ và trao đổi dữ liệu nhưng không hiển thị dữ liệu.

XML có thể trao đổi dữ liệu giữa các hệ thống không tương thích.

3.4.3.4. Cấu trúc tài liệu XML

♣ XML hợp khuôn dạng: khai báo XML và dữ liệu XML.

♣ XML hợp lệ: Là tài liệu được kết hợp với định nghĩa kiểu tư liệu (Document Type Definition) và tuân theo tiêu chuẩn đó.

3.4.3.5. Quy tắc cú pháp ngôn ngữ XML

Các khai báo XML cần được đặt ở dòng đầu tiên của tài liệu.

Mọi phần tử XML đều phải có thẻ đóng: />

Tất cả các tài liệu XML phải có thẻ gốc trong đó thẻ đầu tiên là thẻ gốc.

Các thẻ XML phân biệt hoa_thường và khoảng trắng được giữ lại.

Các giá trị thuộc tính phải luôn đặt trong ngoặc kép.

3.4.3.6. Ưu điểm của XML

Đơn giản, ổn định, linh hoạt và có tính mở rộng cao.

XML được chấp nhận rộng rãi. Rất nhiều công cụ và tiện ích sẵn có đáp ứng nhu cầu phân tích và chuyển đổi dữ liệu XML hoặc hiển thị chúng.

3.4.3.7. Nhược điểm của XML

Sự phức tạp.

Việc chuẩn hóa.

Dung lượng lớn.

3.4.4. Ngôn ngữ mô tả dịch vụ WSDL

3.4.4.1. Khái niệm

WSDL là ngôn ngữ dựa trên XML và mô tả cách thức truy cập Web Service.

WSDL thường được sử dụng với SOAP và cấu trúc XML để cung cấp Web Service qua Internet. Một máy khách kết nối tới Web Service có thể đọc WSDL

để xác định hàm nào hiện đang có trên máy chủ. Khách có thể sử dụng SOAP để gọi một trong nhiều hàm được liệt kê trong WSDL.

⇨ Tóm lại: WSDL mô tả Web Service theo cú pháp tổng quát XML, bao gồm các thông tin: tên service, giao thức và kiểu mã hoá, tham số, kiểu dữ liệu …

WSDL chỉ định các đặc tính vận hành của Web Service. Ngôn ngữ mô tả những khái niệm trả lời cho các câu hỏi sau:

♣ Cái gì (Web Service làm gì) ?

♣ Ở đâu (nơi chứa Web Service) ?

♣ Như thế nào (Web Service có thể kích hoạt bằng cách nào) ?

3.4.4.2. Cấu trúc WSDL

Một WSDL hợp lệ gồm có hai phần:

♣ Phần giao diện mô tả giao diện và giao thức kết nối.

♣ Phần thi hành mô tả thông tin để truy xuất service.

Cả 2 thành phần này sẽ được lưu trong hai tập tin XML, bao gồm:

♣ Tập tin giao diện service (cho phần 1)

♣ Tập tin thi hành service (cho phần 2)

3.4.4.3. Ưu điểm của WSDL

Như một yêu cầu cơ bản đối với ứng dụng của bất cứ Web Service,WSDL là yêu cầu bắt buộc đáp ứng nhu cầu công bố giao tiếp và thỏa thuận cho các dịch vụ khác kích hoạt.

3.4.4.4. Nhược điểm của WSDL

Tài liệu không cung cấp một số thông tin người sử dụng có nhu cầu như :

♣ Ai cung cấp dịch vụ ?

♣ Loại hình kinh doanh cung cấp dịch vụ ?

♣ Các dịch vụ khác cùng do nhà cung cấp dịch vụ này cung cấp ?

♣ Dịch vụ này sẽ cung cấp với chất lượng dịch vụ như thế nào ?

♣ Đây là dịch vụ miễn phí hay có thu phí ?

3.4.5. Tích hợp mô tả trình bày tổng hợp UDDI

3.4.5.1. Khái niệm

UDDI là một chuẩn công nghiệp cho việc công bố và tìm kiếm thông tin về Web Service. Nó định nghĩa một khung thông tin cho phép bạn mô tả và phân loại tổ chức của bạn, dịch vụ của nó và những chi tiết kỹ thuật về giao diện của Web Service mà bạn trình bày.

UDDI chỉ định cách thức lưu trữ và nhận thông tin về các dịch vụ và đặt biệt là nhà cung cấp dịch vụ cùng với các giao tiếp kỹ thuật 35 UDDI dựa vào những chuẩn đã có như là ngôn ngữ đánh dấu mở rộng (XML) và giao thức truy cập đối tượng đơn giản (SOAP). Tất cả các cài đặt của UDDI đều hỗ trợ các đặc tả UDDI.

⇨ Tóm lại: Để có thể sử dụng các dịch vụ, trước tiên khách phải tìm dịch vụ, ghi nhận thông tin về cách sử dụng dịch vụ và biết được đối tượng cung cấp dịch vụ. UDDI định nghĩa thành phần cho biết trước các thông tin này để cho phép máy khác truy tìm và nhận lại những thông tin yêu cầu sử dụng Web Service.

3.4.5.2. Đặc điểm của UDDI

UDDI là phần chứa các thông tin của web service, xây dựng trên nền tảng .NET.

UDDI được miêu tả bởi ngôn ngữ WSDL và giao tiếp thông qua SOAP.

Nhiệm vụ UDDI: tìm đúng dịch vụ và định nghĩa cách kích hoạt dịch vụ.

3.4.5.3. Nội dung của thư mục UDDI

Một nội dung thư mục UDDI là một tệp XML mô tả một nghiệp vụ và các dịch vụ nó chào.

Nội dung trong UDDI có ba phần:

♣ White pages: chứa thông tin liên hệ và các định dạng của Web Service. Những thông tin này cho phép đối tượng khác xác định được dịch vụ.

♣ Yellow pages: chứa thông tin mô tả Web Service. Những thông tin này cho phép các đối tượng thấy được từng loại của Web Service.

♣ Green pages: chứa thông tin kỹ thuật mô tả các hành vi và các chức năng

♣ Loại dịch vụ – tModel: chứa các thông tin về loại dịch vụ được sử dụng.

3.4.5.4. Cấu trúc sổ đăng ký UDDI

UDDI cung cấp 4 cấu trúc dữ liệu mô tả dịch vụ mà nó đưa ra: BusinessEntity, BusinessService, BusinessTemplate và tModels.

♣ BusinessEntity: mô tả nhà cung cấp dịch vụ

♣ BusinessService: chứa các thông tin chung về dịch vụ

♣ BindingTemplate: chứa thông tin kỹ thuật cách thức truy cập vào dịch vụ

♣ tModels (Technical Model- mô hình kỹ thuật): chứa các thông tin về loại Web Service sử dụng. Được sử dụng để lấy thông tin chi tiết về giao diện của Web Service và làm cho chúng có thể sử dụng lại giữa các dịch vụ tương thích.

3.4.5.5. Các kiểu sổ đăng ký UDDI

Mô hình đám mây các nút UDDI: một cơ chế khai triển công khai của tiêu chuẩn UDDI đó là Sổ đăng ký kinh doanh UDDI hoặc UBR. UBR bao gồm vài nút UDDI. Những nút này do các công ty như IBM, Microsoft hay SAP và NTT quản lý. Khi một nhà cung cấp dịch vụ muốn công bố dịch vụ của họ, họ sẽ tới một trong các địa chỉ UBR như: http://uddi.ibm.com. Và sau đó đăng ký rồi công bố dịch vụ của họ. Dữ liệu tiếp tục nhân bản tới các nút khác trong cùng hệ thống UBR.

Nhóm hoặc các sổ đăng ký cộng tác: những triển khai này tập trung vào một số lượng cụ thể các đối tác đã từng biết.

Sổ đăng ký riêng tư: hầu hết các công ty đều hướng tới việc bắt đầu các dự án Web Service thông qua một sổ đăng ký UDDI riêng biệt.

CHƯƠNG 4: CÁC KỸ THUẬT BẢO MẬT WEB SERVICE

4.1. Tổng quan về an toàn Web Service

Từ những giai đoạn đầu tiên của Internet, các doanh nghiệp luôn đòi hỏi rất khắt khe về vấn đề bảo mật trong thương mại điện tử. Những hạn chế của tường lửa như việc giám sát các gói tin được truyền tải dựa trên giao thức HTTP là chưa có; điều này có thể khiến cho máy chủ có nguy cơ bị những cuộc tấn công không hề biết được biết trước. Đã có rất nhiều các thuật toán đưa ra cơ chế và những chuẩn về bảo mật như sự mã hoá khoá thông tin, chữ ký số …; nhưng hầu hết chỉ tập trung vào việc đưa ra các định dạng bảo về dữ liệu trong quá trình trao đổi, không quan tâm đến việc xác định các nghi thức mà các bên cần thực hiện khi tương tác với nhau.

Ngoài ra, những chuẩn chung về việc chỉ ra nghi thức giao tiếp giữa Web Service là chưa có, đã khiến cho các sản phẩm hỗ trợ bảo mật của Web Service không thể tích hợp với nhau, mặc dù các sản phẩm này đều được thiết kế dựa trên chuẩn về bảo mật cho web service.

Một chuẩn an toàn chung cho các hệ thống giao dịch trên mạng thường phải tập trung vào những điều sau :

♣ Identification: định danh được những ai truy cập tài nguyên hệ thống.

♣ Authentication: chứng thực truy cập tài nguyên của người muốn sử dụng.

♣ Authorization: cho phép giao dịch khi đã xác nhận định danh người truy cập.

♣ Integrity: toàn vẹn thông tin trên đường truyền.

♣ Confidentiality: độ an toàn, không ai có thể đọc thông tin trên đường đi.

♣ Auditing: kiểm tra, tất cả các giao dịch đều được lưu lại để kiểm tra.

♣ Non-repudiation: độ mềm dẻo, cho phép chứng thực hợp tính hợp pháp hóa của thông tin đến từ một phía thứ ba ngoài hai phía là người gửi và người nhận.

4.2. Bảo mật Web Service

4.2.1. Khái niệm

Web Service Security là một chuẩn an toàn cho SOAP và cả những phần mở rộng của SOAP, nó được dùng khi muốn xây dựng những web service toàn vẹn và tin cậy của thông điệp.

Web Service Security đảm bảo cho tính an toàn, sự toàn vẹn và tin cậy của thông điệp.

4.2.2. Chứng thực trong một ứng dụng

- Phía máy khách

Máy khách sẽ cung cấp một dấu hiệu an toàn trong tập tin mô tả cũng như phải chỉ rõ một Callback handler để lấy tài khoản và mật khẩu trong thông điệp SOAP và gửi tới máy chủ.

- Phía máy chủ

Để cấu hình máy chủ an toàn cần có một dấu hiệu an toàn hợp lệ cũng như phải chỉ rõ một Callback handler để đọc dấu hiệu an toàn trong SOAP máy khách và xác nhận nó.

4.2.3. Các bước tạo sự an toàn thông tin trong một ứng dụng

- Phía máy khách

Chỉ rõ những thành phần thông điệp mà phải có chữ ký hay một dấu hiệu chứng thực nào đó (nằm ở phần thân thông điệp)

Chỉ rõ một khóa trên hệ thống tập tin mà sẽ ký lên thông điệp. Chỉ những máy khách đã được cấp quyền mới có quyền sở hữu khóa này.

Chỉ rõ những giải thuật sẽ được sử dụng bởi khóa để ký lên thông điệp.

- Phía máy chủ

Chỉ rõ những thành phần của thông điệp cần được ký. Nếu thông điệp đến không có một chữ kí hợp lệ thì yêu cầu sẽ thất bại.

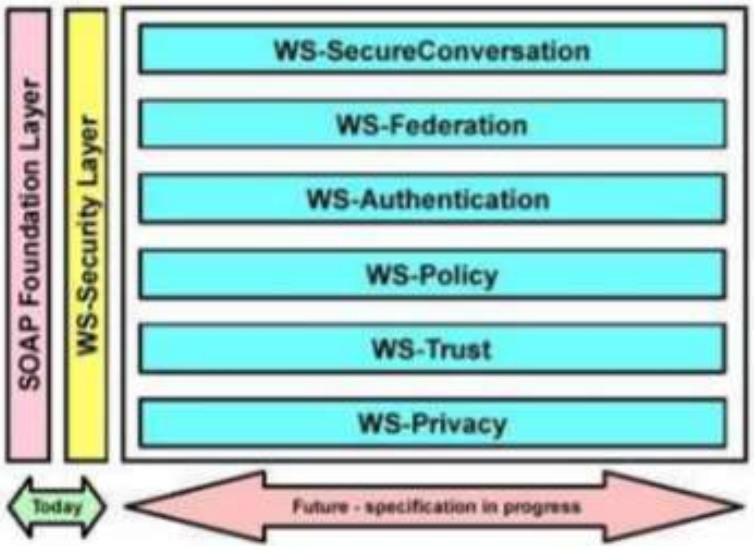
Chỉ rõ một khóa để duyệt chữ kí của thông điệp đến xem có hợp lệ hay không.

Chỉ rõ giải thuật mà khóa sử dụng để đảm bảo toàn vẹn của thông điệp gửi đến.

Thông điệp phản hồi phải được cung cấp thông tin chữ ký khi phản hồi.

4.2.4. Những thành phần mở rộng của Web Service Security

Do web service security chỉ là một lớp trong nhiều lớp của giải pháp an toàn đầy đủ nên cần một mô hình an toàn chung lớn hơn để bao phủ tất cả các khía cạnh an toàn khác như đăng kí và không từ chối.



Mô hình an toàn cho Web service

Trong mô hình này các thành phần quan trọng bao gồm:

WS-SecureConversation Describes: quản lý và xác nhận thông điệp trao đổi giữa các phần, bao gồm sự trao đổi ngữ cảnh, thiết lập, dẫn xuất ra những phiên.

WS-Authentication Describes: quản lý những dữ liệu, chính sách cần chúng thực.

WS-Policy Describes: quản lý những ràng buộc của những chính sách an toàn ở các điểm trung gian và đầu cuối.

WS-Trust Describes: cho phép Web Service an toàn trao đổi, tương tác với nhau.

4.3. Các kỹ thuật Web Service Security

- eXtensible Access Control Markup Language (XACML)

- Security Assertion Markup Language (SAML)

- XML Key Management Specification (XKMS)

- Web Services Policy Framework (WS-Policy)

- eXentisble Rights Markup Language (XrML)

- Secure Socket Layer (SSL)

4.3.1. eXtensible Access Control Markup Language (XACML)

4.3.1.1: Tổng quan XACML

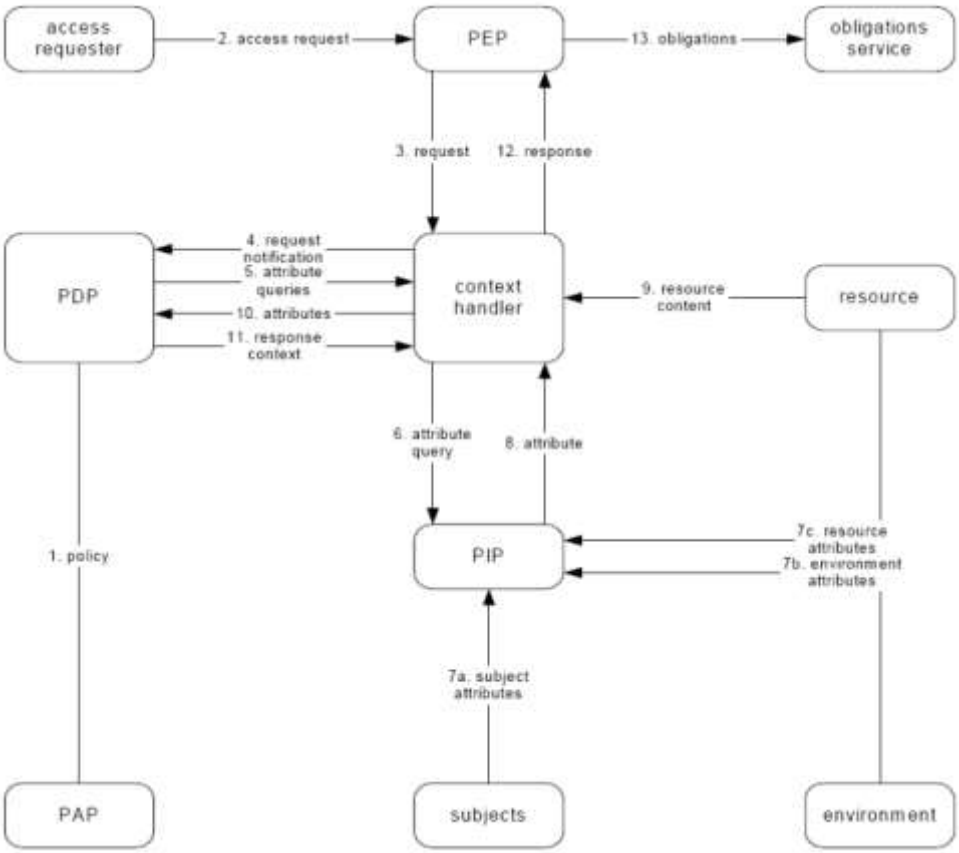
Các chính sách điều khiển truy cập rất phức tạp và phải được thi hành tại nhiều điểm. Trong một môi trường phân phối, ví dụ như thiết lập một dịch vụ web, thực hiện các chính sách điều khiển truy cập bằng cách cấu hình chúng tại mỗi điểm, khiến cho các chính sách trở nên đắt tiền và không đáng tin cậy. Hơn nữa, các chính sách điều khiển truy cập thường được thể hiện thông qua các ngôn ngữ độc quyền và khác nhau.

XACML được hình thành để giải quyết vấn đè này, bằng cách cung cấp một tiêu chuẩn, ngôn ngữ duy nhất để xác định các chính sách điều khiển truy cập. XACML phiên bản 2.0 đã được chấp nhận như một tiêu chuẩn OASIS cùng với sáu cấu hình của XACML: SAML 2.0, XML Digital Signature, Privacy Policy (chính sách bảo mật), Hierarchical Resource (phân cấp tài nguyên) và RBAC (Role-Based Access Control). XACML là một tiêu chuẩn bổ sung của OASIS để đưa ra các quyết định việc điều khiển truy cập.

XACML được thực hiện trong XML.

Các đối tượng của XACML được dùng để tạo ra một tiêu chuẩn cho việc miêu tả các thực thể điều khiển truy cập và các thuộc tính của chúng. Chúng đề nghị nhiều các điều khiển truy cập hơn việc từ chối và cấp quyền truy cập

4.3.1.2: Mô hình của XACML



XACML Architecture

PEP: Policy Enforcement Point: Thực hiện kiểm soát truy cập bằng cách yêu cầu quyết định và thực thi các quyết định ủy quyền.

PAP: Policy Administration Point: Tạo và lưu trữ chính sách bảo mật.

PDP: The Policy Decision Point: Nhận, xem xét yêu cầu. Sau đó áp dụng các chính sách cùng với việc đánh giá các chính sách đó rồi trả về PEP

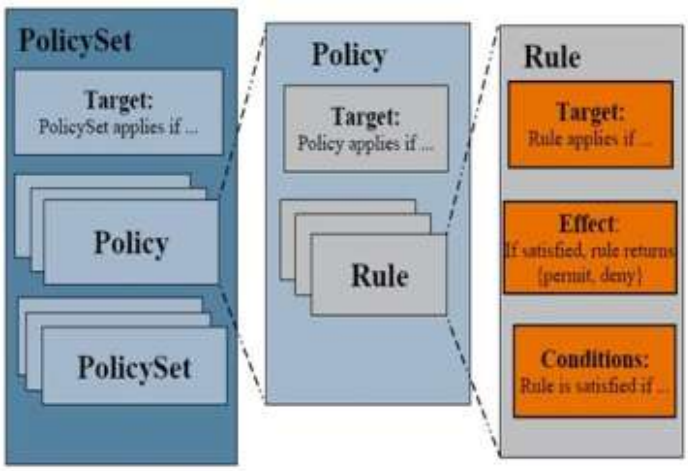
PIP: Policy Information Point: Là nguồn gốc của các giá trị thuộc tính hoặc các dữ liệu cần thiết để đánh giá chính sách.

Context Handler: Xác định để chuyển đổi các yêu cầu theo định dạng gốc của nó với hình thức XACML và chuyển đổi các quyết định ủy quyền theo hình thức XACML sang định dạng gốc.

Các chính sách XACML sẽ được nạp vào PAP, tại đây các chính sách sẽ được gửi tiếp tới PDP. PDP là điểm quyết định sẽ sử dụng chính sách nào cho các yêu cầu truy cập. Khi có một yêu cầu truy cập được gửi tới PEP, nó sẽ tiếp nhận các yêu cầu và thực hiện chúng bằng cách yêu cầu tới các văn bản xử lý. Các văn bản này lại được gửi yêu cầu tới PDP, tại đây các yêu cầu được xử lý và sau đó được gửi phản hồi lại cho Context Handler. Và tiếp tục gửi lại cho PEP – nơi thực hiện các chính sáchh sau khi đã qua quá trình xử lý và thực hiện tại PDP. Sau khi thực thi các chính sách PEP sẽ gửi các chính sách tới các Máy chủ chứng thực và tạo ra các tài nguyên để chia sẻ. Các tài nguyên này kết hợp cùng với PIP được lưu trữ trở lại cho Context Handler phục vụ cho những yêu cầu lần sau.

Các XACML Context Handler sẽ cách ly và xử lý các ứng dụng cho các đầu vào và đầu ra sử dụng PDP. Trong thực tế, đó là các Context Handler dùng để dịch các yêu cầu về truy cập ứng dụng từ định dạng ban đầu của nó sang định dạng theo chuẩn trên. Mấu chốt XACML là xác định các cú pháp cho một ngôn ngữ chính sách bất kỳ, ngữ nghĩa cho các quy tắc chính sách và giao thức nhằm đáp ứng các yêu cầu giữa PEP và PDP.

4.3.1.3: Thành phần của XACML

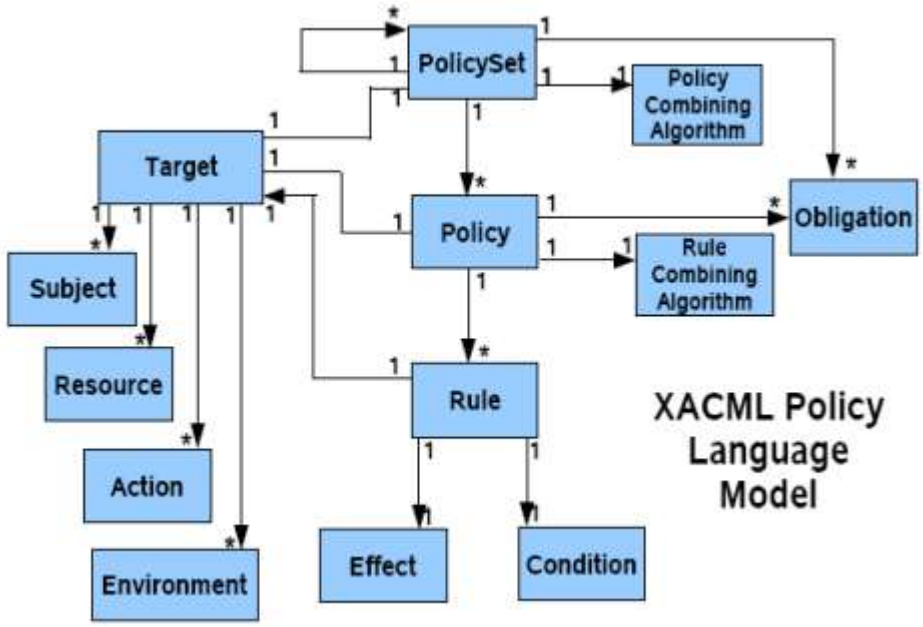


Thành phần của XACML

Một XACML bao gồm 3 thành phần cơ bản sau:

- Rule (quy tắc)
- Policy (chính sách)
- Policy Set (thiết lập chính sách)

4.3.1.4: Mô hình ngôn ngữ XACML



XACML Policy Language Model

Theo như lý thuyết được trình bày bên trên, xuất phát từ Target: bao gồm ba thành phần chính: **suject, action, resource** có mối quan hệ với target cụ thể như trên hình vẽ. Target cũng có mối quan hệ tương tự với Policy Set – Policy – Rule theo tỷ lệ cụ thể như hình vẽ. Giao tiếp giữa Policy Set và Policy thông qua việc kết hợp sử dụng các chính sách và tương tự từ Policy với Rule là việc kết hợp thông qua các quy tắc. Các mối quan hệ được miêu tả cụ thể như trên hình vẽ.

Mô hình cấu trúc XACML là một thể thống nhất trong đó các thành phần có mối quan hệ chặt chẽ với nhau thông qua các quy tắc đã được xác định trước

- **Cấu trúc XACML Request**

  Bao gồm bốn thành phần:

- Thuộc tính đối tượng
- Thuộc tính tài nguyên
- Thuộc tính hành động
- Thuộc tính môi trường

XACML Request

❖ **Cấu trúc XACML Response**

Bao gồm ba thành phần

- Quyết định
- Trạng thái
- Trách nhiệm

XACML Response

4.3.2. Security Assertion Markup Language (SAML)

4.3.2.1: Tổng quan SAML

SAML là sự kết hợp giữa S2ML và AuthML, được phát triển thông qua OASIS. SAML là một tiêu chuẩn dựa trên XML, được hình thành như một khuôn khổ cho việc trao đổi thông tin liên quan đến an ninh, thể hiện dưới các xác nhận và sự tin tưởng giữa các bên tham gia trao đổi, nhằm xác thực giao tiếp người dùng, quyền lợi và các thuộc tính thông tin.

4.3.2.2: Hoạt động của SAML

Hỗ trợ việc khẳng định các chứng thực gốc duy nhất giữa các domain với nhau. Việc khẳng định có thể truyền đạt thông tin về các thuộc tính của đối tượng và có thể quyết định ủy quyền cho đối tượng được phép truy cập tài nguyên nhất định.

- Xác thực tin tưởng
- Chứng thực các vấn đề liên Domain
- Tập trung các vấn đề xác thực liên

SAML hỗ trợ ba loại hình xác nhận:

- Xác thực: Các đối tượng quy định được chứng thực tại thời điểm cụ thể
- Thuộc tính: Các đối tượng quy định có liên quan tới thuộc tính được cung cấp.
- Quyết định ủy quyền: một yêu cầu cho phép đối tượng quy định để truy cập vào tài nguyên quy định đã được cấp hoặc từ chối.

4.3.2.3: Đặc điểm của SAML

Một SAML duy nhất khẳng định có thể chứa một số báo cáo khẳng định về

chứng thực, ủy quyền và các thuộc tính. Khẳng định là do cơ quan SAML, cụ thể là cơ quan thẩm định, cơ quan thuộc tính, hoặc là một điểm quyết định chính sách.Tuy nhiên, nó không cung cấp cơ chế để kiểm tra, thu hồi chứng trỉ. SAML cung cấp bối cảnh chứng thực, được truyền đạt (hoặc tham chiếu) một sự khẳng định của chứng thực đó. Khuôn khổ quy định của SAML là nhằm hỗ trợ nhiều tình huống kinh doanh thực trên thế giới, từ những người mà trong đó khách hàng là một trình duyệt để thêm những phần phức tạp nơi mà Web Service có liên quan.

Bảo mật thông tin SOAP, khẳng định SAML có thể được sử dụng trong thông điệp SOAP để thực hiện vấn đề an ninh và nhận dạng thông tin giữa các hành động trong giao dịch.Các SAML Token của tổ chức WSS OASIS quy định cách xác nhận SAML nên được sử dụng cho mục đích này. The Liberty Alliance's Identity Web Service Frameword (ID-WSF) cũng sử dụng SAML xác nhận như là thẻ an ninh cơ sở để cho phép việc an toàn và tôn trọng sự riêng tư khi tiếp cận với các Web Service

4.3.3. XML Key Management Specification (XKMS)

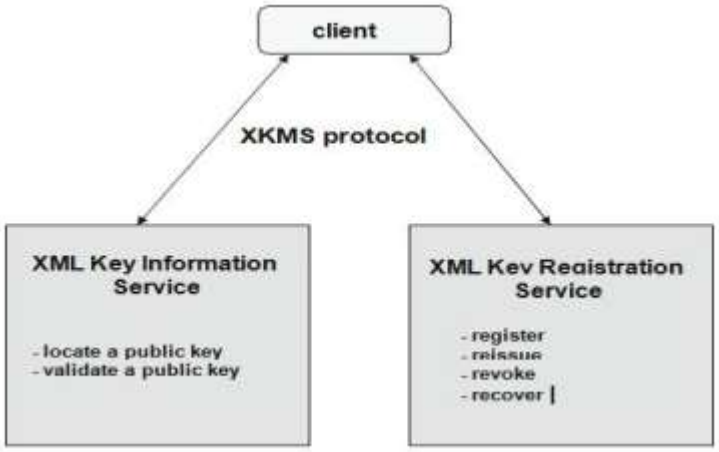
Các khóa công cộng là các khối cơ bản xây dựng cho chữ ký và chứng nhân kỹ thuật số. Khóa công khai quản lý bao gồm việc tạo ra, lưu trữ an toàn, phân phối, sử dụng và hủy bọ chúng. Các khóa công cộng có thể được tạo ra bởi một gói phần mềm chạy trên nền tảng của các ứng dụng khách hàng và sau đó đăng ký một khóa cơ sở hạ tầng công cộng , chứng nhận ủy quyền hoặc ứng dụng khách hàng có thể yêu cầu một chứng nhận tham gia đến một cơ sở hạ tầng để tạo ra các khóa này. Khi một bên sử dụng một khóa công khai, nó có nhu cầu để xác định tính hợp lệ của nó, nghĩa là, nó cần phải xác minh các khóa công cộng chưa hết hạn hoặc đã bị thu hồi bởi nhà cung cấp Web Service. Khóa công cộng có thể được cấp bằng nhiều cách khác nhau, có thể có nhiều hơn một khóa công khai liên quan tới khóa công cộng. Tuy nhiên, các khóa cơ sở hiện nay dựa trên bộ công cụ độc quyền, làm cho tương tác giữa các ứng dụng khách hàng và các hạ tầng trờ nên tốn kém và khó khăn hơn. Hơn nữa, các ứng dụng khách hàng phải tự thực hiện rất tốn kém các hoạt động như xác nhận chữ ký, xác nhận dây chuyền và kiểm tra thu hồi. Do đó cần phải đơn giản hóa các nhiệm vụ của các bên khi chúng ta công khai khóa công cộng, cũng như cho phép các chứng thực khác nhau, hoặc thậm chí khóa khác nhau. Hơn nữa, các khóa công cộng có thể được đại diện trong XML, và là cơ sở của XML Encryption và XML chữ ký. Những vấn đề mô tả ở trên đã dẫn đến việc định nghĩa các chuẩn đối với XML Key Management

Hơn nữa,WS-Security xác định các cơ chế cơ bản cho việc cung cấp thông điệp an toàn, thông điệp SOAP được bảo vệ bởi WS-Security trình bày ba vấn

đề chính đó là: tính không tương thích định dạng bảo mật thẻ;sự khác biệt không gian tên và sự tin cậy an ninh thẻ. Để khắc phục những vấn đề trên, cần thiết phải xác định tiêu chuẩn mở rộng để WS-Security cung cấp các phương pháp nhằm đưa ra, đổi mới và xác nhận thẻ bảo mật và để thiết lập và đánh giá sự xuất hiện, mối quan hệ tin tưởng lẫn nhau.

XKMS là một giao thức được phát triển bởi W3C để mô tả sự phân phối và đăng ký khóa công cộng, nó làm giảm các ứng dụng phức tạp cú pháp của nền tảng được sử dụng để thiết lập các mối quan hệ tin tưởng.

Trong hình vẽ sau, một dịch vụ X-KISS cung cấp cho khách hàng hai chức năng và có thể thực hiện bởi chính các dịch vụ X-KISS hoặc của một khóa cơ sở cơ bản. Đó là chức năng: xác định ví trí và tính xác thực. Đối với mục đích mã hóa, các chức năng cho phép một người gửi không cần biết chính xác liên kết với người nhận để có được thông điệp đó. Các dịch vụ X-KISS không thực hiện bất kỳ sự khẳng định về tính hợp lệ của các liên kết giữa dữ liệu và khóa.



XKMS Services

Đối với việc xác thực của một khóa, những thông tin được cung cấp bởi người ký có thể là chưa đủ cho người nhận có thể thực hiện việc xác minh mật mã và quyết định có nên tin tưởng vào khóa ký kết này hay không, hoặc là các thông tin không thể thực thi trong một định dạng người nhận có thể sử dụng được. Các chức năng xác nhận cho phép các khách hàng để có được từ các dịch vụ X-KISS một sự khẳng định rõ ràng, đó là hiệu lực của sự ràng buộc giữa các khóa

và các dữ liệu công cộng, ví dụ: một danh từ hoặc một tập hợp các thuộc tính mở rộng. Hơn nữa, các dịch vụ X-KISS đại diện cho tất cả các yếu tố dữ liệu mà được liên kết với cùng một khóa công khai.

XKRSS định nghĩa một giao thức cho việc đăng ký và quản lý các khóa thông tin quan trọng. Bạn có thể đăng ký các khóa với một dịch vụ XKMS bằng cách sau: Các dịch vụ XKMS tạo ra một cặp khóa cho khách hàng và đăng ký các khóa công khai của chính cặp khóa đó và gửi các khóa riêng của cặp khóa này cho khách hàng của mình sử dụng chúng. Các khách hàng cũng có thể nói cho dịch vụ XKMS để họ giữ lại các khóa riêng tư nhằm phục vụ cho trường hợp khách hàng khi bị mất.

4.3.4. Web Services Policy Framework (WS-Policy)

Các dịch vụ Web Policy Framework tiêu chuẩn cung cấp một mô hình mở rộng là ngữ pháp cho phép các dịch vụ web mô tả chính sách của chúng. Các tiêu chuẩn WS-Policy đã được hình thành để cung cấp một mô hình chung, phù hợp với việc thể hiện tất cả các loại mô hình chính sách miền cụ thể, từ việc vận chuyển cấp an ninh, chính sách sử dụng nguồn tài nguyên, đặc điểm chất lượng dịch vụ và quy trình kinh doanh end-to-end. Cốt lõi của mô hình là các khái niệm về sự khẳng định chính sách, xác định hành vi, đó là việc yêu cầu một hoặc nhiều hơn một, của một đối tượng chính sách. Ngữ nghĩa của việc xác nhận chính là các miền cụ thể. Cách tiếp cận được thông qua bởi WS-Policy là xác định khẳng định tên miền cụ thể trong thông số kỹ thuật riêng biệt. Chính sách khẳng định có thể được xác định trong thông số kỹ thuật công cộng như WS-SecurityPolicy và WS-PolicyAssertion hoặc bởi các thực thể sở hữu các Web Service. Đáng chú ý là sự khẳng định này có thể làm hài lòng bằng cách sử dụng SOAP Message Security, WS-Security hoặc bằng cách sử dụng cơ các cơ chế khác trong phạm vi bảo đảm thông tin SOAP. Ví dụ: bằng cách gửi tin nhắn trong một giao thức như HTTPs. Các đối tượng mà chính sách này áp dụng cho một thông điệp chính sách đối tượng (thông điệp SOAP) và tiêu chuẩn WS-PolicyAttachment mà thực thể hoặc tổ chức WSDL và UDDI áp dụng.

WS-Policy định nghĩa các điều kiện theo một yêu cầu có thể đáp ứng, tương ứng, khẳng định chính sách của các web servuce đó, giải pháp thay thế chính sách và cuối cùng là toàn bộ các chính sách:

- Một sự khẳng định chính sách được hỗ trợ bằng cách yêu cầu khi và chỉ khi người yêu cầu đáp ứng được các yêu cầu tương ứng để khẳng định.
- Một chính sách được hỗ trợ bằng cách yêu cầu khi và chỉ khi có yêu cầu hỗ trợ ít nhất là một trong những lựa chọn thay thế trong chính sách đó.

Khung chính sách được bổ sung bởi ba tiêu chuẩn:

- WS-Policy Assertion: xác định cấu trúc của một chính sách khẳng định
- WS-Policy Attachment: định nghĩa làm thế nào để chính sách liên kết với web service hoặc bằng cách trực tiếp nhúng nó trong WSDL, định nghĩa hoặc gián tiếp liên kết thông qua UDDI.WS-PolicyAttachment cũng xác định làm thế nào để thực hiện liên kết các chính sách cụ thể với tất cả hoặc một phần của một kiểu cổng WSDL khi tiếp cận từ thực hiện cụ thể.
- WS-Security Policy:xác định một tập các khẳng định chính sách tương ứng với tiêu chuẩn bảo mật thông điệp SOAP, đó là thông điệp khẳng định tính toàn vẹn, tin tưởng bảo mật khẳng định, và tin tưởng an ninh khẳng định. Một chính sách WS-Security tiếp cận thông qua WSDL hoặc UDDI, cho phép người gửi yêu cầu để xác định xem WS-Security là tùy chọn hay bắt buộc đối với web service bất kỳ. Nếu nó là bắt buộc, người yêu cầu có thể xác định kiểu bảo mật mã hóa mà web service cung cấp. Người yêu cầu cũng có thể xác định xem họ cần phải ký tên vào thông điệp hay không và những phần nào để đăng nhập. Cuối cùng, yêu cầu xác định có thể mã hoác các thông điệp và nếu có là những thuật toán sử dụng.

4.3.5. eXentisble Rights Markup Language (XrML)

Kỹ thuật và các công cụ được sử dụng để cung cấp bảo mật hệ thống, chẳng hạn như tường lửa phục vụ việc truy cập vào mạng và hệ thống kiểm soát truy cập hạn chế truy cập dữ liệu được lưu trữ, không thể thực thi các quy định kinh doanh mà cách mọi người sử dụng và phân phối dữ liệu bên ngoài hệ thống.

Việc kiểm soát và thực thi phân phối, sử dụng thông tin số đã được giải quyết bằng cách quản lý bản quyền số (Digital Right Management DRM). Thuật ngữ này thường được gọi bằng luật về quyền tác giả, chủ sở hữu nội dung khi tìm kiếm phương tiện đề kiểm soát sử dụng tài sản trí tuệ của mình.Hệ thống DRM về cơ bản thực hiện hai chức năng chính đó là giám sát và điều khiển truy cập:

- Chức năng giám sát, cho phép việc theo dõi những gì đang thực sự được chuyển giao qua mạng đến tay người nhận.
- Chức năng điều khiển truy cập và sử dụng kiểm soát những gì người dùng có thể hoặc không thể làm gì với nội dung kỹ thuật số chuyển giao cho máy tính của mình.

Các mô tả về hoạt động cho phép cho người dùng trên một nội dung kỹ thuật số là khái niệm tương tự như mô tả về các hoạt động trong chính sách kiểm soát truy cập. Các chính sách kiểm soát truy cập được gắn với các nội dung kỹ thuật số của nó trong một hộp an toàn, để các nội dung kỹ thuật số đi kèm với mô tả của chính sách điều khiển truy cập áp dụng cho nó. Mục đích DRM thi hành việc truy cập cụ thể và chính sách kiểm soát sử dụng kết hợp với các nội dung kỹ thuật số.

XrML là một ngôn ngữ XML mà xác định làm thế nào để mô tả các quyền, lệ phí và điều kiện để sử dụng nội dung kỹ thuật số, với tính toàn vẹn thông điệp và tổ chức chứng thực. XrML đã được hình thành để hỗ trợ thương mại trong các nội dung kỹ thuật số, đó là việc xuất bản và bán sách điện tử, phim kỹ thuật số, kỹ thuật số âm nhạc, trò chơi tương tác, phần mềm máy tính và sáng tạo khác được phân phối dưới dạng kỹ thuật số. XrML dự định hỗ trợ truy cập và đặc điểm

kỹ thuật của việc sử dụng điều khiển các đối tượng an toàn kỹ thuật số trong trường hợp trao đổi tài chính.

Đặc điểm cốt lõi kỹ thuật XrML cũng xác định các tập thường được sử dụng, quyền hạn cụ thể, đặc biệt là các quyền liên quan đến quyền khác, chẳng hạn như vấn đề, thu hồi, ủy quyền. Phần mở rộng cho các XrML có thể định nghĩa về quyền cho việc sử dụng các ứng dụng cụ thể. Ví dụ: nội dung XrML gia hạn xác định quyền thích hợp cho việc sử dụng sản phẩm kỹ thuật số (sử dụng và in quyền). Một thực tế tài nguyên đại diện cho các đối tượng trong đó một bên có thể được cấp cho một người đứng đầu. Một nguồn tài nguyên có thể là một công việc kỹ thuật số, chẳng hạn như âm thanh hoặc tập tin video, hoặc hình ảnh, dịch vụ, chẳng hạn như là dịch vụ email, hoặc thậm chí mẩu thông tin có thể được sử dụng bởi một địa chỉ email, thuộc tài sản nào khác hay thuộc tính.

4.3.6. Giao thức bảo mật SSL

4.3.6.1: Tổng quan về SSL

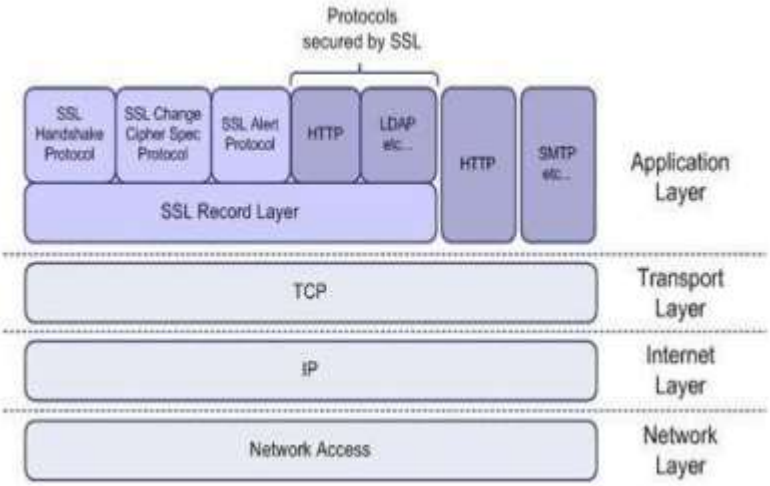
SSL là một sự xuất hiện bổ sung của VPN (Virtual Private Networks).Nó được thiết kế cho giải pháp truy cập từ xa và không cung cấp những kết nối site-to-site. SSL VPNs cung cấp vấn đề bảo mật truy cập đầu tiên những ứng dụng web.

SSL VPNs hoạt động ở tầng phiên của mô hình tiêu chuẩn OSI. Và bởi vì máy khách là một trình duyệt web nên những ứng dụng chúng hỗ trợ trình duyệt web,mặc định, nó sẽ làm việc với một giải pháp VPN. Vì thế những ứng dụng như Telnet, FTP, SMTP, POP3, multimedia, hệ thống điện thoại di động IP, điều khiển desktop từ xa, và những cái khác không làm việc với SSL VPNs bởi vì chúng không sử dụng trình duyệt web cho giao diện đầu cuối người dùng của họ. Tất nhiên, nhiều nhà cung cấp cũng sử dụng cả java hoặc ActiveX để nâng cao SSL VPNs Thêm vào đó để phân phối những thành phần SSL VPNs khác, chẳng hạn như thêm vào những chức năng bảo mật cho việc xóa hết những dấu vết từ một hoạt động của một khách hàng trên máy tính của họ sau khi SSL VPNs đã được kết thúc. Cisco chỉ sự bổ xung SSL VPN như là WebVPN.

SSL được coi là giao thức bảo mật trong lớp vận chuyển (Layer Transport) có tầm quan trọng cao nhất đối với sự bảo mật của các trình ứng dụng trên Web. Và đó là một giao thức đa mục đích được thiết kế để tạo ra các giao tiếp giữa hai chương trình ứng dụng trên một cổng định trước (thông thường là socket 433) nhằm mã hóa toàn bộ thông tin đi/đến, mà ngày nay được sử dụng rộng rãi cho giao dịch điện tử như truyền số hiệu thẻ tin dụng, mật khẩu, số bí mật cá nhân (PIN) trên Internet. Giao thức SSL được hình thành và phát triển đầu tiên vào năm 1994 bởi nhóm nghiên cứu Netscape dẫn dắt bởi Elgammal và ngày nay đã trở thành chuẩn bảo mật thực hành trên mạng Internet. Phiên bản SSL hiện nay là 3.0 và vẫn đang tiếp tụ được bổ sung và hoàn thiện. Chức năng chính là bảo vệ bằng mật mã lưu lượng dữ liệu HTTP.

4.3.6.2 Cấu trúc của một giao thức bảo mật SSL

Cấu trúc và giao thức SSL tương ứng được minh họa trong hình dưới đây. SSL ám chỉ một lớp (bảo mật) trung gian giữa lớp vận chuyển và lớp ứng dụng.SSL được xếp lớp lên trên một dịch vụ vận chuyển định hướng nối kết và đáng tin cậy. Về khả năng, nó có thể cung cấp các dịch vụ bảo mật cho các giao thức ứng dụng tùy ý dựa vào TCP chứ không chỉ HTTP. Thực tế, một ưu điểm chính của các giao thức bảo mật lớp vận chuyển nói chung và giao thức SSL nói riêng là chúng độc lập với ứng dụng theo nghĩa là chúng có thể được sử dụng để bảo vệ bất kỳ giao thức ứng dụng được xếp lớp lên trên TCP một cách trong suốt.SSL có một định hướng máy khách-máy chủ mạnh mẽ và thật sự không đáp ứng các yêu cầu của các giao thức ứng dụng ngang hàng.



Cấu trúc của SSL và giao thức SSL

⇨ Tóm lại: SSL cung cấp sự bảo mật truyền thông vốn có ba đặc tính cơ bản:

- Các bên giao tiếp có thể xác thực nhau bằng cách sử dụng mật mã khóa chung.
- Sự bí mật của lưu lượng dữ liệu được bảo vệ
- Tính xác thực và tính toàn vẹn của lưu lượng dữ liệu cũng được bảo vệ

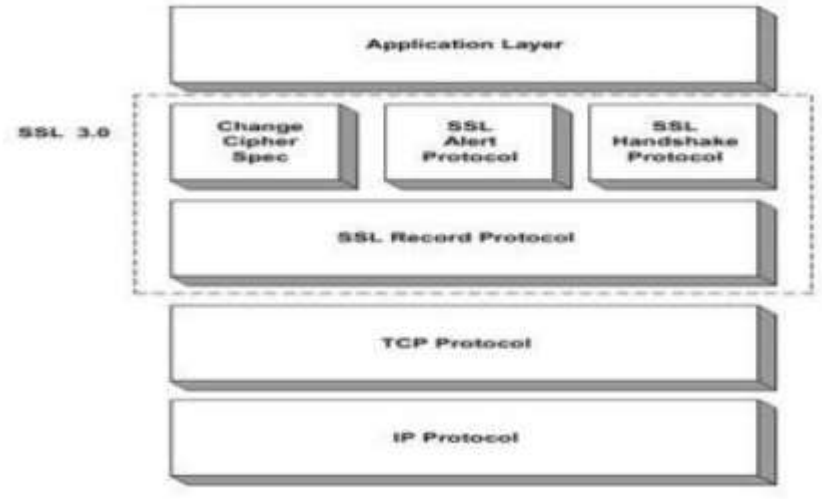
Để sử dụng SSL, máy khách và máy chủ đều phải sử dụng giao thức SSL:

- Sử dụng các số cổng chuyên dụng được dành riêng bởi Internet Asigned
- Numbers Authority (IANA).M ột số cổng riêng biệt phải được gán cho mọi giao thức ứng dụng vốn sử dụng SSL.
- Sử dụng số cổng chuẩn cho mọi giao thức ứng dụng và để thương lượng các tùy chọn bảo mật như là một phần của giao thức ứng dụng
- Sử dụng một tùy chọn TCP để thương lượng việc sử dụng một giao thức bảo mật

4.3.6.3: Các giao thức bảo mật SSL

❖ SSL Record Protocol

SSL Record Protocol nhận dữ liệu từ các giao thức con SSL lớp cao hơn và xử lý việc phân đoạn, nén, xác thực và mã hóa dữ liệu. Chính xác hơn, giao thức này lấy một khối dữ liệu có kích cỡ tùy ý làm dữ liệu nhập và tọa một loạt các đoạn dữ liệu SSL làm dữ liệu xuất (hoặc còn được gọi là các bản ghi) nhỏ hơn hoặc bằng 16,383 byte.



Các bước SSL Record Protocol

Các bước khác nhau của SSL Record Protocol vốn đi từ một đoạn dữ liệu thô đến một bản ghi SSL Plaintext (bước phân đoạn), SSL Compressed (bước nén) và SSL Ciphertext (bước mã hóa). Sau cùng, mỗi bản ghi SSL chứa các trường thông tin sau

- Loại nội dung.
- Số phiên bản của giao thức.
- Chiều dài.
- Tải trọng dữ liệu (được nén và được mã hóa tùy ý).
- MAC.

Loại nội dung xác định giao thức lớp cao hơn vốn phải được sử dụng để sau đó xử lý tải trọng dữ liệu bản ghi SSL (sau khi giải nén và giải mã hóa thích hợp). Số phiên bản của giao thức xác định phiên bản SSL đang sử dụng (thường là version 3.0). Mỗi tải trọng dữ liệu bản ghi SSL được nén và được mã hóa theo phương thức nén hiện hành và thông số mật mã được xác định cho session SSL.

Lúc bắt đầu mỗi session SSL, phương pháp nén và thông số mật mã thường

được xác định là rỗng. Cả hai được xác lập trong suốt quá trình thực thi ban đầu SSL andshake Protocol. Sau cùng, MAC được thêm vào mỗi bản ghi SSL. Nó cung cấp các dịch vụ xác thực nguồn gốc thông báo và tính toàn vẹn dữ liệu. Tương tự như thuật toán mã hóa, thuật toán vốn được sử dụng để tính và xác nhận MAC được xác định rong thông số mật mã của trạng thái session hiện hành. Theo mặc định, SSL Record Protocol sử dụng một cấu trúc MAC vốn tương tự nhưng vẫn khác với cấu trúc HMAC hơn. Có ba điểm khác biệt chính giữa cấu trúc SSL MAC và cấu trúc HMAC:

- Cấu trúc SSL MAC có một số chuỗi trong thông báo trước khi hash để ngăn các hình thức tấn công xem lại riêng biệt.
- Cấu trúc SSL MAC có chiều dài bản ghi.
- Cấu trúc SSL MAC sử dụng các toán tử ghép, trong khi cấu trúc MAC sử dụng module cộng 2.

Tất cả những điểm khác biệt này hiện hữu chủ yếu vì cấu trúc SSL MAC được sử dụng trước cấu trúc HMAC trong hầu như tất cả thông số kỹ thuật giao thức bảo mật Internet. Cấu trúc HMAC cũng được sử dụng cho thông số kỹ thuật giao thức TLS gần đây hơn

Một số giao thức con SSL được xếp lớp trên SSL Record Protocol. Mỗi giao thức con có thể tham chiếu đến các loại thông báo cụ thể vốn được gửi bằng cách sử dụng SSL Record Protocol. Thông số kỹ thuật SSL 3.0 xác định ba giao thức SSL sau đây:

- Alert Protocol: được sử dụng để chuyển các cảnh báo thông qua SSL Record Protocol. Mỗi cảnh báo gồm 2 phần, một mức cảnh báo và một mô tả cảnh báo.

- Handshake Protocol: là giao thức con SSL chính được sử dụng để hỗ trợ xác thực máy khách và máy chủ và để trao đổi một khóa session.
- Change CipherSpec Protocol: được sử dụng để thay đổi giữa một thông số mật mã này và một thông số mật mã khác. Mặc dù thông số mật mã thường được thay đổi ở cuối một sự thiết lập quan hệ SSL, nhưng nó cũng có thể được thay đổi vào bất kỳ thời điểm sau đó

Ngoài những giao thức con SSL này, một SSL Application Data Protocol được sử dụng để chuyển trực tiếp dữ liệu ứng dụng đến SSL Record Protocol.

❖ **SSL Handshake Protocol**

SSL Handshake Protocol[4] là giao thức con SSL chính được xếp lớp trên SSL Record Protocol. Kết quả, các thông báo thiết lập quan hệ SSL được cung cấp cho lớp bản ghi SSL nơi chúng được bao bọc trong một hoặc nhiều bản ghi SSL vốn được xử lý và được chuyển như được xác định bởi phương pháp nén và thông số mật mã của session SSL hiện hành và các khóa bảo mật mã của nối kết SSL tương ứng. Mục đích của SSL Handshake Protocol là yêu cầu một máy khách và máy chủ thiết lập và duy trì hông tin trạng thái vốn được sử dụng để bảo vệ các cuộc liên lạc. Cụ thể hơn, giao thức phải yêu cầu máy khách và máy chủ chấp thuận một phiên bản giao thức SSL chung, chọn phương thức nén và thông số mật mã, tùy ý xác thực nhau và tạo một khóa mật chính mà từ đó các khóa session khác nhau dành cho việc xác thực và mã hóa thông báo có thể được dẫn xuất từ đó.

Các thuật toán mã hóa và xác thực của SSL được sử dụng bao gồm (version3.0):

- DES: chuẩn mã hóa dữ liệu (1977).
- DSA: thuật toán chữ ký điện tử, chuẩn xác thực điện tử.
- KEA: thuật toán trao đổi khóa.
- MD5: thuật toán tạo giá trị "băm".
- RC2, RC4: mã hóa Rivest.
- RSA: thuật toán khóa công khai, cho mã hóa và xác thực.

- RSA key exchange: thuật toán trao đổi khóa cho SSL dựa trên thuật toán RSA.
- SHA-1: thuật toán hàm băm an toàn, phát triển và sử dụng bởi chính phủ Mỹ.
- SKIPJACK: khóa đối xứng phân loại được thực hiện trong phần cứng Fortezza
- Triple-DES: mã hóa DES ba lần.

Cơ sở lý thuyết và cơ chế hoạt động của các thuật toán sử dụng về bảo mật trên hiện nay là phổ biến rộng rãi và công khai, trừ các giải pháp thực hiện trong ứng dụng thực hành vào trong các sản phẩm bảo mật (phần cứng, phần mềm).

Đã có những kết luận cho rằng SSL cung cấp sự bảo mật hoàn hảo ngăn việc nghe lén và những cuộc tấn công thụ động khác, và người thực thi giao thức này sẽ ý thức đến một số cuộc tấn công chủ động tinh vi hơn.

CHƯƠNG 5: KẾT LUẬN

Web Service đã và đang được triển khai và áp dụng trong nhiều lĩnh vực đời sống như ngân hàng, chứng khoán, trao đổi dữ liệu … và ngày càng trở lên phổ biến. Cùng với sự phát triển của nó là những đòi hỏi về tính an toàn, khả năng bảo mật. Bằng việc sử dụng các kỹ thuật đảm bảo an ninh Web Service sẽ giúp cho người sử dụng Web Service trở nên an tâm hơn.

TÀI LIỆU THAM KHẢO

Tiêu chuẩn Web Services Security: SOAP Message Security (link: https://aita.gov.vn/tieu-chuan-web-services-security-soap-message-security-1.1)

[Web Service] Sự khác nhau giữa SOAP và RESTful Web Service trong Java (link: https://tubean.github.io/2018/12/web-service-s%E1%BB%B1-kh%C3%A1c-nhau-gi%E1%BB%AFa-soap-v%C3%A0-restful-web-service-trong-java/)