BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM

NGUYỄN ANH TUẤN

MỘT SỐ KỸ THUẬT KIỂM THỬ AN TOÀN HỆ THỐNG

LUẬN VĂN THẠC SĨ

Chuyên ngành : Công nghệ thông tin

Mã số ngành : 60480201

TP. HỒ CHÍ MINH, tháng 7 năm 2015

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM

NGUYỄN ANH TUẤN

MỘT SỐ KỸ THUẬT KIỂM THỬ AN TOÀN HỆ THỐNG

LUẬN VĂN THẠC SĨ

Chuyên ngành : Công nghệ thông tin

Mã số ngành : 60480201

CÁN BỘ HƯỚNG DẪN KHOA HỌC: PGS.TS LÊ TRỌNG VĨNH

TP. HỒ CHÍ MINH, tháng 7 năm 2015

CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM

Cán bộ hướng dẫn khoa học : PGS.TS Lê Trọng Vĩnh

Luận văn Thạc sĩ được bảo vệ tại Trường Đại học Công nghệ TP. HCM ngày 15 tháng 08 năm 2015.

Thành phần Hội đồng đánh giá Luận văn Thạc sĩ gồm:

TT	Họ và tên	Chức danh Hội đồng
1	PGS.TSKH Nguyễn Xuân Huy	Chủ tịch
2	PGS.TS Lê Hoài Bắc	Phản biện 1
3	TS Trần Đức Khánh	Phản biện 2
4	PGS.TS Đỗ Phúc	Ủy viên
5	TS Võ Đình Bảy	Ủy viên, Thư ký

Xác nhận của Chủ tịch Hội đồng đánh giá Luận sau khi Luận văn đã được sửa chữa (nếu có).

Chủ tịch Hội đồng đánh giá LV

PGS.TSKH Nguyễn Xuân Huy

TRƯỜNG ĐH CÔNG NGHỆ TP. HCM
PHÒNG QLKH – ĐTSĐH

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

TP. HCM, ngày 12 tháng 07 năm 2015

NHIỆM VỤ LUẬN VĂN THẠC SĨ

Họ tên học viên: Nguyễn Anh Tuấn — Giới tính: Nam
Ngày, tháng, năm sinh: 31/01/1980 — Nơi sinh: Đồng Tháp
Chuyên ngành: Công nghệ thông tin — MSHV: 1341860029

I- Tên đề tài:

Một số kỹ thuật kiểm thử an toàn hệ thống

II- Nhiệm vụ và nội dung:

- Các vấn đề liên quan đến an toàn của hệ thống.
- Các công cụ phát hiện các lỗ hổng của hệ thống.
- Một số các lỗ hổng thường gặp.
- Đưa ra các kiến nghị về sự an toàn của hệ thống

III- Ngày giao nhiệm vụ: 19/08/2014

IV- Ngày hoàn thành nhiệm vụ: 10/06/2015

V- Cán bộ hướng dẫn: PGS.TS Lê Trọng Vĩnh

CÁN BỘ HƯỚNG DẪN — **KHOA QUẢN LÝ CHUYÊN NGÀNH**

Lê Trọng Vĩnh

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Các số liệu, kết quả nêu trong Luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Tôi xin cam đoan rằng mọi sự giúp đỡ cho việc thực hiện Luận văn này đã được cảm ơn và các thông tin trích dẫn trong Luận văn đã được chỉ rõ nguồn gốc.

Học viên thực hiện Luận văn

Nguyễn Anh Tuấn

LỜI CÁM ƠN

Tôi chân thành sâu sắc biết ơn thầy PGS.TS Lê Trọng Vĩnh đã hết lòng hướng dẫn tôi trong quá trình thực hiện luận văn.

Tôi xin chân thành cám ơn đến quý thầy, cô Khoa Công nghệ Thông tin Trường Đại học Công nghệ Tp Hồ Chí Minh đã giúp đỡ và tạo điều kiện cho tôi nghiên cứu và học tập để hoàn thành luận văn này.

Tôi xin bày tỏ lòng biết ơn đến gia đình và người thân đã động viên tôi vượt qua khó khăn để hoàn thành khóa học và luận văn này.

Tôi cũng muốn bày tỏ lòng biết ơn đến Ban lãnh đạo Công ty TNHH MTV Cấp nước và môi trường đô thị Đồng Tháp, nơi tôi công tác, đã tạo điều kiện và hỗ trợ tôi hoàn thành khóa học và luận văn này.

Tác giả luận văn

Nguyễn Anh Tuấn

TÓM TẮT

Luận văn giới thiệu các lỗ hổng bảo mật thường gặp, đưa ra một quy trình kiểm thử an toàn hệ thống thông tin sử dụng các công cụ trong bộ công cụ mã nguồn mở Kali Linux với tiêu chí: tin cậy, dễ sử dụng. Dựa vào đó, người quản trị hệ thống thông tin không chuyên về bảo mật có thể dễ dàng tự đánh giá hệ thống họ đang phụ trách và khắc phục điểm yếu nếu có, nhằm giảm bớt nguy cơ và thiệt hại do việc mất an toàn hệ thống gây ra.

ABSTRACT

The thesis introduces common security vulnerabilities, giving a secure testing process of information system by using the tools available in Kali Linux with criteria : reliability , ease of use . Based on this, the nonprofessional information system administrator of security can be easily self-assessment the system they are in charge of and overcomes any weakness if happened, to reduce the risk and damage caused by unsafe system .

MỤC LỤC

DANH MỤC CÁC TỪ VIẾT TẮT

STT	Từ viết tắt	Từ đầy đủ
1	ACL	Access control list
2	AES	Advanced Encryption Standard
3	ARP	Address Resolution Protocol
4	CMS	Content Management System
5	CNTT	Công nghệ thông tin
6	CPU	Central Processing Unit
7	CSDL	Cơ sở dữ liệu
8	CSRF	Cross-Site request forgery
9	DDOS	Distributed Denial of Service
10	DOM	Document Object Model
11	DOS	Denial of Services
12	GPS	Global Positioning System
13	GPU	Graphics processing unit
14	HTML	HyperText Markup Language
15	ICMP	Internet Control Message Protocol
16	IDS	Intrusion detection systems
17	IMAP	Internet Message Access Protocol
18	IP	Internet Protocol
19	IPS	Intrusion detection systems
20	ISSAF	Information Systems Security Assessment Framework
21	LDAP	Lightweight Directory Access Protocol
22	MITM	Man in the middle
23	NFC	Near-Field Communications
24	NIST	National Institute of Standards and Technology
25	OS	Operating system
26	OSINT	Open-source intelligence
27	OSTMM	Open Source Security Testing Methodology Manual
28	OWASP	Open Web Application Security Project
29	POP3	Post Office Protocol version 3
30	RAID	Redundant Arrays of Independent Disks

31	RDP	Remote Desktop Protocol
32	RFID	Radio Frequency Identification
33	Session id	Session identifier
34	SFTP	Secure File Transfer Protocol
35	SIP	Session Initiation Protocol
36	SMTP	Simple Mail Transfer Protoco
37	SSL	Secure Sockets Layer
38	TCP	Transmission Control Protocol
39	TLS	Transport Layer Security
40	URL	Uniform Resource Locator
41	VOIP	Voice over Internet Protocol
42	VPN	Virtual private network
43	WASC - TC	Web Application Security Consortium - Threat Classification
44	WEP	Wireless Encryption Protocol
45	WPA	Wi-Fi protected access
46	XSS	Cross-Site Scripting

DANH MỤC CÁC BẢNG

DANH MỤC BIỂU ĐỒ, ĐỒ THỊ, HÌNH ẢNH

MỞ ĐẦU

1. Lý do chọn đề tài:

Việt Nam trong vài năm gần đây, mức độ tấn công các hệ thống CNTT, nhất là các trang web thương mại điện tử, trang tin điện tử của chính phủ, cũng như tốc độ lây nhiễm mã độc trên máy tính cá nhân ngày càng tăng. Theo các báo cáo về bảo mật từ quý 3 năm 2013 đến quý 2 năm 2014 của Microsoft [12], [13], Việt Nam luôn nằm trong năm quốc gia hàng đầu có sự gia tăng lây nhiễm mã độc mạnh.

Đối với một hệ thống CNTT, việc kiểm tra hệ thống có an toàn trước các cuộc tấn công của tin tặc vào hệ thống thông tin là một việc làm thường xuyên và rất quan trọng. Trên thực tế, người quản trị hệ thống thường đi sau tin tặc trong việc ngăn ngừa xâm nhập, tấn công hệ thống do nhiều nguyên nhân, trong đó có nguyên nhân chính là chậm trễ trong việc cập nhật thông tin, không thường xuyên đánh giá độ an toàn bảo mật của hệ thống. Ngoài ra, đa phần người quản trị hệ thống làm việc trong các cơ quan nhà nước, các doanh nghiệp vừa và nhỏ, các doanh nghiệp không liên quan đến các ngành ngân hàng, công nghệ thông tin thường không chuyên về bảo mật hệ thống. Đó cũng là yếu tố làm gia tăng sự rủi ro khi vận hành hệ thống thông tin.

Đã có nhiều đề tài nghiên cứu như: nâng cao bảo mật hệ thống mạng không dây[1], kiểm thử bảo mật website[2], xây dựng công cụ đánh giá an toàn website [3], nghiên cứu về phương pháp tấn công và ngăn chặn tấn công mạng máy tính[4]. Nhưng chưa có một đề tài nghiên cứu nào đưa ra bộ công cụ kiểm thử an toàn hệ thống thông tin chung (không chỉ dành riêng cho ứng dụng web) cho người quản trị hệ thống không chuyên về bảo mật, đồng thời đưa ra các gợi ý phòng ngừa, ngăn chặn. Chính vì lý do trên, tôi chọn đề tài **"Một số kỹ thuật kiểm thử an toàn hệ thống".**

2. Mục đích, đối tượng và phạm vi nghiên cứu:

2.1 Mục đích nghiên cứu:

Giới thiệu một bộ công cụ dễ sử dụng, nhiều tiện ích cùng đưa ra một quy trình kiểm thử. Dựa vào đó họ tự đánh giá được độ an toàn của hệ thống đang quản

lý hoặc sản phẩm phần mềm do họ tạo ra một cách chính xác, dễ dàng và nhanh chóng.

2.2 Đối tượng nghiên cứu:

- Tổng quan về tình hình an toàn thông tin hiện nay tại Việt Nam.
- Các loại phương thức tấn công, các phương pháp phòng chống, ngăn ngừa.
- Các tiêu chuẩn đánh giá an toàn hệ thống thông tin.
- Bộ công cụ Kali Linux và xây dựng quy trình kiểm thử an toàn hệ thống dùng Kali Linux.

2.3 Phạm vi nghiên cứu:

Xây dựng quy trình kiểm thử an toàn hệ thống thông tin (không thử nghiệm tấn công) dùng một số công cụ trong Kali Linux. Đánh giá thực nghiệm trên hệ thống máy tính Windows trong mạng LAN; máy chủ ứng dụng Web, đưa ra đề xuất các biện pháp phòng chống, ngăn ngừa đối với các lỗ hổng bảo mật phổ biến.

3. Ý nghĩa khoa học và thực tiễn của đề tài:

Đề tài tập trung tìm hiểu về các loại lỗ hổng bảo mật phổ biến hiện nay và các biện pháp phòng chống, ngăn ngừa. Giới thiệu các công cụ kiểm thử an toàn hệ thống theo cách đơn giản, dễ sử dụng, dành cho đối tượng sử dụng là người quản lý hệ thống, người lập trình, người dùng cuối không chuyên về bảo mật.

Dự kiến các đóng góp chính của luận văn:

- Trình bày được các loại phương thức tấn công thông qua lỗ hổng bảo mật phổ biến, các biện pháp phòng chống, ngăn ngừa tương ứng.
- Đưa ra quy trình kiểm thử an toàn hệ thống.
- Nắm rõ và sử dụng các công cụ trên bộ công cụ Kali Linux.
- Thực nghiệm đánh giá mức độ an toàn hệ thống: máy tính Windows trong hệ thống mạng LAN; máy chủ ứng dụng Web.

4. Cấu trúc của luận văn:

Luận văn gồm các phần như sau:

Mở đầu, trình bày lý do chọn đề tài, mục tiêu, phạm vi và những đóng góp chính của luận văn, giới thiệu cấu trúc của luận văn.

Chương 1: Tổng quan, trình bày tình hình chung về an toàn thông tin hiện nay tại Việt Nam. Các khái niệm về an toàn hệ thống thông tin.

Chương 2: Một số tiêu chuẩn kiểm thử an toàn hệ thống thông tin, trình bày về các tiêu chuẩn, quy trình kiểm thử và đánh giá an toàn hệ thống: OWASP, OSTMM, ISSAF, WASC-TC, NIST SP 800-115.

Chương 3: Nguy cơ mất an toàn hệ thống từ lỗi của ứng dụng, trình bày nguy cơ mất an toàn hệ thống từ các ứng dụng phổ biến và cách phòng chống, ngăn ngừa tương ứng.

Chương 4: Sử dụng Kali Linux kiểm thử an toàn hệ thống, đưa ra quy trình kiểm thử an toàn hệ thống và giới thiệu các công cụ có trên Kali Linux. Sử dụng chúng để dò tìm lỗ hổng bảo mật của một hệ thống: máy tính Windows trong mạng LAN; máy chủ ứng dụng Web.

Kết luận và kiến nghị, trình bày tóm lược kết quả của luận văn và các đề nghị liên quan đến luận văn.

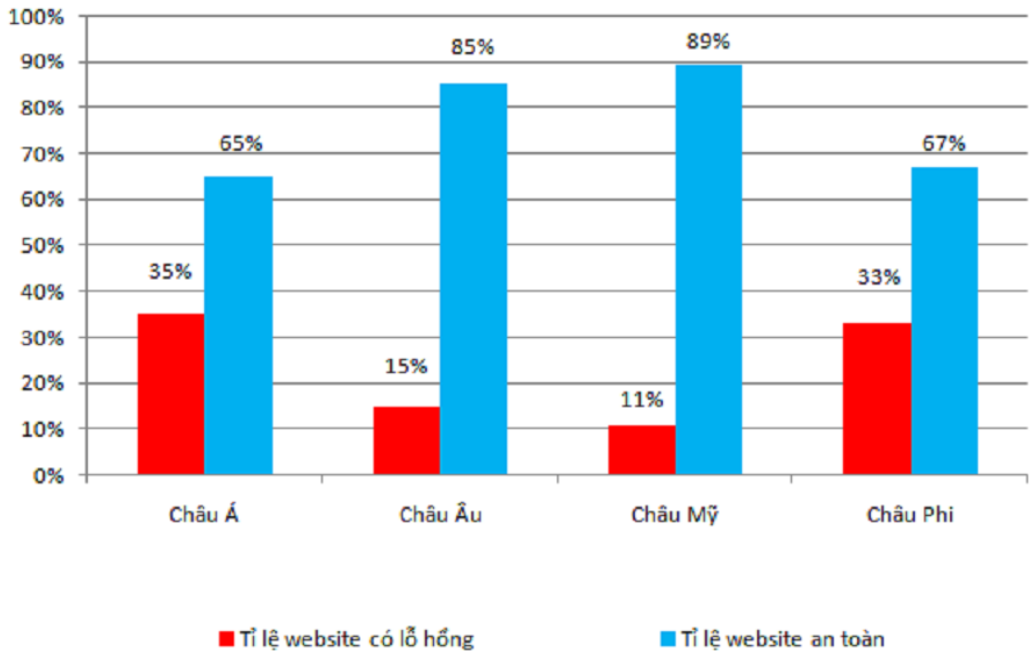
Danh mục tài liệu tham khảo.

Chương 1. TỔNG QUAN

1.1 Tình hình chung về an toàn thông tin hiện nay tại Việt Nam:

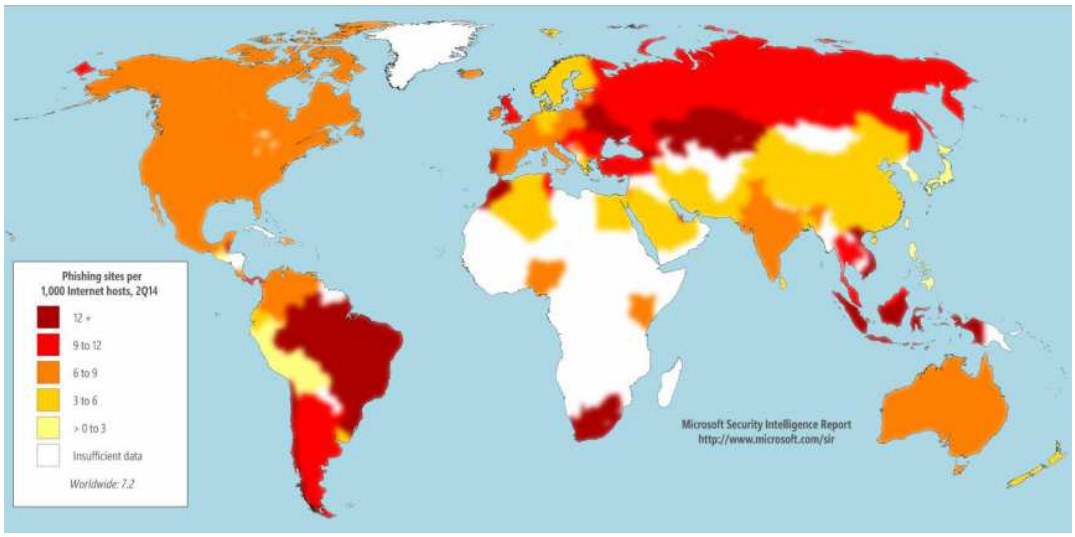
Theo thống kê của Bkav[4], trong năm 2012 có tới 2.203 trang web của các cơ quan doanh nghiệp tại Việt Nam bị tấn công, chủ yếu thông qua các lỗ hổng trên hệ thống mạng. So với năm 2011 (có 2.245 trang web bị tấn công), con số này hầu như không giảm.

Theo một nghiên cứu đánh giá cũng của BKAV công bố vào tháng 03 năm 2014 [5], tỉ lệ số trang web có lỗ hổng bảo mật tại Việt Nam là hơn 40%.

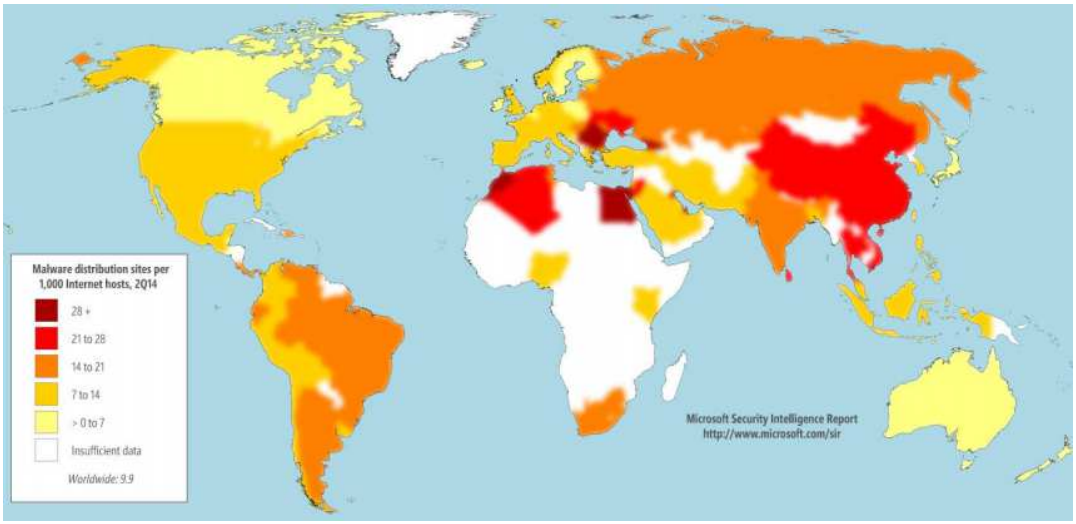


Hình 1.1: Tỉ lệ các trang web có lỗ hổng an ninh theo khu vực [5]

Trong báo cáo về bảo mật của Microsoft năm 2013 [12] và 2014 [13], Việt Nam cũng thuộc những nước có tỉ lệ xuất hiện các trang lừa đảo phising và phân phối malware thuộc loại cao trên thế giới. Tỉ lệ % số máy tính phát hiện có malware trên số máy tính có cài các sản phẩm bảo vệ máy tính của Microsoft (không dưới 100.000 máy tính) tại Việt Nam trong quí 3 và quí 4 năm 2013 lần lượt là 45.31% và 49.22% , trong quí 1 và quí 2 năm 2014 lần lượt là 60,8% và 52% thuộc những nước có tỉ lệ máy tính nhiễm malware cao trên thế giới.



Hình 1.2: Bản đồ tỉ lệ trang web lừa đảo phising, quí 2 – 2014 [13]



Hình 1.3: Bản đồ tỉ lệ trang web phân phối malware, quí 2 – 2014 [13]

Từ thực trạng trên cho thấy các đợt tấn công của tin tặc nhắm vào các doanh nghiệp, trang web hiện đang ngày càng tăng. Việc nghiên cứu và phổ cập kiến thức về an toàn hệ thống là cần thiết cho mọi người, nhất là đối với nhà lập trình, người quản trị các hệ thống thông tin nhỏ.

1.2 Các khái niệm trong lĩnh vực kiểm thử an toàn hệ thống thông tin:

1.2.1 Khái niệm hệ thống:

Hệ thống là một tập hợp các phần tử vật chất và phi vật chất, như: con người, máy móc, dữ liệu, các phương pháp xử lý, các qui tắc, quy trình xử lý… Các phần tử đó tương tác với nhau và cùng hoạt động để hướng tới mục đích chung.

1.2.2 Khái niệm hệ thống thông tin:

Hệ thống thông tin (information system) là hệ thống mà mối liên hệ giữa các thành phần của nó cũng như mối liên hệ giữa nó với các hệ thống thông tin khác là sự trao đổi thông tin. Mục tiêu của hệ thống thông tin là cung cấp thông tin phục vụ cho hoạt động của con người trong một tổ chức nào đó.

1.2.3 Khái niệm an toàn hệ thống thông tin:

Theo Matt Bishop [14], an toàn máy tính xét trên tính bí mật, tính toàn vẹn, tính sẵn sàng. Ba tính đó còn gọi là tam giác C-I-A (**c**onfidentiality, **i**ntegrity, **a**vailability). Đảm bảo an toàn hệ thống thông tin là đảm bảo an toàn của hệ thống thông tin (phần cứng, phần mềm, dữ liệu) trước các mối đe dọa (sự truy cập, sửa đổi, phá hoại dữ liệu bất hợp pháp) bằng các biện pháp kỹ thuật lẫn phi kỹ thuật (mã hóa, kiểm soát truy cập, chính sách…). Một hệ thống thông tin được xem là an toàn khi đảm bảo ít nhất ba mục tiêu cơ bản: tính bí mật, tính toàn vẹn, tính sẵn sàng. Ngoài ra, còn có thể có các mục tiêu khác như: tính không thể chối cãi, tính xác thực,..

1.2.3.1 Tính bí mật (confidentiality):

Đảm bảo tính bí mật của thông tin, nghĩa là thông tin chỉ được phép truy cập (đọc) bởi những đối tượng (người, chương trình máy tính…) được cấp phép. Tính bí mật của thông tin có thể đạt được bằng cách giới hạn truy cập về cả mặt vật lý (tiếp cận trực tiếp tới thiết bị lưu trữ thông tin) hoặc logic (truy cập thông tin đó từ xa qua môi trường mạng).

Một số cách thức đảm bảo tính bí mật:

- Yêu cầu đối tượng cung cấp định danh (cặp username và password) hay đặc điểm về sinh trắc để xác thực.
- Sử dụng tường lửa (firewall) hoặc danh sách cho phép truy cập (ACL) trên router để ngăn chặn truy cập trái phép.

- Mã hóa thông tin sử dụng các giao thức và thuật toán mạnh như SSL/TLS, AES, v.v..

1.2.3.2 Tính toàn vẹn (integrity):

Đảm bảo tính toàn vẹn của thông tin, nghĩa là thông tin chỉ được phép xóa hoặc sửa bởi những đối tượng được phép và phải đảm bảo rằng thông tin vẫn còn chính xác khi được lưu trữ hay truyền đi. Về điểm này, nhiều người thường hay nghĩ tính toàn vẹn đơn giản chỉ là đảm bảo thông tin không bị thay đổi là chưa đầy đủ. Một giải pháp đảm bảo tính toàn vẹn thông tin có thể bao gồm thêm việc xác thực nguồn gốc của thông tin này (thuộc sở hữu của đối tượng nào) để đảm bảo thông tin đến từ một nguồn đáng tin cậy và ta gọi đó là tính xác nhận của thông tin.

Một số trường hợp tính toàn vẹn của thông tin bị phá vỡ:

- Thay đổi giao diện trang chủ của một trang web.
- Chặn đứng hoặc thay đổi nội dung gói tin được gửi qua mạng.
- Chỉnh sửa trái phép các tập tin được lưu trữ trên máy tính.
- Tín hiệu bị nhiễu hoặc suy hao trên đường truyền làm thông tin bị sai lệch.

1.2.3.3 Tính sẵn sàng (availability):

Đảm bảo độ sẵn sàng của thông tin nghĩa là thông tin có thể được truy xuất bởi những người đã được cấp phép vào bất cứ khi nào họ muốn. Hệ thống có tính sẵn sàng cao hướng đến sự sẵn sàng ở mọi thời điểm, tránh được những rủi ro cả về phần cứng, phần mềm như: sự cố mất điện, hỏng phần cứng, cập nhật, nâng cấp hệ thống…

Ví dụ, nếu một máy chủ chỉ bị ngưng hoạt động hay ngừng cung cấp dịch vụ trong vòng 5 phút trên một năm thì độ sẵn sàng của nó là 99,999%.

Để tăng khả năng phòng chống trước các cuộc tấn công cũng như duy trì độ sẵn sàng của hệ thống ta có thể áp dụng một số kỹ thuật như: RAID, Load Balancing, Clustering, Redudancy, Failover…

1.2.4 Khái niệm kiểm thử:

Theo Myers[11], kiểm thử là tiến trình thực thi một chương trình với mục đích tìm ra lỗi. Theo cách nghĩ của Myers, việc kiểm thử được xem là thành công

khi nó tìm ra lỗi phần mềm và cách sửa lỗi. Ngược lại được xem là không thành công khi nó không thực hiện hết các kiểm tra hoặc không tìm ra một lỗi nào, vì một phần mềm không có lỗi thì có vẻ phi thực tế.

1.2.5 Khái niệm kiểm thử an toàn hệ thống thông tin:

Theo khái niệm về an toàn hệ thống thông tin đã nêu ở mục 1.2.3, kiểm thử an toàn hệ thống thông tin là một loại kiểm thử nhằm mục đích kiểm tra xem hệ thống đó có đảm bảo về: tính bí mật, tính toàn vẹn, tính sẵn sàng của thông tin hay không.

Một hệ thống thông tin được cấu tạo từ nhiều thành phần: con người, phần mềm, máy tính, thông tin,dữ liệu, mạng truyền thông, các chính sách liên quan,… nên việc kiểm thử an toàn hệ thống luôn phải xét đến các yếu tố liên quan đến các thành phần trên tùy theo mục đích đánh giá của người kiểm thử.

1.2.6 Đối tượng tấn công:

Thường được gọi là tin tặc, là những cá nhân hoặc các tổ chức sử dụng kiến thức về hệ thống, mạng và các công cụ phá hoại, xâm nhập (phần mềm lẫn phần cứng) để dò tìm các lỗ hổng bảo mật trên hệ thống từ xa qua mạng, thực hiện các hoạt động xâm nhập và chiếm đoạt tài nguyên trên hệ thống trái phép, bao gồm:

- Hacker: là những kẻ xâm nhập vào hệ thống trái phép bằng cách sử dụng các công cụ phá mật khẩu hoặc khai thác các lỗ hổng bảo mật của các thành phần thuộc hệ thống.
- Masquerader: là những kẻ giả mạo thông tin trên mạng. Có một số hình thức như giả mạo địa chỉ IP, tên miền, định danh người dùng, trang web,…
- Eavesdropping: là những đối tượng nghe trộm thông tin trên mạng, sử dụng các công cụ thu thập gói tin trái phép, sau đó dùng các công cụ phân tích các gói tin để lấy được các thông tin có giá trị [6].

1.2.7 Lỗ hổng bảo mật:

Các lỗ hổng bảo mật là những điểm yếu trên hệ thống hoặc ẩn chứa trong một dịch vụ hệ thống đó cung cấp, dựa vào đó tin tặc có thể xâm nhập trái phép để thực hiện các hành động phá hoại hoặc chiếm đoạt tài nguyên bất hợp pháp [6].

Có nhiều nguyên nhân gây ra những lỗ hổng bảo mật: do lỗi bản thân hệ thống, do phần mềm cung cấp, do người quản trị yếu kém không hiểu sâu sắc các dịch vụ cung cấp, do người sử dụng có ý thức bảo mật kém. Điểm yếu ở yếu tố con người cũng được xem là lỗ hổng bảo mật.

Mức độ ảnh hưởng của các lỗ hổng là khác nhau. Có những lỗ hổng chỉ ảnh hưởng đến chất lượng dịch vụ cung cấp, có những lỗ hổng ảnh hưởng nghiêm trọng đến toàn bộ hệ thống… Các lỗ hổng bảo mật sẽ là các điểm yếu có thể tạo ra sự ngưng trệ của dịch vụ, thêm quyền đối với người sử dụng hoặc cho phép các truy cập không hợp pháp vào hệ thống. Các lỗ hổng có thể nằm ngay các dịch vụ hệ thống cung cấp như Email, ứng dụng web, Ftp,.. Nằm trên các hệ điều hành máy tính, hệ điều hành trong các thiết bị router, modem,…Hoặc nằm trên các phần mềm thường được sử dụng như Microsoft Office, Internet Explorer,…

Có ba loại lỗ hổng bảo mật [6]:

- Lỗ hổng loại C: Cho phép thực hiện hình thức tấn công theo kiểu DoS (Denial of Services – Từ chối dịch vụ) làm ảnh hưởng tới chất lượng dịch vụ, ngưng trệ, gián đoạn hệ thống, nhưng không phá hỏng dữ liệu hoặc đoạt được quyền truy cập hệ thống.
- Lỗ hổng loại B: Lỗ hổng cho phép người sử dụng có thêm các quyền trên hệ thống mà không cần kiểm tra tính hợp lệ dẫn đến lộ, lọt thông tin.
- Lỗ hổng loại A: Cho phép người ngoài hệ thống có thể truy cập bất hợp pháp vào hệ thống, có thể phá hủy toàn bộ hệ thống.

1.2.8 Chính sách bảo mật:

Chính sách bảo mật là tập hợp các quy tắc áp dụng cho những người tham gia quản trị mạng, những người có sử dụng các tài nguyên và các dịch vụ mạng [6]. Hoạch định và áp dụng một chính sách bảo mật tốt sẽ giúp giảm thiểu các rủi ro gây mất an toàn hệ thống. Việc kiểm tra việc thực hiện nghiêm túc chính sách bảo mật cần thực hiện một cách thường xuyên và có kế hoạch rõ ràng.

1.2.9 Những mối đe dọa an toàn hệ thống thường gặp:

Hầu hết các hệ thống CNTT có đặc điểm chung là có nhiều người sử dụng, kết nối vào một hệ thống mạng, phân tán về mặt địa lý nên việc bảo vệ các tài nguyên trong môi trường mạng phức tạp hơn nhiều so với môi trường một máy tính đơn lẻ hoặc một người sử dụng.

Người quản trị hệ thống CNTT phải đảm bảo các thông tin trên hệ thống là tin cậy và sử dụng đúng mục đích, đối tượng đồng thời đảm bảo hệ thống hoạt động ổn định, không bị tấn công bởi những kẻ phá hoại, luôn trong trạng thái hoạt động sẵn sàng phục vụ yêu cầu của người dùng hợp pháp.

Có bốn loại mối đe dọa an toàn hệ thống [6]:

- Chặn bắt (Interception): chỉ thành phần không được phép cũng có thể truy cập đến các dịch vụ hay các dữ liệu, "nghe trộm" thông tin đang được truyền đi.
- Đứt đoạn (Interruption): là mối đe dọa mà làm cho dịch vụ hay dữ liệu bị mất mát, bị hỏng, không thể dùng được nữa…
- Thay đổi (Modification): là hiện tượng thay đổi dữ liệu hay can thiệp vào các dịch vụ làm cho chúng không còn giữ được các đặc tính ban đầu.
- Giả mạo (Fabrication): là hiện tượng thêm vào dữ liệu ban đầu các dữ liệu hay hoạt động đặc biệt mà không thể nhận biết được để tấn công hệ thống.

1.2.10 Các nguyên nhân gây mất an ninh thông tin:

Có vô số cách thức để tấn công một hệ thống thông tin, từ cách tấn công vào các điểm yếu thuộc về phần mềm (lỗi thiết kế, lập trình phần mềm, hệ điều hành), các điểm yếu về công nghệ (các lỗi thuộc về phần cứng, giao thức mạng) cho đến các điểm yếu về con người (ý thức bảo mật kém, không hiểu biết về bảo mật).

1.2.10.1. Lỗi và sự bỏ sót, cố tình bỏ qua:

Nguy cơ này được xếp vào loại nguy hiểm nhất. Khi lập trình, các cảnh báo và lỗi do trình biên dịch đưa ra thường bị bỏ qua và nó có thể dẫn đến những sự việc không đáng có, ví dụ như tràn bộ đệm. Khi người dùng vô tình (hay cố ý) sử dụng các đầu vào không hợp lý thì chương trình sẽ xử lý sai, hoặc dẫn đến việc bị khai thác, đổ vỡ. Lập trình viên phải luôn luôn cập nhật thông tin, các lỗi bị khai thác, cách phòng chống, sử dụng phương thức lập trình an toàn.

Một cách tốt nhất để phòng tránh là sử dụng chính sách "lease privilege" (có nghĩa là ít quyền hạn nhất có thể). Người dùng sẽ chỉ được xử lý, truy cập đến một số vùng thông tin nhất định.

1.2.10.2. Lừa đảo và lấy cắp thông tin:

Trong thời đại hiện nay, thông tin là món hàng quý giá và có thể bị đánh cắp dễ dàng. Việc đánh cắp thông tin có thể xảy ra ở mọi nơi, do nhiều đối tượng thực hiện. Việc lấy cắp có thể được thực hiện dưới nhiều hình thức: lấy cắp văn bản in hay lấy cắp thông tin số, cung cấp thông tin nội bộ cho bên ngoài.

Cách tốt nhất để phòng tránh nguy cơ này là phải có những chính sách bảo mật được thiết kế tốt. Những chính sách có thể giúp người quản trị an toàn hệ thống thông tin thu thập thông tin, từ đó điều tra và đưa ra những kết luận chính xác, nhanh chóng. Khi đã có một chính sách tốt, người quản trị có thể sử dụng các kỹ thuật điều tra dấu vết tấn công (forensics) để truy vết các hành động tấn công.

1.2.10.3. Tin tặc:

Có rất nhiều cách tin tặc tấn công hệ thống. Mỗi kẻ tấn công đều có những thủ thuật, công cụ, kiến thức, hiểu biết về hệ thống. Nhìn chung có các bước tấn công hệ thống như sau: trước tiên, tin tặc thu thập thông tin về hệ thống, nhiều nhất có thể. Những thông tin đó có thể là: tên ứng dụng, phiên bản ứng dụng, hệ điều hành, email quản trị… Bước tiếp theo là quét hệ thống để tìm lỗ hổng bảo mật. Từ đó, họ sẽ lợi dụng các lỗ hổng bảo mật tìm được, hoặc sử dụng các tài khoản mặc định nhằm chiếm quyền truy cập vào ứng dụng. Khi đã thành công, tin tặc sẽ cài đặt các phần mềm, mã độc để có thể xâm nhập vào hệ thống trong các lần sau. Bước cuối cùng là xóa vết tấn công.

Để phòng tránh nguy cơ này, cần sử dụng các phần mềm phát hiện truy cập trái phép, rà soát hệ thống thường xuyên xem có phần mềm lạ không, cấu hình tường lửa hợp lý, chính sách truy cập của từng nhóm người dùng,…

1.2.10.4. Lây lan mã độc:

Có rất nhiều loại mã độc có thể kể đến như: virus, sâu máy tính, trojan horse, logic bomb… Khi đã xâm nhập vào máy nạn nhân, mã độc có thể: mở cổng sau

(back door) để kẻ tấn công có thể truy cập và làm mọi việc trên máy nạn nhân; ghi lại thông tin sử dụng máy tính (thao tác bàn phím, thông tin đăng nhập…).

Cài mã độc vào máy tính có thể bằng nhiều cách: lỗ hổng phần mềm; hệ thống đã bị hacker điều khiển; sử dụng phần mềm không có giấy phép sử dụng.

Cách tốt nhất để tránh nguy cơ này là luôn cập nhật phần mềm, hệ điều hành và phần mềm an ninh mạng, diệt virus.

1.2.10.5. Tấn công từ chối dịch vụ:

Nếu một tin tặc không thể cướp quyền truy cập vào một hệ thống, họ sẽ tìm cách tấn công từ chối dịch vụ (làm hệ thống không thể phục vụ người dùng được trong một khoảng thời gian, bằng cách truy cập đến hệ thống liên tục, số lượng lớn, có tổ chức). Việc tấn công từ chối dịch vụ khiến nạn nhân ngưng trệ các hoạt động dịch vụ, nó làm hệ thống mất tính sẵn sàng.

Có 2 kiểu tấn công từ chối dịch vụ:

- DoS (Denny of Service – tấn công từ chối dịch vụ): tấn công này có thể xảy ra với cả ứng dụng trực tuyến và ứng dụng offline. Với ứng dụng trực tuyến, tin tặc sử dụng các công cụ tấn công (tấn công Syn floods, Fin floods, Smurfs, Fraggles) trên một máy tính để tấn công vào hệ thống, khiến nó không thể xử lý được yêu cầu, hoặc làm nghẽn băng thông khiến người dùng khác khó mà truy cập được. Với ứng dụng offline, tin tặc tạo ra những dữ liệu cực lớn, hoặc các dữ liệu xấu (làm cho quá trình xử lý của ứng dụng bị ngưng trệ, treo)
- DDoS (Distributed Denny of Service – tấn công từ chối dịch vụ phân tán): một hình thức cao cấp của DoS, các nguồn tấn công được điều khiển bởi một (hoặc nhiều) máy chủ của tin tặc (gọi là máy chủ điều khiển) cùng tấn công vào hệ thống. Loại tấn công này khó phát hiện ra hơn, giúp tin tặc ẩn mình tốt hơn.

Để chống lại nguy cơ này, hệ thống cần có nhiều máy chủ phục vụ, máy chủ phân tải, cơ chế phát hiện tấn công từ chối dịch vụ hiệu quả.

1.2.10.6. Phương pháp phi kỹ thuật (Social engineering):

Thuật ngữ này khá phổ biến trong công nghệ thông tin. Đây là một kỹ thuật khai thác nhằm vào điểm yếu con người. Con người trực tiếp quản lý phần mềm, hệ

thống, nên họ nắm nhiều thông tin quan trọng. Người tấn công tìm hiểu thói quen, thông tin của nạn nhân để đánh lừa khai thác thông tin cần thiết từ nạn nhân.

Kỹ thuật này ngày càng có tác dụng và có độ chính xác tương đối cao. Điển hình cho hình thức này là tin tặc nổi tiếng: Kevin Mitnick. Trong một lần, anh chỉ cần vài thông tin quan trọng của tổng thống Mỹ, đã gọi điện cho thư ký của ông và lấy được toàn bộ thông tin về thẻ tín dụng của tổng thống [7].

1.2.11 Các phương thức đảm bảo an toàn thông tin trong hệ thống:

Trong hệ thống thông tin, thông tin là thành phần cốt lõi, nó liên quan hoặc hiện diện trong tất cả các thành phần khác của hệ thống, nên vấn đề đảm bảo an toàn thông tin luôn được chú ý. Có bốn loại phương thức đảm bảo an toàn thông tin trong hệ thống:

- Mã hóa (Cryptography): là việc thực hiện chuyển đổi dữ liệu theo một quy tắc nào đó thành dạng mới mà kẻ tấn công không nhận biết được.
- Xác thực (Authentication): là các thao tác để nhận dạng người dùng.
- Ủy quyền (Authorization): chính là việc phân định quyền hạn cho mỗi thành phần đã đăng nhập thành công vào hệ thống. Quyền hạn này là các quyền sử dụng dịch vụ, truy cập dữ liệu…
- Kiểm tra (Auditing): là các phương pháp để xác định được người dùng đã truy cập đến dữ liệu nào và bằng cách nào.

1.3 Tóm tắt nội dung chương:

Trong chương này đã giới thiệu về tình hình an toàn thông tin tại Việt Nam trong những năm gần đây. Đưa ra các khái niệm về hệ thống thông tin, an toàn hệ thống thông tin, kiểm thử an toàn hệ thống thông tin, các khái niệm cơ bản trong lĩnh vực an toàn thông tin. Ngoài ra, còn nêu lên những mối đe dọa đến an toàn, các nguyên nhân gây mất an toàn và phương thức đảm bảo an toàn hệ thống thông tin . Thông qua chương này, người đọc đã có một cái nhìn khái quát về lĩnh vực an toàn thông tin.

Chương 2. MỘT SỐ TIÊU CHUẨN KIỂM THỬ AN TOÀN HỆ THỐNG THÔNG TIN

Khi tiến hành kiểm thử an toàn hệ thống, cần phải xác định phương pháp kiểm thử thực hiện theo tiêu chuẩn nào. Hiện nay có nhiều tiêu chuẩn về phương pháp kiểm thử, một số chuyên về đánh giá ứng dụng web như OWASP, WASC-TC, một số tập trung vào hệ thống mạng và máy tính nối mạng như: OSSTMM, ISSAF. Riêng tài liệu hướng dẫn kiểm tra và đánh giá an toàn thông tin NIST SP 800-115 có thể áp dụng tùy biến cho các hệ thống thông tin. Chương này sẽ giới thiệu cả năm tiêu chuẩn OWASP, WASC-TC, OSSTMM, ISSAF, NIST SP 800-115.

2.1 Dự án nguồn mở đánh giá an toàn ứng dụng web (OWASP):

Đây là dự án được phát triển bởi cộng đồng mở nhằm nâng cao nhận thức về an toàn ứng dụng web trong các tổ chức. OWASP có kho dữ liệu bao gồm hơn 500.000 lỗ hổng bảo mật, ngoài ra còn cung cấp nhiều tài liệu hướng dẫn về các lĩnh vực khác nhau trong việc bảo mật ứng dụng:

- Các vấn đề về bảo mật ứng dụng (Application Security Desk Reference): Tài liệu này cung cấp các định nghĩa và mô tả về tất cả các khái niệm quan trọng, các loại lỗi, lỗ hổng, các phương pháp tấn công, phương pháp kiểm tra, các tác động kỹ thuật và tác động kinh doanh trong bảo mật ứng dụng. Đây là tài liệu tham chiếu cho tất cả các tài liệu hướng dẫn khác của OWASP.
- Hướng dẫn phát triển (Developer's Guide): Tài liệu này bao gồm tất cả các yếu tố bảo mật mà người phát triển ứng dụng cần quan tâm. Trong tài liệu cung cấp hàng trăm loại lỗ hổng phần mềm, có thể được sử dụng như một sách hướng dẫn mạnh mẽ về kiểm soát bảo mật.
- Hướng dẫn kiểm tra (Testing Guide): Là tài liệu cung cấp về các quy trình và công cụ kiểm tra bảo mật ứng dụng. Cách sử dụng tài liệu tốt nhất là áp dụng vào việc kiểm tra lỗ hổng bảo mật của một ứng dụng hoàn thiện.
- Hướng dẫn kiểm tra mã nguồn (Code Review Guide): Kiểm tra ứng dụng bằng cách xem mã nguồn sẽ hỗ trợ phòng tránh cho ứng dụng khỏi các tác động

bên cạnh việc kiểm tra từ bên ngoài. Người kiểm tra có thể chủ động lựa chọn cách thức tiếp cận với ứng dụng phù hợp nhất.

Ngoài ra, cộng đồng OWASP cũng giới thiệu tài liệu "OWASP Top 10". Đây là một dự án tập trung vào phân loại 10 rủi ro về an toàn ứng dụng phổ biến nhất, đồng thời cung cấp các hướng dẫn cụ thể về cách thức kiểm tra, xác minh và khắc phục những lỗ hổng bảo mật dễ gặp phải của ứng dụng. OWASP Top 10 chủ yếu tập trung giải quyết vấn đề về các nguy cơ phổ biến hơn là việc bảo mật trên một ứng dụng web hoàn thiện.

Ưu điểm của phương pháp OWASP:

- Khuyến khích các nhà phát triển thực hành mã hóa an toàn bằng cách tích hợp kiểm tra an ninh ở từng giai đoạn phát triển. Điều này sẽ giúp cho các ứng dụng trong quá trình phát triển tránh được các lỗi và an toàn hơn.
- Tài liệu Hướng dẫn Kiểm tra của OWASP cung cấp chi tiết về các kỹ thuật đánh giá, cung cấp một cái nhìn rộng hơn vào nhiều nền tảng công nghệ giúp người kiểm tra lựa chọn cách thức phù hợp nhất để tiến hành kiểm tra.
- Tài liệu OWASP Top 10 cung cấp các hướng dẫn kỹ thuật giúp chống lại các cuộc tấn công, lỗ hổng bảo mật phổ biến nhất và đảm bảo tính bảo mật, tính toàn vẹn và tính sẵn sàng của ứng dụng (tam giác C-I-A).

Cộng đồng OWASP cũng đã phát triển một số công cụ bảo mật và hướng dẫn sử dụng tập trung vào kiểm tra ứng dụng web một cách tự động như: WebScarab, Wapiti, JBroFuzz và SQLiX.

2.2 Phương pháp kiểm tra an toàn dành cho mạng và hệ thống (OSSTMM):

OSSTMM (www.isecom.org/osstmm) là một tiêu chuẩn quốc tế được công nhận để kiểm tra và phân tích bảo mật và đang được sử dụng bởi nhiều tổ chức. Có những cách thức khác nhau để thực hiện kiểm tra bảo mật theo phương pháp OSSTMM như sau:

- Blind: Kiểm thử mù không yêu cầu phải biết các thông tin về hệ thống mục tiêu trước đó. Tuy nhiên, mục tiêu này đã được thông báo trước khi bắt đầu tiến hành kiểm thử. Tấn công có đạo đức hoặc kiểm thử game là những ví dụ về

kiểm thử mù. Loại kiểm thử này cũng được chấp nhận rộng rãi bởi mục đích đạo đức của nó trong việc thông báo trước mục tiêu.

- Double blind: Trong kiểm thử Double blind, người kiểm tra không cần biết bất kỳ thông tin nào về hệ thống mục tiêu và mục tiêu cũng không được thông báo trước khi tiến hành kiểm tra. Kiểm thử hộp đen và kiểm thử thâm nhập là những ví dụ của kiểm thử cặp mù. Với loại kiểm thử này, người kiểm tra gặp một thách thức lớn trong việc lựa chọn loại công cụ và kỹ thuật tốt nhất để giải quyết được yêu cầu kiểm tra.
- Gray box (hộp xám): Trong kiểm thử hộp xám, người kiểm tra đã biết trước một vài thông tin về hệ thống mục tiêu và mục tiêu được thông báo trước khi kiểm tra được thực hiện. Đánh giá lỗ hổng (vulneralbility) là một trong những ví dụ cơ bản của kiểm thử hộp xám.
- Tandem (song song): Trong kiểm thử song song, người kiểm tra đã có các kiến thức tối thiểu để đánh giá về hệ thống mục tiêu và mục tiêu cũng được thông báo trước khi kiểm tra được thực hiện. Kiểm toán là ví dụ của kiểm thử song song.
- Reversal (đảo ngược): Trong kiểm thử đảo ngược, người kiểm tra biết trước đầy đủ kiến thức về hệ thống mục tiêu và mục tiêu không được biết khi nào kiểm thử được tiến hành.

Ưu điểm của phương pháp OSSTMM:

- OSSTMM làm giảm đáng kể sự xuất hiện của cảnh báo nhầm, bỏ qua nhầm và cung cấp phép đo chính xác đối với bảo mật.
- Phương pháp này thích nghi với nhiều loại kiểm tra bảo mật như kiểm thử thâm nhập, kiểm thử hộp trắng, đánh giá lỗ hổng,…
- Đảm bảo rằng đánh giá được thực hiện triệt để và kết quả được tổng hợp một cách phù hợp, có định lượng và đáng tin cậy.
- Các phương pháp này tuân theo một quá trình bốn bước gồm: bước định nghĩa, bước thông tin, bước pháp lý, và bước tiến hành kiểm thử. Mỗi bước gồm thu thập, đánh giá và xác minh các thông tin liên quan đến môi trường mục tiêu.

- Số liệu của độ đo bảo mật có được bằng cách sử dụng phương pháp RAV (Risk Assessment Values – Giá trị đánh giá rủi ro). RAV tính toán giá trị bảo mật thực tế dựa trên hoạt động an ninh, kiểm soát sự mất mát và những giới hạn. Số điểm RAV thể hiện trạng thái an ninh hiện tại của mục tiêu.
- Các báo cáo được thể hiện dưới định dạng STAR (Security Test Audit Report – Báo cáo kiểm tra bảo mật) để thuận lợi cho việc quản lý cũng như xem xét của đội ngũ kỹ thuật, các giá trị đánh giá rủi ro (RAV) và đầu ra từ mỗi giai đoạn kiểm tra.
- Phương pháp này thường xuyên được cập nhật với các xu hướng mới của kiểm tra bảo mật, quy định và mối quan tâm về đạo đức.
- Các bước của OSSTMM có thể dễ dàng phối hợp với các quy định của ngành công nghiệp, chính sách kinh doanh và pháp luật của chính phủ. Ngoài ra, một chứng nhận kiểm thử cũng có thể hội đủ điều kiện để được công nhận trực tiếp từ ISECOM (Institute for Security and Open Methodologies – Viện An ninh và các phương pháp mở).

2.3 Chuẩn đánh giá an ninh hệ thống thông tin (ISSAF):

ISSAF là một chuẩn phân tích và kiểm tra an toàn hệ thống mã nguồn mở. ISSAF tiến hành các đánh giá theo một thứ tự hợp lý. Mỗi đánh giá này được thực hiện trên các thành phần khác nhau của hệ thống. Bằng cách tiến hành theo một vòng khép kín, ISSAF có thể cung cấp chính xác, đầy đủ, và hiệu quả yêu cầu kiểm định an ninh của tổ chức. ISSAF được phát triển để tập trung vào hai lĩnh vực kiểm tra an toàn hệ thống: kỹ thuật và quản lý. Mặt kỹ thuật thiết lập các quy tắc và thủ tục cốt lõi để tạo ra một quá trình đánh giá đầy đủ về an toàn hệ thống, trong khi mặt quản lý hoàn thành các bước của quá trình quản lý và những công việc nên được thực hiện trong giai đoạn kiểm tra. Phương pháp đánh giá ISSAF bao gồm các giai đoạn: lập kế hoạch, đánh giá, xử lý, cấp phép và bảo trì. Mỗi giai đoạn được tiến hành hiệu quả và linh hoạt tùy theo điều kiện cụ thể. Kết quả cuối cùng là sự kết hợp của các hoạt động nghiệp vụ, các giải pháp làm tăng an toàn hệ thống và một danh sách đầy đủ các lỗ hổng có thể tồn tại trong môi trường được kiểm tra.

ISSAF bao gồm một tập hợp các quy trình, kỹ thuật đánh giá khác nhau và thường xuyên được cập nhật. Người kiểm tra có thể thêm vào các công cụ, các phương pháp, thủ tục,… để bổ sung cho quy trình đánh giá an toàn hệ thống. Cũng có thể kết hợp với phương pháp OSSTMM hoặc bất kỳ phương pháp kiểm tra nào khác nhằm phát huy ưu điểm của mỗi phương pháp.

Ưu điểm của phương pháp ISSAF:

- Là phương pháp hữu ích trong việc đảm bảo an toàn cho hệ thống bằng cách thực hiện các kiểm tra lỗ hổng hệ thống.
- Chỉ ra những thành phần quan trọng trong đánh giá an toàn thông tin như: đánh giá rủi ro, quản lý kinh doanh, tổ chức đánh giá, tiến trình quản lý, phát triển chính sách bảo mật và các thực hành hữu ích.
- Toàn bộ tiến trình đánh giá ISSAF bao gồm các hoạt động quản lý, đánh giá bảo mật vật lý, phương pháp thử nghiệm thâm nhập, quản lý sự cố, quản lý thay đổi, quản lý kinh doanh liên tục, nhận thức về bảo mật, tuân thủ pháp luật và quy định.
- Phương pháp kiển thử xâm nhập ISSAF kiểm tra cả thành phần mạng, hệ thống hoặc ứng dụng. Phương pháp này tập trung vào những công nghệ cụ thể như thiết bị định tuyến (router), chuyển mạch (switch), tường lửa, hệ thống phát hiện và phòng thủ xâm nhập, mạng riêng ảo, máy chủ ứng dụng web, CSDL,…
- Thu hẹp khoảng cách giữa kỹ thuật và quản lý kiểm tra bảo mật bằng cách thực hiện các tác động cần thiết ở cả hai lĩnh vực.
- Giúp người quản lý hiểu về những rủi ro bảo mật và các lỗ hổng có thể ảnh hưởng đến hoạt động của tổ chức.

2.4 Tiêu chuẩn phân loại nguy cơ trong bảo mật ứng dụng web (WASC-TC):

WASC-TC là một tiêu chuẩn mở để đánh giá sự an toàn của các ứng dụng web. Tương tự như tiêu chuẩn OWASP, WASC-TC cũng được phân loại thành các phương pháp tấn công và lỗ hổng bảo mật, nhưng đi vào phân tích một cách sâu sắc hơn. Các tiêu chuẩn tổng thể được trình bày trong ba quan điểm khác nhau để giúp

các nhà phát triển và người kiểm tra bảo mật có được những hiểu biết cần thiết về mối đe dọa bảo mật ứng dụng web.

- Quan điểm liệt kê (Enumeration): Quan điểm này cung cấp cái nhìn cơ bản về tấn công ứng dụng web và lỗ hổng bảo mật. Mỗi loại tấn công và lỗ hổng bảo mật được trình bày với định nghĩa ngắn gọn, phân loại và nhiều ví dụ khác nhau. Có tổng cộng 49 loại tấn công và lỗ hổng bảo mật đối chiếu với số thứ tự trong WASC-ID (1-49).
- Quan điểm phát triển (Development): Quan điểm này giúp nhà phát triển có được cái nhìn toàn diện về các loại tấn công và lỗ hổng bảo mật có thể xảy ra trong các giai đoạn phát triển: thiết kế, thực hiện hoặc triển khai. Các lỗ hổng thiết kế xuất hiện khi ứng dụng không thực hiện các yêu cầu bảo mật ở giai đoạn thu thập ban đầu. Các lỗ hổng thực hiện xảy ra do nguyên tắc mã hóa và thực hành mã hóa không an toàn. Và các lỗ hổng triển khai là kết quả của việc cấu hình sai ứng dụng, máy chủ web và các hệ thống bên ngoài khác. Như vậy, dưới quan điểm phát triển, kết hợp các yếu tố này trong một vòng đời phát triển liên tục là thực hành tốt nhất để đảm bảo an toàn cho ứng dụng.
- Quan điểm Tham chiếu chéo (Taxonomy Cross Reference): Đề cập đến quan điểm tham khảo chéo nhiều tiêu chuẩn bảo mật ứng dụng web để giúp người kiểm tra và các nhà phát triển trình bày các thuật ngữ theo một chuẩn phù hợp. Tuy nhiên, mỗi tiêu chuẩn vẫn có những tiêu chí riêng để đánh giá ứng dụng từ các góc độ khác nhau và các biện pháp đo lường rủi ro riêng. Các phương pháp tấn công và lỗ hổng bảo mật trong WASC-TC được trình bày tham chiếu với OWASP Top 10, CWE (Mitre’s Common Weakness Enumeration), CAPEC (Mitre’s Common Attack Pattern Enumeration and Classification) và SANS-CWE Top 25 List.

Ưu điểm của phương pháp WASC-TC:

- Cung cấp kiến thức chuyên sâu để đánh giá môi trường ứng dụng web nhằm chống lại các cuộc tấn công và lỗ hổng bảo mật phổ biến nhất.

- Các loại tấn công và lỗ hổng bảo mật được trình bày bởi WASC-TC có thể được sử dụng để kiểm tra mọi nền tảng ứng dụng web bằng cách kết hợp với các công cụ trong bộ công cụ Kali Linux.
- Quan điểm tham chiếu chéo giúp WASC-TC tham khảo và tương thích với những chuẩn bảo mật ứng dụng khác.
- WASC-TC đã được chấp nhận ở mức công nghiệp và tính hội nhập này được thể hiện trong các mã nguồn mở và các giải pháp thương mại, chủ yếu là đánh giá lỗ hổng bảo mật và các sản phẩm quản lý.

2.5 Hướng dẫn kiểm tra và đánh giá an toàn thông tin (NIST SP 800-115):

NIST SP 800-115 là một tài liệu hướng dẫn kỹ thuật các phương pháp kỹ thuật kiểm tra và đánh giá an toàn của hệ thống thông tin do Viện Tiêu chuẩn và Kỹ thuật Quốc gia Mỹ (NIST) xây dựng. Theo đó, khuyến cáo quy trình kiểm thử an toàn hệ thống thông tin tối thiểu gồm ba quá trình:

- Lên kế hoạch: thu thập thông tin về hệ thống được đánh giá, các nguy cơ gây mất an toàn hệ thống, xác định các mục tiêu, phạm vi đánh giá cũng như đội ngũ thực hiện.
- Tiến hành kiểm thử: mục tiêu chính là định danh các lỗ hổng bảo mật và xác nhận chúng một cách thích hợp.
- Báo cáo kết quả: phân tích các lỗ hổng bảo mật đã được xác định để tìm ra nguyên nhân tạo ra chúng, đề xuất phương án khắc phục cũng như giảm thiểu thiệt hại đến hệ thống.

Tài liệu này đưa ra các kỹ thuật kiểm thử hệ thống phân theo ba nhóm chính như sau:

- Các phương pháp kỹ thuật nhận định mục tiêu (Review techniques).
- Các phương pháp kỹ thuật xác định và phân tích mục tiêu (Target Identification and Analysis Techniques).
- Các phương pháp kỹ thuật xác nhận điểm yếu của mục tiêu (Target Vulnerability Validation Techniques).

Khi xây dựng một quy trình kiểm thử an toàn hệ thống, dựa vào đặc thù của mỗi hệ thống thông tin và mục tiêu đánh giá, người kiểm thử cần lựa chọn kết hợp các phương pháp kỹ thuật với nhau. Một quy trình kiểm thử an toàn hệ thống đúng nghĩa thông thường sử dụng các phương pháp trong cả ba nhóm phương pháp kỹ thuật nêu trên.

Bảng 2.1: Các phương pháp kỹ thuật dùng kiểm thử an toàn hệ thống

STT	Tên phương pháp kỹ thuật	Phân nhóm
1	Xem xét tài liệu	Các phương pháp nhận định mục tiêu
2	Xem xét tập tin log	
3	Xem xét tập luật qui định luồng vận chuyển thông tin trên hệ thống mạng	
4	Xem xét cấu hình hệ thống	
5	Thăm dò mạng	
6	Kiểm tra toàn vẹn tập tin	
7	Khám phá hệ thống mạng	Các phương pháp xác định và phân tích mục tiêu
8	Xác định cổng và các dịch vụ	
9	Quét lỗ hổng bảo mật	
10	Kiểm tra hệ thống mạng không dây	
11	Bẻ khóa mật khẩu	Các phương pháp xác nhận điểm yếu của mục tiêu
12	Xâm nhập thử vào hệ thống bằng phương pháp kỹ thuật	
13	Xâm nhập thử vào hệ thống bằng các phương pháp phi kỹ thuật	

2.5.1 Các phương pháp kỹ thuật nhận định mục tiêu:

Các kỹ thuật nhận định mục tiêu thường dùng ở giai đoạn đầu trong quá trình kiểm thử. Mục đích là để kiểm tra thụ động các hệ thống nhằm nhận biết hệ thống và tìm ra các điểm yếu về an toàn hệ thống đó [15]. Hầu hết các kỹ thuật này là thụ động nên chúng ít gây nguy cơ cho hệ thống. Các kỹ thuật trong nhóm này gồm có:

2.5.1.1 Xem xét tài liệu:

Tài liệu ở đây là những chính sách và thủ tục an ninh hiện hành. Nó cung cấp tình hình an toàn hệ thống ở mức độ cơ bản nhất, tuy nhiên chúng thường bị bỏ

qua khi thực hiện kiểm thử. Sự xem xét này có thể giúp tìm ra những chi tiết không khớp và điểm yếu có thể gây ra mất an toàn hệ thống. Những điểm yếu thông thường khi xem xét tài liệu có thể là những thủ tục an ninh cho hệ điều hành hoặc những giao thức hoặc dịch vụ không còn dùng. Chú ý là xem xét tài liệu không đảm bảo những kiểm soát về an toàn hệ thống được thực hiện đúng, nó chỉ giúp hỗ trợ hạ tầng đang có. Đồng thời những kết quả từ hoạt động này có thể được dùng cho giai đoạn sau.

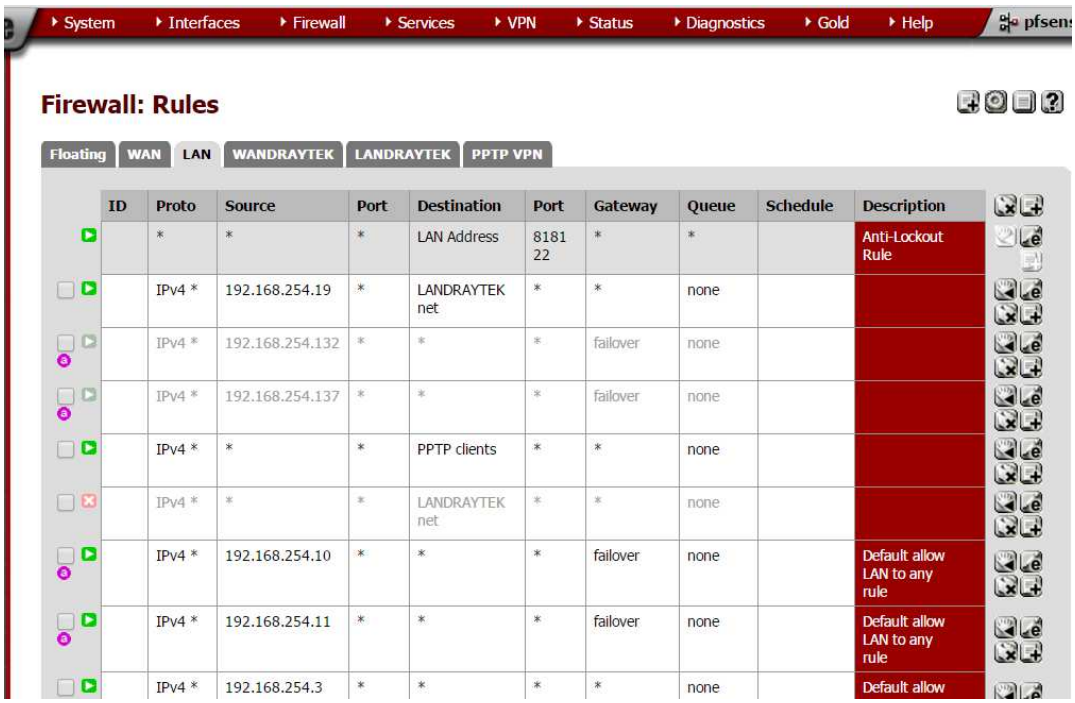
2.5.1.2 Xem xét tập tin log:

Các tập tin log trên hệ thống bao gồm firewall log, IDS log, server log hoặc bất kỳ một tập tin nào ghi lại quá trình hoạt động trên hệ thống. Xem xét tập tin log nhằm xác định những kiểm soát về an toàn hệ thống có ghi lại thông tin một cách chính xác và tổ chức có tuân thủ chính sách quản lý log hay không. Ví dụ, nếu chính sách yêu cầu tất cả những lần đăng nhập vào những máy chủ quan trọng phải được ghi lại, việc xem xét tập tin log sẽ cho phép xác định những thông tin này có được ghi lại đúng như yêu cầu hay không. Ngoài ra, cũng có thể phát hiện những vấn đề như cấu hình sai, truy cập và xâm nhập không phép. Hầu hết dữ liệu log phát sinh rất lớn nên xem xét log bằng phương pháp thủ công sẽ rất tốn thời gian. Việc sử dụng những công cụ tự động sẽ giảm khá nhiều thời gian xem xét và tạo báo cáo tổng hợp. Các công cụ đó cho phép lọc bớt hoặc tìm kiếm các hoạt động cần thiết nhằm tăng hiệu quả xem xét log. Những công cụ phổ biến như Event viewer, Cisco Security Monitoring, Analysis and Response System (MARS), Consul InSight, Netcool/NeuSecure, NetIQ Security Manager, …

2.5.1.3 Xem xét tập luật qui định luồng vận chuyển thông tin trên mạng:

Trên hệ thống tường lửa, bộ định tuyến thông thường sẽ có tập hợp các luật hoặc những dấu hiệu được lưu trữ nhằm mang ra so sánh hoặc đối chiếu với luồng thông tin qua mạng hoặc thiết bị mạng đó để xác định hành động thích hợp gọi là tập luật qui định luồng vận chuyển thông tin trên mạng, ví dụ như một gói tin được phép đi tiếp hoặc bị chặn lại, một cảnh báo phát sinh. Việc xem xét này cho phép phát hiện những điểm yếu và lỗ hổng trên hệ thống, đồng thời cũng cho phép kiểm

soát những ảnh hưởng tiêu cực đến hiệu năng hoạt động của mạng gây ra bởi các luật.



Hình 2.1: Một tập luật trên tường lửa sử dụng PfSense

2.5.1.4 Xem xét cấu hình hệ thống:

Thông qua việc xem xét cấu hình hệ thống có thể bộc lộ những khiếm khuyết về an toàn hệ thống. Ví dụ như hệ thống không được cấu hình hoặc nâng cấp phù hợp với chính sách an ninh; những dịch vụ hoặc ứng dụng không cần thiết; tài khoản người dùng không phù hợp…. Khi xem xét cấu hình hệ thống cần quyền quản trị để xem xét.

Nếu hệ thống không được cấu hình theo đúng chính sách an ninh thì có thể thực hiện những việc sau: Gỡ bỏ các dịch vụ có lỗ hổng nếu chúng không cần thiết; Cấu hình lại hệ thống; Thay đổi luật của tường lửa để hạn chế quyền truy cập vào các hệ thống hoặc dịch vụ có lỗ hổng.

2.5.1.5 Thăm dò mạng:

Là kỹ thuật thụ động để giám sát các luồng thông tin trên mạng, nó bao gồm bắt, nhận biết và giải mã giao thức, khảo sát nội dung gói tin (đọc header và

payload) để phát hiện thông tin cần quan tâm. Các công cụ dùng cho kỹ thuật thăm dò mạng gọi là sniffer. Thăm dò mạng đôi lúc cũng được sử dụng như công cụ phát hiện và phân tích mục tiêu. Nhìn chung, thăm dò mạng ít ảnh hưởng đến hệ thống. Nhược điểm của thăm dò mạng là không đọc được gói tin mã hóa, hoạt động giới hạn trong một phân đoạn mạng, yêu cầu người kiểm thử có trình độ cao.

2.5.1.6 Kiểm tra toàn vẹn tập tin:

Kiểm tra sự toàn vẹn của một tập tin là tạo và lưu trữ một biến phát hiện lỗi checksum cho mọi tập tin được bảo vệ và thiết lập một cơ sở dữ liệu của biến checksum đó. Phương pháp này cung cấp một công cụ cho quản trị hệ thống nhận ra sự thay đổi của các tập tin, đặc biệt là sự thay đổi trái phép. Các biến checksum lưu trữ nên được tính toán lại thường xuyên để so sánh với giá trị hiện tại được lưu trữ, sau đó xác định có sự thay đổi tập tin hay không. Chức năng kiểm tra sự toàn vẹn của tập tin thường được tích hợp vào trong hệ thống phát hiện xâm nhập trên máy chủ thương mại.

Phương pháp kiểm tra sự toàn vẹn là một công cụ hữu ích mà không đòi hỏi sự can thiệp của con người ở mức độ cao, nhưng nó cần phải thực hiện một cách thận trọng để đảm bảo tính hiệu quả. Một số công cụ kiểm tra sự toàn vẹn của dữ liệu như Aide, LANGuard, Tripwire,....

2.5.2 Các phương pháp kỹ thuật xác định và phân tích:

Các kỹ thuật xác định và phân tích mục tiêu tập trung vào việc nhận biết những thiết bị, cổng và dịch vụ liên quan đến thiết bị đó, sau đó phân tích những lỗ hổng bảo mật tiềm tàng . Các kỹ thuật này có thể là công nghệ hoặc phi công nghệ [15]. Một ví dụ của kỹ thuật phi công nghệ là quan sát bằng mắt thường để phát hiện thiết bị trên một mạng. Các kỹ thuật có thể được phân thành các loại sau đây:

2.5.2.1 Khám phá hệ thống mạng:

Là kỹ thuật tìm hiểu mạng và các thành phần hoạt động trên mạng. Có hai kỹ thuật chính dùng để khám phá hệ thống mạng là thụ động (passive) và chủ động (active). Kỹ thuật thụ động thường dùng một thiết bị thăm dò để giám sát luồng thông tin, qua đó cung cấp những thông tin như địa chỉ IP đang dùng; cổng/dịch vụ

đang mở trên đó; Hệ điều hành nào đang được máy chủ sử dụng; quan hệ giữa các máy chủ. Kỹ thuật chủ động cần gửi một vài gói tin đến mục tiêu (ví dụ ICMP ping, SYN, FIN, NULL, …) và thu thập các thông tin phản hồi từ mục tiêu để phân tích. Ưu điểm của kỹ thuật thụ động là không ảnh hưởng đến mạng do không cần gửi những gói tin thăm dò, đây cũng chính là nhược điểm của kỹ thuật chủ động do kỹ thuật này hay phát sinh nhiễu trên mạng. Nhược điểm của kỹ thuật thụ động là tốn nhiều thời gian và thiết bị không truyền thông tin trên mạng thì không bị phát hiện , trong khi đó kỹ thuật chủ động tốn ít thời gian hơn, có thể dùng trên nhiều mạng khác nhau.

Khám phá hệ thống mạng cũng có thể phát hiện những thiết bị, dịch vụ giả mạo trên mạng như các dịch vụ DHCP hay DNS giả mạo.

2.5.2.2 Xác định cổng và dịch vụ:

Kỹ thuật này đào sâu tìm hiểu những thông tin có được từ khảo sát mạng để nhận biết các cổng và dịch vụ đang hoạt động trên những máy chủ và ứng dụng nào đang vận hành dịch vụ đó, ví dụ cổng 80 dùng cho dịch vụ web thì IIS hay Apache được cài đặt. Thông tin này rất hữu ích khi xác định mục tiêu trong kiểm thử an toàn hệ thống.

Tất cả các công cụ cơ bản đều có khả năng nhận biết máy chủ và cổng (dịch vụ) mở trên đó, ngoài ra một vài công cụ còn cung cấp thêm một số thông tin về hệ điều hành thông qua thủ thuật gọi là OS fingerprinting; ứng dụng chạy trên cổng thông qua thủ thuật nhận biết dịch vụ: lưu file danh sách cổng và dịch vụ tương ứng, lưu những đặc trưng hành xử của từng ứng dụng để so sánh với hành xử của ứng dụng trên cổng đang quét; phiên bản của ứng dụng thông qua thủ thuật version scanning, banner grabbing: lấy thông tin banner truyền bởi cổng từ xa khi khởi tạo kết nối.

Những thông tin lấy được từ các kỹ thuật trên không chắc chắn bởi vì người quản trị mạng có kinh nghiệm có thể sửa banner những đặc trưng khác của ứng dụng để che đậy ứng dụng thực sự. Nên lựa chọn công cụ quét tùy tình huống cụ thể, kết quả rất khác nhau phụ thuộc công cụ quét được dùng.

2.5.2.3 Quét lỗ hổng bảo mật:

Giống như quá trình xác định cổng và dịch vụ mạng, nhưng giai đoạn này tập trung vào tìm những lỗ hổng bảo mật trên các cổng và dịch vụ tìm được. Quét lỗ hổng có thể phát hiện những phần mềm hết hạn, những lỗi chưa được cập nhật và cả cấu hình sai. Để nhận biết điểm yếu thì thường so sánh thông tin tìm được về hệ điều hành, ứng dụng với thông tin điểm yếu lưu trong CSDL của phần mềm quét.

Nếu trong hệ thống có phần tử là máy tính thì việc quét mã độc cũng là một phương pháp quét lỗ hổng bảo mật. Mọi hệ thống có phần tử là máy tính đều có nguy cơ nhiễm virus, trojan, sâu,… nếu như chúng kết nối với mạng Internet hoặc sử dụng USB. Có hai dạng chương trình diệt mã độc: chương trình được cài đặt trên hạ tầng mạng và chương trình cài đặt trên máy người sử dụng.

Quét lỗ hổng bảo mật để kiểm tra sự tuân thủ chính sách an ninh và chính sách sử dụng dịch vụ. Ngoài ra còn để cung cấp thông tin mục tiêu cho kiểm thử an toàn hệ thống và giúp xử lý các lỗ hổng bảo mật. Các phần mềm quét lỗ hổng có thể quét từ xa hay cục bộ, quét bên trong hay bên ngoài. Quét lỗ hổng giả lập một mẫu tấn công sau đó xem cách hành xử và phản ứng của hệ thống rồi so sánh với đặc trưng hành xử của những hệ thống có điểm yếu được lưu trữ trong CSDL để xác định sự tồn tại của lỗ hổng được gọi là quét lỗ hổng bảo mật dựa vào đặc trưng. Các công cụ quét lỗ hổng bảo mật thường ấn định mức độ rủi ro của lỗ hổng. Tuy nhiên, lỗ hổng bảo mật độc lập thường có mức độ rủi ro thấp nhưng khi tồn tại nhiều lỗ hổng bảo mật thì có thể mức độ rủi ro rất cao. Ngoài ra các phần mềm quét lỗ hổng cũng có tiêu chí đánh giá và xếp loại mức độ rủi ro, tỉ lệ phát hiện sai cũng khác nhau.

2.5.2.4 Kiểm tra hệ thống mạng không dây:

Mạng không dây là môi trường các thiết bị kết nối với nhau mà không cần dây dẫn vật lý. Ngày nay, với sự phát triển của công nghệ không dây và sự tiện lợi khi sử dụng, các thiết bị không dây ngày càng được sử dụng nhiều nên các tổ chức ngày càng coi trọng kiểm thử mạng không dây để bảo vệ môi trường mạng này Phần mềm quét không dây thường cài đặt trên thiết bị có ăng -ten không dây như

máy tính xách tay, thiết bị cầm tay hoặc thiết bị đặc chủng. Các phần mềm này nên có khả năng quét tất cả các kênh, trên các dải tần số lớn.

Ngoài ra, một số phần mềm cũng tích hợp chức năng vẽ bản đồ hay GPS cho phép định vị thiết bị không dây. Quét không dây có thể dùng kỹ thuật quét thụ động và quét chủ động, đồng thời còn có kỹ thuật truy vết thiết bị không dây để định vị thiết bị đó. Quét thiết bị bluetooth cũng là một kỹ thuật quét không dây cần cho các tổ chức có sử dụng thiết bị Bluetooth.

2.5.3 Các phương pháp kỹ thuật xác nhận điểm yếu của mục tiêu:

Sau khi đã tìm được các điểm yếu được phát hiện trong giai đoạn phát hiện và phân tích mục tiêu thì kỹ thuật xác nhận điểm yếu của mục tiêu để chứng minh các điểm yếu đó tồn tại và mối đe dọa xuất hiện là cần thiết. Kỹ thuật này cần người kiểm thử có nhiều kinh nghiệm và phải thực hiện cẩn thận vì nó ảnh hưởng tiềm tàng đến hệ thống mục tiêu nhiều hơn kỹ thuật khác.

2.5.3.1 Bẻ mật khẩu (password cracking):

Khi mật khẩu được tạo ra, một kỹ thuật mã hóa một chiều sẽ được dùng để biến đổi mật khẩu thành một giá trị gọi là giá trị băm, kỹ thuật mã hóa một chiều đó gọi là kỹ thuật băm. Giá trị băm được lưu trữ trong hệ thống. Khi người dùng nhập mật khẩu, kỹ thuật băm được áp dụng cho mật khẩu đó để phát sinh một giá trị băm mới và so sánh với giá trị băm lưu trữ trong hệ thống. Nếu hai giá trị băm giống nhau thì người dùng được xác thực. Bẻ mật khẩu là quá trình khôi phục lại mật khẩu từ giá trị băm lấy được từ nơi lưu trữ trong hệ thống hoặc bắt được khi truyền trên mạng. Do kỹ thuật băm là kỹ thuật một chiều nên để tìm được mật khẩu thường thì có một phần mềm lấy các mật khẩu được đoán trước thực hiện kỹ thuật băm và so sánh với giá trị băm lấy được. Nếu giống nhau thì mật khẩu được tìm ra.

Có các kỹ thuật bẻ mật khẩu:

- Kỹ thuật vét cạn (brute force): mật khẩu có thể là toàn bộ các kết hợp của các ký tự trong bảng chữ cái và có độ dài cho trước. Cách này tuy cần nhiều thời gian để tìm ra mật khẩu nhưng được xem là chắc chắn tìm ra mật khẩu, miễn là có thời gian và năng lực máy tính mạnh.

- Kỹ thuật tấn công từ điển (dictionary attack): Để thu hẹp phạm vi tìm kiếm, mật khẩu đoán trước có thể lấy các từ trong từ điển vì người dùng thường đặt các mật khẩu là từ có nghĩa để dễ nhớ.
- Kỹ thuật lai (hybrid attack): Kết hợp giữa hai kỹ thuật trên, dùng các từ trong từ điển và thêm các nhóm số hay ký tự đặc biệt (ví dụ: password99, password!@,...).
- Kỹ thuật bảng cầu vồng (rainbow table). Kỹ thuật này dựa trên việc tính toán trước các giá trị băm, lưu thành bảng mật khẩu và giá trị băm tính được. Khi dùng chỉ việc so sánh các giá trị băm trong bảng với giá trị băm lấy được. Kỹ thuật này nhanh nhưng chiếm nhiều không gian lưu trữ.

2.5.3.2 Xâm nhập thử vào hệ thống bằng phương pháp kỹ thuật:

Đây là việc kiểm tra an toàn của hệ thống bằng cách phá vỡ các tính năng an toàn của chúng dựa trên các hiểu biết về thiết kế và hoạt động của hệ thống. Mục đích là để xác định các phương pháp tiếp cận hệ thống thông qua các công cụ và kỹ thuật cơ bản của kẻ tấn công. Việc thâm nhập phải được tiến hành sau khi khảo sát hệ thống một cách cẩn thận, thông báo cho toàn hệ thống và lập kế hoạch thâm nhập đầy đủ.

Việc thử nghiệm thâm nhập chính là tạo ra một mô phỏng cuộc tấn công vào hệ thống, nên có thể bị pháp luật hoặc chính sách bảo mật ngăn cấm. Do đó, trước khi thực hiện phải được sự cho phép và chỉ nên thực hiện như sau:

- Thực hiện trên một địa chỉ hoặc một dải địa chỉ cụ thể.
- Không thực hiện trên một số máy tính bị ngăn cấm.
- Dùng một số các kỹ thuật thâm nhập cho phép.
- Xác định rõ thời gian thực hiện việc thâm nhập.
- Xác định khoảng thời gian hữu hạn cho việc thâm nhập
- Xác định rõ địa chỉ IP từ máy sẽ thực hiện thâm nhập để người quản trị có thể phân biệt cuộc tấn công thử nghiệm với các cuộc tấn công thực sự khác.
- Xử lý các thông tin được thu thập bởi đội thử nghiệm thâm nhập.

2.5.3.3 Xâm nhập thử vào hệ thống bằng phương pháp phi kỹ thuật:

Được sử dụng để kiểm tra nhận thức về an toàn hệ thống của thành phần con người trong hệ thống thông tin, qua đó có thể tiết lộ những yếu kém trong hành vi người dùng, chẳng hạn như không tuân theo quy trình tiêu chuẩn. Kỹ thuật này nhằm cố gắng đánh lừa một ai đó để họ tiết lộ thông tin như mật khẩu, thông tin mạng, cấu hình,… mà thông tin này có thể dùng để tấn công hệ thống.

Có nhiều hình thức thực hiện: gọi điện thoại, gửi email, tin nhắn,…Một phương pháp nổi tiếng là phising. Người dùng bị hấp dẫn bởi các thông tin giới thiệu trong email và nhấn vào các đường link dẫn đến các trang web lừa đảo có hình thức giống trang web của tổ chức đáng tin cậy. Khi người dùng đưa thông tin lên trang web đó sẽ được ghi nhận lại bởi người tấn công.

Ví dụ: Người tấn công gửi cho người dùng một email từ ngân hàng yêu cầu xác nhận lại thông tin cá nhân, bao gồm cả mật khẩu. Nếu làm theo yêu cầu đó, người tấn công sẽ ghi nhận được những thông tin quan trọng để có thể chuyển tiền từ tài khoản ngân hàng của người dùng đó.

2.6 Tóm tắt nội dung chương:

Trong chương này đã giới thiệu về năm tiêu chuẩn kiểm thử an toàn hệ thống thông tin nhằm làm cơ sở để xây dựng quy trình kiểm thử an toàn hệ thống thông tin. Trong đó, tài liệu có đi sâu vào tìm hiểu tài liệu hướng dẫn kiểm tra và đánh giá an toàn thông tin (NIST SP 800-115) do Viện Tiêu chuẩn và Kỹ thuật Quốc gia Mỹ (NIST) xây dựng.

Thông qua chương này, người đọc nắm được một số tiêu chuẩn dùng trong kiểm thử an toàn hệ thống thông tin, qua đó có thể so sánh, chọn lọc, kết hợp chúng để áp dụng cho việc kiểm tra trong thực tế.

Chương 3. NGUY CƠ MẤT AN TOÀN HỆ THỐNG TỪ LỖI CỦA ỨNG DỤNG

Trong chương này sẽ tìm hiểu một số loại phương thức tấn công vào các lỗ hổng bảo mật hay gặp, phương thức khai thác chúng và cách phòng chống. Việc tìm hiểu các lỗ hổng bảo mật giúp người kiểm thử có cái nhìn chính xác hơn về an toàn của hệ thống được kiểm thử, giúp đưa ra được các gợi ý vá lỗi đúng đắn và đầy đủ.

3.1 Injection:

Là loại lỗ hổng bảo mật cho phép người tấn công "tiêm" (injection) một đoạn dữ liệu không tin cậy đến ứng dụng như một phần câu lệnh hoặc câu truy vấn. Thông qua đoạn dữ liệu đó, người tấn công có thể tương tác với dữ liệu của ứng dụng một cách bất hợp pháp [16].

Có nhiều loại Injection: Sql injection, OS injection, LDAP injection,.. Phổ biến nhất là SQL injection nên trong nội dung luận văn sẽ tìm hiểu loại này.

3.1.1 SQL Injection là gì:

SQL Injection là một loại lỗ hổng bảo mật cho phép những kẻ tấn công thi hành các câu lệnh truy vấn SQL bất hợp pháp (người phát triển không lường trước được) bằng cách lợi dụng lỗ hổng trong việc kiểm tra dữ liệu nhập từ các ứng dụng web. Hậu quả này rất tai hại vì nó cho phép kẻ tấn công có toàn quyền, hiệu chỉnh… trên CSDL của ứng dụng. Lỗi này thường xảy ra trên các ứng dụng web có dữ liệu được quản lí bằng các hệ quản trị CSDL như SQL Server, Oracle, DB2, Sybase. Nó có thể tồn tại trong chức năng tìm kiếm (search), đăng nhập (login) hoặc các trang hiển thị tin tức, sản phẩm,…bằng cách truyền tham số (ví dụ: http://www.site.com/view.aspx?id=10).

3.1.2 Nguyên lý thực hiện:

Ví dụ: Xét đoạn mã truy vấn SQL sau:

```
SELECT * FROM Users WHERE Username='$username' AND
Password='$password'
```

Đây là một câu truy vấn thường hay được dùng trong các trình ứng dụng nhằm xác thực người dùng. Nếu câu truy vấn trả về một giá trị nói rằng thông tin về

người dùng đang đăng nhập là đúng và được lưu trong cơ sở dữ liệu, thì người dùng được phép đăng nhập vào hệ thống, ngược lại thì không đăng nhập được.

Thay vì nhập đúng tên đăng nhập và mật khẩu, thử nhập vào các ký tự đặc biệt như:

```
$username = 1’ OR ‘1’ = ‘1
$password = 1’ OR ‘1’ = ‘1
```

Khi đó trong trình ứng dụng phần kiểm tra xác thực người dùng sẽ thực thi câu truy vấn:

```
SELECT * FROM Users WHERE Username=’1’ OR ‘1’ = ‘1’ AND Password=’1’ OR ‘1’ = ‘1’
```

Giả sử rằng giá trị của các tham số được gửi tới máy chủ bằng phương thức GET, thì có một câu lệnh khai thác lỗi như sau:

```
http://www.site.com/index.php?username=1’%20or%20’1’%20=%20’1&password=1’%20or%20’1’%20=%20’1
```

Khi đó, truy vấn sẽ trả về một giá trị (hay một loạt các giá trị) vì điều kiện trên luôn luôn đúng (OR 1=1). Trong trường hợp này tin tặc sẽ đăng nhập được vào hệ thống mà không cần biết tên đăng nhập và mật khẩu. Trường hợp này sẽ rất nguy hiểm nếu dòng đầu tiên trong bảng “Users” là tài khoản của người quản trị (admin) vì tin tặc sẽ đăng nhập vào hệ thống bằng tài khoản đầu tiên trong bảng này.

3.1.3 Một số kiểu tấn công SQL Injection:

Thông thường trải qua các bước: xác định ứng dụng có lỗi SQL injection, khai thác lỗi. Phân chia theo sự phản hồi từ máy chủ có hai kiểu tấn công: hiển thị lỗi cho người tấn công thấy và không hiển thị lỗi.

3.1.3.1 Trường hợp máy chủ hiển thị lỗi:

Trong trường hợp không tắt tính năng debug trên máy chủ ứng dụng, các lỗi khi truy vấn máy chủ SQL sai sẽ hiển thị cho người tấn công thấy. Nếu thêm các ký tự dấu nháy đơn (‘), dấu nháy kép (“), phần trăm 27 (%27), 00 phần trăm (00%) vào

cuối dòng địa chỉ URL (ví dụ: http://www.site.com/view.aspx?id=10') để kiểm tra nếu nó hiện ra lỗi kiểu như :

Microsoft SQL Native Client error '80040e14' Unclosed quotation mark after the character string

Có thể xác định trang này bị lỗi SQL Injection. Trong trường hợp này, người tấn công có thể thao tác trực tiếp trên CSDL thông qua các lệnh SELECT, INSERT, UPDATE, DELETE hay gọi thi hành các Procedure có trên CSDL.

3.1.3.2 Trường hợp máy chủ không hiển thị lỗi:

Khi người tấn công không thấy được các thông tin hiện lỗi. Trường hợp này người tấn công sẽ thử sử dụng phương thức tấn công Blind injection. Trong phương thức này, chỉ đoán được kết quả trả về là True hay False thông qua trạng thái trang trả về hoặc thời gian phản hồi của máy chủ.

Ví dụ thêm OR '1=1' (http://www.site.com/view.aspx?id=10 OR '1=1') vào cuối địa chỉ URL vẫn nhận được trang có nội dung không đổi so với trang gốc, hoặc thêm ;waitfor delay '00:10' -- vào cuối địa chỉ URL (http://www.site.com/view.aspx?id=10; waitfor delay '00:10' --) trang hiển thị chậm hơn bình thường 10 giây thì trang này có thể khai thác lỗi kiểu Blind injection.

3.1.4 Phương pháp phòng chống:

- Không hiển thị thông tin lỗi cho người sử dụng thấy.
- Kiểm tra dữ liệu nhập vào: loại bỏ các ký tự nguy hiểm ('," ,--,/…) hoặc các từ khóa là tên các hàm hệ thống trên CSDL, kiểm tra kiểu dữ liệu truyền vào.
- Hạn chế tối đa quyền truy vấn, cấp quyền truy vấn theo nhu cầu sử dụng, ví dụ: chỉ cấp quyền thực thi lệnh SELECT cho người dùng chỉ cần đọc dữ liệu.
- Sử dụng mô hình n-tier cho ứng dụng, các tham số truyền vào các lớp phải thông qua các câu lệnh tham số hóa.
- Thường xuyên kiểm tra, quét lỗi ứng dụng bằng các công cụ kiểm thử an toàn (ví dụ như công cụ được hướng dẫn trong chương 4).

3.2 Lỗi liên quan đến quá trình quản lý xác thực và phiên truy cập:

Người tấn công từ bên ngoài hệ thống xâm nhập trái phép vào tài nguyên nội bộ trong hệ thống dựa vào lỗ hổng bảo mật nằm trên các phân hệ quản lý xác thực người dùng hoặc quản lý session (phiên truy cập) [16]. Có thể liệt kê số loại lỗ hổng đó như sau:

- Mật khẩu, thông tin xác người dùng không được mã hóa khi lưu trên CSDL hoặc truyền đi trên đường truyền giữa máy người sử dụng và máy chủ hệ thống.
- Phân hệ xác thực người dùng có thể bị bỏ qua hoặc chép đè thông tin xác thực thông qua các chức năng của phân hệ không được kiểm soát chặt chẽ (ví dụ: chức năng tạo tài khoản, phục hồi mật khẩu, thay đổi mật khẩu).
- Điểm yếu trên phân hệ quản lý phiên truy cập, qua đó cho phép tấn công kiểu ấn định phiên truy cập (session fixation), chiếm phiên truy cập (session hijacking).

Trong nội dung phần này sẽ tìm hiểu các kiểu tấn công thông qua lỗ hổng bảo mật liên quan đến quản lý phiên truy cập.

3.2.1 Tấn công kiểu ấn định phiên truy cập:

Là kĩ thuật tấn công cho phép người tấn công mạo danh người dùng hợp lệ bằng cách gửi một định danh phiên truy cập (session ID) hợp lệ đến người dùng (bằng cách ngụy trang, lừa đảo người dùng đó), sau khi người dùng đăng nhập vào hệ thống thành công, hacker sẽ dùng lại session ID đó và nghiễm nhiên trở thành người dùng hợp lệ.

3.2.1.1 Nguyên lý thực hiện:

Thông qua ba bước như sau:

- Người tấn công kết nối đến hệ thống nhằm lấy session ID.
- Đánh lừa người dùng hợp lệ (nạn nhân), là người dùng đã thông qua bước xác thực người dùng, truy cập vào hệ thống thông qua URL có sử dụng session ID đã lấy được ở bước trên.

- Khi nạn nhân đã truy cập vào hệ thống với session ID của người tấn công đưa, người tấn công có thể truy cập hợp lệ vào hệ thống với session ID đó và có quyền được cấp trên hệ thống giống như nạn nhân.

3.2.1.2 Các cách gắn session ID vào trình duyệt của nạn nhân:

Kiểu tấn công này chỉ thành công khi nạn nhân truy cập vào hệ thống với session ID do người tấn công đã tạo trước. Tùy theo hệ thống, có ba cách để người tấn công gắn session ID vào trình duyệt của nạn nhân:

- Gắn session ID trên URL: Người tấn công phải đánh lừa nạn nhân truy cập đến hệ thống thông qua đường dẫn kiểu như : http://www.website.com/login.php?sessionid=12 .
- Gắn session ID trong biến ẩn trên form: trong trường hợp hệ thống lưu session ID trong biến ẩn nằm trên form, người tấn công phải đánh lừa nạn nhân vào URL của trang web hệ thống người tấn công đã vào, hoặc trang có nội dung giống trang web đó với biến ẩn trên form đã lưu session ID định sẵn.
- Gắn session ID trong cookie: trường hợp hệ thống lưu session ID trong cookie, người tấn công đánh lừa nạn nhân truy cập đến hệ thống qua đường dẫn kiểu như: http://www.website.com/<script>document.cookie=”sessionid=ab”; </script>

3.2.2 Tấn công kiểu chiếm phiên truy cập:

Khi người dùng hợp lệ đăng nhập vào hệ thống thành công sẽ được cấp một session ID. Người tấn công sẽ tìm cách đánh cắp session ID này và sử dụng nó, khi đó họ có quyền như nạn nhân trên hệ thống.

Có nhiều cách tiến hành: brute force để dự đoán session ID, nghe lén thông tin trên mạng, sử dụng kiểu tấn công XSS,…

Nhìn chung, kiểu tấn công này tương tự như tấn công kiểu ấn định phiên truy cập, điểm khác biệt là người tấn công chiếm lấy session ID của nạn nhân, còn trong kiểu ấn định phiên truy cập thì người tấn công ép nạn nhân xác thực với session ID được định trước.

3.2.3 Phương pháp phòng chống:

- Mã hóa thông tin người dùng lưu trên CSDL, mã hóa session ID khi truyền đi giữa máy tính của người sử dụng và máy chủ hệ thống, dùng giao thức SSL.
- Kiểm tra chặt chẽ mã nguồn, quy trình của phân hệ quản lý xác thực người dùng, nhất là các chức năng: tạo tài khoản; phục hồi, thay đổi mật khẩu.
- Phòng chống các phương thức tấn công liên quan đến phiên truy cập bằng cách: dùng các giao thức an toàn (ví dụ: https thay vì http) cho ứng dụng, giới hạn thời gian hiệu lực của phiên truy cập, thay đổi session ID khi người dùng đăng nhập thành công. Đối với người dùng cần sử dụng VPN khi dùng mạng nơi công cộng.

3.3 Thực thi đoạn mã trên trình duyệt (XSS):

Lỗ hổng bảo mật XSS là một trong những lỗ hổng phổ biến nhất hiện nay, trong nhiều năm liền được liệt vào danh sách những lỗ hổng bảo mật nguy hiểm nhất với ứng dụng web. Theo dự án "10 mối nguy cơ về bảo mật nguy cấp nhất của ứng dụng Web" (OWASP Top Ten Project) [16], từ năm 2007 đến nay XSS luôn được xếp vào trong ba thứ hạng đầu.

3.3.1 Nguyên lý thực hiện:

Ứng dụng web bị lỗi XSS sẽ cho phép người tấn công thông qua trang web đó gửi đoạn mã độc đến và thực thi trên trình duyệt web của máy tính nạn nhân, nhằm chiếm phiên đăng nhập của người dùng, thay đổi nội dung trang web, chuyển hướng đến trang web khác, chặn và ghi lại dữ liệu người dùng nhập vào…[16]

3.3.2 Một số kiểu tấn công XSS:

Phân loại theo cách thức gửi đoạn mã độc đến và thực thi trên trình duyệt máy tính nạn nhân, chia ra làm hai kiểu tấn công XSS:

- Đoạn mã độc không lưu trữ trên trang web: Đoạn mã độc cần thực thi được nhúng trong URL, ngụy trang và gửi đến nạn nhân.
- Đoạn mã độc được lưu trữ trên trang web: Chèn đoạn mã độc lên trang web đang tồn tại lỗi XSS, bất cứ người nào truy cập vào trang này thì đoạn mã độc sẽ thi hành.

3.3.2.1 Đoạn mã độc không lưu trữ trên trang web:

Người tấn công gửi URL đã được nhúng đoạn mã độc của trang web đang tồn tại lỗi XSS đến máy tính nạn nhân, khi nạn nhân truy cập vào theo URL đó thì đoạn mã độc mới thi hành. Kiểu này còn gọi là kiểu phản xạ (Reflected XSS), bởi vì ngay khi nạn nhân truy cập vào URL đã nhúng đoạn mã độc thì ngay lập tức người tấn công nhận ngay kết quả thực thi đoạn mã đó [8].

Ví dụ kiểu tấn công XSS kiểu không lưu đoạn mã trên trang web:

- Bước 1: Nạn nhân truy cập vào URL đã nhúng đoạn mã do người tấn công gửi đến.

```
http://www.website.com/search.php?giatri= <script language=”javascript” source=”http://hacker.com/doanma.js”> </script>
```

- Bước 2: Máy chủ trang website.com sẽ trả về kết quả bao gồm việc thực thi đoạn mã độc lưu trong tập tin doanma.js chứa trên trang hacker.com của người tấn công.

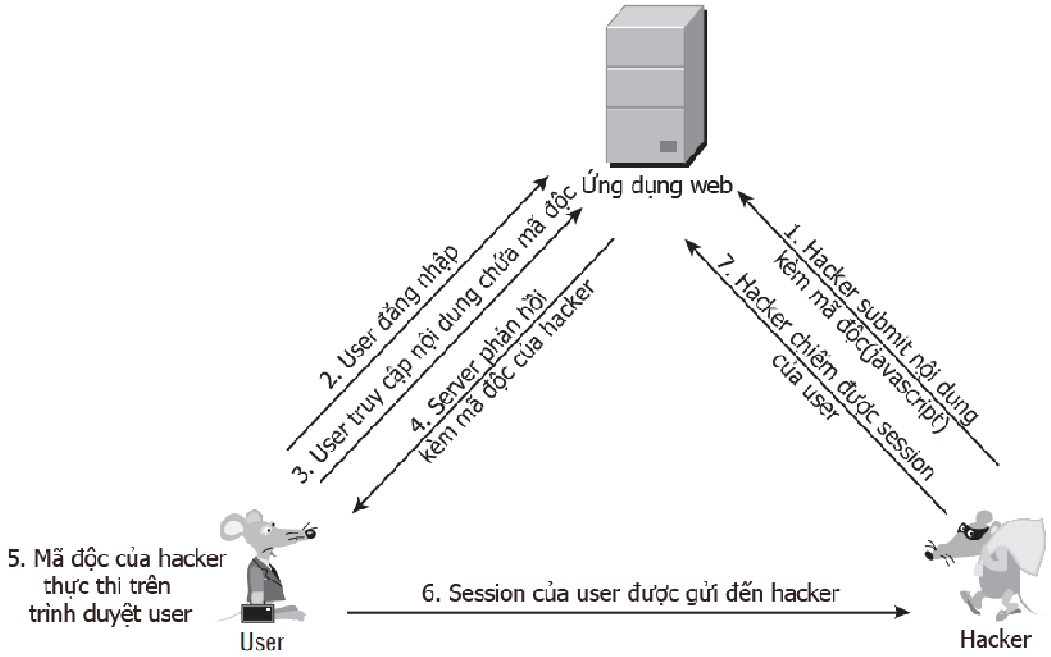
Có thể có trường hợp đoạn mã độc thực thi ngay trên trình duyệt của nạn nhân mà không đợi máy chủ trang web phản hồi về, gọi là DOM base XSS. DOM , viết tắt của Document Object Model, là 1 dạng chuẩn của W3C đưa ra nhằm để truy xuất và thao tác dữ liệu của tài liệu có cấu trúc như HTML, XML. Mô hình này thể hiện tài liệu dưới dạng cấu trúc cây phân cấp. Tất cả các thành phần trong HTML, XML đều được xem như một node [9].

Với kỹ thuật tấn công kiểu DOM base XSS, người tấn công có thể thay đổi giao diện trang web. Ví dụ truyền đoạn mã sau để thêm 1 combobox chọn giới tính vào trong giao diện:

```
http://www.website.com/search.php?giatri= <label class="col-sm-2 control-label">Gender</label><div class="col-sm-4"><select class = "form-control" onchange="java_script_:show()"><option value="Male">Male</option><option value="Female"> Female</option></select> </div><script>function show(){alert();}</script>
```

3.3.2.2 Đoạn mã độc lưu trữ trên trang web (Stored XSS):

Kiểu tấn công này thực hiện được trên các ứng dụng web không kiểm tra kỹ các dữ liệu đầu vào trước khi lưu vào CSDL. Ví dụ như các form góp ý, các nơi đăng bình luận … trên các trang web [10].



Hình 3.1: Quy trình tấn công lấy session ID kiểu Stored XSS

Với kỹ thuật Stored XSS , người tấn công không khai thác trực tiếp mà phải thực hiện tối thiểu qua 2 bước:

- Bước 1, người tấn công sẽ thông qua các điểm đầu vào (form, input, textarea…) không được kiểm tra kỹ để chèn vào CSDL các đoạn mã nguy hiểm.
- Bước 2, khi người dùng truy cập vào trang web và thực hiện các thao tác liên quan đến dữ liệu được lưu này, đoạn mã của người sẽ được thực thi trên trình duyệt người dùng.

Với kiểu tấn công này, nạn nhân có thể là tất cả những người sử dụng trang web đó. Nếu nạn nhân có vai trò quản trị sẽ có nguy cơ bị chiếm quyền điều khiển trang web.

3.3.3 Phương pháp phòng chống:

Cốt lõi của kiểu tấn công XSS là việc thực thi các đoạn mã độc trên trình duyệt nạn nhân, việc vô hiệu hóa việc thực thi các câu lệnh trên trình duyệt người dùng cuối sẽ ngăn chặn được điều này. Để phòng chống tấn công kiểu XSS cần thực hiện các biện pháp trên cả hai phía: máy chủ và người dùng cuối:

- Phía trên máy chủ ứng dụng web: cần phải lọc dữ liệu đầu vào (lọc các kí tự: < > " ' % ;) (& + -) [14] và trước khi hiển thị lại dữ liệu đó ra màn hình trình duyệt trên máy người sử dụng cần mã hóa thay thế các ký tự cần cho việc thực thi đoạn mã độc, chẳng hạn thay ký tự "<" và ">" bằng ">" và "<". Có các hàm chức năng dùng sẵn cho công việc này, chẳng hạn như sql_escape và htmlentities hay bộ lọc HTML Purifier (trong PHP), URLEncode và HTMLEncode (trong ASP.Net). Ngoài ra cần chú ý quản lý phiên truy cập, cấp lại session Id sau khi người dùng đăng nhập, sử dụng giao thức HTTPS, khi lưu dữ liệu quan trọng trong cookie nên mã hóa chúng.
- Phía người dùng cuối: cấu hình trên trình duyệt cho phép thực thi các đoạn mã script ở mức độ nào và có cảnh báo hay không.

3.4 Không mã hóa dữ liệu nhạy cảm:

Đây là trường hợp lỗ hổng bảo mật xảy ra do ý thức người lập trình, người dùng hệ thống. Dữ liệu nhạy cảm của ứng dụng, ví dụ như thông tin tài khoản đăng nhập, thông tin tài khoản ngân hàng,… không được mã hóa khi lưu trữ trên máy chủ hoặc trong quá trình truyền dữ liệu trên mạng. Nó được xếp hạng nguy hiểm thứ 6 trong trong 10 loại nguy cơ gây mất an toàn hệ thống ứng dụng web (dự án The OWASP Top 10 – 2013) [16].

3.4.1 Nguy cơ mất thông tin:

Dữ liệu quan trọng không được mã hóa làm tăng nguy cơ rò rỉ thông tin khi người tấn công xâm nhập được vào hệ thống máy chủ hoặc nghe lén thông tin trên mạng. Có hai trường hợp người tấn công sẽ khai thác:

- Dữ liệu nhạy cảm không được mã hóa khi lưu trữ trên CSDL: Khi người tấn công xâm nhập được vào hệ thống máy chủ, họ có khả năng truy cập được vào

CSDL hoặc các tập tin trên hệ thống. Qua đó, người tấn công dễ dàng lấy được và sử dụng các thông tin như: thông tin xác thực, cấu hình hệ thống, dữ liệu ngân hàng,… nếu chúng không được mã hóa hoặc có mã hóa nhưng thuật toán mã hóa yếu.

- Dữ liệu nhạy cảm không được mã hóa trên đường truyền: Khi người dùng sử dụng các giao thức không mã hóa thông tin xác thực và dữ liệu trước khi chuyển đi trên đường truyền (POP3, IMAP, HTTP,…) nguy cơ bị lộ thông tin đăng nhập, dữ liệu gửi và nhận trên đường truyền mạng là rất cao. Người tấn công trong cùng hệ thống mạng có thể dùng phương pháp nghe lén gói tin truyền trên mạng để lấy các thông tin đó nếu chúng không được mã hóa.

3.4.2 Phương pháp phòng chống:

Đối với phía máy chủ ứng dụng web: sử dụng các thuật toán mã hóa mạnh để mã hóa các dữ liệu nhạy cảm trước khi lưu lại trên hệ thống, chọn giao thức https để phát triển ứng dụng web, mã hóa thông tin xác thực khi truyền đi trên mạng.

Đối với phía người sử dụng: dùng các tiện ích tạo kênh kết nối VPN khi sử dụng hệ thống mạng internet công cộng (ví dụ: phần mềm Hotspot shield).

3.5 Lỗ hổng bảo mật CSRF:

Trang web mắc lỗi CSRF sẽ cho phép người tấn công ép buộc trình duyệt web của nạn nhân gửi các yêu cầu giao thức HTTP đến nó ngoài ý muốn của nạn nhân, trong điều kiện nạn nhân đã đăng nhập (đã được chứng thực). Trong trường hợp phiên truy cập của nạn nhân trên trình duyệt web này chưa hết hạn, kiểu tấn công khai thác lỗi CSRF cho phép người tấn công buộc trình duyệt tạo ra những yêu cầu cho trang web này mà nó không thể biết đây là những yêu cầu giả mạo của người tấn công [16].

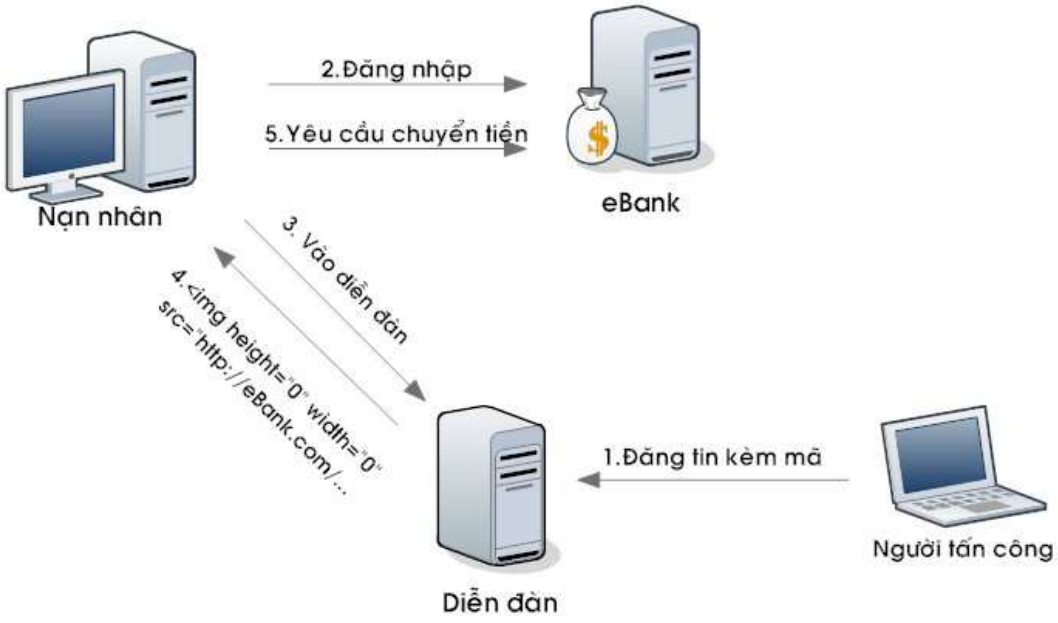
3.5.1 Sự khác nhau giữa hai kiểu tấn công khai thác lỗi XSS và CSRF:

Trong kiểu tấn công khai thác lỗi XSS, trình duyệt trên máy tính nạn nhân nhận đoạn mã nguy hiểm từ trang web bị lỗi XSS gửi về và trình duyệt đó thực thi. Còn đối với kiểu tấn công khai thác lỗi CSRF, trang web bị lỗi CSRF sẽ nhận và

thực hiện yêu cầu (ngoài mong muốn) của nạn nhân từ trình duyệt trên máy tính nạn nhân với chứng thực hợp pháp của nạn nhân đó.

3.5.2 Nguyên lý thực hiện:

- Bước 1: Nạn nhân đăng nhập trang web bị lỗi CSRF.
- Bước 2: Thông qua kiểu tấn công XSS hoặc các phương pháp phi kỹ thuật như gửi link qua email, chat để đánh lừa trình duyệt nạn nhân gửi các yêu cầu ngoài ý muốn của nạn nhân cho trang web đó thực thi với chứng thực của nạn nhân đã đăng nhập trước đó.
- Bước 3: Trang web nhận các yêu cầu đó và thực hiện chúng vì nghĩ đó là yêu cầu hợp pháp đã được gửi từ trình duyệt trên máy tính của nạn nhân (với điều kiện chưa hết hạn phiên truy cập).



Hình 3.3: Một ví dụ tấn công kiểu CSRF

Ví dụ: Người tấn công đăng tin lên diễn đàn có chèn một đoạn mã:

```
<img height=”0″ width=”0″ src=”http://eBank.com/withdraw?
account=hacker_id&amount=1000000&for=victim_ id”/>
```

Nạn nhân sau khi đăng nhập vào trang web của eBank, nếu tình cờ khi phiên truy cập trên trang web của eBank chưa hết hạn mà nạn nhân vào xem tin trên diễn

đàn có kèm đoạn mã trên, trình duyệt của nạn nhân sẽ gửi yêu cầu chuyển tiền đến tài khoản của người tấn công cho máy chủ eBank xử lý.

3.5.3 Phương pháp phòng chống:

Rất khó để phòng chống tấn công kiểu CSRF một cách triệt để, để hạn chế bị tấn công và giảm thiệt hại cho người sử dụng có một số biện pháp như sau:

Về phía máy chủ ứng dụng web:

- Sử dụng Captcha hoặc các thông báo xác nhận ở các form như có chức năng quan trọng: xác thực người dùng, thực thi một lệnh,…
- Sử dụng phương thức Get cho việc truy vấn lấy dữ liệu về, phương thức Post cho mục đích tạo ra thay đổi hệ thống.
- Hạn chế thời gian hiệu lực của phiên truy cập, thời gian ngắn sẽ giảm khả năng bị tấn công.
- Sử dụng Token: Tạo ra một Token tương ứng với mỗi form, Token này sẽ là duy nhất đối với mỗi form và thường thì hàm tạo ra Token này sẽ nhận đối số là phiên truy cập hoặc được lưu thông tin trong phiên truy cập. Khi nhận lệnh Http Post về, hệ thống sẽ thực hiện so khớp giá trị Token này để quyết định có thực hiện hay không.
- Sử dụng cookie riêng biệt cho trang quản trị: Một cookie không thể dùng chung cho các tên miền khác nhau,chính vì vậy việc sử dụng admin.site.com thay vì sử dụng site.com/admin sẽ an toàn hơn.

Về phía người dùng cuối:

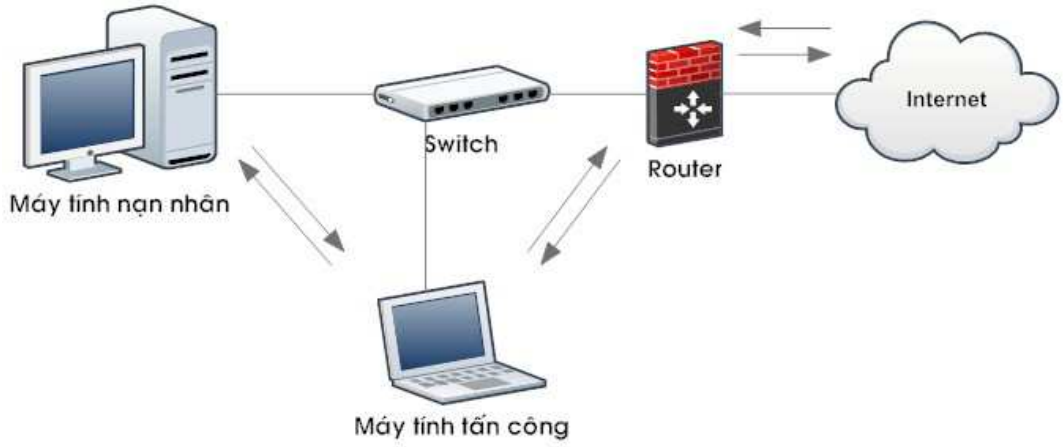
- Trong quá trình đăng nhập vào các trang web quan trọng cần chứng thực như ngân hàng, thanh toán trực tuyến, email… không nên vào cùng lúc các trang web khác và thoát ra sau khi sử dụng xong.
- Không sử dụng tính năng lưu mật khẩu.

3.6 Tấn công kiểu Man in the middle (MITM):

Tấn công kiểu MITM là một dạng nghe lén thông tin trên mạng dạng tích cực (active eavesdropping attack) [18]. Khi người tấn công sử dụng máy tính nằm chung miền mạng với máy tính nạn nhân, người tấn công có thể dùng các phương pháp

chặn bắt gói tin đang vận chuyển trên mạng, qua đó có thể lấy được các thông tin như: thông tin xác thực, session Id, dữ liệu…của các ứng dụng sử dụng các giao thức không mã hóa dữ liệu khi truyền đi trên đường truyền: HTTP, SMTP, POP3, IMAP, FTP..hoặc bắt lấy các phiên truy cập của giao thức https nhằm bỏ qua bước xác thực của các ứng dụng (Facebook, Yahoo,…) để sử dụng các ứng dụng đó với quyền được cấp như chính nạn nhân.

Trong kiểu tấn công MITM, máy tính của người tấn công đứng giữa tiếp nhận và gửi lại các gói tin truyền đi qua lại giữa hai máy tính nạn nhân (hoặc giữa máy tính nạn nhân và gateway ra Internet) [18]. Hầu như nạn nhân sẽ không biết đang bị tấn công.



Hình 3.2: Tấn công kiểu MITM

Có nhiều hình thức tấn công kiểu MITM: ARP Cache Poisoning, DNS Spoofing, DHCP Spoofing,…Trong đó, ARP Cache Poisoning (giả mạo ARP Cache) có lâu đời và được sử dụng nhiều nhất. Trong khuôn khổ tài liệu này sẽ tìm hiểu về tấn công kiểu MITM bằng cách giả mạo ARP Cache.

3.6.1 Tấn công bằng cách giả mạo ARP Cache:

3.6.1.1 Giao thức ARP:

ARP là giao thức ánh xạ địa chỉ IP đến địa chỉ vật lý (MAC) được nhận diện. Khi một máy cần giao tiếp với máy khác, và nó tìm trong bảng ARP Cache của mình, nếu địa chỉ MAC không được tìm thấy trong bảng, giao thức ARP sẽ quảng bá gói ARP request ra toàn miền mạng. Tất cả các máy trong miền mạng sẽ so sánh

địa chỉ IP đến địa chỉ MAC của chúng. Nếu một trong những máy đó, xác định được đó chính là địa chỉ của mình, nó sẽ gửi gói ARP hồi đáp (gói ARP reply) và địa chỉ này sẽ được lưu trong bảng ARP Cache và quá trình giao tiếp diễn ra.

3.6.1.2 ARP Cache:

ARP cache có thể coi như một bảng có chứa một tập tương ứng giữa các địa chỉ MAC và địa chỉ IP. Mỗi một thiết bị trên một mạng nào đó đều có cache riêng. Có hai cách lưu giữ thông tin trong cache để phân giải địa chỉ diễn ra nhanh:

- ARP Cache tĩnh: Các cặp địa chỉ IP và địa chỉ MAC được thêm một cách thủ công vào bảng cache và được duy trì lâu dài.
- ARP Cache động: Các địa chỉ IP và địa chỉ MAC được giữ trong cache bởi phần mềm sau khi nhận được kết quả của việc hoàn thành quá trình phân giải trước đó. Các địa chỉ được giữ tạm thời và sau đó được gỡ bỏ.

3.6.1.3 Nguyên lý tấn công bằng cách giả mạo ARP Cache:

Việc giả mạo bảng ARP chính là lợi dụng tính không an toàn của giao thức ARP. Không giống như các giao thức khác, chẳng hạn như DNS (có thể được cấu hình để chỉ chấp nhận các cập nhật động khá an toàn), các thiết bị sử dụng giao thức ARP sẽ chấp nhận cập nhận bất cứ lúc nào. Điều này có nghĩa rằng bất cứ thiết bị nào có thể gửi gói ARP reply đến một máy tính khác và máy tính này sẽ cập nhật vào bảng ARP cache của nó ngay giá trị mới này. Việc gửi một gói ARP reply khi không có request nào được tạo ra được gọi là việc gửi ARP vu vơ. Khi các ARP reply vu vơ này đến được các máy tính đã gửi request, máy tính request này sẽ nghĩ rằng đó chính là đối tượng mình đang tìm kiếm để truyền thông, tuy nhiên thực chất họ lại đang truyền thông với máy tính người tấn công.

3.6.2 Phương pháp phòng chống tấn công kiểu MTIM:

Người dùng bình thường khó phát hiện bị tấn công kiểu MITM, để phòng chống cũng như hạn chế thiệt hại khi bị tấn công, người dùng cần chú ý:

- Sử dụng các giao thức mã hóa: Https, OpenSSH, SFTP hoặc sử dụng VPN khi sử dụng Internet nơi công cộng.

- Sử dụng bảng ARP Cache dạng tĩnh cho các thiết bị mạng trong trường hợp hệ thống mạng ít thay đổi thiết bị.
- Vô hiệu hóa việc sử dụng giao thức NetBIOS over TCP/IP nhằm làm người tấn công khó dò tìm được địa chỉ IP các máy tính đang có trên mạng.

3.7 Tóm tắt nội dung chương:

Trong chương này đã giới thiệu một số lỗ hổng bảo mật phổ biến như: SQL injection, XSS, CSRF, không mã hóa dữ liệu nhạy cảm, lỗ hổng liên quan đến quản lý phiên truy cập. Ngoài ra cũng giới thiệu các phương pháp tấn công vào các lỗ hổng đó, phương pháp tấn công kiểu MITM và cách phòng chống, ngăn ngừa chúng.

Qua chương này, người đọc có thể nắm được một số cách giúp hệ thống giảm bớt nguy cơ bị tấn công hoặc giảm bớt thiệt hại khi bị tấn công. Đối với người đọc là người dùng thông thường, ít nhất cũng giảm bớt nguy cơ bị tấn công và mất thông tin cá nhân khi sử dụng mạng nơi công cộng.

Chương 4. SỬ DỤNG KALI LINUX KIỂM THỬ AN TOÀN HỆ THỐNG

4.1 Giới thiệu về Kali Linux:

Từ khi xuất hiện vào năm 2006, BackTrack là hệ điều hành được sử dụng nhiều nhất bởi các chuyên gia đánh giá an toàn hệ thống. Trong 7 năm tiếp đó, nó đã không ngừng cải tiến để đạt được một vị trí nhất định trong cộng đồng bảo mật trên khắp thế giới. Vào tháng 03/2013, trung tâm đào tạo bảo mật online Offensive Security giới thiệu Kali Linux thay thế cho Backtrack nhằm ổn định và dễ sử dụng hơn.

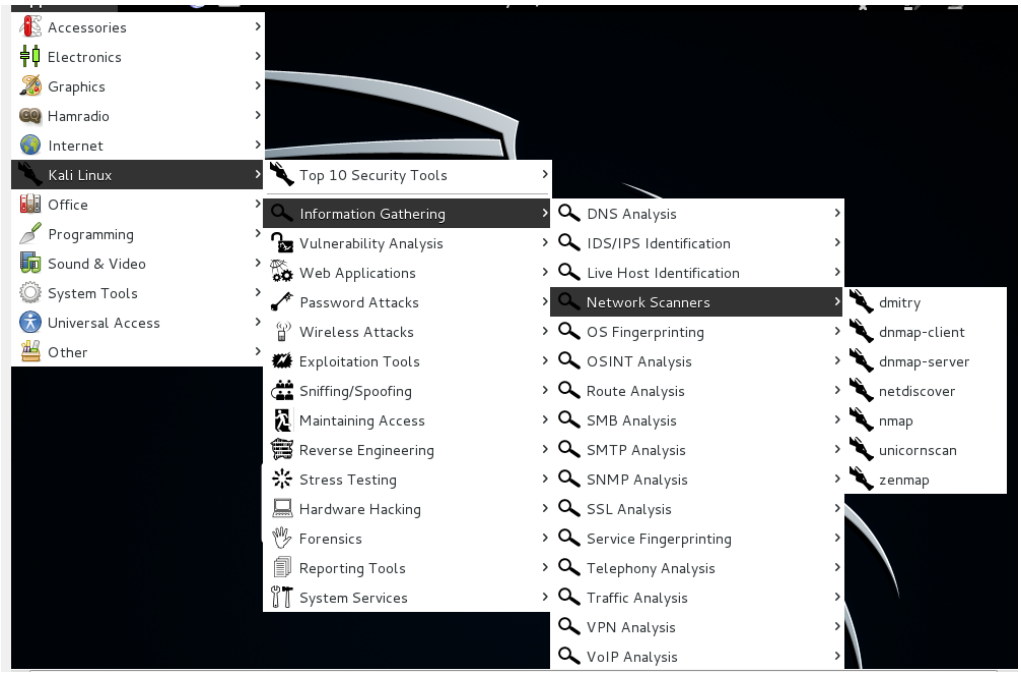
Kali Linux (gọi tắt là Kali) được phát triển trên nền tảng hệ điều hành Debian tích hợp sẵn hàng loạt các công cụ dò tìm lỗ hổng, thâm nhập thử nghiệm, nó có các tính năng sau:

- Miễn phí: Bản thân Kali được phát hành miễn phí. Các công cụ có sẵn khi cài đặt có hai dạng: bản miễn phí hoàn toàn đủ chức năng và bản miễn phí có giới hạn chức năng nhưng đủ dùng cho người kiểm thử không chuyên.
- Tương thích nhiều cấu hình phần cứng: Từ PC đến các máy tính sử dụng chip vi xử lý ARM như: Raspberry Pi, Odroid, Beaglebone…
- Tích hợp sẵn có hơn 300 bộ công cụ kiểm thử với đủ các nhóm chức năng như: thu thập thông tin, phân tích lỗ hổng, tấn công thử nghiệm cho cả hệ thống phần mềm lẫn phần cứng, kiểm tra hiệu năng hệ thống, báo cáo.
- Đa ngôn ngữ và dễ tùy biến: cho phép tùy biến giao diện sử dụng cũng như thêm vào các công cụ kiểm thử không có sẵn theo bộ cài đặt.
- Dễ dàng cập nhật phiên bản mới khi có bản nâng cấp của Kali hay các công cụ kiểm thử trên nó.

4.2 Phân nhóm các công cụ có sẵn trên Kali Linux:

Kèm theo bản cài đặt mặc định là hàng trăm bộ công cụ phục vụ cho nhiều mục đích khác nhau của quá trình kiểm thử. Để dễ dàng cho người sử dụng, trong

Kali phân loại chúng theo mục đích sử dụng thành nhiều nhóm, một công cụ có thể nằm trong nhiều nhóm khác nhau.



Hình 4.1: Phân nhóm công cụ trong Kali

4.2.1 Thu thập thông tin (Information gathering):

Nhóm phân loại này gồm những công cụ tập trung vào việc thu thập thông tin về mục tiêu. Trong phân loại này có một số lượng lớn các công cụ được phân chia theo loại thông tin cần thu thập. Ví dụ: OSINT Analysis (thu thập thông tin đối tượng từ các nguồn thông tin công khai), OS Fingerprinting (thu thập thông tin về hệ điều hành), Network Scanners (dò quét cổng, dò quét mạng, dò quét phiên bản dịch vụ), SSL Analysis (phân tích giao thức SSL), VoIP Analysis (phân tích giao thức VoIP), phân tích VPN và còn nhiều công cụ khác nữa.

Việc thu thập thông tin là một bước rất quan trọng trong quá trình kiểm thử, nên các công cụ trong phần này rất đa dạng về loại thông tin thu thập, về loại mục tiêu (ứng dụng web, thiết bị mạng, hệ quản trị CSDL, hệ điều hành,…).

4.2.2 Phân tích lỗ hổng (Vulnerability analysis):

Nhóm phân loại này gồm những công cụ tập trung vào việc phát hiện các lỗ hổng bảo mật, từ lỗ hổng ứng dụng, hạ tầng mạng cho đến phần cứng chuyên dụng.

Vì vậy ở đây có rất nhiều các công cụ dùng để dò quét lỗ hổng các thiết bị của Cisco, CSDL, ứng dụng web hay hệ thống máy tính.

Đặc biệt trong phần này có các công cụ phân tích lỗ hổng theo thuật toán Fuzzing. Fuzzing là thuật toán trong kiểm thử tiêu cực, thay vì gửi các dữ liệu hợp lý (được xử lý theo ý đồ thiết kế của mã nguồn), hệ thống sẽ nhận được các đầu vào hoặc chuỗi đầu vào không hợp lệ hoặc bán hợp lệ thông qua giao diện tương tác.

Chương trình hoặc framework tạo ra các kiểm thử fuzz (fuzz test) hoặc thực thi các kiểm thử gọi là Fuzzer. Fuzzer có thể được phân loại dựa trên 2 tiêu chí khác biệt nhau: Vector tiêm (injection) hoặc vector tấn công.

Một quy trình fuzzing đơn giản bao gồm một chuỗi các thông điệp được gửi tới hệ thống được kiểm tra. Các kết quả thay đổi và thông điệp gửi tới có thể được phân tích, trong một số trường hợp có thể bị bỏ qua. Kết quả trả về điển hình của một kiểm thử fuzz bao gồm: Đáp trả hợp lệ (Valid response), đáp trả lỗi (Error response), đáp trả bất thường (Anomalous response), sụp đổ hay thất bại (Crash or other failure).

Quá trình fuzzing không chỉ là việc gửi và nhận các thông điệp. Kiểm thử đầu tiên sẽ được tạo ra và gửi tới hệ thống được kiểm tra. Việc giám sát mục tiêu được thực hiện liên tục, tất cả các thất bại đều được ghi lại để đánh giá cho lần sau.

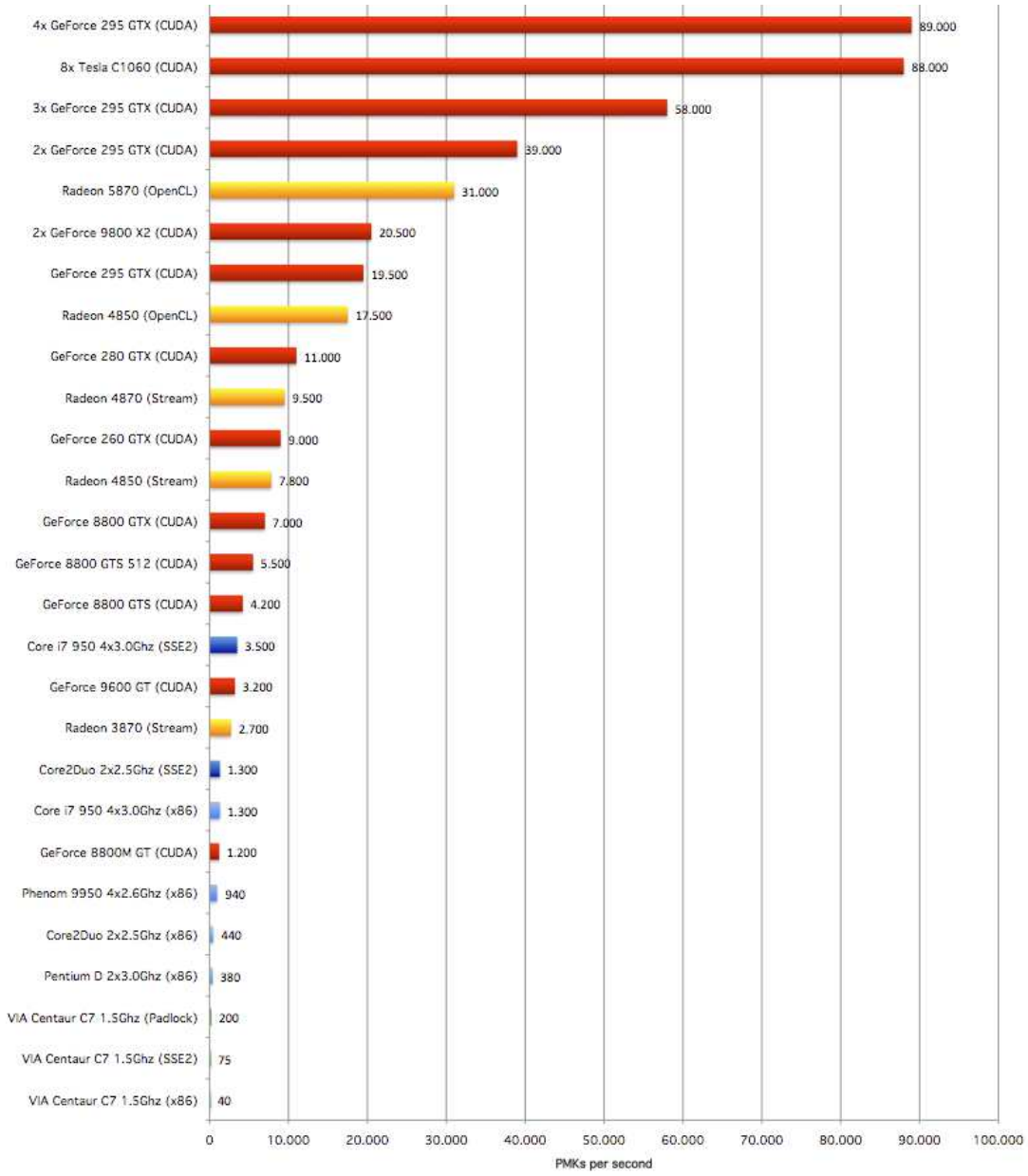
4.2.3 Ứng dụng web (Web applications):

Trong phân loại này gồm những công cụ dùng để dò tìm và tấn công thử nghiệm các ứng dụng web. Từ các công cụ có nhiều chức năng: dùng để lấy thông tin của mục tiêu (CMS identification), dò tìm và khai thác lỗ hổng SQL injection, các công cụ chuyên dò tìm phát hiện và khai thác lỗ hổng bảo mật trên ứng dụng web.

4.2.4 Tấn công mật khẩu (Password attacks):

Bao gồm những công cụ dùng để dò tìm mật khẩu theo từ điển, brute force cả online lẫn offline. Có loại còn giúp dò tìm mật khẩu đã được mã hóa một chiều dạng băm. Các công cụ trong phần này cho phép tấn công theo các giao thức Http, Ftp, Ssh, Rdp, SMTP,… Một số công cụ cho phép tận dụng sức mạnh của GPU (hỗ

trợ GPU của hãng Nvidia và ATI) để tăng tốc độ xử lý (ví dụ như Pyrit) lên gấp nhiều lần so với dùng CPU tính toán.



Hình 4.2: So sánh tốc độ dò tìm cặp khóa PMK trên CPU - GPU bằng Pyrit [19]

Ngoài ra, có các công cụ dùng để tấn công mật khẩu vào các hệ CSDL như MSSQL, MySQL, Oracle hay các thiết bị phần cứng của hãng Cisco.

4.2.5 Tấn công mạng không dây (Wireless attacks):

Có các công cụ hỗ trợ để tấn công các loại giao thức mạng không dây: IEEE 802.11, RFID / NFC hay Bluetooth. Các công cụ này cho phép bẻ khóa mật khẩu mạng Wifi được mã hóa theo chuẩn WEP hay WPA/WPA2, nghe lén, tấn công từ chối dịch vụ hoặc tấn công kiểu MITM.

4.2.6 Khai thác, tấn công thông qua các lỗ hổng (Exploitation tools):

Sau dò tìm được các lỗ hổng bảo mật sẽ sử dụng các công cụ này tấn công hệ thống thông qua lỗ hổng bảo mật đó. Nếu thành công xem như chắc chắn hệ thống tồn tại lỗ hổng đó và cần phải cập nhật, vá lỗi. Có rất nhiều công cụ giúp tấn công vào nhiều dạng lỗ hổng bảo mật khác nhau: SQL injection, XSS, CSRF,… Có cả một framework phục vụ cho việc tấn công hệ thống như bộ Metasploit.

4.2.7 Nghe lén – giả mạo (Sniffing/Spoofing):

Trong phần này bao gồm các công cụ phục vụ cho triển khai tấn công kiểu MITM, chặn bắt các gói tin truyền trên mạng, kể cả chặn bắt các cuộc gọi VoIP, IP Video, tấn công khâu xác thực của hệ thống VoIP sử dụng giao thức SIP. Nó cho phép lọc ra dữ liệu theo yêu cầu: mật khẩu, email, http, cookie.

4.2.8 Duy trì kết nối, quyền truy cập (Maintaining Access):

Tập hợp các công cụ cho phép tạo cửa sau (backdoor) vào hệ thống sau khi đã chiếm được quyền điều khiển hệ thống đó, nhằm mục đích có thể dễ dàng xâm nhập khai thác hệ thống đó về sau. Các công cụ này có khả năng download và upload các tập tin, thực thi các tập lệnh của hệ điều hành trên hệ thống đó, cho phép tạo kênh kết nối trực tiếp từ máy tính người tấn công đến máy tính của nạn nhân.

4.2.9 Dịch ngược mã nguồn (Reverse Engineering):

Tập hợp các công cụ giúp debug hay phân tích mã nguồn các chương trình ứng dụng nhằm tìm ra các lỗ hổng bảo mật trong ứng dụng đó. Chẳng hạn các lỗ hổng kiểu tràn bộ đệm hay các lỗ hổng về xác thực. Nó cho phép so sánh tập tin nhị phân, dò tìm các đoạn mã phá hoại nằm vùng trong ứng dụng do tác giả cố tình để lại. Một số các công cụ này cũng nằm bên nhóm các công cụ "Điều tra dấu vết tấn công"

4.2.10 Kiểm tra hiệu năng (Stress Testing):

Cho phép kiểm tra hiệu năng của mạng LAN, mạng không dây, hệ thống VoIP hay các ứng dụng Web. Thực chất có thể dùng các công cụ này để tấn công kiểu từ chối dịch vụ DOS. Khi sử dụng các công cụ này cần cân nhắc về thời gian thực hiện và độ dài cuộc thử nghiệm để tránh làm ảnh hưởng đến hệ thống.

4.2.11 Chỉnh sửa phần cứng (Hardware Hacking):

Hiện nay trong phần này chỉ mới có một số ít các công cụ thao tác trên các đối tượng là hệ thống Android hay hệ thống vi xử lý mã mở Arduino. Các công cụ trong phần này giúp lập trình Android, phân tích các ứng dụng Android.

4.2.12 Điều tra dấu vết tấn công (Forensic):

Các công cụ trong phần này cho phép giám sát, chặn bắt và phân tích tính toán dữ liệu truyền trên mạng. Còn có các công cụ giám sát hoạt động của hệ thống máy tính, truy tìm dấu vết tấn công thông qua việc tìm các thay đổi của hệ thống tập tin, các dữ liệu đã xóa, hoặc nội dung bộ nhớ RAM.

4.2.13 Báo cáo (Reporting):

Có các công cụ hỗ trợ việc tạo báo cáo kết quả kiểm thử hệ thống từ kết quả thu được khi sử dụng các công cụ trong các phần ở trên.

4.3 Quy trình kiểm thử an toàn hệ thống:

Theo các phương pháp kiểm thử an toàn hệ thống đã giới thiệu ở chương 2, nhìn chung một quy trình kiểm thử an toàn hệ thống gồm các bước như sau:

- Lên kế hoạch.
- Tìm hiểu và thu thập thông tin về mục tiêu.
- Dò tìm các lỗ hổng bảo mật.
- Xác nhận các lỗ hổng bảo mật đã phát hiện.
- Lập báo cáo.

Với mục tiêu sử dụng Kali làm bộ công cụ kiểm thử an toàn hệ thống dành cho người không chuyên về bảo mật. Qua các phương pháp, tiêu chuẩn kiểm thử an toàn hệ thống đã tìm hiểu ở chương 2: OWASP, OSSTMM, ISSAF, WASC-TC, NIST SP 800-115. Nhận thấy xây dựng quy trình kiểm thử an toàn hệ thống sử dụng

Kali Linux theo **Tài liệu hướng dẫn kiểm tra và đánh giá an toàn thông tin (NIST SP 800-115)** phù hợp với mục tiêu đề tài đưa ra vì những lý do sau:

- Áp dụng được cho nhiều loại hệ thống: ứng dụng web, hệ thống mạng LAN, mạng không dây Wifi, máy tính cá nhân. NIST SP 800-115 đưa ra các phương pháp kiểm thử chung cho nhiều loại hệ thống, Kali có các công cụ có chức năng và phân loại phù hợp để thực hiện các phương pháp đó.
- Quy trình đơn giản, dễ hiểu và áp dụng dễ dàng: Kali dễ dàng nâng cấp, thêm ứng dụng và đặc biệt có cộng đồng người sử dụng rộng lớn nên tài liệu hướng dẫn sử dụng nhiều.
- Dữ liệu về lỗ hổng bảo mật và cách khai thác nó được cập nhật thường xuyên, giúp nhanh chóng phát hiện những lỗ hổng mới xuất hiện.

Hình 4.3: Quy trình kiểm thử an toàn hệ thống

Nội dung chi tiết các bước sẽ được nêu ra kèm tên các công cụ tương ứng có trong Kali.

4.3.1 Bước lập kế hoạch:

Đây là bước đầu tiên của quy trình kiểm thử an toàn hệ thống. Bao gồm các công việc: xác định mục tiêu cần kiểm thử, phương pháp kỹ thuật sử dụng, số lượng người tham gia kiểm thử, xác định vai trò trách nhiệm cá nhân, thời gian thực hiện và chuẩn bị tài liệu có nội dung liên quan đến công việc sắp thực hiện (tài liệu hướng dẫn sử dụng, tài liệu kỹ thuật, chính sách bảo mật..). Trong bước này cần thực hiện các công việc đảm bảo tính hợp pháp của quá trình kiểm thử (ký kết các

văn bản có tính pháp lý với chủ đối tượng cần kiểm thử), bởi vì đối với hệ thống được kiểm thử, quá trình kiểm thử tương tự như quá trình tấn công của kẻ xấu.

Các công cụ trong Kali có thể sử dụng cho bước này bao gồm: Casefile (lưu giữ thông tin thể hiện các mối quan hệ dạng đồ họa), Dradis (chia sẻ thông tin), Keep-note (lưu giữ ghi chú), Redmine (quản lý dự án).

4.3.2 Tìm hiểu và thu thập thông tin mục tiêu:

Giai đoạn này bao gồm hai bước nhỏ: Thu thập thông tin liên quan đến mục tiêu, dò tìm và phân tích lỗ hổng bảo mật. Tùy theo mục tiêu đã được giới hạn và các phương pháp kỹ thuật đã lựa chọn trong bước "Lập kế hoạch", chọn các công cụ phù hợp có sẵn trên Kali.

Kết quả thu được từ giai đoạn này rất quan trọng, nó giúp giảm công sức thực hiện các bước sau và tăng tính chính xác kết quả thu được sau khi kiểm thử.

4.3.2.1 Thu thập thông tin liên quan đến mục tiêu:

Thông tin liên quan đến mục tiêu bao gồm:

- Thông tin chủ sở hữu: tên tổ chức, cá nhân, số điện thoại, email..
- Thông tin cá nhân liên quan: thông tin của người lãnh đạo, người quản trị, người sử dụng. Ví dụ: họ tên, địa chỉ email, ngày tháng năm sinh, sở thích,…
- Thông tin hosts (nếu là ứng dụng web): thông tin về DNS, nhà cung cấp dịch vụ, địa chỉ IP các máy chủ liên quan,…
- Thông tin hệ thống: thông tin về hệ điều hành, hệ thống tường lửa, các dịch vụ và ứng dụng đang chạy trên hệ thống đó.

Việc thu thập thông tin một cách đầy đủ và đa dạng cộng với việc đánh giá chính sách bảo mật sẽ giúp có một đánh giá chính xác về những lỗ hổng bảo mật tồn tại ở yếu tố con người trong hệ thống cần kiểm thử. Quá trình thu thập thông tin liên quan đến hệ thống cần kiểm thử có thể thông qua việc đọc các tài liệu đã chuẩn bị ở Bước "Lập kế hoạch" kết hợp sử dụng các công cụ có sẵn trên Kali.

Đa phần các công cụ có sẵn trên Kali phục vụ cho công việc ở bước này được phân nằm trong nhóm "Thu thập thông tin" (Information Gathering), nhóm "Tấn công mạng không dây" (Wireless attacks).

Bảng 4.1: Một số công cụ có sẵn trên Kali sử dụng để thu thập thông tin

STT	Mục đích	Tên công cụ	Phân nhóm trên Kali	Mô tả
1	Thu thập thông tin các đối tượng liên quan đến mục tiêu	Creepy	OSINT Analysis	Xác định vị trí địa lý và thể hiện trên bản đồ người dùng có tài khoản Twitter hoặc Flickr
2		Dmitry	OSINT Analysis	Thu thập thông tin host, địa chỉ Ip các subdomain
3		TheHarvester	OSINT Analysis	Thu thập thông tin host, địa chỉ Ip, email liên quan đến mục tiêu
4		Metagoofil	OSINT Analysis	Dùng Google tìm các tập tin có nội dung liên quan đến mục tiêu, trích xuất thông tin tự động, tổng hợp và xuất thông tin ra dưới dạng HTML
5		Maltego	OSINT Analysis	Tương tự Theharvester nhưng kết quả hiện thị dạng đồ thị
6	Khám phá mạng	Fierce	DNS Analysis	Dò tìm thông tin DNS
7		Fping	Live Host Indenfitication	Gửi gói ICMP đến host, cho phép thực hiện ở tốc độ cao và dò tìm tìm theo dãy IP.
8	Xác định hệ điều hành, các dịch vụ đang chạy	Hping3	Live Host Indenfitication	Gửi gói tin TCP tùy biến, xác định mục tiêu có nằm sau tường lửa hay không, các dịch vụ mạng đang chạy trên mục tiêu
9		Nmap	OS Fingeringting	Phát hiện host, tìm các cổng đang mở, xác định ứng dụng, phiên bản, hệ điều hành chạy trên mục tiêu. Dò tìm một số lỗ hổng bảo mật
10		Uniscan-gui	Web Vulnerability Scanners	Tìm thông tin máy chủ, các dịch vụ đang chạy trên máy chủ đó
11		Whatweb	Web Vulnerability Scanners	Tìm thông tin ứng dụng web
12	Quét mạng không dây	Bluemaho	Bluetooth Tools	Dò tìm các thiết bị Bluetooth, gửi tập tin, có khả năng dò tìm lỗ hổng bảo mật các thiết bị đó
13		Kismet	802.11Wireless Tools	Dò tìm thông tin hệ thống Wifi ,phát hiện các hệ thống ẩn SSID, nghe lén thông tin trên mạng Wifi

4.3.2.2 Dò tìm và phân tích lỗ hổng bảo mật:

Dựa trên thông tin đã thu thập ở bước trước, đặc biệt là thông tin về hệ điều hành của hệ thống, các cổng đang mở, các dịch vụ mạng, ứng dụng đang chạy trên hệ thống, phiên bản các ứng dụng, sẽ lựa chọn công cụ phù hợp có sẵn trên Kali để tiết kiệm thời gian và cho kết quả ở mức chính xác cao nhất.

Ví dụ: Xác định được ứng dụng web là Joomla (một CMS mã nguồn mở viết bằng ngôn ngữ PHP) sẽ dùng công cụ joomscan để dò tìm lỗ hổng bảo mật.

Hầu hết các công cụ này nằm trong nhóm "Phân tích lỗ hổng" (Vulnerability analysis), nhóm Ứng dụng web (Web applications), một số ít các công cụ phải cài đặt thêm như Nexpose và Nessus.

Bảng 4.2: Một số công cụ có sẵn trên Kali sử dụng để phân tích lỗ hổng

STT	Mục đích	Tên công cụ	Phân nhóm trên Kali	Mô tả
1	Dò tìm lỗ hổng bảo mật cho nhiều loại hệ thống	Nessus	Cài đặt thêm	Bản miễn phí cho phép dò tìm tối đa 16 địa chỉ IP, sử dụng mã nguồn của Openvas, có nhiều cấu hình dành cho nhiều loại hệ thống, kể cả điện thoại Android. Đánh giá mức độ nguy hiểm của lỗ hổng
2		Nexpose	Cài đặt thêm	Bản miễn phí cho phép dò tìm tối đa 32 địa chỉ IP. Chỉ ra phương pháp khai thác lỗi.
3		Nmap	Misc Scanners	Dùng với cú pháp --vuln sẽ chạy các script kiểm tra lỗ hổng trên mục tiêu
4		Openvas	Openvas	Miễn phí, có định danh và mô tả lỗ hổng chi tiết, đánh giá mức độ nguy hiểm của lỗ hổng
5	Dò tìm lỗ hổng bảo mật hệ CSDL	jSQL	Database Assessment	Quét thông tin, dò tìm lỗi SQL hệ CSDL, lỗi SQL injection của ứng dụng web.Hỗ trợ các hệ CSDL:MySQL,DB2 ,PostgreSQL,MSSQL,Firebird,MaxDB
6		SQLmap	Database Assessment	Quét thông tin, dò tìm lỗi SQL hệ CSDL, lỗi SQL injection của ứng dụng web.Hỗ trợ các hệ CSDL:MySQL,DB2 ,Oracle, PostgreSQL,MSSQL,Access, SQLite,Firebird, Sybase, MaxDB

7		SQLninja	Database Assessment	Quét thông tin, dò tìm lỗi SQL hệ CSDL, ứng dụng web. Hỗ trợ hệ CSDL MSSQL
8		Oscanner	Database Assessment	Dò tìm thông tin hệ CSDL Oracle
9	Dò tìm lỗ hổng bảo mật hệ thống mạng	Cisco-global-exploiter	Cisco Tools	Dò tìm khoảng 14 loại lỗ hổng bảo mật trên các thiết bị Cisco
10		Yersinia	Cisco Tools	Hoạt động trên layer-2, tấn công thử nghiệm các giao thức: STP,VTP, HSRP,DTP,IEEE 802.1X,CDP, DHCP,ISL,MPLS
11		Nmap	Misc Scanners	Dùng với cú pháp --script=sniffer-detect để dò tìm đối tượng đang nghe lén trên hệ thống mạng
12	Dò tìm lỗ hổng bảo mật ứng dụng web	Grabber	Web Vulnerability Scanners	Dò tìm lỗi XSS, SQL injection, file inclusion, phân tích mã Javascript,…
13		Nikto	Misc Scanners	Dò tìm lỗi cấu hình, kiểm tra phiên bản phần mềm, nhiều loại lỗ hổng bảo mật.
14		Owasp-zap	Web Vulnerability Scanners	Bộ công cụ toàn diện nhất hiện nay để đánh giá ứng dụng web: đánh giá chức năng, dò tìm lỗ hổng bảo mật
15		Joomscan	Web Vulnerability Scanners	Dò tìm lỗ hổng bảo mật trang web sử dụng Joomla
16		Vega	Web Vulnerability Scanners	Bộ công cụ dò tìm lỗ hổng bảo mật giao diện đồ họa
17		Xsser	Web Vulnerability Scanners	Tự động dò tìm, khai thác, báo cáo lỗi XSS
18		W3af	Web Vulnerability Scanners	Bộ công cụ dò tìm lỗ hổng bảo mật giao diện đồ họa
19		Wfuzz	Web Vulnerability Scanners	Dò tìm và khai thác các lỗi injection ,XSS. Hỗ trợ bruteforce, tấn công vào form đăng nhập theo kiểu bruteforce
20		Wpscan	Web Vulnerability Scanners	Dò tìm lỗ hổng bảo mật trang web sử dụng Wordpress
21	Dò tìm kiểu fuzzing	bed	Fuzzing Tools	Dò tìm các lỗi buffer overflows, format string bugs, integer overflows,..
22		siparmyknife	Fuzzing Tools	Dò tìm các lỗi XSS, SQL injection, log injection, format strings, buffer overflows,..

Fuzzing là thuật toán trong kiểm thử tiêu cực, trái ngược với kiểm tra chức năng, kiểm tra khả năng của hệ thống. Trong kiểm thử tiêu cực, thay vì gửi các dữ

liệu hợp lý (được xử lý trong mã nguồn), hệ thống được kiểm thử sẽ nhận được các đầu vào hoặc chuỗi đầu vào không hợp lệ thông qua giao diện tương tác.

Một quy trình fuzzing đơn giản bao gồm một chuỗi các thông điệp được gửi tới hệ thống được kiểm thử. Các kết quả thay đổi và thông điệp gửi tới có thể được phân tích, trong một số trường hợp có thể bị bỏ qua. Kết quả trả về điển hình của một kiểm thử fuzz bao gồm: Đáp trả hợp lệ (Valid response), đáp trả lỗi (Error response), đáp trả bất thường (Anomalous response), sụp đổ hay thất bại (Crash or other failure).

Quá trình fuzzing không chỉ là việc gửi và nhận các thông điệp. Kiểm thử đầu tiên sẽ được tạo ra và gửi tới hệ thống được kiểm thử. Việc giám sát mục tiêu cần được thực hiện liên tục, tất cả các thất bại đều được ghi lại để đánh giá trong các lần sau. Một phần quan trọng của quá trình fuzzing là giám sát mã lệnh, khi nó xử lý một đầu vào không hợp lệ.

Một số công cụ vừa có chức năng dò tìm lỗ hổng vừa có chức năng khai thác lỗ hổng đó. Khi sử dụng các công cụ này cần chú ý vấn đề toàn vẹn dữ liệu của mục tiêu, nên backup dữ liệu, tránh gây hư hỏng, mất mát dữ liệu.

4.3.3 Bước xác nhận lỗ hổng bảo mật:

Sau khi có được danh sách các lỗ hổng bảo mật tìm được ở bước trước, cần sử dụng các phương pháp khai thác các lỗ hổng đó để xác nhận chúng có tồn tại thật sự hay không. Tuy nhiên, do các phương pháp này yêu cầu nhiều kiến thức chuyên sâu và có thể gây hại đến hệ thống, nên người kiểm thử không chuyên có thể cân nhắc bỏ qua và xem như các lỗ hổng đó có tồn tại và cần kiểm tra, vá lỗi.

Một số công cụ vừa có chức năng dò tìm lỗ hổng bảo mật vừa có chức năng khai thác lỗ hổng đó, nên các công cụ đó sẽ được phân nhóm trên Kali trong cả hai mục: 4.2.2 và 4.2.6. Vì mục đích của tài liệu hướng đến việc tự kiểm tra hệ thống dành cho người quản trị, người lập trình không chuyên về bảo mật tự đánh giá được hệ thống, nên những công cụ chuyên tấn công dạng: tấn công mật khẩu, tấn công kiểu Dos và DDos, chiếm quyền điều khiển sẽ không được giới thiệu.

Bảng 4.3: Một số công cụ có sẵn trên Kali sử dụng để khai thác lỗ hổng

STT	Mục đích	Tên công cụ	Phân nhóm trên Kali	Mô tả
1	Khai thác nhiều loại lỗ hổng	Metasploit	Exploitation Tools	Là Framework tập hợp rất nhiều công cụ phục vụ cho việc tấn công, kiểm tra độ an toàn hệ thống
2		Social Engineering Toolkit	Exploitation Tools	Bộ công cụ giúp tấn công theo các phương pháp phi kỹ thuật
3		Websploit	Web Vulnerability Scanners	Bộ công cụ phục vụ cho việc tấn công ứng dụng web lẫn tấn công theo kiểu MITM, tấn công mạng Wifi, thiết bị Bluetooth
4	Khai thác lỗ hổng bảo mật liên quan đến CSDL	jSQL	Database Assessment	Khai thác lỗi SQL injection của ứng dụng web. Hỗ trợ các hệ CSDL:MySQL,PostgreSQL,DB2, MSSQL,Firebird,MaxDB
5		SQLmap	Database Assessment	Khai thác lỗi SQL injection của ứng dụng web. Hỗ trợ các hệ CSDL: MySQL,Oracle,DB2, PostgreSQL, MSSQL,Access, SQLite,Firebird, Sybase, MaxDB
6		SQLninja	Database Assessment	Khai thác lỗi SQL injection của ứng dụng web dùng MSSQL
7	Khai thác lỗ hổng bảo mật hệ thống mạng	cisco-global-exploiter	Cisco Tools	Khai thác lỗ hổng bảo mật các thiết bị Cisco
8		Yersinia	Cisco Tools	Hoạt động trên layer-2, tấn công thử nghiệm các giao thức: STP,VTP,HSRP,DTP, IEEE 802.1X,CDP, DHCP, ISL,MPLS
9	Khai thác lỗ hổng bảo mật ứng dụng web	BeEF	Exploitation Tools	Công cụ chuyên khai thác lỗi XSS
10		Xsser	Web Vulnerability Scanners	Tự động dò tìm, khai thác, báo cáo lỗi XSS
11		W3af	Web Vulnerability Scanners	Khai thác lỗ hổng bảo mật giao diện đồ họa
12		Wfuzz	Web Vulnerability Scanners	Dò tìm và khai thác các lỗi XSS, injection. Hỗ trợ bruteforce, tấn công vào form đăng nhập theo kiểu bruteforce

4.3.4 Bước lập báo cáo:

Sau khi đã xác nhận được đúng các lỗ hổng đang tồn tại trên mục tiêu, cần tổng hợp thông tin từ bước lên kế hoạch, thu thập thông tin đến bước xác nhận lỗ hổng thành một báo cáo hoàn chỉnh. Các công cụ sử dụng ở bước lên kế hoạch có thể tạo ra báo cáo: Casefile, Dradis. Để ghi video lại quá trình thực hiện việc kiểm tra sử dụng recordmydesktop.

4.4 Thực nghiệm:

Trong phần thực nghiệm sẽ hành kiểm thử an toàn hệ thống đối với ba trường hợp: máy tính dùng hệ điều hành Windows trong mạng LAN, máy chủ ứng dụng web, mạng không dây, dựa theo quy trình và các công cụ đã nêu ra ở mục 4.3 của tài liệu.

4.4.1 Kiểm thử hệ thống mạng LAN:

Tiến hành kiểm thử mạng LAN tại văn phòng công ty Dowasen với mục đích tìm lỗ hổng bảo mật trên máy tính nối mạng LAN và tìm các máy tính đang nghe lén trên mạng LAN. Các bước tiến hành: chỉ thực hiện bước dò tìm lỗ hổng với công cụ nmap. Máy tính người kiểm thử nằm trong mạng LAN.

Trong nmap có sẵn 73 đoạn mã dò tìm lỗ hổng bảo mật, từ loại dành cho các ứng dụng ftp, web đến các giao thức ssl, dịch vụ samba..Trong phần này sẽ thực thi 05 đoạn mã sau:

- **Rdp-vuln-ms12-020**: Dò tìm lỗ hổng bảo mật (lỗ hổng MS12-020) cho phép người tấn công từ xa có thể kiểm soát và làm tê liệt hoàn toàn máy tính bằng cách cài đặt các mã độc đối với các máy tính sử dụng hệ điều hành Micrsoft Windows, kể cả phiên bản máy bàn và phiên bản máy chủ, có mở dịch vụ Remote Desktop thông qua giao thức Remote Desktop Protocol (RDP) [20].
- **Samba-vuln-cve-2012-1182**: Dò tìm lỗ hổng bảo mật trên các thiết bị có sử dụng dịch vụ Samba dùng để chia sẻ tập tin (lỗ hổng CVE-2012-1182). Loại lỗ hổng này cho phép người tấn công thực thi các đoạn mã độc từ xa.
- **Smb-check-vulns**: Dò tìm 05 loại lỗ hổng bảo mật: MS08-067 (sâu Conficker phát tán dựa trên lỗ hổng này), CVE-2009-3103 (cho phép thi hành

mã độc từ xa), MS06-025,MS07-029, regsvc DoS (lỗ hổng cho phép tấn công từ chối dịch vụ vào Remote Registry Service của hệ điều hành Windows).

- **Smb-vuln-ms10-054**: Dò tìm lỗ hổng bảo mật MS10-054 của các máy tính chạy hệ điều hành Windows.
- **Smb-vuln-ms10-061**: Dò tìm lỗ hổng bảo mật MS10-061 nằm trên dịch vụ in ấn của các máy tính chạy hệ điều hành Windows. Sâu Stuxnet khai thác lỗ hổng này.

Cú pháp câu lệnh như sau:

```
nmap  --script=tendoanma  192.168.254.0/24  -oN  tentaptin.log
```

Trong đó:

- tendoanma: Tên các đoạn mã cần thực thi đã nêu ở trên (ví dụ: smb-vuln-ms10-054,…).
- tentaptin : Tên tập tin lưu kết quả dò tìm.

Kết quả quét hệ thống mạng LAN tại văn phòng công ty Dowasen như sau:

- Số thiết bị nối mạng: 98 (bao gồm máy tính, điện thoại, máy tính bảng,..)
- Số thiết bị tìm thấy lỗ hổng bảo mật:
    - Lỗ hổng MS12-020: 02 máy tính.
    - Lỗ hổng CVE-2009-3103: 15 máy tính.
    - Lỗ hổng CVE-2012-1182: 01 đầu phát HD player.

Để phát hiện có kẻ nghe lén thông tin trên mạng, sử dụng câu lệnh sau:

```
nmap  --script=sniffer-detect  192.168.254.0/24
```

Thực chất câu lệnh trên sẽ dò tìm các máy tính đang bật chế độ promiscuous cho card Ethernet trên máy tính đó. Ở chế độ này mới thực hiện được công việc nghe lén trên mạng, nhưng không phải máy tính đang ở chế độ promiscuous cũng đều đang thực hiện nghe lén.

4.4.2 Kiểm thử ứng dụng web:

Trong phần này sẽ tiến hành kiểm thử trang web Thư viện tổng hợp Tp Hồ Chí Minh tại tên miền: thuvientphcm.gov.vn. Các bước tiến hành bao gồm: thu thập

thông tin và dò tìm lỗ hổng được thực thi từ phía bên ngoài vào hệ thống. Trong quá trình kiểm thử bỏ qua bước xác nhận lỗ hổng.

4.4.2.1 Thu thập thông tin:

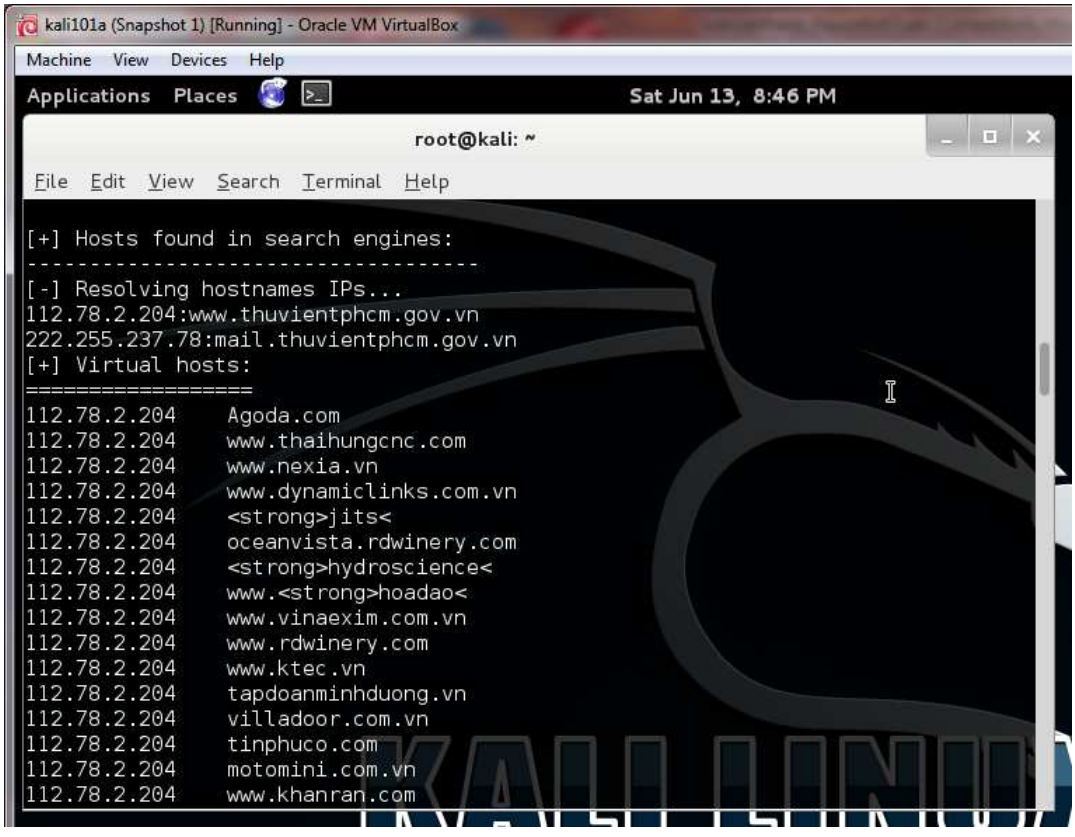
Sử dụng công cụ TheHarvester thu thập thông tin với cú pháp câu lệnh như sau:

```
theharvester -d tenmien -l 500 -b all
```

Trong đó:

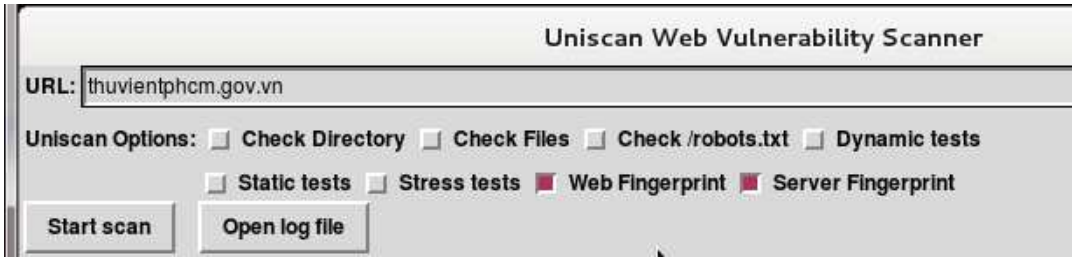
- tenmien: Tên miền cần thu thập thông tin.
- 500 : Lấy 500 kết quả đầu tiên từ kết quả trả về của các bộ máy tìm kiếm.
- all : Tìm kiếm từ tất cả các bộ máy tìm kiếm mà công cụ hỗ trợ như Google, Bing, hoặc các CSDL của Linkedin, Twitter…

Kết quả trả về như sau:



Hình 4.4: Kết quả tìm thông tin bằng TheHarvester

Cho thấy trang web này đang nằm chung máy chủ với nhiều trang web khác. Sử dụng công cụ Uniscan-gui để thu thập thông tin về máy chủ và ứng dụng web trên đó. Đây là công cụ có giao diện đồ họa, chọn thu thập thông tin ứng dụng web và máy chủ như trong hình 4.5.



Hình 4.5: Tìm thông tin máy chủ web bằng công cụ Uniscan-gui

Kết quả thu được nhiều thông tin về DNS, địa chỉ IP của máy chủ ứng dụng web, máy chủ email, phần mềm, hệ điều hành và các cổng đang mở trên máy chủ đó. Trong đó, có một số thông tin đáng chú ý như sau:

- Máy chủ sử dụng hệ điều hành Windows Server 2008 R2 Enterprise SP 1
- Hệ quản trị CSDL : Microsoft SQL Server 2008 R2 RTM sử dụng port 1433/tcp
- Hệ quản trị CSDL : MySQL 1.76-community sử dụng port 3306/tcp
- Chương trình ứng dụng web: Microsoft IIS 7.5
- Ứng dụng web: Joomla
- Các thông tin về địa chỉ IP.

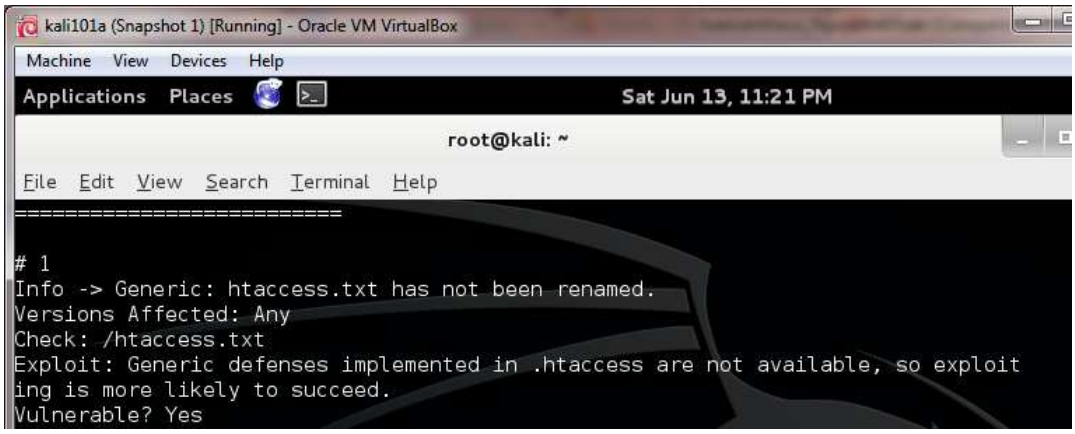
4.4.2.2 Dò tìm lỗ hổng:

Trong giai đoạn này sẽ sử dụng các công cụ sau để dò tìm lỗ hổng trên trang web của Thư viện Tp Hồ Chí Minh: Joomscan, W3af, Openvas.

Đầu tiên, chạy công cụ Joomscan với dòng lệnh như sau:

```
joomscan  -u thuvientphcm.gov.vn
```

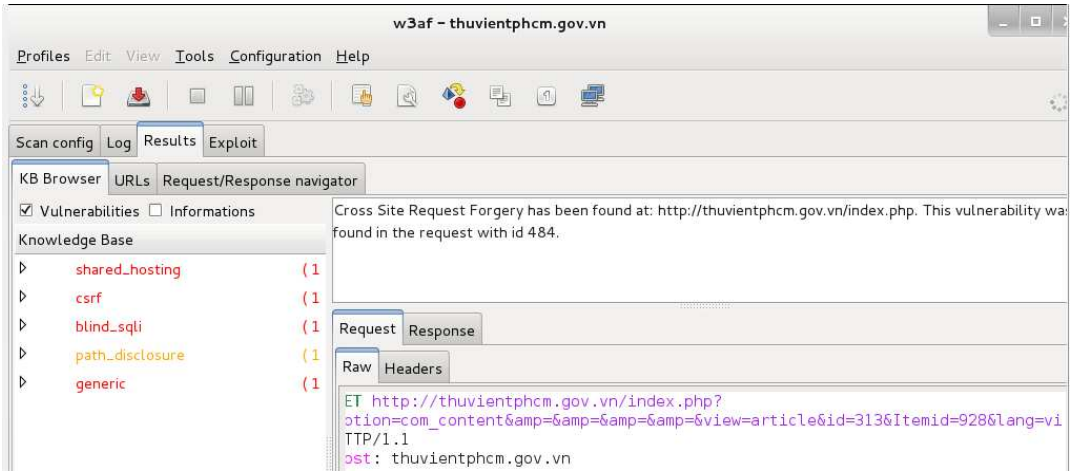
Kết quả tìm thấy 1 lỗ hổng: không đổi tên tập tin htaccess.txt và cho phép truy cập trực tiếp vào nội dung tập tin đó.



Hình 4.6: Kết quả chạy công cụ Joomscan

Tiếp theo, quét lỗ hổng với công cụ W3af, chọn cấu hình quét là OWASP_TOP 10 để phát hiện các lỗ hổng bảo mật được xếp loại theo dự án "10 mối nguy cơ về bảo mật nguy cấp nhất của ứng dụng Web" (OWASP Top Ten Project) [16].

Kết quả thu được có hai loại lỗ hổng bảo mật đáng chú ý là CRSF và Blind Sql Injection đang tồn tại trên trang web này.



Hình 4.7: Kết quả chạy công cụ W3af

Sử dụng công cụ Openvas quét tìm lỗ hổng bảo mật, thu được một số lỗ hổng bảo mật có khả năng tồn tại đáng chú ý như sau :

- DCE Services Enumeration (cổng 135/tcp): liệt kê các dịch vụ ứng dụng phân tán.

- MS Sql DrDos (cổng 1434/udp): tấn công từ chối dịch vụ hệ quản trị CSDL MS Sql.
- OpenSSL CCS MITM (cổng 25, 110, 143, 366, 465, 993, 995/tcp): cho phép người tấn công thông qua phương pháp tấn công MITM nghe lén các kết nối mã hóa giữa máy chủ và các khách có thể bắt được các kết nối mã hóa, cho phép họ giải mã, đọc hay thao tác sửa đổi dữ liệu.
- Poodle SSL 3.0 (cổng 465, 993, 995, 8443/tcp): lỗ hổng bảo mật trong giao thức mã hóa web SSL 3.0, cho phép người tấn công thông qua phương pháp tấn công MITM có thể kiểm soát các tài khoản email, mạng xã hội và tài khoản ngân hàng của nạn nhân.
- Dịch vụ SMTP đang chạy trên cổng không chuẩn (cổng 366/tcp): Có khả năng tồn tại chương trình backdoor trên máy chủ.

4.4.2.3 Đánh giá kiểm thử:

Tuy chưa thực hiện bước xác nhận lỗ hổng, nhưng thông qua kết quả chạy một số công cụ dò tìm lỗ hổng bảo mật như Joomscan, W3af và Openvas cũng đã giúp người quản trị thấy hai mối nguy cơ cần được kiểm tra ngay:

- Kiểm tra dịch vụ đang chạy trên cổng 366/tcp.
- Khóa cổng 1434/udp nhằm chống tấn công từ chối dịch vụ hệ CSDL MS Sql.

Các lỗ hổng bảo mật khác nếu có tồn tại, người tấn công cần một số điều kiện mới có thể khai thác được.

4.5 Tóm tắt nội dung chương:

Trong chương này đã đưa ra một quy trình kiểm thử chung sử dụng các công cụ trên Kali Linux để kiểm thử cho nhiều loại hệ thống. Ngoài ra, còn giới thiệu với người đọc tên một số công cụ hay sử dụng và phân chia chúng theo từng bước tiến hành của quy trình kiểm thử. Tài liệu không đi sâu vào tìm hiểu cách sử dụng các công cụ đó, thông tin này người đọc có thể tham khảo trang web: http:///www.kali.org.

KẾT LUẬN VÀ KIẾN NGHỊ

Chương 1 luận văn đã giới thiệu các kiến thức cơ bản về an toàn hệ thống thông tin và kiểm thử an toàn hệ thống thông tin. Trong chương 2 đã giới thiệu 5 tiêu chuẩn kiểm thử an toàn hệ thống thông tin nhằm đưa ra được quy trình kiểm thử ở trong chương sau như mục tiêu của đề tài hướng đến. Chương 3 nêu lên một số lỗ hổng bảo mật và cách phòng chống chúng nhằm giúp người đọc hiểu, đánh giá được mức độ nguy hiểm và định hướng được cách giải quyết vấn đề khi phát hiện lỗ hổng bảo mật đó xuất hiện trên hệ thống mình quản lý. Ở chương 4, luận văn giới thiệu bộ công cụ Kali Linux và đưa ra quy trình kiểm thử an toàn hệ thống thông tin sử dụng các công cụ có trên bộ Kali Linux đó. Các công cụ đó được phân chia theo từng bước tiến hành của quy trình kiểm thử. Nội dung trình bày bám sát theo mục tiêu đề ra ban đầu của luận văn **"Một số kỹ thuật kiểm thử an toàn hệ thống".**

1. Kết quả đạt được:

Luận văn đã đáp ứng nhiệm vụ đề ra của đề tài:

- Trình bày được các loại phương thức tấn công thông qua lỗ hổng bảo mật phổ biến, các biện pháp phòng chống, ngăn ngừa tương ứng.
- Đưa ra quy trình kiểm thử an toàn hệ thống.
- Nắm rõ và sử dụng các công cụ trên bộ công cụ Kali Linux.
- Thực nghiệm đánh giá mức độ an toàn bảo mật hệ thống: máy tính Windows trong hệ thống mạng LAN; máy chủ ứng dụng Web.

Hạn chế của luận văn là chưa giới thiệu chi tiết cách sử dụng các công cụ kiểm thử. Nhưng nhìn chung, luận văn có thể trở thành một tài liệu tham khảo tin cậy cho các nhà quản trị hệ thống nhỏ, người lập trình ứng dụng không chuyên về bảo mật nhằm giúp họ có thể tự đánh giá được hệ thống họ phụ trách quản lý, lập trình có bảo mật hay không.

2. Hướng phát triển:

Để luận văn hoàn chỉnh hơn và có thể áp dụng cho nhiều loại hệ thống, cần bổ sung thêm những nội dung sau:

- Giới thiệu thêm nhiều loại lỗ hổng bảo mật

- Hướng dẫn sử dụng các công cụ có trong Kali
- Thực nghiệm thêm nội dung kiểm thử mạng không dây.

3. Lời kết:

Từ nhu cầu thực tế của cá nhân tôi (một người không chuyên về bảo mật) là cần kiểm thử mức độ an toàn của các hệ thống tôi đang quản lý đã dẫn đến việc thực hiện luận văn này. Qua quá trình thực hiện luận văn, nhất là quá trình thực nghiệm, tôi đã hiểu biết nhiều hơn về kiểm thử an toàn hệ thống thông tin, tìm ra được những lỗ hổng bảo mật đang tồn tại trên hệ thống tôi phụ trách. Tôi tin chắc rằng luận văn sẽ giúp được nhiều người quản lý hệ thống, nhất là tại các đơn vị nhà nước, hạn chế được những thiệt hại do mất an toàn hệ thống thông tin gây ra.

TÀI LIỆU THAM KHẢO

TÀI LIỆU TIẾNG VIỆT

[1] Vũ Anh Tuấn (2009), *Giải pháp nâng cao độ an ninh thông tin trong mạng Lan không dây chuẩn IEEE 802.11i* . Luận văn (thạc sỹ), Khoa Công nghệ thông tin, Đại học Thái Nguyên.

[2] Đinh Thị Thiên Anh (2011), *Nghiên cứu kiểm thử bảo mật website*. Luận văn (thạc sỹ), Khoa Công nghệ thông tin, Đại học Đà nẵng.

[3] Lê Ngọc Thức (2012), *Xây dựng công cụ đánh giá an toàn website*. Luận văn (thạc sỹ), Khoa Công nghệ thông tin, Đại học Lạc Hồng.

[4] Nguyễn Văn Thịnh (2013), *Nghiên cứu phân tích một số phương pháp tấn công điển hình trên mạng máy tính và phương pháp ngăn chặn*, Luận văn (thạc sỹ), Khoa Công nghệ thông tin, Học viện công nghệ bưu chính viễn thông.

[5] BKAV (2014). 40% website Việt Nam tồn tại lỗ hổng [online], viewed 12 March 2015, from:< http://www.bkav.com.vn/tieu-diem/-/view_content/content/224719/40-website-viet-nam-ton-tai-lo-hong >.

[6] Nguyen Cuong (2013). Tổng quan về an toàn hệ thống thông tin [online], viewed 12 March 2015, from:< http://voer.edu.vn/m/tong-quan-ve-an-toan-he-thong-thong-tin/b728064d >.

[7] SecurityDaily.NET (2015). Hiểu an ninh thông tin và các khái niệm liên quan [online], viewed 12 March 2015, from:< http://securitydaily.net/hieu-an-ninh-thong-tin-va-cac-khai-niem-lien-quan/ >.

[8] SecurityDaily.NET (2014). Các kiểu khai thác XSS – Phần 1: Reflected XSS [online], viewed 18 March 2015, from:< http://securitydaily.net/ky-thuat-khai-thac-xss-phan-1-reflected-xss/ >.

[9] SecurityDaily.NET (2014). Các kiểu khai thác XSS – Phần 3: DOM base XSS [online], viewed 18 March 2015, from:< http://securitydaily.net/cac-kieu-khai-thac-xss-phan-3-dom-based-xss/ >.

[10] SecurityDaily.NET (2014). Các kiểu khai thác XSS – Phần 2: Stored XSS [online], viewed 18 March 2015, from:< http://securitydaily.net/cac-kieu-khai-thac-xss-phan-2-stored-xss/ >.

TÀI LIỆU TIẾNG ANH

[11] Glenford J. Myers (2011). *Microsoft The art of software testing – third edition.* John Wiley & Sons, USA.

[12] Microsoft (2013). *Microsoft Security Intelligence Report - Volume 16.* Microsoft, USA.

[13] Microsoft (2014). *Microsoft Security Intelligence Report - Volume 17.* Microsoft, USA.

[14] Matt Bishop (2004), *Introduction to Computer Security.* Addison-Wesley, USA.

[15] NIST - National Institute of Standards and Technology (2011), *Technical Guide to Information Security Testing and Assessment.* National Institute of Standards and Technology (NIST), Gaithersburg.

[16] The OWASP Foundation (2013), *OWASP Top 10 – 2013 – The ten most critical web application security risks.* The OWASP Foundation, USA.

[17] Microsoft (2006). How to prevent cross-site scripting security issues [online], viewed 19 March 2015, from:< https://support.microsoft.com/en-us/kb/252985/en-us >.

[18] Chris Sanders (2010). Understanding Man-in-the-Middle Attacks – ARP Cache Poisoning [online], viewed 20 March 2015, from:< http://www.windowsecurity.com/articles-tutorials/authentication_and_encryption /Understanding-Man-in-the-Middle-Attacks-ARP-Part1.html>.

[19] Lukas Lueg (2010). The Pyrit open source project [online], viewed 2 April 2015, from <https://code.google.com/p/pyrit/>.

[20] Microsoft (2012). Vulnerabilities in Remote Desktop Could Allow Remote Code Execution (2671387) [online], viewed 25 March 2015, from:

< https://technet.microsoft.com/library/security/ms12-020>.