HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

LUẬN VĂN THẠC SĨ KỸ THUẬT

ĐỀ TÀI:

NGHIÊN CỨU GIẢI PHÁP NÂNG CAO BẢO MẬT CHO TRUNG TÂM TÍCH HỢP DỮ LIỆU CỤC DỰ TRỮ NHÀ NƯỚC KHU VỰC HÀ NỘI

CHUYÊN NGÀNH: KỸ THUẬT VIỄN THÔNG

Giáo viên hướng dẫn : **TS. HOÀNG TRỌNG MINH**

Người thực hiện : NGUYỄN THẾ ANH

Mã số : 8.52.02.08

HÀ NỘI – 2021

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

LUẬN VĂN THẠC SĨ KỸ THUẬT

ĐỀ TÀI:

NGHIÊN CỨU GIẢI PHÁP BẢO MẬT CHO TRUNG TÂM TÍCH HỢP DỮ LIỆU CỤC DỰ TRỮ NHÀ NƯỚC KHU VỰC HÀ NỘI

CHUYÊN NGÀNH: KỸ THUẬT VIỄN THÔNG

Giáo viên hướng dẫn : **TS. HOÀNG TRỌNG MINH**

Người thực hiện : NGUYỄN THẾ ANH

Mã số : 8.52.02.08

HÀ NỘI – 2021

LỜI CAM ĐOAN

Tôi xin cam đoan luận văn này là kết quả nghiên cứu của riêng tôi. Việc sử dụng kết quả, trích dẫn tài liệu tham khảo trên các tạp chí, các trang web tham khảo đảm bảo theo đúng quy định. Các nội dung trích dẫn và tham khảo các tài liệu, sách báo, thông tin được đăng tải trên các tác phẩm, tạp chí và trang web theo danh mục tài liệu tham khảo của luận văn.

Tối xin chịu hoàn toàn trách nhiệm cho lời cam đoan của mình.

Tác giả luận văn

Nguyễn Thế Anh

LỜI CẢM ƠN

Lời đầu tiên, em xin chân thành cảm ơn Thầy **Hoàng Trọng Minh.**

Sau thời gian học tập và trải nghiệm, em đã rút ra được rất nhiều kinh nghiệm về cách thức học tập, cách tiếp cận môn học, kỹ năng làm việc mà không chỉ đơn giản là đọc trong sách vở có thể có được và một lần nữa, em xin gửi lời cảm ơn sâu sắc nhất đến Thầy đã dạy bảo và hướng dẫn những kiến thức chuyên môn cần có để em áp dụng tốt nhất những gì đã được học.

Trong quá trình thực hiện, do còn thiếu nhiều kinh nghiệm thực tế nên không tránh khỏi những sai sót. Em rất mong nhận được những ý kiến đóng góp của Thầy để giúp em trong lĩnh vực này được hoàn thiện hơn. Đó là hành trang quý giá giúp em hoàn thiện kiến thức của mình sau này.

Em xin chân thành cảm ơn và trân trọng kính chào!

MỤC LỤC

DANH MỤC HÌNH

DANH MỤC CÁC TỪ VIẾT TẮT

Tên tiếng Việt	Tên tiếng Anh	Từ viết tắt
Cơ sở dữ liệu		CSDL
Mạng riêng ảo	Vitual Private Network	VNP
Bộ lưu trữ điện dự phòng	Uninterruptible Power Supplier	UPS
Ngôn ngữ đánh dấu mở rộng	Extensible Markup Language	XML
Bộ giao thức liên mạng	Internet protocol suite	TCP/IP
Mạng cục bộ	Local Area Network	LAN
Chuẩn mã hóa tiên tiến	Advanced Encryption Standard	AES
	Windows Communication Foundation	WCF
Hệ quản trị cơ sở dữ liệu	DataBase Management System	DBMS

MỞ ĐẦU

Ngày nay, lĩnh vực bảo mật thông tin đang được nghiên cứu, phát triển và ứng dụng rộng rãi trong nhiều hệ thống thông tin nhằm đảm bảo một hệ thống có tính bảo mật, tin cậy và sẵn sàng. Đặc biệt là những hệ thống cơ sở dữ liệu lưu trữ lớn hoặc Trung tâm tích hợp dữ liệu cần phải có những giải pháp đảm bảo an toàn và bí mật trong lĩnh vực ngân hàng, tài chính, bảo hiểm. Người ta đều xây dựng và triển khai các Trung tâm tích hợp dữ liệu nhưng việc nghiên cứu các giải pháp đảm bảo an toàn và bảo mật thì chưa được quan tâm nhiều. Nhưng vấn đề này ở Việt Nam là một trong vấn đề mới cần được quan tâm đầu tư nghiên cứu, chính vì vậy việc nghiên cứu giải pháp đảm bảo an toàn và bảo mật cho Trung tâm tích hợp dữ liệu là rất cần thiết. Để triển khai nội dung trên cần tập trung nghiên cứu các vấn đề an toàn bảo mật thông tin, về các công nghệ triển khai cho Trung tâm tích hợp dữ liệu về mô hình hệ thống Trung tâm tích hợp dữ liệu. Trên cơ sở đó đề xuất giải pháp đảm bảo bảo mật thông tin cho Trung tâm tích hợp dữ liệu. Triển khai xây dựng giải pháp thử nghiệm cho một số ứng dụng đảm bảo bảo mật thông tin cho Trung tâm tích hợp dữ liệu của Cục Dự trữ Nhà nước khu vực Hà Nội.

Nội dung của luận văn là:

Nghiên cứu về giải pháp đảm bảo bảo mật cho Trung tâm tích hợp dữ liệu là lĩnh vực nghiên cứu mới ở Việt Nam. Để giải quyết các vấn đề trên thì đề tài tập trung vào các vấn đề sau:

- Nghiên cứu cơ bản về an ninh mạng, thuật ngỡ bảo mật cơ bản.
- Mô tả các cuộc tấn công mạng phổ biến nhất, bao gồm tấn công Session, tấn công vi rút, phần mềm mã độc Trojan, từ chối dịch vụ và tràn bộ đệm, xác định các biện pháp phòng thủ cơ bản chống lại các cuộc tấn công đó
- Đề xuất được những giải pháp đảm bảo an toàn và bảo mật cho Trung tâm tích hợp dữ liệu.
- Trung tâm CNTT Triển khai xây dựng ứng dụng thử nghiệm đảm bảo bảo mật thông tin cho Trung tâm tích hợp dữ liệu của Cục Dự trữ Nhà nước khu vực Hà Nội.

CHƯƠNG 1 GIỚI THIỆU VỀ AN NINH MẠNG

Mục tiêu của chương

- Xác định các mối nguy hiểm phổ biến nhất đối với mạng.
- Hiểu cơ bản về mạng.
- Sử dụng thuật ngữ bảo mật cơ bản.
- Tìm cách tiếp cận tốt nhất để bảo mật mạng cho Cục Dự trữ Nhà nước khu vực Hà Nội.
- Đánh giá các vấn đề pháp lý sẽ ảnh hưởng đến công việc của bạn với tư cách là quản trị viên mạng.
- Sử dụng các tài nguyên có sẵn để bảo mật mạng.

1.1 Giới thiệu

Việc tìm kiếm một tuần mà không có một số vi phạm an ninh lớn trong tin tức là rất khó. Máy chủ web của trường đại học bị tấn công, máy tính của chính phủ bị tấn công, dữ liệu của ngân hàng bị xâm phạm, thông tin sức khỏe bị lộ — danh sách vẫn tiếp tục. Có vẻ như mỗi năm lại tập trung nhiều hơn vào vấn đề này. Rất khó để tìm thấy bất kỳ ai ở bất kỳ quốc gia nào chưa từng nghe về những điều như trang web bị tấn công và danh tính bị đánh cắp.

Hiện nay cũng có nhiều địa điểm đào tạo hơn. Nhiều trường đại học cung cấp các bằng cấp về Đảm bảo thông tin từ cấp độ cử nhân cho đến cấp độ tiến sĩ. Có rất nhiều chương trình đào tạo chứng chỉ ngành, bao gồm CISSP, CEH của Hội đồng EC, Mile2 Security, SANS và Security+ của CompTIA. Hiện nay cũng có một số trường đại học cấp bằng về an ninh mạng, bao gồm cả bằng đào tạo từ xa.

Bất chấp sự chú ý này của giới truyền thông và các cơ hội được đào tạo về bảo mật, quá nhiều chuyên gia máy tính - bao gồm một số lượng đáng ngạc nhiên quản trị mạng - không hiểu rõ về loại mối đe dọa mà hệ thống mạng bị phơi nhiễm hoặc những mối đe dọa nào có nhiều khả năng thực sự xảy ra. Các phương tiện truyền thông chính thống tập trung sự chú ý vào các vi phạm bảo mật máy tính nghiêm trọng nhất hơn là đưa ra một bức tranh chính xác về các tình huống đe dọa hợp lý nhất.

Chương này xem xét các mối đe dọa gây ra cho mạng, xác định thuật ngữ bảo mật cơ bản và đặt nền tảng cho các khái niệm được đề cập trong các chương tiếp theo.

Các bước cần thiết để đảm bảo tính toàn vẹn và bảo mật của mạng của bạn là có phương pháp và phần lớn đã được phác thảo. Vào thời điểm hoàn thành đề tài này, có thể xác định các cuộc tấn công phổ biến nhất, giải thích cách chúng được thực hiện để ngăn chặn chúng và hiểu cách bảo mật việc truyền dữ liệu.

+ Kiến thức cơ bản về mạng

Trước khi đi sâu vào cách bảo vệ mạng của Đơn vị, hãy khám phá mạng là gì có lẽ là một ý tưởng hay. Đối với nhiều độc giả, phần này sẽ là một bài đánh giá, nhưng đối với một số người, nó có thể là tài liệu mới. Cho dù đây là bài đánh giá hay thông tin mới, việc tìm hiểu kỹ về mạng cơ bản trước khi cố gắng nghiên cứu an ninh mạng là rất quan trọng. Ngoài ra, hãy lưu ý rằng đây chỉ là một giới thiệu ngắn gọn về các khái niệm mạng cơ bản. Nhiều chi tiết khác không được khám phá trong phần này.

Mạng chỉ đơn giản là một cách để máy móc/máy tính giao tiếp. Ở cấp độ vật lý, nó bao gồm tất cả các máy tính muốn kết nối và các thiết bị bạn sử dụng để kết nối chúng. Các máy riêng lẻ được kết nối bằng kết nối vật lý (cáp loại 5 đi vào card giao diện mạng, hoặc NIC) hoặc không dây. Để kết nối nhiều máy với nhau, mỗi máy phải kết nối với một trung tâm hoặc công tắc, sau đó các trung tâm/công tắc đó phải kết nối với nhau. Trong các mạng lớn hơn, mỗi mạng con được kết nối với các mạng khác bằng một bộ định tuyến[1]. Xem xét nhiều cuộc tấn công trong cuốn sách này (bao gồm một số cuộc tấn công trong Chương 2, “Các loại tấn công”) tập trung vào các thiết bị kết nối các máy với nhau trên mạng (đó là bộ định tuyến, trung tâm và thiết bị chuyển mạch). Nếu bạn thấy chương này không đủ, tài nguyên này có thể được hỗ trợ: http://compnetworking.about.com/od/basicnetworkingconcepts/Networking_Basics_Key_Concepts_in_Computer_Networking.htm.

+ Cấu trúc mạng cơ bản

Một số điểm kết nối phải tồn tại giữa mạng nội bộ và thế giới bên ngoài. Một rào cản được thiết lập giữa mạng đó và Internet, thường ở dạng tường lửa. Nhiều cuộc tấn công được thảo luận trong cuốn sách này hoạt động để vượt qua tường lửa và xâm nhập vào mạng.

Bản chất thực sự của mạng là giao tiếp - cho phép một máy giao tiếp với máy khác. Tuy nhiên, mọi con đường giao tiếp cũng là một con đường tấn công. Bước đầu

tiên để hiểu cách bảo vệ mạng là hiểu biết chi tiết về cách các máy tính giao tiếp qua mạng.

Các thẻ giao diện mạng, bộ chuyển mạch, bộ định tuyến, trung tâm và tường lửa đã đề cập trước đây là những phần vật lý cơ bản của một mạng. Cách chúng được kết nối và định dạng chúng sử dụng để giao tiếp là kiến trúc mạng.

+ Các gói dữ liệu

Sau khi bạn đã thiết lập kết nối với mạng (dù là mạng vật lý hay không dây), khi cần gửi dữ liệu. Phần đầu tiên là xác định nơi bạn muốn gửi nó. Sẽ bắt đầu thảo luận về địa chỉ IP phiên bản 4; chúng ta sẽ xem xét IPv6 một chút sau trong chương này. Tất cả các máy tính (cũng như bộ định tuyến) đều có địa chỉ IP là một chuỗi bốn số từ 0 đến 255 và được phân tách bằng dấu chấm, chẳng hạn như 192.0.0.5 (lưu ý rằng đây là địa chỉ IPv4). Phần thứ hai là định dạng dữ liệu để truyền. Tất cả dữ liệu cuối cùng ở dạng nhị phân (1 và 0). Dữ liệu nhị phân này được đưa vào các gói, tất cả nhỏ hơn khoảng 65.000 byte[1]. Một vài byte đầu tiên là tiêu đề. Tiêu đề đó cho biết gói đang đi đâu, đến từ đâu và có bao nhiêu gói nữa sẽ đến như một phần của quá trình truyền này. Thực tế có nhiều hơn một tiêu đề, nhưng hiện tại, sẽ chỉ thảo luận về tiêu đề như một thực thể duy nhất. Một số cuộc tấn công mà sẽ nghiên cứu (ví dụ: giả mạo IP) cố gắng thay đổi tiêu đề của các gói để cung cấp thông tin sai lệch. Các phương pháp tấn công khác chỉ đơn giản là cố gắng đánh chặn các gói và đọc nội dung (do đó làm ảnh hưởng đến dữ liệu).

Một gói có thể có nhiều tiêu đề. Trên thực tế, hầu hết các gói sẽ có ít nhất ba tiêu đề. Tiêu đề IP có thông tin như địa chỉ IP cho nguồn và đích, cũng như gói tin là giao thức nào. Tiêu đề TCP có thông tin như số cổng. Tiêu đề Ethernet có thông tin như địa chỉ MAC cho nguồn và đích. Nếu một gói được mã hóa bằng Bảo mật lớp truyền tải (TLS), nó cũng sẽ có tiêu đề TLS.

+ Các địa chỉ IP

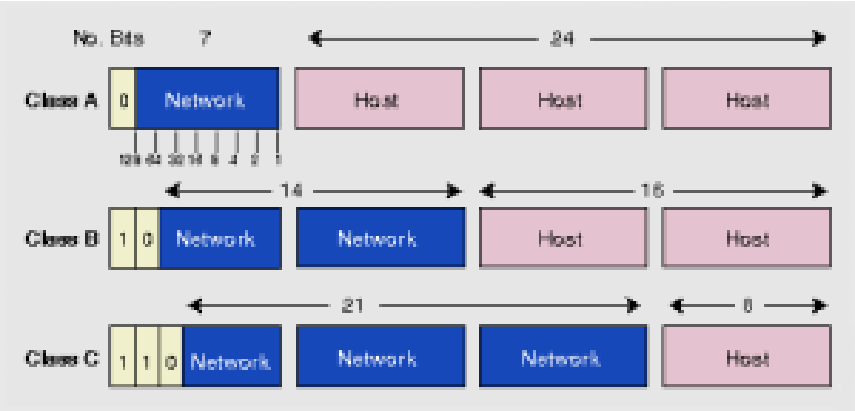
Vấn đề chính đầu tiên cần hiểu là làm thế nào để đưa các gói đến đích thích hợp của chúng. Ngay cả một mạng nhỏ cũng có nhiều máy tính có khả năng là đích cuối cùng của bất kỳ gói tin nào được gửi đi. Internet có hàng triệu máy tính trải dài trên toàn cầu. Làm thế nào để bạn đảm bảo rằng một gói tin đến được đích thích hợp của

nó? Vấn đề không giống như việc giải quyết một bức thư và đảm bảo nó đến đúng điểm đến. Hãy bắt đầu bằng cách xem xét địa chỉ IP phiên bản 4 vì nó là địa chỉ phổ biến nhất được sử dụng ngày nay, nhưng phần này cũng thảo luận ngắn gọn về IP phiên bản 6.

Địa chỉ IP phiên bản 4 là một chuỗi bốn số có ba chữ số được phân tách bằng dấu chấm. (Ví dụ là 107.22.98.198.) Mỗi số có ba chữ số phải nằm trong khoảng từ 0 đến 255. Bạn có thể thấy rằng địa chỉ 107.22.98.466 sẽ không phải là địa chỉ hợp lệ. Lý do cho quy tắc này là các địa chỉ này thực sự là bốn số nhị phân: Máy tính chỉ hiển thị ở định dạng thập phân. Nhớ lại rằng 1 byte là 8 bit (1s và 0s) và số nhị phân 8 bit được chuyển đổi sang định dạng thập phân sẽ nằm trong khoảng từ 0 đến 255. Tổng số 32 bit có nghĩa là có khoảng 4,2 tỷ địa chỉ IP phiên bản 4 tồn tại.

Địa chỉ IP của một máy tính cho bạn biết rất nhiều điều về máy tính đó. Byte đầu tiên (hoặc số thập phân đầu tiên) trong một địa chỉ cho biết máy đó thuộc lớp mạng nào. Bảng 1-1 tóm tắt năm lớp mạng.

BẢNG 1-1 Các lớp mạng



- **Lớp A.** 8 bit network trong đó 1 bit đầu bằng 0 – 24 bit host: 0.0.0.0 đến 126.255.255.255, như vậy sẽ có 127 dải mạng, mỗi dải mạng lớp A có đến 16,777,216 host.
- **Lớp B.** 16 bit network trong đó 2 bit đầu bằng 10 – 16 bit host: 128.0.0.0 đến 191.255.255.255, lớp B có 16,384 dải mạng, mỗi dải mạng lớp B sẽ có tối đa 65,536 host.

- **Lớp C.** 24 bit network trong đó 3 bit đầu bằng 110 – 8 bit host: 192.0.0.0 đến 223.255.255.255, và lớp C có 2,097,152 dải mạng, mỗi dải mạng lớp C sẽ có tối đa 256 host.
- **Lớp D.** 4 bit đầu bằng 1110 – 28 bit dùng cho multicast. 224.0.0.0 đến 239.255.255.255
- **Lớp E.** 4 bit đầu bằng 1111 – 28 bit còn lại chưa rõ. 240.0.0.0 đến 255.255.255.255

Năm lớp mạng này sẽ trở nên quan trọng hơn ở phần sau của cuốn sách này (hoặc nên quyết định nghiên cứu về mạng ở cấp độ sâu hơn). Quan sát Bảng 1-1 một cách cẩn thận, có thể sẽ phát hiện ra rằng dải IP là 127 không được liệt kê. Sự thiếu sót này là do phạm vi đó được dành riêng cho thử nghiệm. Địa chỉ IP của 127.0.0.1 chỉ định máy đang sử dụng, bất kể địa chỉ IP được chỉ định của máy đó là gì. Địa chỉ này thường được gọi là địa chỉ lặp lại. Địa chỉ đó sẽ được sử dụng thường xuyên trong việc kiểm tra máy và NIC của chúng ta. Chúng ta sẽ xem xét việc sử dụng nó ở phần sau của chương này trong phần về các tiện ích mạng[1].

Các lớp cụ thể này rất quan trọng vì chúng cho biết phần nào của địa chỉ đại diện cho mạng và phần nào đại diện cho nút. Ví dụ, trong địa chỉ Lớp A, octet đầu tiên đại diện cho mạng và ba phần còn lại đại diện cho nút. Trong địa chỉ Lớp B, hai octet đầu tiên đại diện cho mạng và hai octet thứ hai đại diện cho nút. Và cuối cùng, trong địa chỉ Lớp C, ba octet đầu tiên đại diện cho mạng và cuối cùng đại diện cho nút.

Ngoài ra còn có một số địa chỉ IP và dải địa chỉ IP rất cụ thể cần biết. Đầu tiên, như đã đề cập trước đó, là 127.0.0.1, hoặc địa chỉ lặp lại. Nó là một cách khác để chỉ card giao diện mạng của máy tính đang sử dụng.

Địa chỉ IP riêng là một vấn đề khác cần lưu ý. Một số dải địa chỉ IP nhất định đã được chỉ định để sử dụng trong các mạng. Chúng không thể được sử dụng làm địa chỉ IP công cộng nhưng có thể được sử dụng cho các máy trạm và máy chủ nội bộ. Các địa chỉ IP đó là

- 10.0.0.10 đến 10.255.255.255
- 172.16.0.0 đến 172.31.255.255

- 192.168.0.0 đến 192.168.255.255

Đôi khi những người mới sử dụng mạng gặp một số khó khăn khi hiểu các địa chỉ IP công cộng và riêng tư. Một tương tự tốt là một tòa nhà văn phòng. Trong một tòa nhà văn phòng, mỗi số văn phòng phải là duy nhất. Chỉ có thể có một 305. Và trong tòa nhà đó, nếu bạn thảo luận về văn phòng 305, ngay lập tức bạn sẽ rõ những gì đang nói. Nhưng có những tòa nhà văn phòng khác, nhiều tòa nhà có văn phòng riêng 305 và có thể coi địa chỉ IP riêng là số văn phòng. Chúng phải là duy nhất trong mạng của chúng, nhưng có thể có các mạng khác có cùng IP riêng.

Địa chỉ IP công cộng giống như địa chỉ gửi thư truyền thống. Chúng phải là duy nhất trên toàn thế giới. Khi liên lạc từ văn phòng này sang văn phòng khác, có thể sử dụng số văn phòng, nhưng để nhận được một lá thư đến một tòa nhà khác, phải sử dụng địa chỉ gửi thư đầy đủ. Nó cũng giống như mạng. Có thể giao tiếp trong mạng bằng địa chỉ IP riêng, nhưng để giao tiếp với bất kỳ máy tính nào bên ngoài mạng, phải sử dụng địa chỉ IP công cộng.

Một trong những vai trò của bộ định tuyến cổng là thực hiện cái được gọi là dịch địa chỉ mạng (NAT). Sử dụng NAT, một bộ định tuyến lấy địa chỉ IP riêng trên các gói gửi đi và thay thế nó bằng địa chỉ IP công cộng của bộ định tuyến cổng để gói có thể được định tuyến qua Internet.

Như đã thảo luận về địa chỉ mạng IP phiên bản 4; bây giờ chúng ta hãy chuyển sự chú ý đến mạng con. Nếu đã quen thuộc với chủ đề này, vui lòng bỏ qua phần này. Vì một số lý do mà chủ đề này có xu hướng mang lại cho sinh viên mạng nhiều rắc rối. Vì vậy, chúng ta sẽ bắt đầu với sự hiểu biết về khái niệm. *Subnetting (đặt mặt nạ mạng con)* chỉ đơn giản là chia nhỏ mạng thành các phần nhỏ hơn. Ví dụ: nếu bạn có mạng sử dụng địa chỉ IP 192.168.1.X (X là bất kỳ địa chỉ nào dành cho máy tính cụ thể), thì đã cấp phát 255 địa chỉ IP có thể. Điều gì xảy ra nếu bạn muốn chia nó thành hai mạng con riêng biệt? Subnetting là cách làm điều đó.

Về mặt kỹ thuật hơn, mặt nạ mạng con là một số 32 bit được gán cho mỗi máy chủ lưu trữ để chia địa chỉ IP nhị phân 32 bit thành các phần mạng và nút. Cũng không thể chỉ nhập bất kỳ số nào muốn. Giá trị đầu tiên của mặt nạ mạng con phải là 255; ba giá trị còn lại có thể là 255, 254, 252, 248, 240, 224 hoặc 128. Máy tính sẽ lấy địa chỉ

IP mạng và mặt nạ mạng con và sử dụng phép toán AND nhị phân để kết hợp chúng.

Có thể sẽ ngạc nhiên khi biết rằng đã có mặt nạ mạng con ngay cả khi chưa tạo mạng con. Nếu bạn có địa chỉ IP Lớp C, thì mặt nạ mạng con là 255.255.255.0. Nếu bạn có địa chỉ IP Lớp B, thì mặt nạ mạng con là 255.255.0.0. Và cuối cùng, nếu đó là Lớp A, mặt nạ mạng con là 255.0.0.0[1].

Bây giờ hãy nghĩ về những con số này trong mối quan hệ với số nhị phân. Giá trị thập phân 255 chuyển đổi thành 11111111 trong hệ nhị phân. Vì vậy, theo nghĩa đen, bạn đang "che" phần địa chỉ mạng được sử dụng để xác định mạng và phần còn lại được sử dụng để xác định các nút riêng lẻ. Bây giờ nếu bạn muốn có ít hơn 255 nút trong mạng con của mình, cần một cái gì đó như 255.255.255.240 cho mạng con của mình. Nếu bạn chuyển đổi 240 sang nhị phân, nó là 11110000. Điều đó có nghĩa là ba octet đầu tiên và 4 bit đầu tiên của octet cuối cùng xác định mạng. 4 bit cuối cùng của octet cuối cùng xác định nút. Điều đó có nghĩa là có thể có tới 1111 (ở dạng nhị phân) hoặc 15 (ở dạng thập phân) trên mạng con này. Đây là bản chất cơ bản của mạng con.

Mạng con chỉ cho phép bạn sử dụng một số mạng con giới hạn. Một cách tiếp cận khác là CIDR, hoặc định tuyến liên miền không phân lớp. Thay vì xác định mặt nạ mạng con, bạn có địa chỉ IP theo sau là dấu gạch chéo và số. Số đó có thể là bất kỳ số nào trong khoảng từ 0 đến 32, dẫn đến các địa chỉ IP như sau:

192.168.1.10/24 (về cơ bản là địa chỉ IP Lớp C)

192.168.1.10/31 (giống địa chỉ IP Lớp C với mặt nạ mạng con)

Khi bạn sử dụng điều này, thay vì có các lớp với mạng con, bạn có mặt nạ mạng con có độ dài thay đổi (VLSM) cung cấp địa chỉ IP không phân lớp. Đây là cách phổ biến nhất để xác định địa chỉ IP mạng ngày nay.

Bạn không nên lo lắng rằng các địa chỉ IP mới có thể sắp hết. Tiêu chuẩn IP phiên bản 6 đã có sẵn và đã có các phương pháp để mở rộng việc sử dụng địa chỉ IPv4. Địa chỉ IP có hai nhóm: công khai và riêng tư. Địa chỉ IP *công cộng* dành cho các máy tính được kết nối với Internet. Không có hai địa chỉ IP công cộng nào có thể giống nhau. Tuy nhiên, địa chỉ IP *riêng*, chẳng hạn như địa chỉ trên mạng công ty tư nhân, phải là duy nhất trong mạng đó. Không thành vấn đề nếu các máy tính khác trên thế

giới có cùng địa chỉ IP, bởi vì máy tính này không bao giờ được kết nối với các máy tính khác trên toàn thế giới. Quản trị viên mạng thường sử dụng địa chỉ IP riêng bắt đầu bằng số 10, chẳng hạn như 10.102.230.17. Các địa chỉ IP riêng khác là 172.16.0.0–172.31.255.255 và 192.168.0.0–192.168.255.255.

Cũng lưu ý rằng một ISP thường sẽ mua một nhóm các địa chỉ IP công cộng và gán chúng khi đăng nhập. Vì vậy, một ISP có thể sở hữu 1.000 địa chỉ IP công cộng và có 10.000 khách hàng. Bởi vì tất cả 10.000 khách hàng sẽ không trực tuyến cùng một lúc, ISP chỉ cần gán địa chỉ IP cho khách hàng khi họ đăng nhập và ISP sẽ bỏ chỉ định địa chỉ IP khi khách hàng đăng xuất[2].

IPv6 sử dụng địa chỉ 128 bit (thay vì 32) và sử dụng phương pháp đánh số hex để tránh các địa chỉ dài như 132.64.34.26.64.156.143.57.1.3.7.44.122.111.201.5. Ví dụ: Định dạng địa chỉ hex xuất hiện ở dạng 3FFE:B00:800:2:: C. Điều này cung cấp cho bạn 2^{128} địa chỉ có thể có (nhiều nghìn tỷ địa chỉ), vì vậy không có khả năng xảy ra tình trạng hết địa chỉ IP trong tương lai gần.

Không có mạng con trong IPv6. Thay vào đó, nó chỉ sử dụng CIDR. Phần mạng được biểu thị bằng dấu gạch chéo theo sau là số bit trong địa chỉ được gán cho phần mạng, chẳng hạn như:

/48

/64

Có một địa chỉ lặp lại cho IPv6 và nó có thể được viết là ::/128. Những khác biệt khác giữa IPv4 và IPv6 được mô tả tại đây:

- Liên kết/máy-cục bộ.
- Phiên bản IPv6 của APIPA của IPv4 hoặc Định địa chỉ IP riêng tự động. Vì vậy, nếu máy được cấu hình cho các địa chỉ được gán động và không thể giao tiếp với máy chủ DHCP, nó sẽ tự gán cho mình một địa chỉ IP chung. DHCP, hoặc Giao thức cấu hình máy chủ động, được sử dụng để chỉ định động các địa chỉ IP trong mạng[2].
- Địa chỉ IP liên kết IPv6/máy-cục bộ đều bắt đầu bằng fe80::. Vì vậy, nếu máy tính của bạn có địa chỉ này, điều đó có nghĩa là nó không thể truy cập vào máy chủ DHCP và do đó tạo ra địa chỉ IP chung của riêng nó.

- Trang web/mạng cục bộ.
- Phiên bản IPv6 của địa chỉ riêng IPv4. Nói cách khác, đây là những địa chỉ IP thực, nhưng chúng chỉ hoạt động trên mạng cục bộ này. Chúng không thể định tuyến trên Internet.
- Tất cả địa chỉ IP trang web/mạng cục bộ bắt đầu bằng FE và có C đến F cho chữ số thập lục phân thứ ba: FEC, FED, FEE hoặc FEF.
- DHCPv6 sử dụng Cờ cấu hình địa chỉ được quản lý (cờ M).
- Khi được đặt thành 1, thiết bị sẽ sử dụng DHCPv6 để có được địa chỉ IPv6 trạng thái.
- Cờ cấu hình trạng thái khác (cờ O).
- Khi được đặt thành 1, thiết bị sẽ sử dụng DHCPv6 để nhận các cài đặt cấu hình TCP/IP khác. Nói cách khác, nó nên sử dụng máy chủ DHCP để đặt những thứ như địa chỉ IP của cổng và máy chủ DNS.

+ Bộ định vị tài nguyên thống nhất

Đối với hầu hết mọi người, mục đích chính để truy cập Internet là các trang web (nhưng có những thứ khác như e-mail và tải xuống tệp). Nếu phải nhớ địa chỉ IP và nhập chúng vào, thì việc lướt mạng sẽ rất phức tạp. May mắn thay, không cần phải làm như vậy, nhập những tên miền có ý nghĩa đối với con người và những tên miền đó được dịch thành địa chỉ IP. Ví dụ: ta có thể nhập www.chuckeasttom.com để truy cập trang web. Máy tính hoặc ISP phải dịch tên đã nhập (được gọi là *Bộ định vị tài nguyên thống nhất* hoặc URL) thành địa chỉ IP. Giao thức DNS (Dịch vụ tên miền), được giới thiệu cùng với các giao thức khác một chút sau trong Bảng 1-2, xử lý quá trình dịch này. Vì vậy, khi gõ một cái tên có ý nghĩa đối với con người, nhưng máy tính của người sử dụng đang sử dụng một địa chỉ IP tương ứng để kết nối. Nếu địa chỉ đó được tìm thấy, trình duyệt sẽ gửi một gói (sử dụng giao thức HTTP) đến cổng TCP 80. Nếu máy tính đích đó có phần mềm lắng nghe và phản hồi các yêu cầu đó (như phần mềm máy chủ web như Apache hoặc Microsoft Internet Information Services) , sau đó máy tính đích sẽ trả lời yêu cầu của trình duyệt của người sử dụng và giao tiếp sẽ được thiết lập. Phương pháp này là cách các trang web được xem. Nếu đã từng nhận được Lỗi 404: Không tìm thấy tệp, những gì bạn đang thấy là trình duyệt của bạn đã nhận lại một gói (từ máy chủ web) với mã lỗi 404, biểu thị rằng không thể tìm thấy trang web

bạn yêu cầu. Máy chủ web có thể gửi lại một loạt thông báo lỗi cho trình duyệt web, cho biết các tình huống khác nhau.

E-mail hoạt động giống như cách truy cập các trang web. Ứng dụng e-mail của bạn sẽ tìm kiếm địa chỉ của máy chủ e-mail của bạn. Sau đó, ứng dụng e-mail của bạn sẽ sử dụng POP3 để lấy e-mail đến hoặc SMTP để gửi e-mail đi của bạn. Máy chủ e-mail của bạn (có thể là tại ISP của bạn hoặc công ty của bạn) sau đó sẽ cố gắng giải quyết địa chỉ mà bạn đang gửi đến. Nếu bạn gửi một cái gì đó đến chuckeasttom@yahoo.com và máy chủ e-mail của bạn sẽ dịch địa chỉ e-mail đó thành địa chỉ IP cho máy chủ e-mail tại yahoo.com, và sau đó máy chủ của bạn sẽ gửi e-mail của bạn đến đó. Lưu ý rằng các giao thức e-mail mới hơn đã có sẵn; tuy nhiên, POP3 vẫn được sử dụng phổ biến nhất[2].

IMAP hiện cũng được sử dụng rộng rãi. Giao thức truy cập tin nhắn Internet hoạt động trên cổng 143. Ưu điểm chính của IMAP so với POP3 là nó cho phép máy khách chỉ tải xuống các tiêu đề về máy và sau đó người dùng có thể chọn tải đầy đủ thông báo nào. Điều này đặc biệt hữu ích cho điện thoại thông minh.

+ Các địa chỉ MAC

Các địa chỉ MAC là một chủ đề thú vị. (Bạn có thể nhận thấy rằng MAC cũng là một lớp con của lớp liên kết dữ liệu của mô hình OSI.) Địa chỉ MAC là một địa chỉ duy nhất cho một NIC. Mọi NIC trên thế giới đều có một địa chỉ duy nhất được biểu thị bằng số thập lục phân sáu byte. Giao thức phân giải địa chỉ (ARP) được sử dụng để chuyển đổi địa chỉ IP thành địa chỉ MAC. Vì vậy, khi bạn nhập địa chỉ web, giao thức DNS sẽ được sử dụng để dịch địa chỉ đó thành địa chỉ IP. Sau đó, giao thức ARP chuyển địa chỉ IP đó thành địa chỉ MAC cụ thể của một NIC riêng lẻ.

+ Các giao thức

Các loại truyền thông khác nhau tồn tại cho các mục đích khác nhau. Các kiểu truyền thông mạng khác nhau được gọi là *các giao thức*. Về cơ bản, một giao thức là một phương thức liên lạc đã được thỏa thuận. Trên thực tế, định nghĩa này chính xác là cách từ ngữ *giao thức* được sử dụng trong cách sử dụng tiêu chuẩn, không dùng máy tính. Mỗi giao thức có một mục đích cụ thể và thường hoạt động trên một cổng nhất định (nhiều cổng hơn một chút). Bảng 1-2 liệt kê một số giao thức quan trọng nhất.

TABLE 1-2 Các cổng và giao thức logic

Giao thức	Mục đích	Cổng
FTP (Giao thức truyền tệp)	Để chuyển các tập tin giữa các máy tính.	20 & 21 SSH Secure Shell. Một cách an toàn/được mã hóa để truyền tệp. 22
Telnet	Được sử dụng để đăng nhập từ xa vào hệ thống.	Sau đó, bạn có thể sử dụng 23 dấu nhắc lệnh hoặc trình bao để thực hiện các lệnh trên hệ thống đó. Phổ biến với các quản trị viên mạng.
SMTP (Giao thức truyền thư đơn giản)	Gửi e-mail.	25
WhoIS	Lệnh truy vấn địa chỉ IP mục tiêu để biết thông tin.	43
DNS (Dịch vụ tên miền)	Dịch các URL thành địa chỉ web.	53
tFTP (Giao thức truyền tệp tầm thường)	Một dạng FTP nhanh hơn nhưng kém tin cậy hơn.	69
HTTP (Giao thức truyền siêu văn bản)	Hiển thị các trang web.	80

Giao thức	Mục đích	Cổng
POP3 (Giao thức Bưu điện Phiên bản 3)	Truy xuất e-mail.	110
NNTP (Giao thức truyền tin tức mạng)	Dùng cho nhóm tin mạng (nhóm tin Usenet). Bạn có thể truy cập các nhóm này trên web qua www.google.com	119
NetBIOS	Một giao thức cũ hơn của Microsoft để đặt tên hệ thống trên mạng cục bộ.	137, 138, 139
IRC (Trò chuyện chuyển tiếp Internet)	Phòng trò chuyện.	194

HTTPS (Bảo mật giao thức truyền siêu văn bản)	HTTP được mã hóa bằng SSL hoặc TLS.	443
SMB (Khối thông báo máy chủ)	Được sử dụng bởi Microsoft Active Directory.	445
ICMP (Giao thức thông báo điều khiển Internet)	Đây chỉ đơn giản là các gói chứa thông báo lỗi, thông báo thông tin và thông báo điều khiển.	Không có cổng cụ thể

Cần lưu ý rằng danh sách này không đầy đủ. Hàng trăm giao thức khác tồn tại, nhưng bây giờ thảo luận về những giao thức này sẽ đủ. Tất cả các giao thức này là một phần của bộ giao thức được gọi là TCP/IP (Trans- Mission Control Protocol/Internet Protocol). Điều quan trọng nhất để bạn nhận ra là giao tiếp trên mạng diễn ra thông qua các gói tin và các gói tin đó được truyền theo các giao thức nhất định, tùy thuộc vào loại hình giao tiếp đang diễn ra. Bạn có thể tự hỏi cổng là gì. Đừng nhầm lẫn loại cổng này với các kết nối ở mặt sau máy tính của bạn, chẳng hạn như cổng nối tiếp hoặc cổng song song[2]. Một cổng trong thuật ngữ mạng là một tay cầm, một điểm kết nối. Nó là một ký hiệu số cho một con đường liên lạc cụ thể. Tất cả giao tiếp mạng, bất kể cổng được sử dụng, đi vào máy tính của bạn thông qua kết nối trên NIC của bạn. Bạn có thể coi một cổng là một kênh trên TV của mình. Bạn có thể có một dây cáp vào TV nhưng bạn có thể xem nhiều kênh. Bạn có một cáp đi vào máy tính của mình, nhưng bạn có thể giao tiếp trên nhiều cổng khác nhau.

Vì vậy, bức tranh mà chúng ta đã vẽ cho đến nay về mạng là một trong những máy móc được kết nối với nhau qua cáp và có lẽ với các trung tâm/bộ chuyển mạch/bộ định tuyến. Mạng truyền thông tin nhị phân trong các gói sử dụng các giao thức và cổng nhất định. Đây là một bức tranh chính xác về truyền thông mạng, mặc dù là một bức tranh đơn giản.

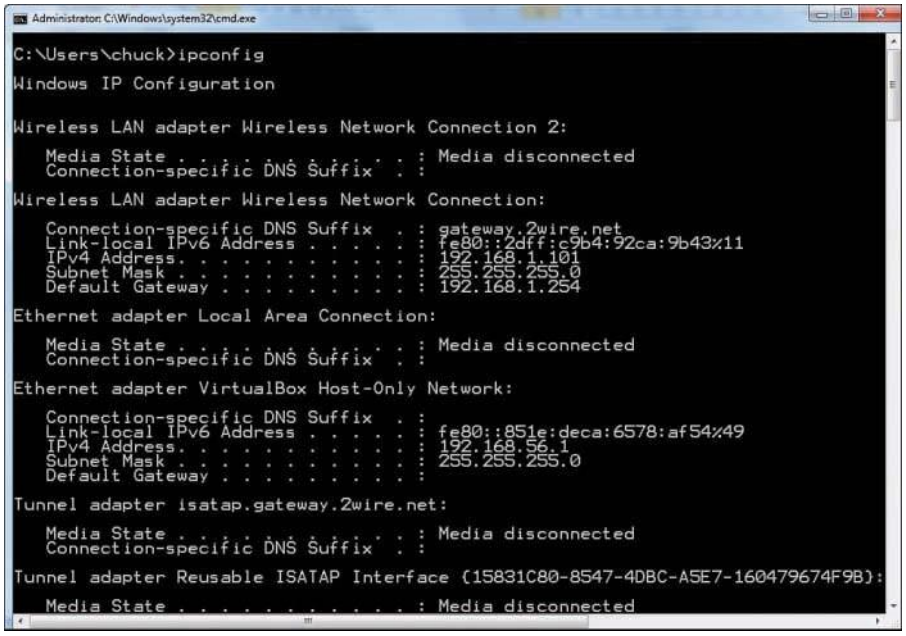
+ Các tiện ích mạng cơ bản

Bây giờ bạn đã biết địa chỉ IP và URL là gì, bạn cần phải làm quen với một số tiện ích mạng cơ bản. Bạn có thể thực thi một số tiện ích mạng từ dấu nhắc lệnh (Windows) hoặc từ trình bao (Unix/Linux). Nhiều người đọc đã quen thuộc với Windows, vì vậy cuộc thảo luận của văn bản sẽ tập trung vào cách thực thi các lệnh và

thảo luận chúng từ góc độ dấu nhắc lệnh của Windows. Tuy nhiên, cần phải nhấn mạnh rằng những tiện ích này có sẵn trong tất cả các hệ điều hành. Phần này bao gồm các tiện ích cần thiết hoặc phổ biến.

+ ipconfig

Điều đầu tiên bạn muốn làm là lấy thông tin về hệ thống của riêng bạn. Để hoàn thành nhiệm vụ tìm kiếm sự thật này, bạn phải nhận được một dấu nhắc lệnh. Trong Windows, bạn thực hiện việc này bằng cách đi tới menu Bắt đầu, chọn Tất cả chương trình, sau đó chọn Phụ kiện. Bạn cũng có thể vào Start, Run và gõ **cmd** để nhận dấu nhắc lệnh. Trong Windows 10, bạn truy cập Tìm kiếm và nhập **cmd**. Bây giờ bạn có thể nhập **ipconfig**. (Bạn có thể nhập cùng một lệnh trong Unix hoặc Linux bằng cách nhập **ifconfig** từ shell.) Sau khi nhập **ipconfig** (**ifconfig** trong Linux), bạn sẽ thấy một cái gì đó giống như Hình 1-1.

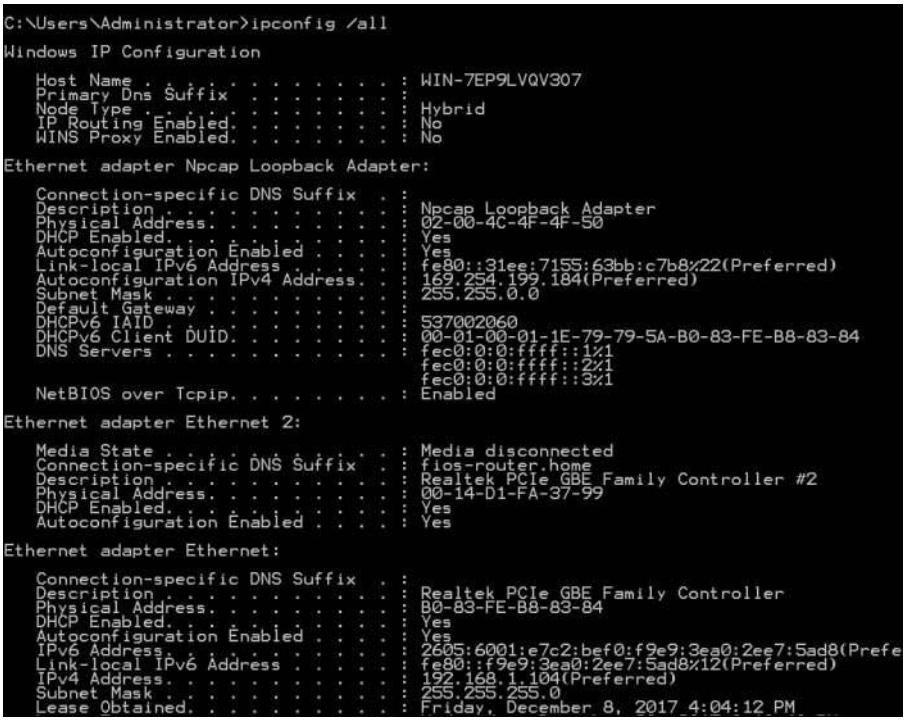


HÌNH 1-1 Lệnh ipconfig

Lệnh này cung cấp cho bạn một số thông tin về kết nối của bạn với mạng (hoặc với Internet). Quan trọng nhất là bạn tìm ra địa chỉ IP của chính mình. Lệnh này cũng có địa chỉ IP cho cổng mặc định của bạn, là kết nối của Người dùng với thế giới bên ngoài. Chạy lệnh ipconfig là bước đầu tiên để xác định cấu hình mạng hệ thống của bạn. Hầu hết các lệnh mà văn bản này đề cập, bao gồm ipconfig, có một số tham số hoặc cờ, có thể được chuyển cho các lệnh để làm cho máy tính hoạt động theo một

cách nhất định. Bạn có thể tìm hiểu các lệnh này là gì bằng cách nhập lệnh, theo sau là dấu cách, sau đó nhập dấu chấm hỏi dấu gạch ngang: **-?**.

Như bạn có thể thấy, bạn có thể sử dụng một số tùy chọn để tìm hiểu các chi tiết khác nhau về cấu hình máy tính của mình. Phương thức được sử dụng phổ biến nhất có lẽ là ipconfig/all, được hiển thị trong Hình 1-2.

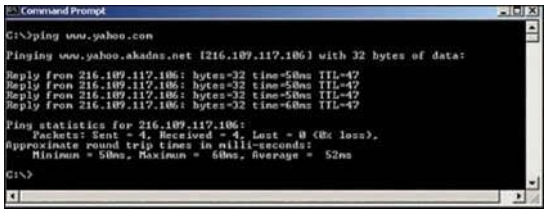


HÌNH 1-2 Lệnh ipconfig/all

Bạn có thể thấy rằng tùy chọn này cung cấp cho bạn nhiều thông tin hơn. Ví dụ: ipconfig/all cung cấp tên máy tính của bạn, khi máy tính của bạn lấy được địa chỉ IP, v.v.

+ Lệnh ping

Một lệnh khác thường được sử dụng là ping. ping được sử dụng để gửi một gói kiểm tra, hoặc gói tiếng vọng, đến một máy để tìm hiểu xem máy có thể truy cập được hay không và mất bao lâu để gói tin đến được máy. Công cụ chẩn đoán hữu ích này có thể được sử dụng trong các kỹ thuật hack cơ bản. Hình 1-3 cho thấy lệnh.



HÌNH 1-3 Lệnh ping

Hình này cho bạn biết rằng một gói echo 32 byte đã được gửi đến đích và được trả về. Ttl có nghĩa là "thời gian để sống." Đơn vị thời gian đó là bao nhiêu bước trung gian, hoặc bước nhảy, gói tin phải đi đến đích trước khi từ bỏ. Hãy nhớ rằng Internet là một tập đoàn rộng lớn của các mạng liên kết với nhau. Gói của bạn có thể sẽ không đi thẳng đến đích của nó. Nó sẽ phải mất vài bước để đến đó. Như với ipconfig, bạn có thể nhập **ping -?** để tìm ra nhiều cách khác nhau mà bạn có thể tinh chỉnh ping của mình.

1.2 Mô hình OSI

Mô hình Kết nối Hệ thống Mở (OSI) mô tả cách các mạng giao tiếp (xem Bảng 1-3). Mô tả các giao thức và hoạt động khác nhau và cho biết các giao thức và hoạt động liên quan với nhau như thế nào. Mô hình này được chia thành bảy lớp[2]. Ban đầu nó được phát triển bởi Tổ chức Tiêu chuẩn hóa Quốc tế (ISO) vào những năm 1980.

BẢNG 1-3 Mô hình OSI

Lớp	Mô tả	Các giao thức
Application	Lớp này giao diện trực tiếp với các ứng dụng và thực hiện các dịch vụ ứng dụng chung cho các quy trình ứng dụng.	POP, SMTP, DNS, FTP, Telnet
Presentation	Lớp trình bày giải tỏa mối quan tâm của lớp ứng dụng liên quan đến sự khác biệt cú pháp trong biểu diễn dữ liệu trong hệ thống người dùng cuối.	Telnet, Biểu diễn dữ liệu mạng (NDR), Giao thức trình bày nhẹ (LPP)
Session	Lớp phiên cung cấp cơ chế quản lý cuộc đối thoại giữa các quy trình ứng dụng của người dùng cuối.	NetBIOS
Transport	Lớp này cung cấp khả năng kiểm soát giao	TCP, UDP

	tiếp đầu cuối.	
Network	Lớp này định tuyến thông tin trong mạng.	IP, ARP, ICMP
Data link	Lớp này mô tả tổ chức hợp lý của các bit dữ liệu được truyền trên một phương tiện cụ thể. Lớp liên kết dữ liệu được chia thành hai lớp con: lớp Điều khiển truy cập phương tiện (MAC) và lớp Điều khiển liên kết lôgic (LLC).	SLIP, PPP
Physical	Lớp này mô tả các đặc tính vật lý của các phương tiện truyền thông khác nhau, cũng như các đặc tính điện và diễn giải các tín hiệu được trao đổi. Nói cách khác, lớp vật lý là NIC thực tế, cáp Ethernet, v.v.	IEEE 1394, DSL, ISDN

Nhiều sinh viên mạng ghi nhớ mô hình này. Ít nhất là ghi nhớ tên của bảy lớp và hiểu cơ bản những gì chúng làm là tốt. Từ góc độ bảo mật, bạn càng hiểu nhiều về truyền thông mạng, thì khả năng phòng thủ của bạn càng tinh vi hơn. Điều quan trọng nhất để bạn hiểu là mô hình này mô tả một hệ thống phân cấp của giao tiếp. Một lớp chỉ giao tiếp với lớp ngay trên nó hoặc bên dưới nó.

Điều này có ý nghĩa gì đối với an ninh mạng?

Luận văn này đề cập đến vấn đề bảo mật từ nhiều góc độ, nhưng cuối cùng chỉ có ba địa điểm tồn tại để tấn công và do đó ba địa điểm bảo mật (lưu ý đây không phải là về các vectơ tấn công, trong số đó có rất nhiều):

- **Bản thân dữ liệu:** Sau khi dữ liệu rời khỏi mạng , các gói rất dễ bị đánh chặn và thậm chí bị thay đổi. Phần sau của cuốn sách này, trong quá trình thảo luận về mã hóa và mạng riêng ảo, chúng ta sẽ học cách bảo mật dữ liệu này. Dữ liệu cũng có thể bị tấn công khi được lưu trữ trên máy tính.
- **Các điểm kết nối mạng:** Cho dù đó là bộ định tuyến hay tường lửa, bất kỳ nơi nào mà máy tính này kết nối với máy tính khác đều là nơi có thể bị tấn công và phải được bảo vệ. Khi xem xét tính bảo mật của hệ thống, trước tiên chúng ta nên xem xét các điểm kết nối.
- **Con người:** Con người luôn tiềm ẩn nguy cơ an ninh. Có thể do thiếu hiểu biết, có ý đồ xấu hoặc do lỗi đơn giản, mọi người trên hệ thống có thể xâm phạm tính bảo mật của hệ thống.

Khi bạn tiếp tục cuốn sách này, đừng quên mục đích cơ bản, đó là bảo mật các mạng và dữ liệu chúng lưu trữ và truyền.

1.3 Đánh giá các mối đe dọa có thể xảy ra đối với mạng

Trước khi có thể khám phá chủ đề bảo mật máy tính, trước tiên phải hình thành một đánh giá thực tế về các mối đe dọa đối với các hệ thống đó. Từ khóa là *thực tế*. Rõ ràng người ta có thể hình dung ra một số nguy hiểm tiềm ẩn rất phức tạp và kỹ thuật cao. Tuy nhiên, là một chuyên gia an ninh mạng, bạn phải tập trung sự chú ý và nguồn lực của mình - vào những nguy cơ có thể xảy ra. Trước khi đi sâu vào các mối đe dọa cụ thể, hãy cùng tìm hiểu về khả năng xảy ra các cuộc tấn công, thuộc bất kỳ loại nào, trên hệ thống[2] .

Về vấn đề này, dường như có hai thái độ cực đoan đối với bảo mật máy tính. Quan điểm đầu tiên cho rằng có rất ít mối nguy hiểm hoặc mối đe dọa thực sự tồn tại đối với hệ thống máy tính và phần lớn tin tức tiêu cực chỉ đơn giản là sự phản ánh của sự hoảng loạn không chính đáng. Những người có thái độ này thường nghĩ rằng chỉ thực hiện các biện pháp phòng ngừa bảo mật tối thiểu sẽ đảm bảo an toàn cho hệ thống của họ. Thật không may, một số người ở các vị trí ra quyết định lại giữ quan điểm này. Tâm lý phổ biến của những người này là, "Nếu máy tính/tổ chức của chúng tôi chưa bị tấn công cho đến nay, chúng tôi phải được bảo mật."

+ Phân loại các mối đe dọa

Mạng của bạn chắc chắn phải đối mặt với các mối đe dọa bảo mật thực sự và những mối đe dọa này có thể tự biểu hiện dưới nhiều hình thức khác nhau. Có nhiều cách khác nhau mà người ta có thể chọn để phân loại các mối đe dọa khác nhau đối với hệ thống của bạn. Bạn có thể chọn phân loại chúng theo thiệt hại gây ra, mức độ kỹ năng cần thiết để thực hiện cuộc tấn công, hoặc thậm chí có thể theo động cơ đằng sau cuộc tấn công. Vì mục đích của mình, chúng tôi phân loại các cuộc tấn công bằng những gì chúng thực sự làm. Dựa trên triết lý đó, hầu hết các cuộc tấn công có thể được phân loại thành một trong ba lớp rộng:

- Xâm nhập
- Chặn
- Phần mềm độc hại

Hình 1-6 cho thấy ba loại. Danh mục xâm nhập bao gồm các cuộc tấn công

nhằm vi phạm bảo mật và truy cập trái phép vào hệ thống. Nhóm tấn công này bao gồm bất kỳ nỗ lực nào nhằm đạt được quyền truy cập trái phép vào hệ thống. Đây thường là những gì tin tặc làm. Loại tấn công thứ hai, ngăn chặn, bao gồm các cuộc tấn công được thiết kế để ngăn chặn truy cập hợp pháp vào hệ thống. Các cuộc tấn công chặn thường được gọi là tấn công từ chối dịch vụ (hoặc đơn giản là DoS). Trong những kiểu tấn công này, mục đích không phải là thực sự xâm nhập vào hệ thống của bạn mà chỉ đơn giản là chặn người dùng hợp pháp có được quyền truy cập.

+ Từ chối dịch vụ

Loại tấn công thứ ba là ngăn chặn các cuộc tấn công, một ví dụ là tấn công từ chối dịch vụ (DoS). Trong cuộc tấn công này, kẻ tấn công không thực sự truy cập vào hệ thống mà chỉ đơn giản là chặn quyền truy cập vào hệ thống từ những người dùng hợp pháp. Theo lời của Trung tâm điều phối CERT (Nhóm ứng cứu khẩn cấp máy tính) (nhóm ứng phó sự cố bảo mật máy tính đầu tiên), “Một cuộc tấn công 'từ chối dịch vụ' được đặc trưng bởi một nỗ lực rõ ràng của những kẻ tấn công nhằm ngăn chặn những người dùng hợp pháp của một dịch vụ bằng cách sử dụng dịch vụ đó. ” Một phương pháp chặn thường được sử dụng là làm ngập hệ thống được nhắm mục tiêu với rất nhiều yêu cầu kết nối sai đến mức nó không thể phản hồi các yêu cầu hợp pháp. DoS là một phương thức tấn công cực kỳ phổ biến, chỉ đứng sau phần mềm độc hại.

1.4 Chọn Phương pháp Tiếp cận An ninh Mạng

Các tổ chức có thể chọn từ một số cách tiếp cận đối với an ninh mạng. Một cách tiếp cận hoặc mô hình cụ thể sẽ ảnh hưởng đến tất cả các quyết định bảo mật tiếp theo và thiết lập âm thanh cho cơ sở hạ tầng an ninh mạng của toàn tổ chức. Các mô hình an ninh mạng có thể được phân loại theo phạm vi của các biện pháp bảo mật được thực hiện (ngoại vi, phân lớp) hoặc mức độ chủ động của hệ thống[2].

+ Phương pháp tiếp cận bảo mật ngoại vi

Trong cách tiếp cận bảo mật theo chu vi, phần lớn các nỗ lực bảo mật đều tập trung vào chu vi của mạng. Trọng tâm này có thể bao gồm tường lửa, máy chủ proxy, chính sách mật khẩu và bất kỳ công nghệ hoặc quy trình nào khiến khả năng truy cập trái phép vào mạng ít xảy ra hơn. Ít hoặc không có nỗ lực nào được thực hiện để bảo mật các hệ thống trong mạng. Trong cách tiếp cận này, vành đai được bảo đảm, nhưng

các hệ thống khác nhau trong vành đai đó thường dễ bị tổn thương.

Cách tiếp cận chu vi này rõ ràng là thiếu sót. Vậy tại sao một số công ty lại sử dụng nó? Một tổ chức nhỏ có thể sử dụng cách tiếp cận chu vi nếu họ gặp khó khăn về ngân sách hoặc quản trị viên mạng thiếu kinh nghiệm. Phương pháp này có thể phù hợp với các tổ chức nhỏ không lưu trữ dữ liệu nhạy cảm, nhưng nó hiếm khi hoạt động trong môi trường công ty lớn hơn.

1.5 Kết luận chương

Các mối đe dọa đối với mạng ngày càng tăng. Chúng tôi đang thấy sự gia tăng về số lượng các cuộc tấn công hack và vi rút, cũng như các hình thức tấn công khác. Mối nguy ngày càng gia tăng này cùng với áp lực pháp lý ngày càng tăng (chẳng hạn như HIPAA và SOX) và các quản trị viên mạng có nhu cầu ngày càng cao về an ninh mạng.

Chương này đã giới thiệu cho bạn các khái niệm cơ bản về an ninh mạng, các lớp nguy hiểm chung và thuật ngữ bảo mật cơ bản. Các chương tiếp theo sẽ trình bày chi tiết về thông tin này.

CHƯƠNG 2 TẤN CÔNG MẠNG

Mục tiêu của chương

- Mô tả các cuộc tấn công mạng phổ biến nhất, bao gồm tấn công Session, tấn công vi rút, phần mềm mã độc Trojan, từ chối dịch vụ và tràn bộ đệm.
- Giải thích cách thực hiện các cuộc tấn công này.
- Xác định các biện pháp phòng thủ cơ bản chống lại các cuộc tấn công đó.
- Cấu hình hệ thống để ngăn chặn các cuộc tấn công từ chối dịch vụ.
- Cấu hình một hệ thống để bảo vệ chống lại các cuộc tấn công.
- Định cấu hình hệ thống để bảo vệ khỏi các cuộc tấn công tràn bộ đệm.

2.1 Giới thiệu

Chương 1, “Giới thiệu về An ninh mạng”, đã giới thiệu một số mối nguy hiểm chung đối với hệ thống máy tính và cung cấp tổng quan về an ninh mạng. Chương này xem xét các loại tấn công cụ thể kỹ hơn nhiều. Phân tích cách các hệ thống bị tấn công phổ biến nhất. Đặc biệt sẽ chú ý đến cuộc tấn công từ chối dịch vụ (DoS). Mối đe dọa này là một trong những phương thức tấn công phổ biến nhất trên Internet, vì vậy hiểu cách thức hoạt động và cách bảo vệ hệ thống chống lại chúng là điều cần thận trọng đối với các quản trị viên.

Chương này cũng mô tả các cuộc tấn công bằng vi rút, cuộc tấn công bằng phần mềm mã độc Trojan và một số phương pháp tấn công ít phổ biến hơn, chẳng hạn như tấn công Session và đào đường hầm. Trong bảo mật thông tin, câu ngạn ngữ cổ "kiến thức là sức mạnh" không chỉ là lời khuyên tốt mà còn là tiên đề để xây dựng toàn bộ triển vọng bảo mật.

+ Hiểu các cuộc tấn công từ chối dịch vụ

Loại tấn công đầu tiên cần kiểm tra là từ chối dịch vụ (DoS). Nhắc lại từ Chương 1 rằng tấn công từ chối dịch vụ là bất kỳ cuộc tấn công nào nhằm mục đích tước quyền sử dụng hệ thống mục tiêu của người dùng hợp pháp. Loại tấn công này không thực sự cố gắng xâm nhập vào hệ thống hoặc lấy thông tin nhạy cảm. Nó chỉ nhằm mục đích ngăn người dùng hợp pháp truy cập vào một hệ thống nhất định. Kiểu tấn công này là một trong những kiểu tấn công phổ biến nhất. Nhiều chuyên gia cho rằng nó quá phổ biến vì hầu hết các hình thức tấn công từ chối dịch vụ đều khá dễ thực

hiện. Sự dễ dàng mà các cuộc tấn công này có thể được thực hiện có nghĩa là ngay cả những kẻ tấn công với kỹ năng kỹ thuật tối thiểu thường có thể thực hiện thành công việc từ chối dịch vụ.

Khái niệm cơ bản của cuộc tấn công từ chối dịch vụ dựa trên thực tế là bất kỳ thiết bị nào cũng có giới hạn hoạt động. Thực tế này áp dụng cho tất cả các thiết bị, không chỉ hệ thống máy tính. Ví dụ, các cây cầu được thiết kế để giữ trọng lượng ở một giới hạn nhất định, máy bay có giới hạn về khoảng cách chúng có thể di chuyển mà không cần tiếp nhiên liệu, và ô tô chỉ có thể tăng tốc đến một điểm nhất định. Tất cả các thiết bị khác nhau này đều có chung một đặc điểm: Chúng có những giới hạn về năng lực thực hiện công việc. Máy tính không khác gì những máy này, hoặc bất kỳ máy nào khác; chúng cũng có giới hạn. Bất kỳ hệ thống máy tính, máy chủ web hoặc mạng nào cũng chỉ có thể xử lý một tải hữu hạn[2].

Cách xác định khối lượng công việc (và các giới hạn của nó) khác nhau giữa các máy này với máy khác. Khối lượng công việc cho một hệ thống máy tính có thể được xác định theo một số cách khác nhau, bao gồm theo số lượng người dùng phổ biến, kích thước tệp, tốc độ truyền dữ liệu hoặc lượng dữ liệu được lưu trữ. Vượt quá bất kỳ giới hạn nào trong số này sẽ khiến hệ thống ngừng phản hồi. Ví dụ: nếu bạn có thể làm ngập một máy chủ web với nhiều yêu cầu hơn mức nó có thể xử lý, nó sẽ bị quá tải và không còn có thể phản hồi các yêu cầu khác nữa. Thực tế này làm nền tảng cho cuộc tấn công DoS. Chỉ cần làm quá tải hệ thống với các yêu cầu và nó sẽ không còn có thể phản hồi những người dùng hợp pháp đang cố gắng truy cập vào máy chủ web.

+ DoS in Action

Khái niệm về một cuộc tấn công từ chối dịch vụ rất đơn giản; tuy nhiên, hầu hết các nguyên tắc sẽ dễ nắm bắt hơn nếu người ta có thể xem một ví dụ cụ thể. Trong trường hợp này, bạn cần một cách an toàn để mô phỏng một cuộc tấn công DoS trong môi trường lớp học hoặc phòng thí nghiệm. Một cách đơn giản để minh họa một cuộc tấn công DoS, đặc biệt là trong bối cảnh lớp học, liên quan đến việc sử dụng lệnh ping cùng với một số tham số nhất định. (Nhớ lại rằng gõ **ping/h** hoặc **ping**/? Sẽ hiển thị cho bạn tất cả các tùy chọn cho lệnh ping.) Bước đầu tiên là khởi động dịch vụ máy

chủ web chạy trên máy tính sẽ được sử dụng làm mục tiêu cho cuộc tấn công này. Bạn có thể sử dụng bất kỳ hệ thống hoạt động nào và bất kỳ máy chủ web nào bạn thích (chẳng hạn như Microsoft Internet Information Services hoặc Apache HTTP Server). Apache là bản tải xuống miễn phí từ www.apache.org. Microsoft Windows 10 đi kèm với Dịch vụ Thông tin Internet, vì vậy bạn sẽ không gặp khó khăn gì khi tìm một máy chủ web để cài đặt và chạy[3]. Đối với mục đích của phòng thí nghiệm này, bạn muốn sử dụng có mục đích máy công suất thấp. Một chiếc máy cũ hơn, có lẽ là một chiếc máy tính xách tay cũ hơn, sẽ là lý tưởng. Bạn muốn chọn một máy sẽ dễ bị quá tải. Về bản chất, bạn đang tìm kiếm điều hoàn toàn ngược lại với những gì bạn tìm kiếm khi thiết lập một máy chủ web thực.

Thiết lập một máy chủ web thực sự khá đơn giản. Vì Apache có sẵn dưới dạng bản tải xuống miễn phí cho cả Linux và Windows (www.apache.org), chúng ta hãy kiểm tra nó. Làm theo các bước sau để cài đặt và cấu hình Apache trên hệ thống của bạn.

Cài đặt cho Windows

1. Tải xuống Apache cho Windows từ www.apache.org .
2. Tìm trong C: \ Program Files \ Apache Group \ Apache2 \ conf để tìm tệp httpd.conf và mở nó.
3. Đặt ServerName = localhost.
4. Lưu tệp.
5. Từ dấu nhắc lệnh, gõ **httpd start**.

Bây giờ bạn có thể mở trình duyệt và xem trang web Apache mặc định.

Cài đặt cho Linux

1. Tải xuống Apache cho Linux (hoặc trong nhiều bản phân phối Linux, bạn có thể chỉ cần thêm gói máy chủ web Apache) từ www.apache.org .
2. Tìm tệp httpd.conf trong / etc / httpd / conf. Khi bạn tìm thấy nó, hãy nhấp chuột phải vào nó và mở nó bằng trình soạn thảo văn bản.
3. Đặt ServerName = localhost.
4. Lưu tệp.
5. Từ một shell, gõ **/etc/init.d/httpd start**. Máy chủ sẽ khởi động và bạn nhận được thông báo OK.

6. Mở trình duyệt của bạn và truy cập http: // localhost /.

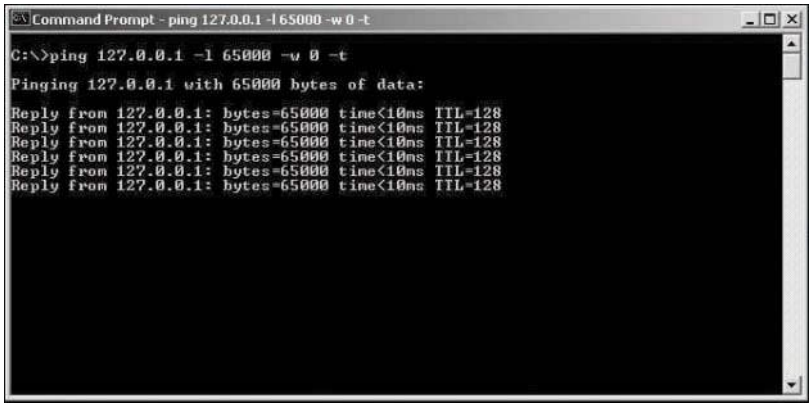
Bạn sẽ thấy trang web mặc định của Apache.

Khi bạn đã sẵn sàng đặt máy chủ của mình hoạt động (có thể truy cập được từ các PC khác), trong tệp /etc/httpd/conf/httpd.conf, hãy thay đổi các cài đặt sau:

Nếu bạn đang sử dụng các phiên bản Máy chủ Windows 7 hoặc 2008 hoặc 2012, bạn cũng có thể chọn sử dụng Dịch vụ Thông tin Internet của Microsoft làm máy chủ web của mình.

Bạn thực hiện cuộc tấn công thực sự bằng lệnh ping. Nếu bạn không nhớ cách sử dụng lệnh ping, bạn nên lưu ý rằng việc nhập **ping/h** tại dấu nhắc lệnh sẽ hiển thị tất cả các tùy chọn cho lệnh ping. Các tùy chọn để sử dụng trong bài tập này là -w và -t. Tùy chọn -w xác định bao nhiêu mili giây tiện ích ping sẽ đợi phản hồi từ mục tiêu. Trong trường hợp này, hãy đặt tùy chọn đó thành -0, vì vậy nó không chờ đợi gì cả. Tùy chọn -t hướng dẫn tiện ích ping tiếp tục gửi các gói cho đến khi được yêu cầu dừng một cách rõ ràng[3]. Một tùy chọn bổ sung, tùy chọn -l, cho phép người dùng thay đổi kích thước của gói mà bạn có thể gửi. Hãy nhớ rằng một gói TCP chỉ có thể có kích thước hữu hạn, vì vậy bạn sẽ đặt các gói này gần như lớn nhất mà bạn có thể gửi.

Tại dấu nhắc lệnh trong Windows 10 (đó là trình bao trong Unix / Linux), nhập **ping** *<địa chỉ của máy đích ở đây>* **-l 65000 -w 0 -t**. Phản hồi của máy phải tương tự như phản hồi trong Hình 2-1. Lưu ý rằng trong hình, tôi đang ping địa chỉ loopback cho máy của chính tôi. Bạn sẽ muốn thay thế địa chỉ của máy mà bạn đang chạy máy chủ web.



HÌNH 2-1 Ping từ dấu nhắc lệnh

Điều gì đang xảy ra khi chuỗi ping này đang được thực thi là một máy duy nhất này liên tục ping đến máy mục tiêu. Tại thời điểm này, việc chỉ có một máy trong lớp học hoặc phòng thí nghiệm ping trên máy chủ web sẽ không ảnh hưởng xấu đến máy chủ web. Điều này là do mức lưu lượng đó nằm trong khả năng của máy chủ web mục tiêu. Tuy nhiên, sau khi khiến các máy khác ping máy chủ theo cách tương tự, bạn sẽ bắt đầu đánh thuế dung lượng của máy mục tiêu. Nếu bạn có đủ số máy ping đến mục tiêu, cuối cùng bạn sẽ đạt đến ngưỡng mà máy mục tiêu sẽ ngừng phản hồi các yêu cầu và bạn sẽ không thể truy cập trang web được nữa. Số lượng máy sẽ cần để đạt đến ngưỡng này tùy thuộc vào máy chủ web bạn đang sử dụng. Tác giả này đã tiến hành thí nghiệm cụ thể này trong các lớp học. Trong những tình huống đó, máy chủ web Apache đang được chạy trên máy tính xách tay Pentium III chạy Windows 7, chỉ với 1 gigabyte. Trong trường hợp đó, chỉ cần khoảng 25 máy đồng thời ping để làm cho máy chủ web ngừng phản hồi các yêu cầu hợp pháp. Ngay cả khi thử nghiệm này không làm hỏng máy, ít nhất nó sẽ khiến nó phản hồi chậm hơn.

DDoS đang trở nên phổ biến hơn; trên thực tế, nó hiện là kiểu tấn công DoS phổ biến nhất. Hầu hết các ví dụ trong thế giới thực mà chúng ta sẽ xem xét ở phần sau của chương này là các cuộc tấn công DDoS. Hai lý do mà hình thức tấn công từ chối dịch vụ này đang trở nên phổ biến hơn bao gồm:

- Quá tải một hệ thống mục tiêu sẽ dễ thực hiện hơn nếu bạn có nhiều hơn một máy đang tấn công nó. Với các máy chủ mới hơn có khả năng xử lý khối lượng công việc cao hơn nhiều, việc thực hiện một cuộc tấn công DoS chỉ từ một máy trở nên khó khăn hơn.
- Nó cho phép kẻ tấn công thực hiện cuộc tấn công từ máy của người khác, do đó bảo vệ tính ẩn danh của anh ta. Việc khởi động một cuộc tấn công từ chính máy của một người có thể gặp rủi ro vì mỗi gói tin đều có khả năng bị truy ngược về nguồn của nó. Điều này có nghĩa là gần như chắc chắn bị bắt bởi các cơ quan chức năng.

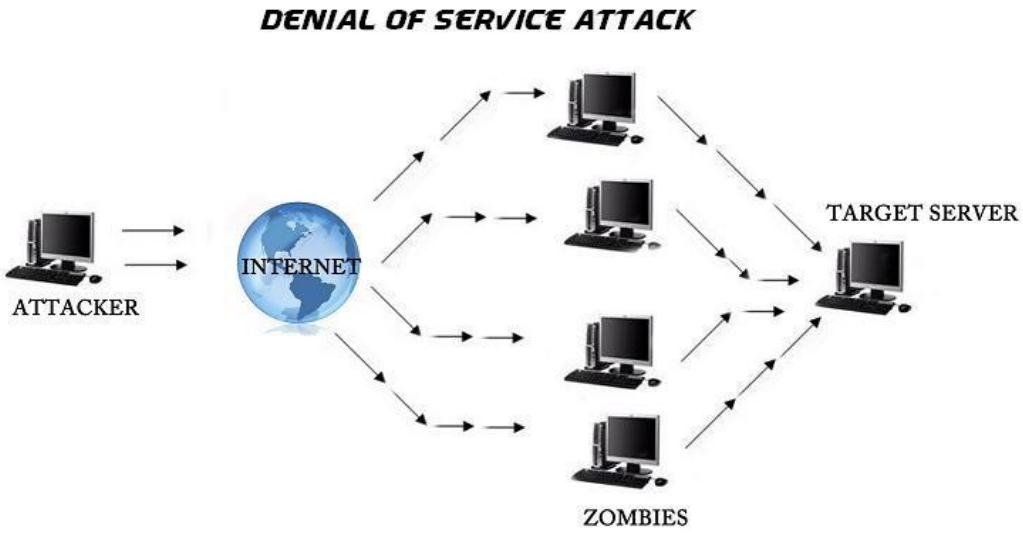
Khái niệm cơ bản đằng sau một cuộc tấn công DoS rất đơn giản. Vấn đề thực sự đối với kẻ tấn công là tránh bị bắt. Phần tiếp theo xem xét một số kiểu tấn công DoS cụ thể và xem xét các nghiên cứu điển hình cụ thể.

+ Bảo vệ bằng Các khối vi mô (Micro Block)

Các khối vi mô tìm cách tránh SYN Lụt bằng cách thay đổi cách máy chủ phân bổ bộ nhớ cho bất kỳ yêu cầu kết nối nhất định nào. Thay vì cấp phát một đối tượng kết nối hoàn chỉnh, máy chủ được thay đổi để nó chỉ cấp phát một bản ghi vi mô. Các triển khai mới hơn của kỹ thuật này phân bổ ít nhất là 16 byte cho đối tượng SYN đến. Các chi tiết cụ thể về cách thiết lập các khối vi mô là dành riêng cho một hệ điều hành nhất định[4]. Đây là một kỹ thuật ít phổ biến hơn. Nhiều quản trị viên mạng thậm chí không nhận thức được khả năng này.

+ Tấn công Smurf

Cuộc tấn công Smurf là một kiểu tấn công DoS phổ biến. Nó được đặt tên theo ứng dụng lần đầu tiên được sử dụng để thực hiện cuộc tấn công này. ách tấn công này vừa thông minh vừa đơn giản. Khó khăn lớn nhất là bắt đầu các gói trên mạng đích. Điều này có thể được thực hiện thông qua một số phần mềm như vi-rút hoặc phần mềm mã độc Trojan sẽ bắt đầu gửi các gói tin. Hình 2-4 minh họa cuộc tấn công này.



HÌNH 2-2 Cuộc tấn công Smurf

Cuộc tấn công của Smurf là một ví dụ về sự sáng tạo mà một số bên độc hại có thể sử dụng. Đôi khi nó được coi là tương đương kỹ thuật số của quá trình sinh học trong bệnh rối loạn tự miễn dịch. Với những rối loạn như vậy, hệ thống miễn dịch sẽ tấn công chính cơ thể của bệnh nhân. Trong một cuộc tấn công Smurf, mạng thực hiện một cuộc tấn công DoS trên một trong các hệ thống của chính nó. Sự thông minh của

phương pháp này cho thấy lý do tại sao điều quan trọng là bạn phải cố gắng làm việc một cách sáng tạo và có tư duy cầu tiến nếu bạn chịu trách nhiệm về bảo mật hệ thống trong mạng của mình. Thủ phạm của các cuộc tấn công máy tính là những người sáng tạo, liên tục phát triển các kỹ thuật mới. Nếu cách phòng thủ của bạn kém sáng tạo và khôn khéo hơn so với hành vi tấn công của kẻ tấn công, thì việc hệ thống của bạn bị xâm phạm chỉ là vấn đề thời gian. Bạn có thể bảo vệ khỏi cuộc tấn công của Smurf theo hai cách:

- Phương pháp trực tiếp nhất là cấu hình tất cả các bộ định tuyến của bạn để chúng không chuyển tiếp bất kỳ gói tin quảng bá trực tiếp nào. Các gói tin này là nền tảng của cuộc tấn công Smurf, và nếu các bộ định tuyến không chuyển tiếp chúng, thì cuộc tấn công được chứa trong một mạng con[4].
- Cách tiếp cận thứ hai được đề cập sâu hơn ở phần sau của chương này. Bởi vì cuộc tấn công Smurf được khởi chạy từ phần mềm được phân phối thông qua một phần mềm mã độc Trojan, việc ngăn chặn việc phân phối ban đầu đó sẽ ngăn chặn cuộc tấn công[5]. Các chính sách cấm nhân viên tải xuống ứng dụng và bảo vệ hệ thống bằng máy quét vi rút phù hợp có thể đi một chặng đường dài để bảo vệ hệ thống khỏi phần mềm mã độc Trojan, và do đó là cuộc tấn công của Smurf.

Sử dụng máy chủ proxy cũng là điều bắt buộc. Máy chủ proxy có thể ẩn địa chỉ IP nội bộ của máy tính của bạn, điều này giúp hệ thống của bạn ít bị tấn công hơn rất nhiều trước cuộc tấn công của Smurf, khám phá chi tiết các máy chủ proxy và tường lửa, một công cụ quan trọng khác.

+ Từ chối dịch vụ phản ánh phân tán

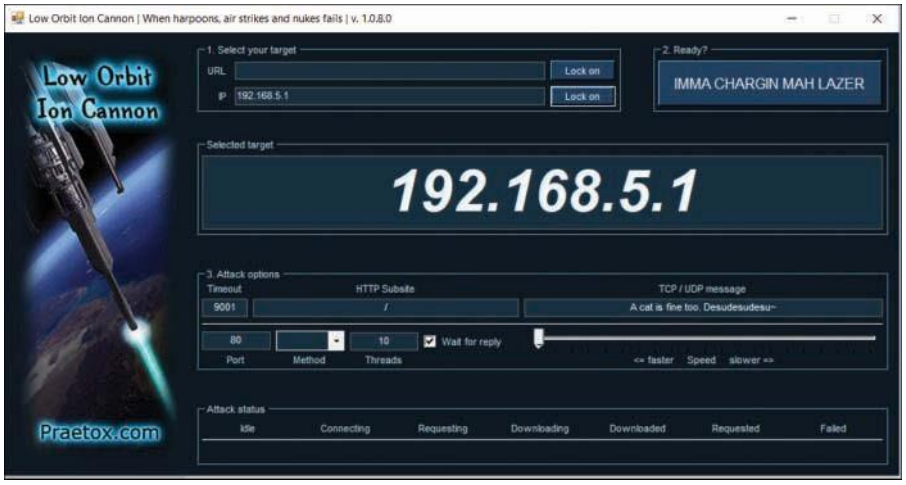
Như đã nêu trước đây, các cuộc tấn công từ chối dịch vụ phân tán đang trở nên phổ biến hơn. Hầu hết các cuộc tấn công như vậy dựa vào việc có được nhiều máy khác nhau (máy chủ hoặc máy trạm) để tấn công mục tiêu. Từ chối dịch vụ phản ánh phân tán (DRDoS) là một kiểu tấn công DoS đặc biệt. Như với tất cả các cuộc tấn công như vậy, nó được thực hiện bằng cách hacker có được một số máy để tấn công mục tiêu đã chọn. Tuy nhiên, cuộc tấn công này hoạt động hơi khác so với các cuộc tấn công DoS khác. Thay vì khiến máy tính tấn công mục tiêu, phương pháp này đánh lừa các bộ định tuyến Internet tấn công mục tiêu.

Một lý do khiến các cuộc tấn công DoS ngày càng trở nên phổ biến là do một số công cụ có sẵn để thực hiện các cuộc tấn công DoS. Các công cụ này có sẵn rộng rãi trên Internet và trong hầu hết các trường hợp đều được tải xuống miễn phí. Điều này có nghĩa là bất kỳ quản trị viên thận trọng nào cũng nên biết về chúng. Ngoài việc sử dụng rõ ràng chúng như một công cụ tấn công, chúng cũng có thể hữu ích để kiểm tra các biện pháp bảo mật chống DoS[5].

+ Tấn công Low Orbit Ion Cannon (LOIC)

Đây có lẽ là công cụ nổi tiếng nhất, và chắc chắn là một trong những công cụ DoS đơn giản nhất ở bất cứ đâu. Một tìm kiếm đơn giản trên Internet sẽ hiển thị cho bạn nhiều trang web mà bạn có thể tải xuống LOIC từ đó.

Trước tiên, bạn đặt URL hoặc địa chỉ IP vào hộp đích. Sau đó nhấp vào nút Lock On. Bạn có thể thay đổi cài đặt liên quan đến phương pháp bạn chọn, tốc độ, bao nhiêu luồng và có đợi trả lời hay không. Sau đó, chỉ cần nhấp vào nút IMMA CHARGIN MAH LAZER và cuộc tấn công đang được tiến hành.



HÌNH 2-3 LOIC

+ Bảo vệ chống lại các cuộc tấn công DoS

Không có cách nào đảm bảo tồn tại để ngăn chặn tất cả các cuộc tấn công DoS, cũng như không có cách nào đảm bảo tồn tại để ngăn chặn bất kỳ nỗ lực hack hoặc một cuộc tấn công mạng nào. Tuy nhiên, bạn có thể thực hiện các bước để giảm thiểu nguy hiểm. Phần này xem xét một số bước mà quản trị viên có thể thực hiện để làm cho hệ thống của họ ít bị tấn công DoS hơn ngoài việc sử dụng các cookie SYN và RST đã được thảo luận trước đó.

Một trong những điều đầu tiên cần xem xét là cách thức thực hiện các cuộc tấn công này. Chúng có thể được thực thi thông qua các gói ICMP được sử dụng để gửi thông báo lỗi trên Internet hoặc được gửi bởi các tiện ích ping và theo dõi. Đơn giản chỉ cần cấu hình tường lửa của bạn để từ chối các gói ICMP từ bên ngoài mạng của bạn sẽ là một bước quan trọng trong việc bảo vệ mạng của bạn khỏi các cuộc tấn công DoS. Vì các cuộc tấn công DoS / DDoS có thể được thực hiện thông qua nhiều giao thức khác nhau, bạn cũng có thể định cấu hình tường lửa của mình để không cho phép bất kỳ lưu lượng nào đến, bất kể nó xảy ra trên giao thức hoặc cổng nào. Đây có vẻ là một bước cấp tiến, nhưng nó chắc chắn là một bước an toàn.

Nếu mạng của bạn đủ lớn để có các bộ định tuyến nội bộ, thì bạn có thể định cấu hình các bộ định tuyến đó để không cho phép bất kỳ lưu lượng nào không bắt nguồn từ mạng của bạn. Theo cách đó, nếu các gói vượt qua được tường lửa của bạn, chúng sẽ không được truyền khắp mạng. Bởi vì tất cả các gói TCP đều có địa chỉ IP nguồn, việc xác định xem một gói có nguồn gốc trong mạng hay từ bên ngoài mạng không khó. Một khả năng khác là vô hiệu hóa các chương trình phát sóng IP được định hướng trên tất cả các bộ định tuyến. Điều này ngăn không cho bộ định tuyến gửi các gói quảng bá đến tất cả các máy trong mạng, do đó ngăn chặn nhiều cuộc tấn công DoS.

Vì nhiều cuộc tấn công DoS phân tán phụ thuộc vào việc các máy tính "không chủ ý" được sử dụng làm điểm khởi chạy, một cách để giảm các cuộc tấn công như vậy là bảo vệ máy tính của bạn chống lại các cuộc tấn công của vi-rút / sâu và phần mềm mã độc Trojan. Bảo vệ chống lại những cuộc tấn công này sẽ được thảo luận ở phần sau của chương này, nhưng hiện tại ba điểm quan trọng cần nhớ là:

- Luôn sử dụng phần mềm quét vi-rút và cập nhật phần mềm này.
- Luôn cập nhật các bản vá hệ điều hành và phần mềm.
- Có chính sách tổ chức quy định rằng nhân viên không thể tải bất cứ thứ gì xuống máy của họ trừ khi nhân viên CNTT đã xóa nội dung tải xuống.

Không có bước nào trong số những bước này sẽ giúp mạng của bạn hoàn toàn an toàn khỏi trở thành nạn nhân của một cuộc tấn công DoS hoặc là điểm khởi động cho một cuộc tấn công, nhưng chúng sẽ giúp giảm khả năng xảy ra một trong hai bước

này. Một nguồn tốt cho chủ đề này là trang web của Viện SANS tại www.sans.org/dosstep/. Trang web này có nhiều mẹo hay về việc ngăn chặn các cuộc tấn công DoS.

+ Bảo vệ chống lại các cuộc tấn công tràn bộ đệm

Các cuộc tấn công bằng virus, DoS và Phần mềm mã độc Trojan có lẽ là những cách phổ biến nhất để tấn công hệ thống, nhưng chúng không phải là những phương pháp tấn công duy nhất hiện có. Một cách khác để tấn công hệ thống được gọi là tấn công tràn bộ đệm (hay tràn bộ đệm). Một số chuyên gia cho rằng tràn bộ đệm xảy ra thường xuyên, nếu không muốn nói là thường xuyên hơn so với cuộc tấn công DoS, nhưng điều này hiện nay ít đúng hơn so với cách đây vài năm. Tấn công tràn bộ đệm được thiết kế để đưa nhiều dữ liệu vào bộ đệm hơn so với bộ đệm được thiết kế để lưu giữ. Tuy nhiên, hãy nhớ lại rằng ít nhất một con sâu đã sử dụng tràn bộ đệm để lây nhiễm cho các máy được nhắm mục tiêu. Điều này có nghĩa là mặc dù mối đe dọa này có thể ít hơn trước đây, nhưng nó vẫn là một mối đe dọa rất thực tế[5].

Bất kỳ chương trình nào giao tiếp với Internet hoặc mạng riêng đều phải nhận một số dữ liệu. Dữ liệu này được lưu trữ, ít nhất là tạm thời, trong một không gian trong bộ nhớ được gọi là bộ đệm. Nếu lập trình viên đã viết ứng dụng cẩn thận, bộ đệm sẽ cắt bớt hoặc từ chối bất kỳ thông tin nào vượt quá giới hạn bộ đệm. Với số lượng ứng dụng có thể đang chạy trên hệ thống đích và số lượng bộ đệm trong mỗi ứng dụng, khả năng có ít nhất một bộ đệm không được ghi đúng cách là đủ đáng kể để gây ra bất kỳ quản trị viên hệ thống thận trọng nào. Một người có kỹ năng lập trình vừa phải có thể viết một chương trình có mục đích ghi nhiều dữ liệu vào bộ đệm hơn mức có thể chứa. Ví dụ: nếu bộ đệm có thể chứa 1024 byte dữ liệu và bạn cố gắng lấp đầy nó với 2048 byte, thì 1024 byte bổ sung sau đó chỉ được tải vào bộ nhớ.

Nếu dữ liệu bổ sung thực sự là một chương trình độc hại, thì nó vừa được tải vào bộ nhớ và đang chạy trên hệ thống đích. Hoặc có lẽ thủ phạm chỉ muốn làm ngập bộ nhớ của máy mục tiêu, do đó ghi đè các mục khác hiện có trong bộ nhớ và khiến chúng bị hỏng. Dù bằng cách nào, tràn bộ đệm là một cuộc tấn công rất nghiêm trọng.

May mắn thay, các cuộc tấn công tràn bộ đệm khó thực hiện hơn một chút so với DoS hoặc virus tập lệnh MS Outlook đơn giản. Để tạo ra một cuộc tấn công tràn bộ đệm,

một tin tặc phải có kiến thức làm việc tốt về một số ngôn ngữ lập trình (C hoặc C ++ thường được chọn) và hiểu hệ điều hành/ứng dụng mục tiêu đủ để biết liệu nó có điểm yếu về tràn bộ đệm hay không và làm thế nào anh ấy có thể khai thác điểm yếu.

Khả năng bị tấn công tràn bộ đệm hoàn toàn phụ thuộc vào các lỗi phần mềm. Một chương trình được viết hoàn hảo sẽ không cho phép tràn bộ đệm. Bởi vì sự hoàn hảo là không thể, cách phòng thủ tốt nhất chống lại các cuộc tấn công tràn bộ đệm là thường xuyên vá phần mềm để các lỗ hổng được sửa chữa khi nhà cung cấp phát hiện ra lỗ hổng..

+ Bảo vệ chống lại xâm nhập Session

Một hình thức tấn công khác là tấn công Session hoặc chiếm quyền điều khiển. Chiếm quyền điều khiển phiên TCP là một quá trình mà tin tặc chiếm quyền điều khiển phiên TCP giữa hai máy. Bởi vì xác thực thường chỉ được thực hiện khi bắt đầu phiên TCP, điều này cho phép hacker xâm nhập vào luồng giao tiếp và kiểm soát phiên. Ví dụ, một người có thể đăng nhập vào một máy tính từ xa. Sau khi cô ấy đã thiết lập kết nối với máy chủ, tin tặc có thể sử dụng tấn công Session để chiếm lấy phiên đó và do đó có quyền truy cập vào máy mục tiêu.

Một phương pháp phổ biến để tấn công Session là sử dụng các gói IP được định tuyến nguồn. Điều này cho phép tin tặc tại điểm A trên mạng tham gia vào cuộc trò chuyện giữa B và C bằng cách khuyến khích các gói IP đi qua máy của tin tặc.

Loại tấn công Session phổ biến nhất là “tấn công man-in-the-middle”. Trong trường hợp này, một tin tặc sử dụng một số loại chương trình đánh hơi gói tin để chỉ cần nghe các đường truyền giữa hai máy tính, lấy bất kỳ thông tin nào mà anh ta hoặc cô ta muốn nhưng không thực sự làm gián đoạn cuộc trò chuyện.

Một thành phần phổ biến của một cuộc tấn công như vậy là thực hiện một cuộc tấn công DoS chống lại một điểm cuối để ngăn nó phản hồi. Bởi vì điểm cuối đó không còn phản hồi nữa, tin tặc giờ đây có thể can thiệp vào máy của chính mình để duy trì điểm cuối đó. Mục đích của việc chiếm quyền điều khiển kết nối là khai thác sự tin tưởng và giành quyền truy cập vào một hệ thống mà nếu không thì một hệ thống sẽ không có quyền truy cập.

2.2 Các Virút

Theo định nghĩa, virus máy tính là một chương trình tự sao chép. Nói chung, virus cũng có một số chức năng khó chịu khác, nhưng khả năng tự sao chép và lây lan nhanh chóng là những dấu hiệu nổi bật của nó.

Hãy xem xét virus Slammer khét tiếng và tác động của việc quét nhanh, khối lượng lớn của nó. Bất kỳ loại vi-rút nào lây lan nhanh đều có thể làm giảm chức năng và khả năng đáp ứng của mạng[7]. Nó có thể dẫn đến quá nhiều lưu lượng mạng và ngăn mạng hoạt động bình thường. Chỉ đơn giản bằng cách vượt quá tải lưu lượng mà một mạng được thiết kế để thực hiện, mạng có thể tạm thời không hoạt động.

+ Virus lây lan như thế nào?

Bạn đã thấy vi-rút có thể tác động đến hệ thống bị nhiễm như thế nào và đã xem xét một vài trường hợp thực tế. Rõ ràng chìa khóa để ngăn chặn vi rút máy tính là ngăn không cho nó lây lan sang các máy tính khác. Để làm được điều này, bạn phải hiểu rõ về cách thức lây lan của vi rút. Virus thường lây lan theo một trong hai cách.

Đầu tiên là quét máy tính để tìm các kết nối với mạng và sau đó sao chép chính nó sang các máy khác trên mạng mà máy đó có quyền truy cập. Đây là cách hiệu quả nhất để vi rút lây lan và là phương thức lây lan điển hình của sâu. Tuy nhiên, phương pháp này đòi hỏi kỹ năng lập trình cao hơn các phương pháp khác. Phương pháp thứ hai và phổ biến hơn là đọc sổ địa chỉ e-mail và tự gửi cho mọi người trong đó. Lập trình cho loại virus này là một công việc nhỏ nhặt, điều này giải thích tại sao việc sử dụng nó lại phổ biến như vậy.

Phương pháp thứ hai, cho đến nay, là phương pháp phổ biến nhất để lây lan vi-rút và Microsoft Outlook có thể là một chương trình e-mail thường gặp phải các cuộc tấn công vi-rút như vậy. Lý do không phải là quá nhiều lỗi bảo mật trong Outlook vì nó là sự dễ dàng làm việc với Outlook. Tất cả các sản phẩm Microsoft Office đều được tạo ra để một lập trình viên hợp pháp có thể truy cập vào nhiều đối tượng bên trong của ứng dụng đó và do đó dễ dàng tạo các ứng dụng tích hợp các ứng dụng trong bộ Microsoft Office. Ví dụ: một lập trình viên có thể viết một ứng dụng truy cập tài liệu Word, nhập bảng tính Excel, sau đó sử dụng Outlook để tự động gửi tài liệu kết quả qua e-mail cho các bên quan tâm. Microsoft đã làm rất tốt khi làm cho quá trình này

trở nên rất dễ dàng. Để hoàn thành các tác vụ này thường chỉ cần một lượng lập trình tối thiểu. Trong trường hợp của Outlook, việc tham chiếu Outlook và gửi e-mail cần ít hơn năm dòng mã để thực hiện[7]. Điều này có nghĩa là một chương trình thực sự có thể khiến chính Outlook gửi e-mail mà người dùng không hề hay biết. Nhiều ví dụ mã trên Internet cho thấy chính xác cách thực hiện việc này, miễn phí.

Tuy nhiên, một loại vi rút đến, sau khi nó ở trên một hệ thống, nó sẽ cố gắng lây lan. Trong nhiều trường hợp, vi-rút cũng cố gắng gây ra một số tác hại cho hệ thống. Sau khi vi-rút có trong hệ thống, nó có thể làm bất cứ điều gì mà một chương trình hợp pháp có thể làm. Điều đó có nghĩa là nó có thể có khả năng xóa tệp, thay đổi cài đặt hệ thống hoặc gây ra tác hại khác. Không thể phóng đại mối đe dọa từ sự tấn công của virus. Hãy dành chút thời gian để xem xét một số đợt bùng phát vi rút, xem cách chúng hoạt động và mô tả thiệt hại mà chúng gây ra. Một số trong số này là cũ hơn, một số gần đây.

+ Virus Sobig

Vi rút Sobig là một vi rút cũ hơn nhưng nó là một ví dụ điển hình về cách vi rút lây lan. Một điều thú vị về loại virus này là cách tiếp cận đa phương thức mà nó lây lan. Nói cách khác, nó đã sử dụng nhiều hơn một cơ chế để lây lan và lây nhiễm cho các máy mới. Sobig tự sao chép vào bất kỳ ổ đĩa dùng chung nào trên mạng và nó tự gửi qua e-mail cho mọi người trong sổ địa chỉ. Do cách tiếp cận này, Sobig có thể được phân loại như một loại sâu chứ không chỉ đơn giản là một loại vi rút. Khả năng lây lan đa phương thức này có nghĩa là Sobig đặc biệt *độc hại* - một thuật ngữ biểu thị rằng virus lây lan nhanh chóng và dễ dàng lây nhiễm các mục tiêu mới.

Khả năng lây lan đa phương thức này là lý do tại sao việc đảm bảo rằng mỗi người trong tổ chức của bạn được cảnh báo về các chính sách và quy trình bảo mật phù hợp là rất quan trọng. Chỉ cần một người trong mạng không may mở được e-mail có chứa virus Sobig, nó không chỉ lây nhiễm cho máy đó mà còn cho mọi ổ đĩa dùng chung trên mạng mà người này có thể truy cập.

Giống như hầu hết các cuộc tấn công bằng vi-rút phân tán qua e-mail, cuộc tấn công này có các dấu hiệu nhận biết trong chủ đề hoặc tiêu đề của e-mail có thể được sử dụng để xác định e-mail là một email bị nhiễm vi rút. Email sẽ có tiêu đề như “đây

là mẫu" hoặc "tài liệu" và khuyến khích bạn mở tệp đính kèm. Sau đó, virus đã tự sao chép vào thư mục hệ thống Windows. Một số biến thể của Sobig khiến máy tính tải xuống tệp từ Internet, sau đó sẽ gây ra sự cố in. Một số máy in mạng sẽ chỉ bắt đầu in rác. Biến thể Sobig.E thậm chí còn được ghi vào sổ đăng ký Windows, khiến vi rút được đưa vào khởi động máy tính. Những đặc điểm phức tạp này chỉ ra rằng người tạo ra Sobig biết cách truy cập sổ đăng ký Windows, truy cập các ổ đĩa dùng chung, thay đổi khởi động Windows và truy cập Outlook.

Một phương pháp mà cá nhân tôi sử dụng và khuyến nghị cho tất cả các quản trị viên bảo mật là thường xuyên gửi e-mail cho mọi người trong tổ chức của bạn để thông báo cho họ biết các dấu hiệu cần cảnh giác trong e-mail. Các trang web như www.f-secure.com liệt kê các loại virus hiện tại và những gì cần tìm trong e-mail. Tôi tóm tắt danh sách này và gửi một hoặc hai lần mỗi tháng cho mọi người trong tổ chức của tôi. Bằng cách đó, tất cả các thành viên của tổ chức đều nhận thức được những e-mail mà họ chắc chắn không nên mở. Nếu bạn kết hợp điều này với việc thận trọng trước những e-mail bất ngờ, bạn có thể giảm đáng kể khả năng bị nhiễm vi-rút.

Loại virus đặc biệt này đã lan rộng và lây nhiễm rất nhiều mạng đến nỗi chỉ riêng việc sao chép nhiều lần của virus đã đủ khiến một số mạng rơi vào bế tắc. Virus này không phá hủy các tập tin hoặc làm hỏng hệ thống, nhưng nó tạo ra đủ lưu lượng truy cập để phá hủy các mạng bị nhiễm virus. Bản thân vi rút có mức độ nguy biện trung bình. Sau khi nó ra mắt, nhiều biến thể bắt đầu mọc lên, làm phức tạp thêm tình hình.

2.3 Giải pháp bảo mật trong SQL Server

2.3.1 Giới thiệu CSDL SQL Server

SQL Server là một hệ thống quản trị cơ sở dữ liệu quan hệ (RDBMS-Relation Database Management System) do Microsoft phát hành, sử dụng các lệnh Transact-SQL để trao đổi dữ liệu giữa Client PC và Server.

Một số đặc tính của SQL Server:

Cho phép quản trị một hệ CSDL lớn (lên đến vài TB., có tốc độ xử lý dữ liệu nhanh và đáp ứng yêu cầu trong thời gian cho phép.

Cho phép nhiều người dùng cùng lúc khai thác đối với một CSDL và quản trị CSDL (hàng ngàn user).

Có hệ thống phân quyền bảo mật tương thích đối với hệ thống bảo mật của công nghệ NT (Network Technology) và tích hợp với hệ thống bảo mật của Windows NT hoặc sử dụng hệ thống bảo vệ độc lập của SQLServer.

Hỗ trợ trong việc triển khai CSDL phân tán và phát triển ứng dụng trên mạng internet.

Cho phép lập trình kết nối với nhiều ngôn ngữ khác thông dụng để xây dựng ứng dụng như C#, VB,...

Sử dụng các câu lệnh truy vấn dữ liệu Transaction-SQL.

b. Mô hình hoạt động của SQL Server trên mạng máy tính

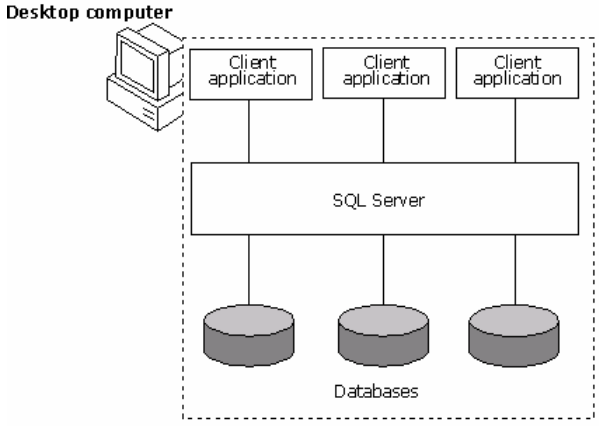
SQL server là hệ quản trị CSDL hoạt động trên mạng và có thể thực hiện trao đổi dữ liệu theo nhiều mô hình mạng khác nhau, theo nhiều giao thức và phương thức truyền tin khác nhau. Được chia thành:

Kết nối trên desktop: Có thể ở trên cùng một máy tính với SQL Server hoặc kết nối qua mạng LAN.

Kết nối qua mạng diện rộng: Thông qua đường truyền trên mạng xa để kết nối đến SQL Server.

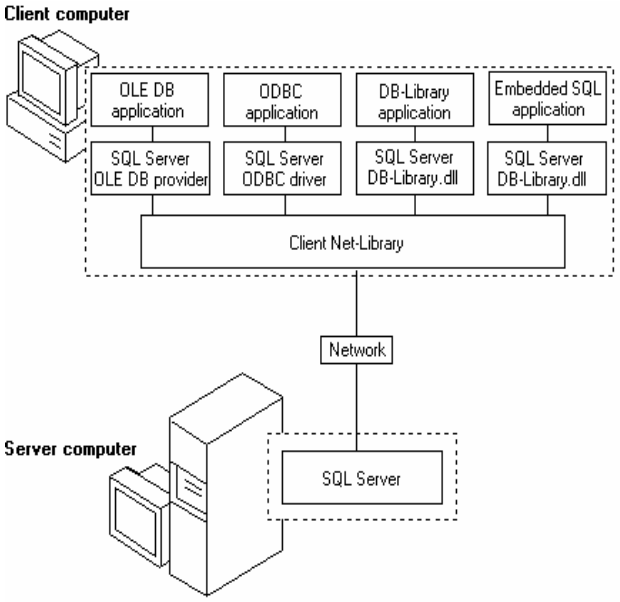
Kết nối qua mạng Internet: Các ứng dụng kết nối thông qua máy chủ Internet, các dịch vụ IIS để thực hiện ứng dụng trên Internet (ví dụ C#.NET, VB.NET,...).

Trên một máy desktop thì sơ đồ kết nối trao đổi dữ liệu được thể hiện:



Hình 2.4 Mô hình desktop.

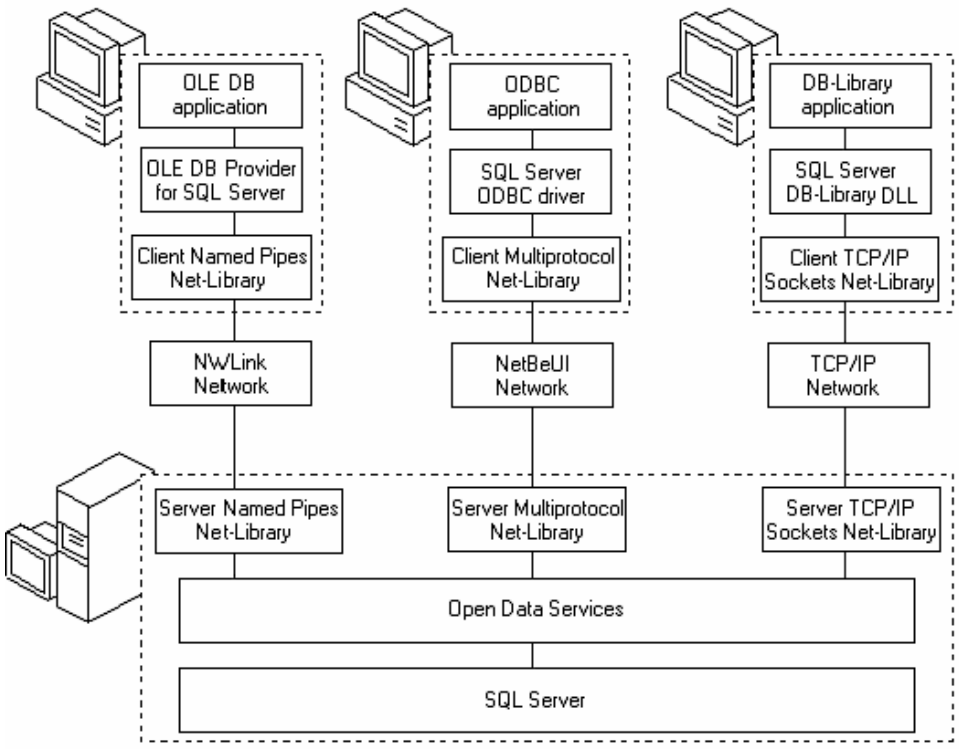
Theo mô hình này thì trên một desktop có nhiều ứng dụng và mỗi ứng dụng có thể thực hiện thao tác với nhiều CSDL khác nhau.



Hình 2.5 Mô hình client/server

Với mô hình client/server thì ứng dụng trao đổi với SQL Server theo sơ đồ sau:

SQL Server cho phép các ứng dụng kết nối theo các phương thức sau: ODBC, OLEDB, DB-Library, Embedded SQL, đó là những phương thức kết nối hữu ích cho các nhà phát triển ứng dụng. Nhưng nếu xét cụ thể hơn:

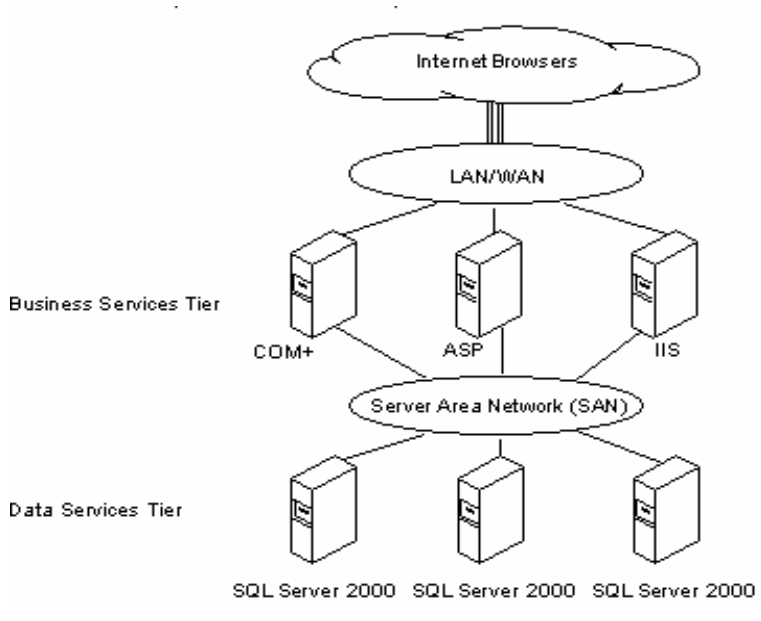


Hình 2.6 Mô hình client/server chi tiết.

SQL Server có thể thực hiện trao đổi dữ liệu với các ứng dụng theo nhiều giao thức truyền tin khác nhau như TCP/IP, Name Pipes..., và các ứng dụng có thể sử dụng nhiều phương thức kết nối khác nhau (OLEDB, ODBC,...).

Mô hình kết nối ứng dụng trên mạng Internet:

Nếu xét riêng các ứng dụng kết nối với SQL Server trên mạng internet thì các máy chủ SQL Server sẽ được quản lý thông qua các hệ thống máy chủ mạng và hệ điều hành mạng, các ứng dụng (IIS, COM+, ASP.NET..) sẽ thông qua máy chủ mạng kết nối đến SQL Server[7], mô hình này có thể được áp dụng trên mạng nội bộ, mạng



diện rộng, ứng dụng được khai thác trên trình duyệt….:

Hình 2.7 Mô hình kết nối ứng dụng trên mạng Internet.

Các vấn đề bảo mật trong CSDL SQL Server:

Các mô hình thực hiện trao đổi dữ liệu của SQL Server theo nhiều mô hình mạng khác nhau, theo nhiều giao thức và phương thức truyền tin khác nhau, chính vì vậy cần phải quan tâm đến vấn đề bảo mật trong CSDL SQL Server bao gồm có:

Bảo mật trong môi trường mạng.

Bảo mật tại chỗ.

Bảo mật trong môi trường mạng:

SQL Server sử dụng kỹ thuật SSL để thực hiện việc kết nối giữa client và server. Được phát triển bởi Netscape, ngày nay giao thức Secure Socket Layer (SSL) đã được sử dụng rộng rãi trên World Wide Web trong việc xác thực và mã hoá thông tin giữa client và server. Tổ chức IETF (Internet Engineering Task Force) đã chuẩn hoá SSL và đặt lại tên là TLS (Transport Layer Security). Mặc dù là có sự thay đổi về tên nhưng TSL chỉ là một phiên bản mới của SSL. Phiên bản TSL 1.0 tương đương với phiên bản SSL 3.1. Tuy nhiên SSL là thuật ngữ được sử dụng rộng rãi hơn. SSL được thiết kế như là một giao thức riêng cho vấn đề bảo mật có thể hỗ trợ cho rất nhiều ứng dụng. Giao thức SSL hoạt động bên trên TCP/IP và bên dưới các giao thức ứng dụng tầng cao hơn như là HTTP (Hyper Text Transport Protocol), IMAP (Internet Messaging Access Protocol) và FTP (File Transport Protocol). Trong khi SSL có thể sử dụng để hỗ trợ các giao dịch an toàn cho rất nhiều ứng dụng khác nhau trên Internet, thì hiện nay SSL được sử dụng chính cho các giao dịch trên Web.

SSL không phải là một giao thức đơn lẻ, mà là một tập các thủ tục đã được chuẩn hoá để thực hiện các nhiệm vụ bảo mật sau:

1. Xác thực server: Cho phép người sử dụng xác thực được server muốn kết nối. Lúc này, phía browser sử dụng các kỹ thuật mã hoá công khai để chắc chắn rằng certificate và public ID của server là có giá trị và được cấp phát bởi một CA (certificate authority) trong danh sách các CA đáng tin cậy của client. Điều này rất quan trọng đối với người dùng. Ví dụ như khi gửi mã số credit card qua mạng thì

người dùng thực sự muốn kiểm tra liệu server sẽ nhận thông tin này có đúng là server mà họ định gửi đến không.

2. Xác thực Client: Cho phép phía server xác thực được người sử dụng muốn kết nối. Phía server cũng sử dụng các kỹ thuật mã hoá công khai để kiểm tra xem certificate và public ID của server có giá trị hay không và được cấp phát bởi một CA (certificate authority) trong danh sách các CA đáng tin cậy của server không. Điều này rất quan trọng đối với các nhà cung cấp. Ví dụ như khi một ngân hàng định gửi các thông tin tài chính mang tính bảo mật tới khách hàng thì họ rất muốn kiểm tra định danh của người nhận.

3. Mã hoá kết nối: Tất cả các thông tin trao đổi giữa client và server được mã hoá trên đường truyền nhằm nâng cao khả năng bảo mật. Điều này rất quan trọng đối với cả hai bên khi có các giao dịch mang tính riêng tư. Ngoài ra, tất cả các dữ liệu được gửi đi trên một kết nối SSL đã được mã hoá còn được bảo vệ nhờ cơ chế tự động phát hiện các xáo trộn, thay đổi trong dữ liệu. (đó là các thuật toán băm - hash algorithm).

4. Giao thức SSL bao gồm 2 giao thức con: giao thức SSL record và giao thức SSL handshake. Giao thức SSL record xác định các định dạng dùng để truyền dữ liệu. Giao thức SSL handshake (gọi là giao thức bắt tay) sẽ sử dụng SSL record protocol để trao đổi một số thông tin giữa server và client vào lần đầu tiên thiết lập kết nối SSL.

Các thuật toán mã hoá dùng trong SSL:

Các thuật toán mã hoá (cryptographic algorithm hay còn gọi là cipher) là các hàm toán học được sử dụng để mã hoá và giải mã thông tin. Giao thức SSL hỗ trợ rất nhiều các thuật toán mã hoá, được sử dụng để thực hiện các công việc trong quá trình xác thực server và client, truyền tải các certificates và thiết lập các khoá của từng phiên giao dịch (sesion key). Client và server có thể hỗ trợ các bộ mật mã (cipher suite) khác nhau tuỳ thuộc vào nhiều yếu tố nhoị phiên bản SSL đang dùng, chính sách của công ty về độ dài khoá mà họ cảm thấy chấp nhận được - điều này liên quan đến mức độ bảo mật của thông tin,…

Các thuật toán SSL hỗ trợ:

- DES (Data Encryption Standard): Là một thuật toán mã hoá có chiều dài khoá là 56 bit.

- 3-DES (Triple-DES): Là thuật toán mã hoá có độ dài khoá gấp 3 lần độ dài khoá trong mã hoá DES.
- DSA (Digital Signature Algorithm): Là một phần trong chuẩn về xác thực số đang được chính phủ Mỹ sử dụng.
- KEA (Key Exchange Algorithm): Là một thuật toán trao đổi khoá đang được chính phủ Mỹ sử dụng.
- MD5 (Message Digest algorithm): Được phát thiển bởi Rivest.
- RSA: là thuật toán mã hoá công khai dùng cho cả quá trình xác thực và mã hoá dữ liệu được Rivest, Shamir, and Adleman phát triển.
- RSA key exchange: Là thuật toán trao đổi khoá dùng trong SSL dựa trên thuật toán RSA.
- RC2 and RC4: Là các thuật toán mã hoá được phát triển bởi Rivest dùng cho RSA Data Security.
- SHA-1 (Secure Hash Algorithm): Là một thuật toán băm đang được chính phủ Mỹ sử dụng.

Các thuật toán trao đổi khoá như KEA, RSA key exchange được sử dụng để 2 bên client và server xác lập khoá đối xứng mà họ sẽ sử dụng trong suốt phiên giao dịch SSL. Và thuật toán được sử dụng phổ biến là RSA key exchange.

Các phiên bản SSL 2.0 và SSL 3.0 hỗ trợ cho hầu hết các bộ mã hoá. Người quản trị có thể tuỳ chọn bộ mã hoá sẽ dùng cho cả client và server. Khi một client và server trao đổi thông tin trong giai đoạn bắt tay (handshake), họ sẽ xác định bộ mã hoá mạnh nhất có thể và sử dụng chúng trong phiên giao dịch SSL.

Các bộ mã hoá sử dụng thuật toán trao đổi khoá RSA:

Đây là danh sách các bộ mã hoá được hỗ trợ trong SSL mà sử dụng thuật toán trao đổi khoá RSA và được liệt kê khả năng bảo mật từ mạnh đến yếu.

Mạnh nhất:

Thuật toán mã hoá 3- DES, thuật toán xác thực SHA-1.

Mạnh:

- Thuật toán mã hoá RC4 (với độ dài khoá 128 bit), thuật toán xác thực MD5.
- Thuật toán mã hoá RC2 (với độ dài khoá 128 bit), thuật toán xác thực MD5.
- Thuật toán mã hoá DES (với độ dài khoá 56 bit), thuật toán xác thực SHA1.

Tương đối mạnh:

- Thuật toán mã hoá RC4 (với độ dài khoá 40 bit), thuật toán xác thực MD5. Thuật toán mã hoá RC2 (với độ dài khoá 40 bit), thuật toán xác thực MD5.

Yếu nhất:

- Không mã hoá thông tin, chỉ dùng thuật toán xác thực MD5.

Chú ý: Khi nói các thuật toán mã hoá RC4 và RC2 có độ dài khoá mã hoá là 40 bit thì thực chất độ dài khoá vẫn là 128 bit nhưng chỉ có 40 bit được dùng để mã hoá.

SSL handshake

Giao thức SSL sử dụng kết hợp 2 loại mã hoá đối xứng và công khai. Sử dụng mã hoá đối xứng nhanh hơn rất nhiều so với mã hoá công khai khi truyền dữ liệu, nhưng mã hoá công khai lại là giải pháp tốt nhất trong qúa trình xác thực. Một giao dịch SSL thường bắt đầu bởi quá trình "bắt tay" giữa hai bên (SSL handshake). Các bước trong quá trình "bắt tay" có thể tóm tắt như sau:

1. Client sẽ gửi cho server số phiên bản SSL đang dùng, các tham số của thuật toán mã hoá, dữ liệu được tạo ra ngẫu nhiên (đó chính là digital signature) và một số thông tin khác mà server cần để thiết lập kết nối với client.

2. Server gửi cho client số phiên bản SSL đang dùng, các tham số của thuật toán mã hoá, dữ liệu được tạo ra ngẫu nhiên và một số thông tin khác mà client cần để thiết lập kết nối với server. Ngoài ra server cũng gửi certificate của nó đến client, và yêu cầu certificate của client nếu cần.

3. Client sử dụng một số thông tin mà server gửi đến để xác thực server. Nếu như server không được xác thực thì người sử dụng sẽ được cảnh báo và khi đó kết nối không được thiết lập. Còn nếu như xác thực được server thì phía client sẽ thực hiện tiếp bước 4.

4. Sử dụng tất cả các thông tin được tạo ra trong giai đoạn bắt tay ở trên, client (cùng với sự cộng tác của server và phụ thuộc vào thuật toán được sử dụng) sẽ tạo ra

premaster secret cho phiên làm việc, mã hoá bằng khoá công khai (public key) mà server gửi đến trong certificate ở bước 2, và gửi đến server.

5. Nếu server có yêu cầu xác thực client, thì phía client sẽ đánh dấu vào phần thông tin riêng chỉ liên quan đến quá trình "bắt tay" này mà hai bên đều biết. Trong trường hợp này, client sẽ gửi cả thông tin được đánh dấu và certificate của mình cùng với premaster secret đã được mã hoá tới server.

6. Server sẽ xác thực client. Trường hợp client không được xác thực, phiên làm việc sẽ bị ngắt. Còn nếu client được xác thực thành công, server sẽ sử dụng khoá bí mật (private key) để giải mã premaster secret, sau đó thực hiện một số bước để tạo ra master secret.

7. Client và server sẽ sử dụng master secret để tạo ra các session key, đó chính là các khoá đối xứng được sử dụng để mã hoá và giải mã các thông tin trong phiên làm việc và kiểm tra tính toàn vẹn dữ liệu.

8. Client sẽ gửi một lời nhắn đến server thông báo rằng các message tiếp theo sẽ được mã hoá bằng session key. Sau đó nó gửi một lời nhắn đã được mã hoá để thông báo rằng phía client đã kết thúc giai đoạn "bắt tay".

9. Server cũng gửi một lời nhắn đến client thông báo rằng các message tiếp theo sẽ được mã hoá bằng session key. Sau đó nó gửi một lời nhắn đã được mã hóa để thông báo rằng server đã kết thúc giai đoạn "bắt tay".

10. Lúc này giai đoạn "bắt tay" đã hoàn thành, và phiên làm việc SSL bắt đầu. Cả hai phía client và server sẽ sử dụng các session key để mã hoá và giải mã thông tin trao đổi giữa hai bên, và kiểm tra tính toàn vẹn dữ liệu[8].

Chứng chỉ khoá công khai X.509:

Chứng chỉ X.509 là định dạng chứng chỉ được sử dụng phổ biến và được hầu hết các nhà cung cấp sản phẩm PKI triển khai.

Chứng chỉ khoá công khai X.509 được Hội viễn thông quốc tế (ITU) đưa ra lần đầu tiên năm 1988 như là một bộ phận của dịch vụ thư mục X.500.

Chứng chỉ gồm 2 phần: Phần đầu là những trường cơ bản cần thiết phải có trong chứng chỉ. Phần thứ hai chứa thêm một số trường phụ, những trường phụ này được gọi

là trường mở rộng dùng để xác định và đáp ứng những yêu cầu bổ sung của hệ thống.

Những trường cơ bản của chứng chỉ X.509:

- Version: Xác định số phiên bản của chứng chỉ.
- Certificate Serial Number: Do CA gán, là định danh duy nhất của chứng chỉ.
- Signature Algorithm ID: Chỉ ra thuật toán CA sử dụng để ký số chứng chỉ. Có thể là thuật toán RSA hay DSA…
- Issuer: Chỉ ra CA cấp và ký chứng chỉ.
- Validity Period: Khoảng thời gian chứng chỉ có hiệu lực. Trường này xác định thời gian chứng chỉ bắt đầu có hiệu lực và thời điểm hết hạn.
- Subject: Xác định thực thể mà khoá công khai của thực thể này được xác nhận.

Tên của subject phải duy nhất đối với mỗi thực thể CA xác nhận.

- Subject public key information: Chứa khoá công khai và những tham số liên quan; xác định thuật toán (ví dụ RSA hay DSA. được sử dụng cùng với khoá.
- Issuer Unique ID (Optional): Là trường không bắt buộc, trường này cho phép sử dụng lại tên người cấp. Trường này hiếm được sử dụng trong triển khai thực tế.
- Extensions (Optional): Chỉ có trong chứng chỉ v.3.
- Certification Authority‟s Digital Signature: Chữ ký số của CA được tính từ những thông tin trên chứng chỉ với khoá riêng và thuật toán ký số được chỉ ra trong trường Signature Algorithm Identifier của chứng chỉ.

Ngoài ra chứng chỉ X.509 còn một số trường mở rộng, phần mở rộng là những thông tin về các thuộc tính cần thiết được đưa vào để gắn những thuộc tính này với người sử dụng hay khoá công khai. Những thông tin trong phần mở rộng thường được dùng để quản lý xác thực phân cấp, chính sách chứng chỉ, thông tin về chứng chỉ bị thu hồi…Nó cũng có thể được sử dụng để định nghĩa phần mở rộng riêng chứa những thông tin đặc trưng cho cộng đồng nhất định. Mỗi trường mở rộng trong chứng chỉ được thiết kế với cờ “critical” hoặc “uncritical”.

Tính toàn vẹn của chứng chỉ được đảm bảo bằng chữ ký số của CA trên chứng chỉ. Khoá công khai của CA được phân phối đến người sử dụng chứng chỉ theo một số cơ chế bảo mật trước khi thực hiện các thao tác PKI. Người sử dụng kiểm tra hiệu lực

của chứng chỉ được cấp với chữ ký số của CA và khoá công khai của CA.

Người sử dụng bất kỳ có thể trao đổi với CA để chứng nhận, và cũng chỉ có CA mới có thể sửa chứng nhận. Vì không thể bị giả mạo nên chứng nhận có thể được đặt trong thư mục công cộng. Nếu cả hai người sử dụng chia sẻ chung CA thì họ được giả thiết là biết khóa công khai của CA đó. Và các CA cũng cần phải tạo nên sơ đồ phân cấp để trao đổi chứng nhận với nhau. Sử dụng chứng nhận liên kết các thành viên của sơ đồ để có được chứng nhận của các CA khác. Mỗi CA có thể gửi tiếp các chứng nhận của mình cho các Client và có thể gửi lại chứng nhận của mình cho cha của nó. Mỗi Client tin tưởng các chứng nhận của cha. Có thể kiểm chứng nhận bất kỳ của một CA cho người sử dụng bằng các CA trong sơ đồ phân cấp. Giấy chứng nhận có chu kỳ sử dụng, có thể thu hồi trước thời hạn trong những trường hợp cần thiết như khóa riêng của người sử dụng bị lộ, người dùng không tiếp tục được chứng nhận bởi CA đó, giấy chứng nhận của CA bị làm hại. Nói chung CA bảo trì danh sách các chứng nhận bị thu hồi. Người sử dụng có thể kiểm tra lại các chứng nhận đã bị thu hồi. X509 bao gồm ba thủ tục xác thực tùy chọn là: Xác thực một chiều, xác thực hai chiều, xác thực ba chiều. Mọi thủ tục đều sử dụng các chữ ký khóa công khai.

Xác thực 1 chiều: Một chiều X→Y được sử dụng để thiết lập danh tính của X và rằng bản tin là từ X, bản tin được gửi cho Y tính toàn vẹn nguồn của bản tin. Bản tin có thể bao gồm cả nhãn thời gian, ký hiệu đặc trưng của bản tin, danh tính của Y và nó được ký bởi X. Có thể bao gồm một số thông tin bổ sung cho Y như khóa phiên.

Xác thực 2 chiều: Hai bản tin X→Y và Y→X được thiết lập, ngoài bản tin từ X đến Y, còn có thêm việc xác thực danh tính của Y và trả lời từ Y, trả lời này dành cho X, tính toàn vẹn và nguồn gốc của trả lời. Trả lời bao gồm cả ký hiệu đặc trọng của mẫu tin từ X, cả nhãn thời gian và ký hiệu đặc trưng trả lời từ Y. Có thể bao gồm một số bổ sung cho X.

Xác thực 3 chiều: Ba mẫu tin X→Y, Y→X và X→Y được thiết lập nhưng không có đồng hồ đồng bộ. Ngoài hai chiều như trên còn có trả lời lại từ X đến Y chứa bản sao ký hiệu đặc trưng của trả lời từ Y,tức là các nhãn thời gian mà không cần kiểm tra.

Bảo mật tại chỗ:

Khi người dùng muốn kết nối tới CSDL thì quá trình này sẽ được thực hiện

theo hai giai đoạn sau

1. Kiểm tra kết nối, giai đoạn này xảy ra khi người dùng cố gắng kết nối vào CSDL thì server sẽ chấp nhận hay từ chối kết nối này dựa trên việc nhận dạng được người dùng. Và server chỉ chấp nhận nếu thông tin này gửi đến servẻ là hợp lệ thì server mới chấp nhận, ngược lại thì server sẽ cấm truy cập. Nếu được chấp nhận, kết nối sẽ chuyển sang giai đoạn chờ đợi yêu cầu phía client. Phía client sẽ cung cấp thông tin cần thiết để server xác thực người dùng bao gồm có user name, password và địa chỉ server.

2. Kiểm tra yêu cầu, giai đoạn này sẽ xảy ra khi mà người dùng muốn thực hiện các yêu cầu đến CSDL, sau khi thiết lập kết nối, mỗi yêu cầu sẽ phải đi qua tiến trình và do đó sẽ giới hạn làm việc. Khi có yêu cầu được đưa ra thì SQL Server sẽ kiểm tra xem user đã được phân quyền ở mức user chưa, nếu có thì người dùng được phép làm việc trên CSDL của SQL Server.

Các biện pháp tác nghiệp chỉ mang tín chất vận hành, các chính sách thao tác, điều khoản,.. của người dùng đối với CSDL. Có một số biện pháp như:

1. Vô hiệu các tài khoản mặc định (điển hình như tài khoản sa..
2. Mọi người dùng truy cập CSDL bắt buộc phải có tài khoản.
3. Thay đổi cấu hình cổng TCP/IP mặc định đến CSDL (mặc định là 1433).
4. Không cấp các quyền processor, super cho người dùng không phải là quản trị.
5. Không cấp các đặc quyền truy cập file cho người dùng không phải quản trị.

2.4. Kết luận chương

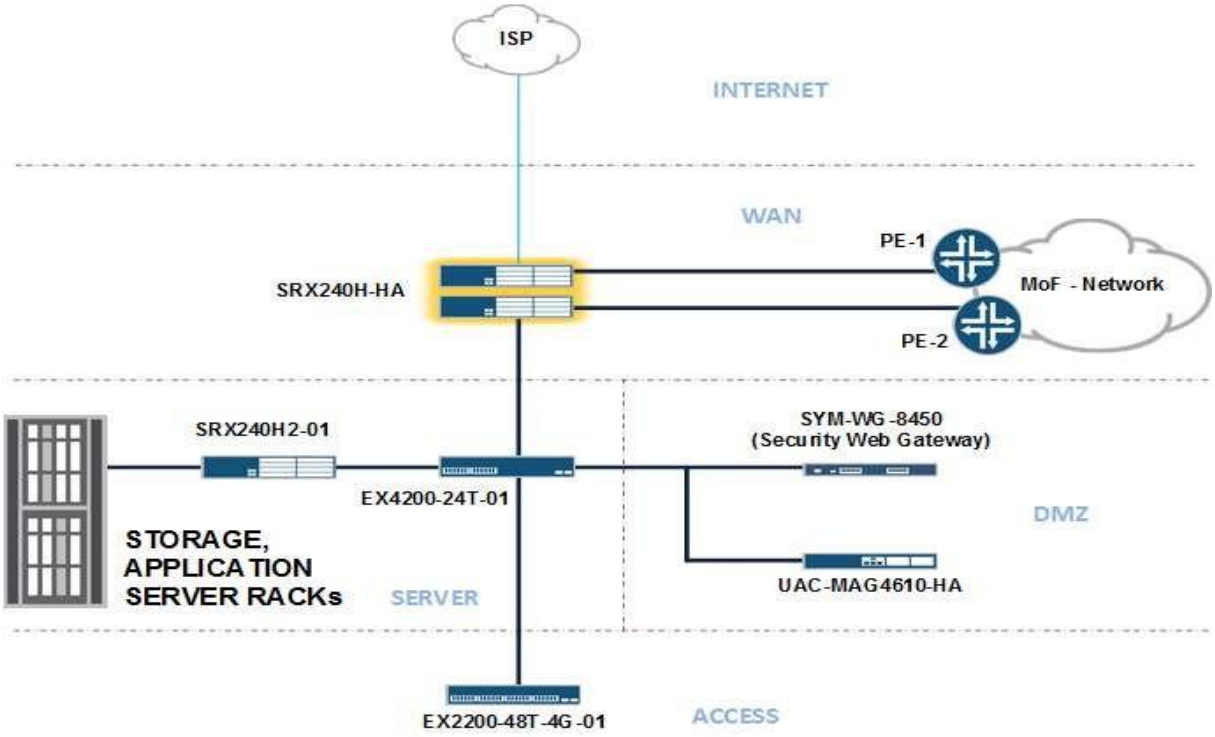
Chương này đã xem xét các mối đe dọa phổ biến nhất đối với hệ thống của bạn: tấn công vi rút, tấn công từ chối dịch vụ, phần mềm mã độc Trojan, chiếm quyền điều khiển phiên và tấn công tràn bộ đệm. Các mối nguy hiểm khác như đánh cắp danh tính và lừa đảo (sử dụng e-mail giả và các trang web để thu thập thông tin người dùng cuối có thể được sử dụng để đánh cắp danh tính và gian lận) đang xảy ra thường xuyên hơn, nhưng không đe dọa trực tiếp đến tổ chức mạng như họ làm với các cá nhân. Đó là lý do tại sao chương này tập trung vào các cuộc tấn công mà nó đã gây ra — chúng là mối quan tâm nhất đối với an ninh mạng.

Trong mỗi trường hợp, các cơ chế phòng vệ khác nhau thuộc một trong hai loại: kỹ

thuật hoặc thủ tục. Phòng vệ kỹ thuật là những hạng mục bạn có thể cài đặt hoặc cấu hình để làm cho hệ thống của mình an toàn hơn. Điều này bao gồm những thứ như khối vi mô, cookie RST, tinh chỉnh ngăn xếp và phần mềm chống vi-rút. Các biện pháp bảo vệ theo thủ tục liên quan đến việc sửa đổi hành vi của người dùng cuối để tăng cường bảo mật. Các biện pháp như vậy bao gồm không tải xuống các tệp đáng ngờ và không mở các tệp đính kèm chưa được xác minh. Khi bạn đọc qua cuốn sách này, bạn sẽ phát hiện ra rằng phòng thủ mạng phải được tiếp cận từ cả hai góc độ. Các chương sau cung cấp thảo luận chi tiết về các biện pháp bảo vệ kỹ thuật (tường lửa, trình quét vi-rút, v.v.) và toàn bộ các chương được dành cho các biện pháp phòng thủ theo thủ tục (chính sách và thủ tục). Hiểu rằng việc sử dụng cả hai cách tiếp cận là cần thiết để bảo mật mạng của bạn là rất quan trọng.

CHƯƠNG 3: XÂY DỰNG ỨNG DỤNG BẢO MẬT CƠ SỞ DỮ LIỆU

3.1. Giới thiệu hệ thống tích hợp dữ liệu



Hình 3.1 Mô hình hệ thống Data Center.

Trong hình 3.1 đưa ra cách bố trí vận hành trong Data Center – tập hợp trung tâm tài nguyên công nghệ cao, có chức năng xử lý, lưu trữ dữ liệu ổn định nhanh chóng. Mô hình Data Center chuẩn bao gồm các thành phần sau đây:

Cơ sở vật chất: không gian đặt máy chủ trung tâm cần đảm bảo sạch sẽ đạt chuẩn về an toàn vệ sinh, chống cháy.

Các thiết thị hỗ trợ nhằm duy trì sự sẵn sàng ở mức độ cao nhất: nguồn điện (UPS – Uninterruptible Power Sources), máy điều hòa không khí, thông gió, hệ thống làm mát và ống xả, các hệ thống an ninh vật lý, …

Các thiết bị IT hỗ trợ cho hoạt động công nghệ thông tin và lưu dữ liệu cho doanh nghiệp.

Nhân viên điều hành giám sát hoạt động của thiết bị và cơ sở hạ tầng.

Thực tế điều hòa không khí là thiết bị để làm mát Data Center và duy trì nhiệt

độ Data Center luôn ở mức 26 độ, có các máy chủ CSDL và sao lưu phục vụ cho việc lưu trữ thông tin, chia sẻ dữ liệu. Ngoài ra thì có máy chủ Web chứa Website của cục, máy chủ Web Service dịch vụ. Hệ thống cáp mạng kết nối các máy chủ CSDL, máy chủ Web,… tới trung tâm vận hành hệ thống.

Mục đích của luận văn cũng chỉ tập trung vào phần CSDL (máy chủ CSDL), bảo mật thông tin CSDL này.

Phòng mạng có nhiều router, switch, hub,… Kết nối tới tất cả các phòng trong Cục Dự trữ Nhà nước khu vực Hà Nội, kết nối với các máy chủ Web, CSDL... trong Data Center.

Như vậy, qua tìm hiểu sơ lược về trung tâm tích hợp dữ liệu, chúng ta đã có một cái nhìn cơ bản, hình dung được về một trung tâm tích hợp dữ liệu. Hiểu một cách đơn giản thì trung tâm tích hợp dữ liệu là một khu vực mà trong đó tích hợp CSDL của các ứng dụng trong một cơ quan, tổ chức, doanh nghiệp,…và trong đó có các kết nối mạng, máy chủ chạy ứng dụng CSDL…thực hiện việc trao đổi dữ liệu giữa tổ chức,...với CSDL trung tâm qua mạng cục bộ, mạng Internet…

3.2. Giải pháp đảm bảo an toàn tại trung tâm

Qua việc tìm hiểu về trung tâm tích hợp dữ liệu tại Cục Dự trữ Nhà nước khu vực Hà Nội thì chúng ta thấy rằng việc áp dụng các giải pháp đảm bảo an toàn và bảo mật cho trung tâm dữ liệu đã được quan tâm và ứng dụng bao gồm:

Vấn đề về an toàn vật lý:

Do trung tâm của Cục được đặt ngay tại trụ sở nên một số vấn đề về bão lũ đã được khắc phục, hệ thống chống sét thì được xây dựng chung cùng với tòa nhà trung tâm nên không có vấn đề gì đáng ngại, hệ thống điều hòa không khí thì đảm bảo cho trung tâm luôn giữ được nhiệt độ ổn định, hệ thống cáp mạng, tủ rack được đầu tư phù hợp.

Hệ thống cung cấp điện thì đã có hệ thống UPS dự phòng, tuy nhiên chưa có hệ thống máy nổ, cũng dễ hiểu vì trung tâm đặt tại Cục, hệ thống điện lưới luôn ổn định.

Ngoài ra đã có các phương tiện phòng chống cháy, hỏa hoạn, nhưng chưa có cơ chế phát hiện ra sự cố, đã có hệ thống an ninh, camera quan sát và nhân viên bảo vệ trực

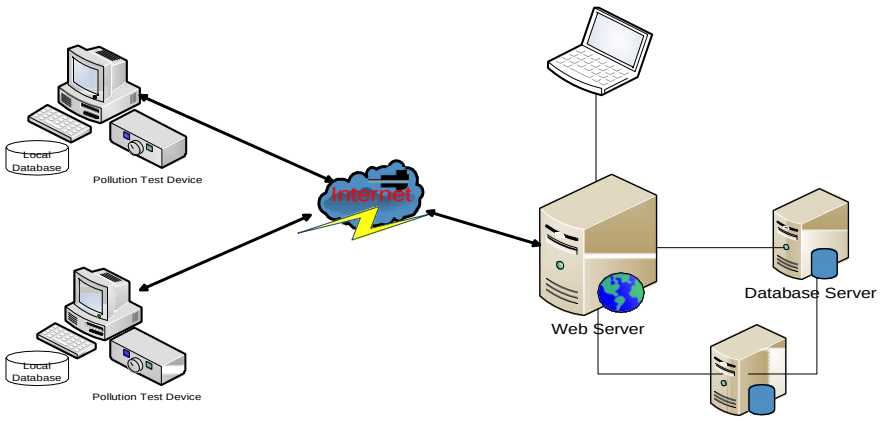
tại các phòng Server…

Vấn đề về dữ liệu: Trung tâm đã xây dựng, cấu hình mạng riêng ảo VPN cho tất cả các phòng trong trung tâm, có firewall, chính sách truy cập tới các máy trong các phòng, đảm bảo được một số vấn đề về bảo mật dữ liệu…

Tuy nhiên thì trong hệ thống CSDL có một số ứng dụng hiện tại chưa được bảo vệ về dữ liệu, các thông tin lưu trữ trong CSDL này thì chưa được bảo vệ, có thể chỉnh sửa được...

Và luận văn tìm hiểu các vấn đề ảnh hưởng đến an ninh CSDL để áp dụng bảo vệ CSDL ứng dụng này.

Hệ thống phần mềm bao gồm: phần mềm kiểm định khí thải tại Cơ sở kiểm định khí thải mô tô, xe máy (gọi là Trạm) và phần mềm quản lý hệ thống kiểm định khí thải tại Cục Dự trữ Nhà nước khu vực Hà Nội (gọi là Cục), là công cụ thực hiện kiểm định và quản lý hoạt động kiểm định khí thải mô tô, xe máy của các Trạm, quản lý kết quả kiểm định, thông tin hành chính và kỹ thuật của xe mô tô, xe máy của các Trạm, quản lý kết quả kiểm định, thông tin hành chính và kỹ thuật của mô tô, xe máy đang lưu hành và một số vấn đề khác có liên quan đến kiểm soát khí thải mô tô, xe máy tham gia lưu thông.



Hình 3.2 Mô hình kiến trúc hệ thống.

Hình 3.3 mô tả kiến trúc chung của ứng dụng, trong đó chia làm hai phần chính. Trạm (Stations) chịu trách nhiệm đo khí thải phương tiện thông qua việc đọc tín hiệu từ máy đo và tính toán kết quả đo lường. Kết quả đo được lưu tạm thời vào cơ sở dữ liệu Access cục bộ phục vụ việc in phiếu kiểm định. Thông tin về kết quả đo được

gửi lên Cục thông qua Web Service qua mạng Internet, Máy chủ lưu thông tin kết quả kiểm định vào cơ sở dữ liệu chính (Main Database Server) và có cơ sở dữ liệu sao lưu.

Tất cả các trao đổi thông tin giữa ứng dụng và CSDL đều thực hiện trên mạng công khai, cho nên nguy cơ các thông tin này bị mất, bị lộ, sửa đổi trái phép,…trong quá trình truyền đi là rất cao, ngoài ra kẻ gian có thể giả mạo Client để tương tác với Server để thực hiện thay đổi thông tin,…bất hợp pháp, chính vì vậy đặt ra vấn đề bảo vệ thông tin trao đổi giữa Client và CSDL và xác thực giữa Client và Server CSDL.

3.3. Phân tích ứng dụng

Do ứng dụng được xây dựng trên nền Web và được chia làm hai phần là cơ sử dữ liệu máy trạm và máy chủ, trao đổi thông tin giữa Client và Server CSDL:

Phân tích một cách chi tiết hơn, đối với mô hình trao đổi thông tin giữa ứng dụng và CSDL thì có một số vấn đề gặp phải như sau:

Như vậy đối với ứng dụng này đặt ra các bài toán chính sau:

- Xác thực ứng dụng Client với Server.
- Bảo vệ ứng dụng truy cập CSDL.
- Bảo vệ thông tin trên đường truyền.
- Bảo mật thông tin tại CSDL.

➢ Giải quyết được bài toán xác thực ứng dụng Client với Server CSDL nhằm ngăn chặn việc sử dụng các phương thức dịch vụ Web được công khai để kẻ gian có thể lợi dụng để thay đổi, chèn, xem, xóa,… các thông tin bất hợp pháp.

➢ Giải quyết được bài toán bảo vệ ứng dụng truy cập CSDL nhằm ngăn chặn kẻ gian lợi dụng sơ sở của ứng dụng để thực hiện tấn công SQL Injection (phía Client) để thực hiện các câu lệnh SQL không mong muốn và việc đánh cắp thông tin kết nối CSDL trong file cấu hình Web Service, từ đó truy cập CSDL trái phép nên cần phải mã hóa thông tin này, nói cách khác là ngăn chặn kẻ gian tấn công CSDL qua môi trường ứng dụng.

➢ Giải quyết được bài toán bảo vệ thông tin trên đường truyền nhằm ngăn chặn kẻ gian tấn công CSLD qua môi trường mạng.

➢ Giải quyết được bài toán bảo mật thông tin tại CSDL nhằm ngăn chặn nguy cơ kẻ

gian truy cập dữ liệu không được phép.

Như vậy giải quyết được các bài toán trên thì đã hạn chế tối thiểu các nguy cơ cơ sở dữ liệu ứng dụng bị tấn công.

3.4. Bảo vệ cơ sở dữ liệu ứng dụng

Trong chiến lược bảo mật dữ liệu, đa số các công ty hiện nay tập trung nguồn lực vào bảo vệ dữ liệu trên đường truyền. Trong khi đó vấn đề bảo vệ dữ liệu nằm trong cơ sở dữ liệu (CSDL, database) chưa được quan tâm đúng mức.

Thực tế cho thấy, sự cố về an ninh xảy ra với CSDL có thể ảnh hưởng nghiêm trọng đến danh tiếng của công ty và quan hệ với khách hàng. Sự cố an ninh mất an ninh của khách hàng gần đây xảy ra với đã phần nào gia tăng sự chú ý đến các giải pháp bảo mật CSDL.

Giải pháp đơn giản nhất bảo vệ dữ liệu trong CSDL ở mức độ tập tin, chống lại sự truy cập trái phép vào các tập tin CSDL là hình thức mã hóa. Tuy nhiên, mã hóa dữ liệu ở mức độ này là giải pháp mang tính "được ăn cả, ngã về không", giải pháp này không cung cấp mức độ bảo mật truy cập đến CSDL ở mức độ bảng (table), cột (column) và dòng (row). Một điểm yếu nữa của giải pháp này là bất cứ ai với quyền truy xuất CSDL đều có thể truy cập vào tất cả dữ liệu trong CSDL. Điều này phát sinh một nguy cơ nghiêm trọng, cho phép các đối tượng với quyền quản trị (admin) truy cập tất cả các dữ liệu nhạy cảm. Thêm vào đó, giải pháp này bị hạn chế vì không cho phép phân quyền khác nhau cho người sử dụng CSDL[7].

Giải pháp thứ hai, đối nghịch với giải pháp mã hóa cấp tập tin nêu trên, giải quyết vấn đề mã hóa ở mức ứng dụng. Giải pháp này xử lý mã hóa dữ liệu trước khi truyền dữ liệu vào CSDL. Những vấn đề về quản lý khóa và quyền truy cập được hỗ trợ bởi ứng dụng. Truy vấn dữ liệu đến CSDL sẽ trả kết quả dữ liệu ở dạng mã hóa và dữ liệu này sẽ được giải mã bởi ứng dụng. Giải pháp này giải quyết được vấn đề phân tách quyền an ninh và hỗ trợ các chính sách an ninh dựa trên vai trò (Role Based Access Control – RBAC). Tuy nhiên, xử lý mã hóa trên tầng ứng dụng đòi hỏi sự thay đổi toàn diện kiến trúc của ứng dụng, thậm chí đòi hỏi ứng dụng phải được viết lại. Đây là một vấn đề đáng kể cho các công ty có nhiều ứng dụng chạy trên nhiều nền CSDL khác nhau.

Từ những phân tích hai giải pháp nêu trên, có thể dễ dàng nhận thấy một giải pháp bảo mật CSDL tối ưu cần hỗ trợ các yếu tố chính sau:

1. Hỗ trợ mã hóa tại các mức dữ liệu cấp bảng, cột, hàng.

2. Hỗ trợ chính sách an ninh phân quyền truy cập đến mức dữ liệu cột, hỗ trợ RBAC.

3. Cơ chế mã hóa không ảnh hưởng đến các ứng dụng hiện tại.

Dưới đây là hai mô hình thỏa mãn các yêu cầu trên, đặc biệt là yêu cầu thứ ba.

Xây dựng tầng CSDL trung gian

Trong mô hình này, một CSDL trung gian (proxy) được xây dựng giữa ứng dụng và CSDL gốc (Sơ đồ 1). CSDL trung gian này có vai trò mã hóa dữ liệu trước khi cập nhật vào CSDL gốc, đồng thời giải mã dữ liệu trước khi cung cấp cho ứng dụng. CSDL trung gian đồng thời cung cấp thêm các chức năng quản lý khóa, xác thực người dùng và cấp phép truy cập.

Giải pháp này cho phép tạo thêm nhiều chức năng về bảo mật cho CSDL. Tuy nhiên, mô hình CSDL trung gian đòi hỏi xây dựng một ứng dụng CSDL tái tạo tất cả các chức năng của CSDL gốc.

Hiện tại, trên thị trường sản phẩm mã hóa CSDL, Secure.Data của công ty Protegrity (www. Protegrity.com) sử dụng mô hình proxy nêu trên.

Sử dụng cơ chế sẵn có trong CSDL

Mô hình này giải quyết các vấn đề mã hóa cột dựa trên các cơ chế sau:

a. Các hàm Stored Procedure trong CSDL cho chức năng mã hóa và giải mã

b. Sử dụng cơ chế View trong CSDL tạo các bảng ảo, thay thế các bảng thật đã được mã hóa.

c. Cơ chế “instead of” trigger được sử dụng nhằm tự động hóa quá trình mã hóa từ View đến bảng gốc.

Trong mô hình này, dữ liệu trong các bảng gốc sẽ được mã hóa, tên của bảng gốc được thay đổi. Một bảng ảo (View) được tạo ra mang tên của bảng gốc, ứng dụng sẽ truy cập đến bảng ảo này.

Các truy xuất dữ liệu đến bảng gốc sẽ được thay thế bằng truy xuất đến bảng ảo.

Bảng ảo được tạo ra để mô phỏng dữ liệu trong bảng gốc. Khi thực thi lệnh “select”, dữ liệu sẽ được giải mã cho bảng ảo từ bảng gốc (đã được mã hóa). Khi thực thi lệnh “Insert, Update”, “instead of” trigger sẽ được thi hành và mã hóa dữ liệu xuống bảng gốc[8].

Quản lý phân quyền truy cập đến các cột sẽ được quản lý ở các bảng ảo. Ngoài các quyền cơ bản do CSDL cung cấp, hai quyền truy cập mới được định nghĩa:

1. Người sử dụng chỉ được quyền đọc dữ liệu ở dạng mã hóa (ciphertext). Quyền này phù hợp với những đối tượng cần quản lý CSDL mà không cần đọc nội dung dữ liệu.

2. Người sử dụng được quyền đọc dữ liệu ở dạng giải mã (plaintext).

Giải pháp nêu trên có lợi điểm đơn giản, dễ phát triển. Tuy nhiên, do các giới hạn về cơ chế view, trigger và cách thức quản trị dữ liệu, giải pháp này có những hạn chế sau:

Những cột index không thể được mã hóa, do đó hạn chế các ứng dụng cần hỗ trợ index, dữ liệu mã hóa có kích thước lớn so với dữ liệu gốc. Sự chênh lệch này không đáng kể đối với các dữ liệu chữ (text), nhưng rất đáng kể đối với các dữ liệu số và dạng nhị phân. Ví dụ, dữ liệu số 1 byte sẽ bị tăng lên 2 byte sau khi mã hóa.

Tốc độ truy cập CSDL giảm do quá trình thực thi tầng mã hóa

Bảo vệ ứng dụng truy xuất CSDL

Ở đây chúng ta hiểu là ứng dụng truy xuất đến CSDL gồm hai phần chính là Client (bên sử dụng dịch vụ) và Web Service (bên cung cấp dịch vụ).

1. Nguy cơ CSDL bị tấn công SQL Injection:

Bên sử dụng dịch vụ Client sẽ thực hiện việc xác thực người dùng trước khi được phép truy cập hệ thống để thực hiện các chức năng, yêu cầu xác thực bao gồm có những thông tin cơ bản như user name và password, do đó lợi dụng “lỗ hổng” này thì kẻ gian có thể tấn công Cơ sở dữ liệu sử dụng tấn công SQL Injection.

SQL Injection chính là việc chèn đoạn mã SQL độc hại (thực hiện các truy vấn không mong muốn) thông qua giao diện ứng dụng, có thể là ứng dụng Web hoặc

winform**.** Khi đó kẻ gian có thể truy cập vào hệ thống, thực thi các lệnh SQL để thay đổi thông tin như thêm user, hoặc sửa đổi thông tin trái phép và nghiêm trọng hơn có thể xóa tất cả các thông tin quan trọng... Ví dụ chúng ta có đoạn mã SQL thực hiện việc truy vấn đến Cơ sở dữ liệu thông qua giao diện đăng nhập. Bình thường thì câu lệnh truy vấn xem có người nào trong Cơ sở dữ liệu ứng với thông tin đã nhập vào không là:

string sqlcmd= “SELECT * FROM tblUser WHERE username= ‘“”

+ username +”‘“” AND password = ‘“”+ password + ‘“”;

Và kết quả ta mong muốn là:

SELECT * FROM tblUser WHERE username= „nguyenvana‟

AND password = „nva123‟;

Chú ý là giá trị khi truyền vào sẽ ở dạng nằm trong dấu nháy đơn. Khi đó nếu có thông tin đăng nhập trong Cơ sở dữ liệu thì sẽ trả về ngược lại thì không. Câu lệnh truy vấn đơn giản hơn là:

SELECT COUNT(*) FROM tblUser WHERE username= „nguyenvana‟

AND password = „nva123‟;

Kẻ tấn công có thể nhập bất kỳ tên đăng nhập nào (đúng hoặc sai) và mật khẩu có chứa phần như Bb‟ OR „‟=‟‟, khi đó câu lệnh truy vấn trên trở thành:

SELECT COUNT(*) FROM tblUser WHERE username= „anyname‟ AND password = „Bb‟ OR „‟=‟‟;

Chú ý là khi đó mệnh đề điều kiện WHERE thành ? and F or T => F or T => T (do phép AND được ưu tiên trước OR) và tất cả các dòng username/password sẽ trả về ứng dụng. Nếu ứng dụng kiểm tra thấy nếu số dòng là >0 thì CSDL bị tấn công.

Như vậy chúng ta thấy rằng kẻ tấn công đã lợi dụng cách thức câu lệnh truy vấn SQL thực hiện để truyền tham số vào và thay đổi cách thực thi câu lệnh. Do đó việc ứng dụng thiếu việc kiểm tra và kiểm soát thông số đầu vào, cụ thể ở đây là không kiểm soát chuỗi đầu vào sẽ làm tăng nguy cơ Cơ sở dữ liệu bị tấn công. Do đó để tránh việc bị tấn công Injection thì ứng dụng phải có khả năng kiểm tra được đầu vào như

sử dụng các tham số Parameters (trong .NET sử dụng các Parameters khi làm việc với Sql Command hoặc OleDb Command) chứ không sử dụng các câu lệnh SQL trực tiếp. khi đó .NET sẽ tự động validate kiểu dữ liệu và nội dung trước khi thực thi câu lệnh[8].

Ví dụ như trong ứng dung sử dụng câu lệnh SQL:

String strSQL= “SELECT * FROM tblUser WHERE “+ “username = „”+ obj.getUserName() + ““” + “password = „”+ obj.getPassword() + ““”;

Các hàm obj.getUserName() và obj.getPassword() sẽ kiểm tra các thông tin đầu vào, lọc các ký tự không mong muốn mà nó có nguy cơ gây ra việc thực thi một câu lệnh SQL tiếp sau…

Ngoài ra cũng cần phải kiểm soát các thông báo lỗi, mặc định thông báo lỗi sẽ không hiển thị chi tiết. Tránh việc sử dụng các dấu nháy trong đầu vào (nhập thông tin). Có một số ký tự khác có thể gây những vấn đề như:

-- // Ký tự chú giải SQL.

; // Ký tự kết thúc một câu lệnh SQL.

% //Ký tự trong câu lệnh LIKE của SQL.

Do đó cần phải kiểm tra đầu vào xem có những ký tự đặc biệt không được phép không.

Ví dụ tấn công SQL Injection sử dụng biến, nhoị để đăng nhập: Kẻ tấn công có thể nhập bất kỳ username nào và password là „OR 1=1-- (chú ý là ký tự -- là ký tự chú giải, nó sẽ kết thúc phạm vi thực hiện câu lệnh phía sau đó). Khi đó câu lệnh truy vấn trên trở thành:

SELECT * FROM tblUser WHERE username=“anyname“

AND password=““ OR 1=1-- „;

Chú ý là “OR 1=1” có thể đính kèm bất kỳ đầu vào nào và được đánh giá là TRUE.

Ví dụ khác có thể gây nguy hiểm cho Cơ sở dữ liệu như kẻ tấn công thực hiện việc xóa thông tin trong Cơ sở dữ liệu như: kẻ tấn công có thể nhập bất kỳ username nào và password là:

DELETE FROM tblUser WHERE username LIKE;

Khi đó câu lệnh truy vấn khi thực thi sẽ trở thành:

SELECT * FROM tblUser WHERE username=‟‟anyname‟‟ AND password=‟‟abc‟‟;

DELETE FROM tblUser WHERE username LIKE „„%‟‟;

Khi đó hệ thống sẽ thực thi hai câu lệnh:

*SELECT * FROM tblUser WHERE username="anyname"*

AND password="abc"; //Câu lệnh này sẽ không trả về gì.

DELETE FROM tblUser WHERE username LIKE „„%"; // Tùy thuộc vào quyền hạn thực thi câu lệnh của người dùng thì nguy cơ Cơ sở dữ liệu bị tấn công nằm ở đây.

Nghiêm trọng hơn nữa, lợi dụng kiến thức về ODBC (Open Database Connectivity, giữa bất kỳ ngôn ngữ nào và bất kỳ DBMS nào) cho phép thực thi câu lệnh sử dụng ký tự „„|‟‟ ví dụ như „„|shell(“cmd /c echo “& char(124) & “format c:”)|‟‟ câu lệnh này có thể sẽ xóa toàn bộ ổ C: và làm cho hệ thống bị sụp đổ.

Ngoài ra giả sử một trang Web nhận được một thông tin của một trạm nào đó có dạng như: http://www.siteapp.com/ThongTinTramKiemDinh.aspx?ID=2.

Khi đó thì chúng ta sẽ liên tưởng đến câu lệnh SQL như: SELECT * FROM tblThongTinTramKiemDinh WHERE id=5 và nếu ta thêm thông tin vào địa chỉ như: http://www.siteapp.com/ThongTinTramKiemDinh.aspx?ID=2 AND 1=1, nếu kết quả có trả về thông tin của trạm đó thì chắc chắn ứng dụng có lỗ hổng, hoặc chúng ta có thể gửi:

http://www.siteapp.com/ThongTinTramKiemDinh.aspx?ID=2

AND user_name()=‟‟dbo‟‟, nếu có kết quả trả về thì chúng ta biết chắc là quyền người dùng là dbo…

2. Bảo vệ thông tin truy xuất Cơ sở dữ liệu của Web Service

Ứng dụng Web Service sẽ cần sử dụng thông tin kết nối tới Cơ sở dữ liệu để thực hiện truy xuất thông tin, từ đó đáp ứng các yêu cầu bên Client. Do đó nguy cơ tiềm tàng chính là việc thông tin kết nối bị kẻ gian đánh cắp, vì chuỗi kết nối chứa thông tin chi tiết về Server, thông tin đăng nhập vào Cơ sở dữ liệu, nếu thông tin này

bị lộ ra thì kẻ gian dễ dàng truy cập được vào Cơ sở dữ liệu, và khi đó thì hậu quả sẽ không lường trước, vì vậy để giảm những nguy cơ này thì thông tin kết nối cần phải được mã hóa. Thông tin kết nối Cơ sở dữ liệu trong ứng dung .NET được lưu trong file cấu hình Web.config và có dạng như sau:

```
<connectionStrings>

<add name="dbconnection" connectionString="Data Source=[AddressServer];Integrated Security=true;Initial Catalog=[NameDB]; User name=[tendangnhap]; Password=[matkhau]"/>
```

Trong .NET cung cấp sẵn các chức năng cấu hình bảo mật để mã hóa và giải mã một số phần trong file cấu hình Web.config bao gồm:

RSA Protected Configuration Provider: mặc định trong .NET sử dụng thuật toán mã hóa khóa công khai RSA để mã hóa và giải mã thông tin.

Data Protection Confguration Provider: cung cấp giao diện ứng dụng để mã hóa và giải mã thông tin.

Trường hợp trên là mã hóa dưới dạng File System, nếu deployed ứng dụng lên host chẳng hạn (chạy trên IIS) thì chúng ta sử dung cú pháp sau:

Để mã hóa: aspnet_regiis.exe -pe “connectionStrings” -app “/[Web Service]” trong đó tham số -pe chỉ ra ứng dụng được xây dựng trên nền IIS (ứng dụng đã được triển khai trên IIS).

Để giải mã: aspnet_regiis.exe -pd “connectionStrings” -app “/[Web Service]”.

Bảo vệ ứng dụng Máy trạm

+ Phát hiện và ngăn chặn chương trình độc hại:

Phát hiện và ngăn chặn chương trình độc hại thông thường được thực hiện bởi các ứng dụng bảo mật riêng rẽ như tường lửa, chống virus và chống spyware. Tường lửa và khả năng kiểm soát chương trình - là chức năng quan trọng nhất bởi vì nó có thể kiểm soát được luồng tin đi vào/ra ở vùng nhân. Chỉ có tường lửa mới có thể ngăn được truy cập trái phép như mã độc, kiểm soát chương trình nào được phép truy cập mạng, và làm cho các máy trạm trở nên “vô hình” đối với hacker. Khả năng chống virus được sử dụng để phát hiện và ngăn chặn sự lây nhiễm của virus. Khả năng chống spyware là ngăn chặn sự lây nhiễm của sâu, trojans, adware, và keystroke loggers. Nó có thể cung cấp sự bảo vệ trong thời gian thực, chống việc cài

đặt spyware trên máy trạm, đồng thời có thể phát hiện và gỡ bỏ spyware đã được cài trước đó.

Điều quan trọng đối với tất cả các khả năng trên là người quản trị có thể kiểm soát tập trung và theo dõi được các máy trạm để đảm bảo chúng tuân thủ với chính sách an ninh.

+ Mã hóa và chống thất thoát dữ liệu:

Bảo vệ dữ liệu trên máy trạm là rất quan trọng bởi vì nó rất dễ bị đánh cắp hay bị mất. Kiểm soát để bảo vệ dữ liệu là bao gồm mã hóa toàn bộ ổ cứng, mã hóa thiết bị nhớ di động và kiểm soát các cổng/thiết bị trên máy trạm. Mã hóa có thể áp dụng các file riêng rẽ, thư mục, toàn bộ ổ cứng hoặc cho thiết bị nhớ di động.

```
private string SafeSqlLikeClauseLiteral(string inputSQL)
{
 // Thực hiện các thay thế sau đây:
 // ' becomes ''
 // [ becomes [[]
 // % becomes [%]
 // _ becomes [_]
 string s =
 inputSQL;
 s = inputSQL.Replace("'", "''");
 s = s.Replace("[", "[[]");
 s = s.Replace("%", "[%]");
 s = s.Replace("_",
 "[_]"); return s;
}
```

Kiểm soát cổng/thiết bị ngoại vi là công nghệ tương đối mới, cho phép các tổ chức quản lý một cách tập trung cổng, thiết bị ngoại vi nào được phép sử dụng trên từng máy trạm. Một lợi ích thực tế là ngăn cản việc copy dữ liệu được bảo vệ từ máy tính tới một thiết bị nhớ như USB. Kiểm soát cổng cũng ngăn chặn được việc lây lan virus từ các thiết bị nhớ ngoài vào máy tính rồi từ đó lây lan ra trong mạng công ty. Ví dụ như có hàm kiểm tra ký tự nhập:

Bảo vệ ứng dụng Web Service

Ngoài phía Client ra thì phía Web Service cung cấp những phương thức mà Client sử dụng, trong đó thực thi câu lệnh SQL hoặc gọi Store Procedure, cho nên việc thực thi các câu lệnh SQL này sử dụng store procedure trong trường hợp câu lệnh SQL không có tham số thì không cần phải quan tâm đến việc chống lại SQL Injection, tuy nhiên thì đa số các store procedure này cần tham số truyền vào, cho nên cần phải chú ý các tham số này. Ví dụ như có đoạn mã sau sử dụng tham số truyền vào:

```
using System.Data;

using System.Data.SqlClient;

using (SqlConnection connection = new SqlConnection(connectionString))

{

  DataSet userDataset = new DataSet();

  SqlDataAdapter myCommand = new SqlDataAdapter("LoginStoredProcedure", connection);
  myCommand.SelectCommand.CommandType  = CommandType.StoredProcedure;
  myCommand.SelectCommand.Parameters.Add("@u_id",  SqlDbType.VarChar, 11);
  myCommand.SelectCommand.Parameters["@u_id"].Value  = SSN.Text;

}
```

Tham số @u_id là tham số đầu vào và nó như một giá trị, không là mã thực thi, ngoài ra tham số này cũng được kiểm tra độ dài không quá 11 ký tự, nếu quá sẽ thông báo lỗi.

```
CREATE PROC
dbo.runQuerySQL @var text

AS

exec sp_executesql @var GO
```

Ứng dụng sử dụng các store procedure thực thi có dạng như:

Một ứng dụng sử dụng thủ tục như dạng này sẽ có những lỗ hổng như nếu tham số của thủ tục truyền vào biến @var có dạng như DROP TABLE tblTram; thì trong trường hợp này, bảng tblTram sẽ bị xóa; Thủ tục thực thi dưới quyền dbo, và tên thủ tục (runQuerySQL) sẽ cho kẻ gian biết được mục đích sử dụng và sẽ lợi dụng để thực thi các câu lênh SQL khác.

Một vấn đề nữa cũng cần chú ý đối vơi Web Service là thông tin truy xuất

CSDL nằm trong file cấu hình Web.config, cho nên việc bảo vệ thông tin này rất quan trọng nhằm ngăn chặn kẻ gian đánh cắp thông tin và truy xuất vào CSDL, cho nên cần phải mã hóa thông tin này. .NET đã hỗ trợ việc mã hóa, giải mã này, cho nên việc cần làm là chỉ việc thực thi câu lệnh đối với ứng dụng[8].

aspnet_regiis.exe -pef “connectionStrings”“D:\Projects 2011\VRService 2.0”

Chú ý là việc mã hóa này không cần phải thay đổi lại mã của chương trình, .NET hỗ trợ hết khi chương trình gọi đến chuỗi kết nối, nó sẽ được tự động giải mã.

Bảo mật Web Service

Do ứng dụng thực hiện trao đổi thông tin trên mô hình Web Service nên việc bảo mật Web Service có thể áp dụng ở ba cấp độ:

Platform/transport-level (point-to-point) security.

Application-level (custom) security.

Message-level (end-to-end) security.

Mỗi phương pháp tiếp cận đều có những ưu, nhược điểm khác nhau, việc lựa chọn phương pháp phụ thuộc vào các đặc điểm của kiến trúc và nền tảng liên quan đến việc trao đổi thông điệp.

1. Platform/Transport Level (Point-to-Point) Security:

Khi sử dụng cách này, giả sử là môi trường sử dụng là hệ điều hành Windows, ví dụ như mạng công ty:

- Web Server (IIS) cung cấp cách xác thực Basic, Digest, Integrated và Cerrtificate.
- ASP.NET Web service kế thừa một số đặc tính của ASP.NET là xác thực và ủy quyền.
- SSL và/hoặc IPSec có thể được sử dụng để cung cấp tính toàn vẹn và bảo mật thông điệp.

Sử dụng:

Mô hình vận chuyển, mức độ bảo mật đơn giản, đầy đủ (chủ yếu là trong mạng nội bộ), trong đó cơ chế vận chuyển và cấu hình đầu cuối có thể được kiểm soát chặt chẽ.

Một số vấn đề chính đối với mức độ bảo mật vận chuyển là: An ninh chặt chẽ và phụ thuộc vào nền tảng, cơ chế vận tải, bên cung cấp dịch vụ an ninh (NTLM, Kerberos,...) An ninh được áp dụng từ điểm-điểm, cho nên không cung cấp cho việc định tuyến thông qua các nút ứng dụng trung gian. Do các thông điệp cần được mã hóa nên cách tiếp cận này có thể tốn kém.

2. Application-level (custom) security:

Với cách tiếp cận này, ứng dụng có các đặc tính bảo mật tùy chỉnh ví dụ như:

- Một ứng dụng có thể sử dụng tiêu đề SOAP tùy chỉnh để xác thực thông tin với mỗi yêu cầu của Web service. Một cách chung hơn là qua ticket (hay giấy phép) trong tựa đề SOAP.
- Ứng dụng có sự linh hoạt trong việc tạo ra đối tượng Iprincipal chứa các vai trò. Điều này có thể là một lớp tùy chỉnh của lớp GenericPrincipal được cung cấp bởi .NET Framework.
- Ứng dụng có thể lựa chọn mã hóa những gì cần thiết, việc này cần phải có yêu cầu về an toàn lưu trữ khóa và nhà phát triển cần phải có kiến thức về các API mật mã.

Một kỹ thuật thay thế là sử dụng SSL để cung cấp bảo mật và toàn vẹn kết hợp với các header SOAP tùy chỉnh để thực hiện xác thực.

Sử dụng:

Sử dụng phương pháp này khi chúng ta muốn tận dụng lợi thế của một lược đồ Cơ sở dữ liệu hiện có của người dùng và vai trò, được sử dụng trong ứng dụng hiện tại.

Chúng ta muốn mã hóa một phần của thông điệp, chứ không phải toàn bộ.

3. Message-level (end-to-end) security:

Đây là cách tiếp cận linh hoạt mềm dẻo và mạnh mẽ nhất và được sử dụng trong các Web service Exhancements cho .NET

WS-Security mô tả thông số kỹ thuật cải tiến thông điệp SOAP, cung cấp tính toàn vẹn, bảo mật thông điệp và xác thực.

Xác thực được cung cấp trong thẻ bảo mật theo tiêu đề SOAP. Không có kiểu đặc tả nào của token được yêu cầu bởi WS-Security. Các token bảo mật có thể bào

gồm có Keberos ticket, X509 certificate hay XML hoặc mã thông báo nhị phân. Thông tin liên lạc an toàn được cung cấp bởi chữ ký số XML để đảm bảo cho tính toàn vẹn thông điệp và mã hóa XML cho bảo mật tin nhắn[8].

Sử dụng: WS-Security có thể được sử dụng để xây dựng một khuôn khổ cho việc trao đổi thông điệp an toàn trong môi trường không đồng bộ Web service.

Mức độ bảo mật: Độc lập vận chuyển, cho phép kiến trúc an ninh không đồng nhất, cung cấp bảo mật điểm-điểm và định tuyến bản tin thông qua các nút ứng dụng trung gian, hỗ trợ nhiều công nghệ mã hóa, hỗ trợ không thoái thác (non-repudiation).

Do đó giải pháp tối ưu là áp dụng Message-level (end-to-end) security cho việc bảo mật Web service.

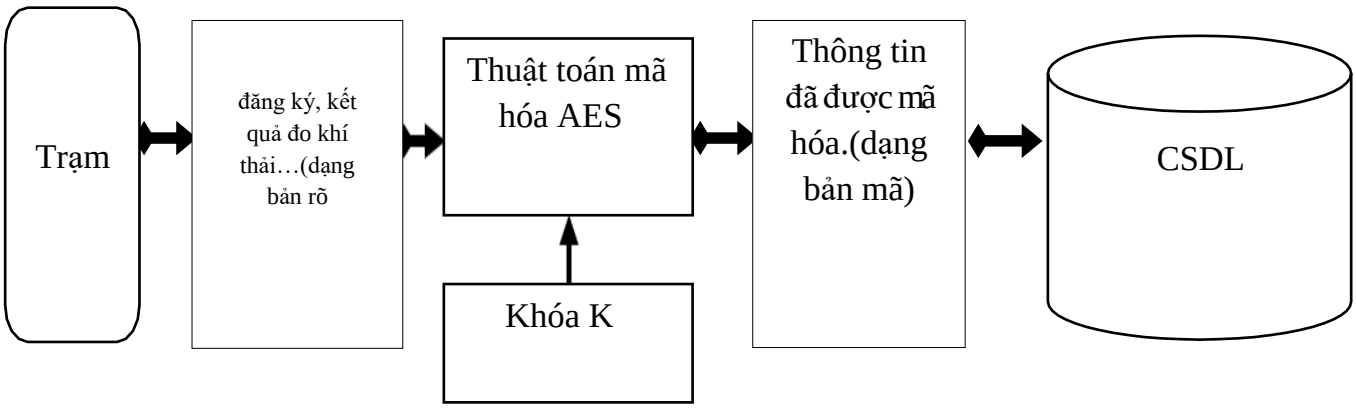
Với ứng dụng Web Service, xây dựng dựa trên công nghệ của Microsoft là Windows Communication Foundation (WCF).

Bảo mật dữ liệu tại Cơ sở dữ liệu

Bảo mật dữ liệu tại Cơ sở dữ liệu tức là những thông tin được lưu trong Cơ sở dữ liệu phải được mã hóa để những người không có thẩm quyền không thể xem được. Do đó ứng dụng cần phải đáp ứng được những công việc như:

- Thứ nhất mã hóa dữ liệu trước khi chèn vào Cơ sở dữ liệu.
- Thứ hai là cho phép hiển thị được dữ liệu đã mã hóa có trong Cơ sở dữ liệu dưới dạng bản mã.
- Thứ ba là hiển thị dữ liệu trong Cơ sở dữ liệu sau khi đã được mã hóa.

Mã hóa dữ liệu trước khi chèn vào CSDL:



Hình 3.3 Mã hóa dữ liệu trước khi chèn vào Cơ sở dữ liệu.

Tất cả các thông tin về đăng ký kiểm định, thông tin liên quan đến phương tiện đăng ký và các kết quả đo khí thải sẽ được mã hóa bằng thuật toán AES và lưu trong Cơ sở dữ liệu.

Quá trình chèn dữ liệu vào CSDL gồm có 3 giai đoạn chính:

1. Giai đoạn lấy thông tin từ bên Client gửi đến.
2. Giai đoạn sử dụng thuật toán mã hóa AES để mã hóa dữ liệu gửi đến với khóa K.
3. Giai đoạn lấy dữ liệu đã được mã hóa chèn vào CSDL.

Hiển thị dữ liệu trong Cơ sở dữ liệu sau khi đã được mã hóa:

Khi Trạm cần truy xuất thông tin đã đăng ký của một phương tiện thì cần truy xuất thông tin của phương tiện đó trên Văn phòng Cục, Thông tin trên cục do Web Service gửi về dưới dạng mã hóa (do thông tin lưu trong Cơ sở dữ liệu ở dưới dạng mã hóa). Do đó để hiển thị được những thông tin này thì cần phải giải mã với khóa.

Do đó quá trình thực hiện bao gồm có những bước sau:

1. Thứ nhất là lấy dữ liệu dưới dạng bãn mã trong Cơ sở dữ liệu.
2. Thứ hai là sử dụng thuật toán giải mã AES để giải mã dữ liệu vừa lấy với khóa giải mã K‟ người dùng nhập vào.
3. Thứ ba là hiển thị dữ liệu đã được giải mã.

Lấy thông tin khóa bí mật:

Thông tin khóa bí mật K sẽ được lưu dưới dạng file mã hóa trong các thiết bị lưu trữ như USB, CD, thẻ nhớ,...hoặc có thể nhập trực tiếp khóa khi chương trình yêu cầu.

Trường hợp nhập trực tiếp khóa vào trong chương trình, chương trình sẽ yêu cầu nhập khóa sau khi xác thực thành công.

Trường hợp chương trình yêu cầu chọn thiết bị lưu trữ file khóa, chọn thiết bị như USB, thẻ nhớ. Chương trình sẽ xác thực thông tin, giải mã lấy ra khóa được lưu trong file khóa. Đối với những thiết bị lưu trữ không có số Serial thì việc xác thực file khóa sẽ phải dùng đến thông tin GUID (Globally Unique IDentifier) trong chương trình.

Cụ thể như sau:

a. Quá trình tạo file khóa lưu trong thiết bị có serial như thẻ nhớ, USB,…

Lấy ra ID của thiết bị. Chọn ra một khóa được chỉ định kết hợp với thông tin GUID của ứng dụng rồi lưu vào file khóa (key.txt).

Mã hóa file khóa bằng thuật toán AES với khóa là ID của thiết bị.

Lưu file khóa lại ứng với thiết bị đó.

b. Quá trình lấy ra khóa trong thiết bị

Lấy ra ID của thiết bị. Giải mã file khóa bằng ID của thiết bị sau đó tách nội dung trong file khóa lấy ra GUID và so sánh với GUID của ứng dụng, nếu khớp nhau thì file khóa đượcxác thực. Điều này sẽ ngăn được tình trạng file khóa được sao chép sang các thiết bị lưu trữ khác nhau do Serial của mỗi thiết bị lưu trữ là duy nhất.

Lấy khóa trong file khóa đã được giải mã.

3.5 DEMO chương trình

- Nhập tên đăng nhập, mật khẩu, thông tin trạm đo:

Hình 3.4 Đăng nhập.

Quá trình xác thực diễn ra, quá trình này gồm có hai công đoạn chính là:

1. Xác thực Máy trạm với Máy chủ sử dụng kỹ thuật xác thực SSL.
2. Xác thực người dùng.

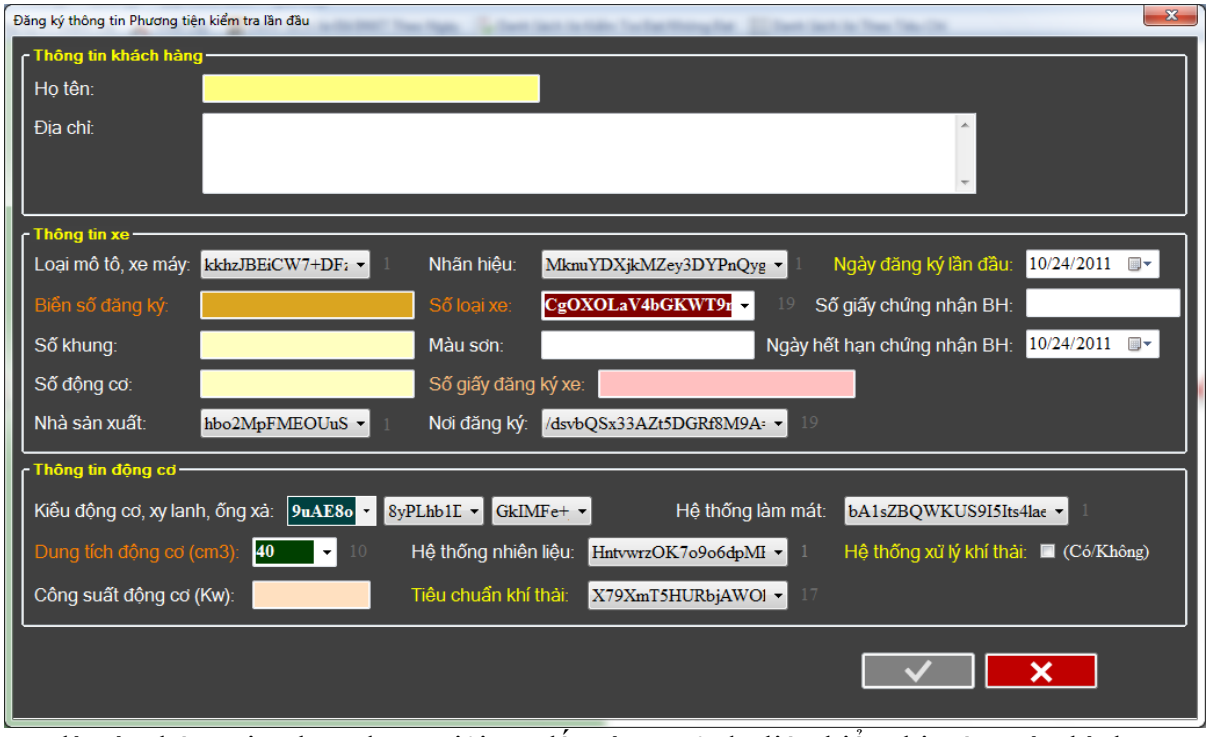
Nếu quá trình này thành công thì vào được chương trình, không thì không được vào. Khi đăng nhập thành công thì chương trình hiển thị hộp thoại yêu cầu nhập khóa bí mật. Việc nhập này có hai cách là nhập trực tiếp bằng bàn phím hoặc thông qua files khóa lưu trong thiết bị lưu trữ như USB, Memory Card…Việc nhập trực tiếp đòi hòi phải nhớ khóa bí mật và nếu khóa phức tạp thì việc nhập khóa sẽ rất khó khăn, đôi khi việc xác thực đúng khóa bí mật cũng là một vấn đề cần chú ý vì nếu nhập sai thì thông tin mã hóa hoặc giải mã phía Server phục vụ cho việc hiển thị trên Website sẽ không

đúng hoặc không giải mã được, cho nên cách nhập khóa được lưu trong thiết bị lưu trữ sẽ khắc phục được điều này.

Nếu bỏ qua bước nhập khóa bí mật hoặc nhập không đúng (khóa bí mật sẽ không được xác thực) thì vẫn vào được chương trình nhưng sẽ không thể thực hiện được chức năng nào như đăng ký mới phương tiện đến kiểm định hoặc đo khí thải của phương tiện.

➢ Màn hình đăng ký thông tin:

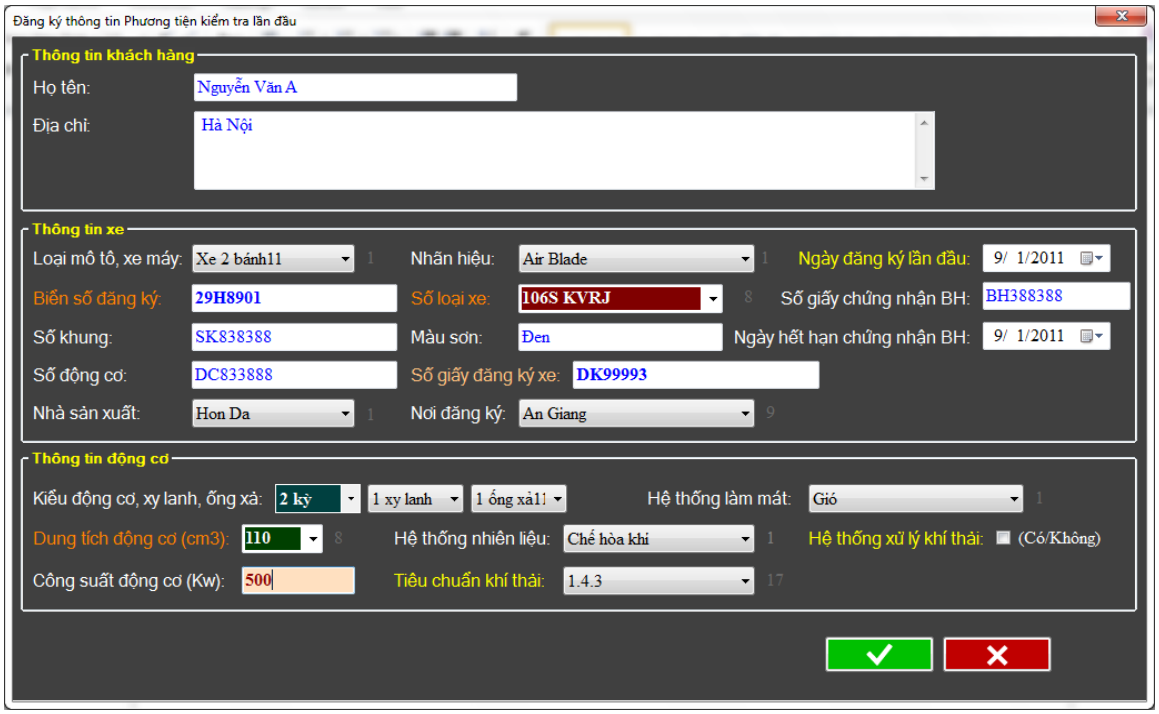
Trường hợp khóa bí mật không được nhập. Màn hình hiển thị thông tin đăng ký



sẽ là các thông tin chưa được giải mã lấy từ cơ sở dữ liệu hiển thị trên màn hình:

Hình 3.5 Đăng ký thông tin phương tiện mới chưa giải mã.

Trường hợp khóa bí mật được nhập. Màn hình hiển thị thông tin đăng ký sẽ là các thông tin đã được giải mã sau khi lấy thông tin đã được mã hóa từ cơ sở dữ liệu về Client và giải mã hiển thị trên màn hình:



Hình 3.6 Đăng ký thông tin phương tiện mới đã giải mã.

Sau khi đăng ký thông tin và tiến hành lưu lại thì thông tin đăng ký sẽ được mã hóa và gửi qua Web Service lên cục và tiến hành lưu lại thông tin này.

- Thông tin được mã hóa trong cơ sở dữ liệu:
- Tiến hành đăng ký kiểm định:

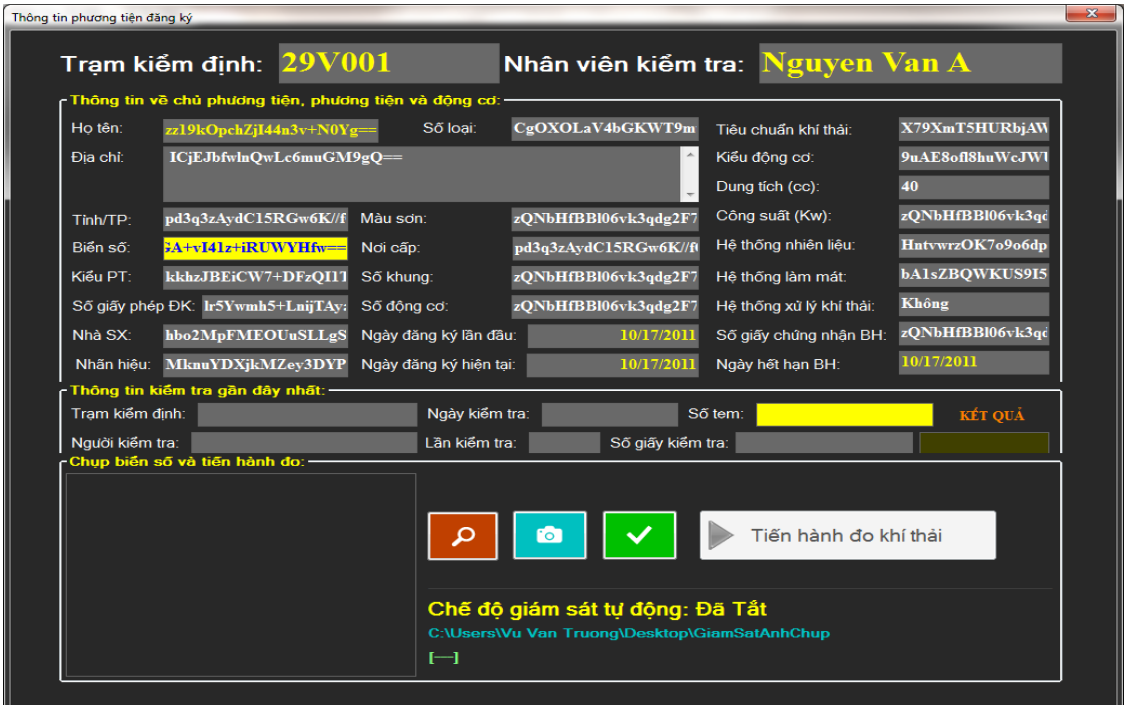
Hình 3.7 Đăng ký kiểm định.

- Thông tin phương tiện đã đăng ký:

Đối với trường hợp chưa nhập khóa bí mật thì do thông tin về biển số của phương tiện được lưu dưới dạng mã hóa trong cơ sở dữ liệu cho nên ở đây giả sử biết

được thông tin biển số đã được mã hóa của một phương tiện bất kỳ thì khi nhập thông tin biển số (dạng đã mã hóa của phương tiện như xHna9ja==) thì thông tin đăng ký của phương tiện sẽ lấy ở cục về chưa được giải mã được hiển thị.

Ngược lại đối với trường hợp khóa bí mật được nhập chính xác thì để tiến hành kiểm định thì chỉ việc nhập biển số thì thông tin biển số sẽ được mã hóa bằng khóa bí mật và so sánh trong cơ sở dữ liệu, nếu đúng thì Server sẽ trả về thông tin đăng ký của phương tiện dưới dạng mã hóa và thông tin này sau đó sẽ được giải mã bằng khóa bí mật để hiển thị trên màn hình.



Hình 3.8 Thông tin phương tiện đã đăng ký chưa giải mã.

Thông tin phương tiện đã đăng ký là thông tin được lấy từ Văn phòng Cục (đã mã hóa) về Trạm, sau đó được giải mã để hiển thị.

Hình 3.9 Thông tin phương tiện đã đăng ký đã giải mã.

Tương tự, sau khi đo kiểm định khí thải thì thông tin về kết quả kiểm định cũng được mã hóa trước khi gửi lên mạng qua Web Service, sau đó thông tin mã hóa được gửi lên Văn phòng Cục và lưu trong Cơ sở dữ liệu.

3.6 Kết luận chương

Trong những năm gần đây việc kiểm định và kiểm định khí thải ô tô, xe máy là việc rất quan trọng, năm trong chương trình làm giảm hiệu ứng nhà kính, giảm bớt xe ô tô, xe máy đã quá hạn sử dụng, không đảm bảo chất lượng khí thải khi tham gia giao thông. Trong phạm vi hạn chế của luận văn đã chỉ ra và góp phần nhỏ bé vào việc kiểm soát chất lượng khí thải, trong chương trình làm giảm khí thải nhà kính của Chính phủ.

Chương này đã tìm hiểu sơ lược về một trung tâm tích hợp dữ liệu, các vấn đề an ninh an toàn thường gặp phải, nghiên cứu một số giải pháp đã áp dụng trong việc đảm bảo an ninh cho trung tâm và cuối cùng là mục đích chính của luận văn.

Chương 3 ứng dụng một số giải pháp được nghiên cứu tại chương 2 để xây dựng ứng dụng, đảm bảo một số yêu cầu cơ bản về bảo vệ, bảo mật thông tin (ở đây là bảo mật CSDL ứng dụng).

KẾT LUẬN

1. Tính năng của luận văn:

Bảo đảm Bảo mật cho Trung tâm tích hợp dữ liệu là một vấn đề đã và đang được quan tâm. Luận văn đã nghiên cứu một số giải pháp đảm bảo bảo mật cho Trung tâm tích hợp dữ liệu và ứng dụng một trong số những giải pháp đó xây dựng ứng dụng đảm bảo bảo mật cho chương trình kiểm định khí thải cho Cục Dự trữ Nhà nước khu vực Hà Nội. Trong khi tiến hành xây dựng ứng dụng thì kết quả phân tích dựa vào việc đi khảo sát thực trạng ứng dụng hiện tại tại trung tâm tích hợp dữ liệu Cục Dự trữ Nhà nước khu vực Hà Nội.

Kết quả đạt được của luận văn:

Đã xây dựng được ứng dụng thỏa mãn một số yêu cầu đặt ra như:

- Xác thực ứng dụng Máy trạm với Máy chủ.
- Bảo vệ ứng dụng truy cập Cơ sở dữ liệu.
- Bảo vệ dữ liệu trên đường truyền mạng công cộng.
- Bảo vệ dữ liệu tại Cơ sở dữ liệu.
- Các tính năng của chương trình (cập nhật, đăng ký, gửi nhận thông tin,...).

2. Hạn chế cần khắc phục:

Tuy chương trình đã đạt được một số yêu cầu đặt ra. Nhưng việc bảo vệ thông tin chưa tiến hành triệt để như triển khai VPN để bảo vệ thông tin theo hướng truyền,... chương trình chỉ dừng lại việc thiết lập bảo vệ Cơ sở dữ liệu riêng lẻ, chưa có đồng bộ giữa các Máy chủ Cơ sở dữ liệu.

3. Hướng phát triển của luận văn:

Mục đích của đề tài mới chỉ tập trung vào nghiên cứu một số vấn đề đảm bảo an toàn và bảo mật thông tin trong một trung tâm tích hợp dữ liệu nhỏ, ở đây là trung tâm của Cục Dự trữ Nhà nước khu vực Hà Nội. Trong tương lai sẽ cải tiến các tính năng của chương trình đáp ứng được một số vấn đề đã nêu trong phần hạn chế.

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1]. Đặng Trần Khánh, Phan Trọng Nhân, "Bảo vệ tính riêng tư cho các dịch vụ dựa trên vị trí" Nhà xuất bản Đại học Quốc gia TP Hồ Chí Minh.

[2]. Thái Hồng Nhị, Phạm Minh Việt (2011), *Giáo trình An Toàn Thông tin*, Nhà xuất bản Khoa học và Kỹ thuật.

[3]. Pgs.Ts Nguyễn Khanh Văn "An Toàn Và Bảo Mật Hệ Thống Thông Tin" - ĐH Bách Khoa Hà Nội.

[4]. Trung tâm khoa học Tự nhiên và Công nghệ Quốc gia (2004), "Thiết kế và xây dựng mạng LAN, WAN".

Tiếng Anh

[5]. M. Gertz, S. Jajodia, "Handbook of Database Security: Applications and Trends, Springer Verlag, ISBN 978-0-387-48532-4, 2008 Sách tham khảo:

[6]. D.C. Knox, "Effective Oracle Database 10g Security by Design", Oracle Press, ISBN 0-07-223130-0, 2004

[7]. S. Castano, M. Fugini, G. Martella, and P. Samarati. "Database Security", ACM Press & Addison-Wesley, ISBN 0-201-59375-0, 1995

[8]. W. Mao, "Modern Cryptography: Theory and Practice", 3rd Ed., Prentice Hall, ISBN 0-13-066943-1, 2003

[9]. T.R. Peltier, J. Peltier, J. Blackley, "Information Security Fundamentals", Auerbach Publications, ISBN 0-8493-1957-9, 2005