**MỤC LỤC**

[**Lời mở đầu 2**](#_Toc8892220)

[**CHƯƠNG 1. TỔNG QUAN VỀ WEBSITE, CÁC DỊCH VỤ CỦA WEBSITE VÀ LỖI BẢO MẬT THÔNG DỤNG 3**](#_Toc8892221)

[**1.1.Mô tả Website và cách hoạt động 3**](#_Toc8892222)

[**1.2.Các dịch vụ và ứng dụng trên nền web 4**](#_Toc8892223)

[**CHƯƠNG 2. CÁC LOẠI TẤN CÔNG VÀ BẢO MẬT ỨNG DỤNG WEB PHỔ BIẾN 5**](#_Toc8892224)

[**2.1. Local Attack 5**](#_Toc8892225)

[**2.1.1. Tìm hiểu về Local Attack 5**](#_Toc8892226)

[**2.1.2. Tiến hành Attack 5**](#_Toc8892227)

[**2.1.3. Cách bảo mật cho Local Attack 8**](#_Toc8892228)

[**2.2. Tấn công từ chối dịch vụ - (Denial Of Service) 11**](#_Toc8892230)

[**2.2.1. DOS(Denial Of Service) 11**](#_Toc8892231)

[**2.2.2. Ddos(Distributed Denial of Service) 14**](#_Toc8892232)

[**2.2.3. Tấn công từ chối dịch vụ phản xạ nhiều vùng DRDoS (Distributed Reflection Denial of Service) 20**](#_Toc8892233)

[**2.3. SQL Injection 21**](#_Toc8892234)

[**2.3.1. Tấn công SQL injection 21**](#_Toc8892235)

[**2.3.2.Cách Phòng Tránh SQL Injection 28**](#_Toc8892236)

[**CHƯƠNG 3. DEMO, ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN ĐỀ TÀI 33**](#_Toc8892237)

[**3.1. Demo 33**](#_Toc8892238)

[**3.2. Kết luận 34**](#_Toc8892239)

[**3.2.1. Các vấn đề đạt được 34**](#_Toc8892240)

**Phân công công việc…………………………………………………………..35**

# **Lời mở đầu**

Cùng với sự phát triển của công nghệ thông tin, công nghệ mạng máy tính và sự phát triển của mạng internet ngày càng phát triển đa dạng và phong phú. Các dịch vụ trên mạng đã thâm nhập vào hầu hết các lĩnh vực trong đời sống xã hội. Các thông tin trên Internet cũng đa dạng về nội dung và hình thức, trong đó có rất nhiều thông tin cần được bảo mật cao hơn bởi tính kinh tế, tính chính xác và tính tin cậy của nó. Bên cạnh đó, các hình thức phá hoại mạng cũng trở nên tinh vi và phức tạp hơn. Do đó đối với mỗi hệ thống, nhiệm vụ bảo mật được đặt ra cho người quản trị mạng là hết sức quan trọng và cần thiết. Xuất phát từ những thực tế đó, chúng ta sẽ tìm hiểu về các cách tấn công phổ biến nhất hiện nay và các phòng chống các loại tấn công này. Chính vì vậy, thông qua việc nghiên cứu một số phương pháp tấn công và cách bảo mật các lọa tấn công này, nhóm mong muốn góp một phần nhỏ vào việc nghiên cứu và tìm hiểu về các vấn đề an ninh mạng giúp cho việc học tập và nghiên cứu.

**1 .Lý do chọn đề tài**

Trong những năm gần đây, Việt Nam ngày càng phát triển và nhất là về mặt công nghệ thông tin. Đặc biệt là về ứng dụng web, hầu như mọi người ai cũng từng nghe và làm việc trên ứng dụng web. Website trở nên phổ biến và trở thành một phần quan trọng của mọi người và nhất là các doanh nghiệp, công ty. Bên cạnh đó lý do an toàn bảo mật cho ứng dụng web luôn là vấn đề nan giải của mọi người.Vì vậy chúng ta sẽ đi tìm hiểu ứng dụng web và cách thức tấn công và bảo mật web.

**2. Mục tiêu**

Giúp chúng ta có thể hiểu hơn về các ứng dụng website, các mối đe dọa về vấn đề an toàn thông tin khi chúng ta làm việc trên ứng dụng web hàng ngày, hiểu rõ hơn về các kỹ thuật tấn công và bảo mật web.

**3. Phạm vi**

Tìm hiểu các kỹ thuật tấn công phổ biến nhất hiện nay như SQL Injection, Denial Of Service, Local Attack,…Cách bảo mật, phòng thủ các loại tấn công phổ biến trên một cách tổng quan nhất.

# **CHƯƠNG 1. TỔNG QUAN VỀ WEBSITE, CÁC DỊCH VỤ CỦA WEBSITE VÀ LỖI BẢO MẬT THÔNG DỤNG**

* 1. **Mô tả Website và cách hoạt động**

Website là một “trang web” trên mạng Internet, đây là nơi giới thiệu những thông tin, hình ảnh về doanh nghiệp và sản phẩm, dịch vụ của doanh nghiệp (hay giới thiệu bất cứ thông tin gì) để khách hàng có thể truy cập ở bất kỳ nơi đâu, bất cứ lúc nào.

Website là tập hợp nhiều trang (web page). Khi doanh nghiệp xây dựng website nghĩa là đang xây dựng nhiều trang thông tin, catalog sản phẩm, dịch vụ....Để tạo nên một website cần phải có 3 yếu tố cơ bản:

- Cần phải có tên miền (domain).

- Nơi lưu trữ website (hosting).

- Nội dung các trang thông tin (web page).

* **Một số thuật ngữ cơ bản:**

Website động (Dynamic website) là website có cơ sở dữ liệu, được cung cấp công cụ quản lý website (Admin Tool). Đặc điểm của website động là tính linh hoạt và có thể cập nhật thông tin thường xuyên, quản lý các thành phần trên website dễ dàng.

Loại website này thường được viết bằng các ngôn ngữ lập trình như PHP, Asp.net, JSP, Perl,..., quản trị Cơ sở dữ liệu bằng SQL hoặc MySQL...

Website tĩnh do lập trình bằng ngôn ngữ HTML theo từng trang như brochure, không có cơ sở dữ liệu và không có công cụ quản lý thông tin trên website. Thông thường website tĩnh được thiết kế bằng các phần mềm như FrontPage, Dreamwaver,...

Đặc điểm của website tĩnh là ít thay đổi nội dung, sự thay đổi nội dung này thường liên quan đến sự thay đổi các văn bản đi kèm thể hiện nội dung trên đó.

Hiện nay, hầu hết các doanh nghiệp đều sử dụng website động, thế hệ công nghệ website được mọi người biết đến là web 2.0.

- Tên miền (domain): Tên miền chính là địa chỉ website, trên internet chỉ tồn tại duy nhất một địa chỉ (tức là tồn tại duy nhất một tên miền). Có 2 loại tên miền:

- Tên miền Quốc tế: là tên miền có dạng .com; .net; .org; .biz; .name ...

- Tên miền Việt Nam: là tên miền có dạng .vn; .com.vn; .net.vn; org.vn; .gov.vn;...

- Lưu trữ website: Dữ liệu thông tin của website phải được lưu trữ trên một máy tính (máy chủ - server) luôn hoạt động và kết nối với mạng Internet. Một server có thể lưu trữ nhiều website, nếu server này bị sự cố chẳng hạn tắt trong một thời điểm nào đó thì không ai có thể truy cập được những website lưu trữ trên server tại thời điểm bị sự cố.

- Tùy theo nhu cầu lưu trữ thông tin mà doanh nghiệp có thể thuê dung lượng thích hợp cho website (thuê dung lượng host).

- Dung lượng host: Là nơi để lưu cơ sở trữ dữ liệu của website (hình ảnh, thông tin…), đơn vị đo dung lượng thường là Mb hoặc Gb.

- Băng thông hay dung lượng đường truyền truyền: Là tổng số Mb dữ liệu tải lên máy chủ hoặc tải về từ máy chủ (download, upload) nơi đặt website, đơn vị đo thông thường là Mb/Tháng.

* 1. **Các dịch vụ và ứng dụng trên nền web**

Với công nghệ hiện nay, website không chỉ đơn giản là một trang tin cung cấp các tin bài đơn giản. Những ứng dụng viết trên nền web không chỉ được gọi là một phần của website nữa, giờ đây chúng được gọi là phần mềm viết trên nền web.

Có rất nhiều phần mềm chạy trên nền web như Google word (xử lý văn bản), Google spreadsheets (xử lý bảng tính), Email ,…

*Một số ưu điểm của phần mềm hay ứng dụng chạy trên nền web:*

- Mọi người đều có trình duyệt và bạn chỉ cần trình duyệt để chạy phần mềm.

- Phần mềm luôn luôn được cập nhật vì chúng chạy trên server.

- Luôn sẵn sàng 24/7.

- Dễ dàng backup dữ liệu thường xuyên.

- Có thể truy cập mọi lúc, mọi nơi, miễn là bạn có mạng.

- Chi phí triển khai cực rẻ so với phần mềm chạy trên desktop.

Hãy hình dung bạn có một phần mềm quản lý bán hàng hay quản lý công việc ở công ty. Không phải lúc nào bạn cũng ở công ty, với phần mềm viết trên nền web, bạn có thể vào kiểm tra, điều hành ở bất cứ đâu, thậm chí bạn chỉ cần một chiếc điện thoại chạy được trình duyệt như IPhone mà không cần đến một chiếc máy tính.

# **CHƯƠNG 2. CÁC LOẠI TẤN CÔNG VÀ BẢO MẬT ỨNG DỤNG WEB PHỔ BIẾN**

## **2.1. Local Attack**

### **2.1.1. Tìm hiểu về Local Attack**

- Local attack là một trong những kiểu hack rất phổ biến và không được khuyên. Đối một web server thông thường khi bạn đăng ký một tài khoản trên server nào đó bạn sẽ được cấp một tài khoản trên server đó và một thư mục để quản lý site của mình. Ví dụ : tenserver/tentaikhoancuaban. Và như vậy cũng có một tài khoản của người dùng khác tương tự như : tenserver/taikhoan1.Giả sử tài khoản 1 bị hacker chiếm được thì hacker có thể dùng các thủ thuật,các đoạn scrip,các đoạn mã lệnh để truy cập sang thư mục chứa site của bạn là tenserver/taikhoancuaban. Và cũng theo cách này hacker có thể tấn công sang các site của người dùng khác và có thể lấy thông tin admin, database, các thông tin bảo mật khác hoặc chèn các đoạn mã độc vào trang index của site bạn. Dạng tấn công trên gọi là Local Attack.

- Thông thường nhất, Local Attack được sử dụng để đọc lấy thông tin config từ victim, sau đó dựa vào thông tin ở config và mục đích của hacker để phá hoại website.

### **2.1.2. Tiến hành Attack**

- Sau khi chúng ta chuản bị xong, tức là đã upload được con shell lên 1 server nào đó. Chúng ta bắt đầu tìm các website cùng server mà bạn đã up shell lên, thông thường các hacker thường sử dụng Reverse Ip domain mà hacker đã upload shell để xem các website cùng server.

- Sau khi tìm được danh sách website ,lần lượt check xem site nào bị lỗi và có thể local sang được.

- Các lệnh thường dùng trong shell để Local Attack

*Xem tên domain trên cùng 1 host*

ls -la /etc/valiases

cd /etc/vdomainaliases;ls –lia

- Trường hợp đặc biệt khi không thể xem user nằm cùng host thì ta thêm && vào

cd /etc/vdomainaliases && ls –lia

- Muốn biết tên user thì dùng lệnh :

cat /etc/passwd/

Hoặc

less /etc/passwd

+ local sang victim, tức là local sang site khác

ví dụ hiện tại con shell chúng ta đang ở :

/home/abcd/public\_html/

thì chúng ta sẽ local sang như sau :

dir home/tên user cần local/public\_html

- Muốn biết tên user cần local sang thì chúng ta sử dụng Reverse Ip để lấy danh sách user trên cùng một server.Muốn biết user đó có tồn tại hay không chúng ta mở trình duyệt web lên và đánh đoạn : Ip của server/~ tên user (Ví dụ : 203.166.222.121/~doanchuyennganh). Nếu trình duyệt hiện lên trang index của website thì tức là user đó tồn tại.

+ Xem nội dung của file

cat /home/tên user cần local/public\_html/index.php

Hoặc

Chúng ta muốn xem config của 1 forum thì dùng

ln -s /home/tên user cần local/public\_html/forum/includes/config.php

doanchuyennganh.txt Với doanchuyennganh.txt ở đây là file chúng ta tạo ra trên host của chúng ta để xem file của người khác ! Nếu không sử dụng được các lệnh trên tức là server đã disable chức năng đó.

Thêm 1 số lệnh shell trong linux :

- pwd: đưa ra ngoài màn hình thư mục đang hoạt động (ví dụ: /etc/ssh).

- cd: thay đổi thư mục (ví dụ: cd .. – ra một cấp thư mục hiện tại; cd vidu – vào thư mục /vidu).

- ls: đưa ra danh sách nội dung thư mục.

- mkdir: tạo thư mục mới (mkdir tên\_thumuc).

- touch: tạo file mới (touch ten\_file).

- rmdir: bỏ một thư mục (rmdir ten\_thumuc).

- cp: copy file hoặc thư mục (cp file\_nguồn file\_đích).

- mv: di chuyển file hoặc thư mục; cũng được dùng để đặt lại tên file hoặc thư mục (mv vị\_trí\_cũ vị\_trí\_mới hoặc mv tên\_cũ tên\_mới).

- rm: loại bỏ file (rm tên\_file).

- Để tìm kiếm file, bạn có thể dùng: - find : dùng cho các tên file. - grep <>: để tìm nội dung trong file.

Để xem một file, bạn có thể dùng:

- more : hiển thị file theo từng trang.

- cat <>: hiển thị tất cả file.

- Nếu muốn kết nối tới một host từ xa, sử dụng lệnh ssh. Cú pháp là ssh <tên\_host>.

Quản lý hệ thống:

- ps: hiển thị các chương trình hiện thời đang chạy (rất hữu ích: ps là cái nhìn toàn bộ về tất cả các chương trình).

- Trong danh sách đưa ra khi thực hiện lệnh ps, bạn sẽ thấy có số PID (Processidentification - nhân dạng tiến trình). Con số này sẽ được hỏi đến khi muốn ngừng một dịch vụ hay ứng dụng, dùng lệnh kill

- top: hoạt động khá giống như Task Manager trong Windows. Nó đưa ra thông tin về tất cả tài nguyên hệ thống, các tiến trình đang chạy, tốc độ load trung bình… Lệnh top

- d <delay> thiết lập khoảng thời gian làm tươi lại hệ thống. Bạn có thể đặt bất kỳ giá trị nào, từ .1 (tức 10 mili giây) tới 100 (tức 100 giây) hoặc thậm chí lớn hơn.

- uptime: thể hiện thời gian của hệ thống và tốc độ load trung bình trong khoảng thời gian đó, trước đây là 5 phút và 15 phút.

Thông thường tốc độ load trung bình được tính toán theo phần trăm tài nguyên hệ thống (vi xử lý, RAM, ổ cứng vào/ra, tốc độ load mạng) được dùng tại một thời điểm. Nếu tốc độ được tính toán là 0.37, tức có 37% tài nguyên được sử dụng. Giá trị lớn hơn như 2.35 nghĩa là hệ thống phải đợi một số dữ liệu, khi đó nó sẽ tính toán nhanh hơn 235% mà không gặp phải vấn đề gì. Nhưng giữa các phân phối có thể khác nhau một chút.

- free: hiển thị thông tin trên bộ nhớ hệ thống.

- ifconfig <tên\_giao\_diện>: để xem thông tin chi tiết về các giao diện mạng; thông thường giao diện mạng ethernet có tên là eth(). Bạn có thể cài đặt các thiết lập mạng như địa chỉ IP hoặc bằng cách dùng lệnh này (xem man ifconfig). Nếu có điều gì đó chưa chính xác, bạn có thể stop hoặc start (tức ngừng hoặc khởi\_động) giao diện bằng cách dùng lệnh ifconfig <tên\_giao\_diện> up/down.

- passwd: cho phép bạn thay đổi mật khẩu (passwd người\_dùng\_sở\_hữu\_mật\_khẩu hoặc tên người dùng khác nếu bạn đăng nhập hệ thống với vai trò root).

- useradd: cho phép bạn thêm người dùng mới (xem man useradd). Dù ở phân phối nào, bạn cũng có thể dùng phím TAB để tự động hoàn chỉnh một lệnh hoặc tên file. Điều này rất hữu ích khi bạn quen với các lệnh. Bạn cũng có thể sử dụng các phím lên, xuống để cuộn xem các lệnh đã nhập. Bạn có thể dùng lệnh đa dòng trên một dòng. Ví dụ như, nếu muốn tạo ba thư mục chỉ trên một dòng, cú pháp có thể là: mkdir thư\_mục\_1 ; mkdir thư\_mục\_2 ; mkdir thư\_mục\_3.

Một điều thú vị khác nữa là các lệnh dạng pipe. Bạn có thể xuất một lệnh thông qua lệnh khác. Ví dụ: man mkdir | tail sẽ đưa ra thông tin các dòng cuối cùng trong trang xem "thủ công" của lệnh mkdir.

Nếu lúc nào đó được yêu cầu phải đăng nhập với tài khoản gốc (tức "siêu" admin của hệ thống), bạn có thể đăng nhập tạm thời bằng cách dùng lệnh su. Tham số -1 (su-1) dùng để thay đổi thư mục chủ và cho các lệnh đã hoặc đang dùng. Chú ý là bạn cũng sẽ được nhắc một mật khẩu. Để thoát hay đóng : gõ exit hoặc logout.

### **2.1.3. Cách bảo mật cho Local Attack**

Để hạn chế Local Attack, chúng ta nên Chmod filemanager ,di chuyển file

config.php và sửa đổi file htaccess và nhất là thường xuyên backup dữ liệu.

-Chmod File Manager:

+ CHMOD thư mục Public\_html thành 710 thay vì 750 mặc định việc này sẽ giúp bạn bảo vệ được cấu trúc Website của mình.

+ CHMOD tiếp các thư mục con , CHMOD thư mục thành 701, rồi CHMOD tiếp các thư mục con trong thư mục web thành 701.

+ CHMOD toàn bộ file thành 404.

Với CHMOD chắc chắn khi run shell sẽ hiện ra thông báo lỗi:

Not Acceptable An appropriate representation of the requested resource/test.php could not be found on this server. Additionally, a 404 Not Found error was encountered while trying to use an ErrorDocument to handle the request. Attacker sẽ không view được.

- Ngoài ra , một số site thì bạn truy cập bằng subdomain của nó mà không là dạng doanchuyenganh.com/diendan , cái này có nhiều ý nghĩa, nhưng trong bảo mật thì nó sẽ rất khác.

+ CHMOD thư mục là 701 và cố gắng đừng bao giờ CHMOD 777, có một số folder ko quan trọng, bạn có thể CHMOD 755 để có thể hiện thị đúng và đầy đủ một số nội dung trong Folder đó. Chú ý thế này, một số Server hỗ trợ CHMOD thư mục được 101, nếu Server của bạn hỗ trợ cái này thì hãy sử dụng nó, vì biện pháp CHMOD này rất an toàn, đến ngay cả Owner cũng ko thể xem được cấu trúc Folder ngay cả khi vào FTP. Hiện chỉ có Server của Eshockhost.net là hỗ trợ cái này.

+ CHMOD File là 604 và đừng bao giờ để là 666 nếu có việc cần 666 thì chúng ta CHMOD tạm để sử dụng lúc đó, sau đó hãy CHMOD lại ngay. Đối với các Server hỗ trợ CHMOD file 404 chúng ta hãy CHMOD như vậy, ví dụ Server Eshockhost.net

- Thay đổi cấu trúc, tên file mặc định có chứa các thông tin quan trọng . Nếu có thể hãy thay đổi cả cấu trúc CSDL nếu bạn làm được .

-Chống local bằng cách bật safe-mode (dành cho root):

Như chúng ta đã biết, đối với các webshell - PHP, trong PHP Configuration có những option để hạn chế tính năng của nó (đặc biệt là r57 - tự động by pass) nên công việc đầu tiên của các root account là phải cập nhật các phiên bản PHP mới nhất và config lại php.ini : [i]PHP safe mode là phương pháp để giải quyết vấn đề bảo mật cho nhưng nơi server chia sẻ hosting cho nhiều accounts (shared-server). Nó là do thiết kế 1 cách sai lạc của từng cấp PHP. Hiện nay, nhiều người đã chọn phương pháp bật safe-mode để bảo mật, đặc biệt là các ISP

- Các hướng dẫn về cấu hình Security and Safe Mode :

Code:

safe\_mode: mặc định : "0" sửa dưới phân quyền : PHP\_INI\_SYSTEM

safe\_mode\_gid: mặc định :"0"sửa dưới phân quyền : PHP\_INI\_SYSTEM

safe\_mode\_include\_dir: mặc định :NULL sửa dưới phân quyền :PHP\_INI\_SYSTEM

safe\_mode\_exec\_dir: mặc định :""sửa dưới PHP\_INI\_SYSTEM

safe\_mode\_allowed\_env\_vars: mặc định :"PHP\_"sửa dưới PHP\_INI\_SYSTEM

safe\_mode\_protected\_env\_vars: mặc định :"LD\_LIBRARY\_PATH"sửa dưới

PHP\_INI\_SYSTEM

open\_basedir: mặc định :NULL sửa dưới PHP\_INI\_SYSTEM

disable\_functions: mặc định :"" sửa dưới php.ini

disable\_classes : mặc định : ""sửa dưới php.ini

- Sau đây là cách đề đặc chính cấu hình server để bật chế độ safe mode :

Trong file php.ini :

safe\_mode = Off chuyển thành safe\_mode = On

- disabled\_functions nên chứa những function sau :

PHP Code:

readfile, system, exec, shell\_exec, passthru, pcntl\_exec, putenv, proc\_close,proc\_get\_status, proc\_nice, proc\_open, proc\_terminate, popen, pclose, set\_time\_limit,escapeshellcmd, escapeshellarg, dl, curl\_exec, parse\_ini\_file, show\_source,ini\_alter, virtual, openlog

- Khi đó, ta ví dụ :

PHP Code: -rw-rw-r-- 1 doanchuyennganh doanchuyennganh 33 Jul 1 19:20 script.php

-rw-r--r-- 1 root root 1116 May 26 18:01 /etc/passwd

- Trong script.php là :

PHP Code:

<?php

readfile('/etc/passwd');

?> - Kết quả :

PHP Code:

Warning: readfile() has been disabled for security reasons in /docroot/script.php on

line 2

- Vài lợi điểm của việc bật safe mode:

- Thường khi upload file, file sẽ vào /tmp/ với những người có quyền không phải là owner.

- Bật safe-mode sẽ có những bất lợi với người lập trình code PHP, do đó, họ thường có:

PHP Code:

<?php

// Kiểm tra safe mode

if( ini\_get('safe\_mode') ){

// Code theo bật safe\_mode

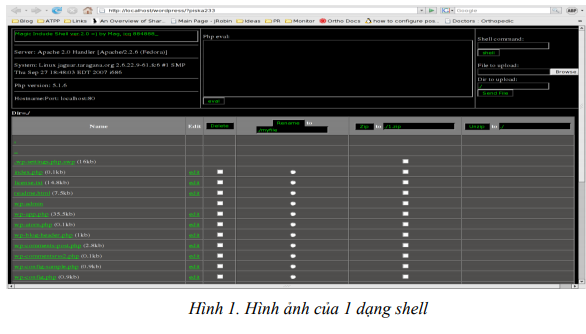
}else{

// Code theo tắt safe\_mode

}

### ?>**2.1.4. Các công cụ hỗ trợ**

-Công cụ hỗ trợ Local Attack phổ biến và hay dùng nhất là các con shell.Các loại shell thường sử dụng là R57,C99,..



## **2.2. Tấn công từ chối dịch vụ - (Denial Of Service)**

### **2.2.1. DOS(Denial Of Service)**

**2.2.1.1. Giới thiệu khái quát về DoS:**

DoS (Denial of Service) có thể mô tả như hành động ngăn cản những người dùng hợp pháp của một dịch vụ nào đó truy cập và sử dụng dịch vụ đó. Nó bao gồm cả việc làm tràn ngập mạng, làm mất kết nối với dịch vụ… mà mục đích cuối cùng là làm cho server không thể đáp ứng được các yêu cầu sử dụng dịch vụ từ các client. DoS có thể làm ngưng hoạt động của một máy tính, một mạng nội bộ, thậm chí cả một hệ thống mạng rất lớn. Thực chất của DoS là kẻ tấn công sẽ chiếm dụng một lượng lớn tài nguyên mạng như băng thông, bộ nhớ… và làm mất khả năng xử lý các yêu cầu dịch vụ đến từ các client khác.

**2.2.1.2. Các cách thức tấn công:**

+ Phá hoại dựa trên tính giới hạn hoặc không thể phục hồi của tài nguyên mạng.

- Thông qua kết nối:

Tấn công kiểu SYN flood:

￹FPRIVATE "TYPE=PICT;ALT="

Lợi dụng các thức hoạt động của kết nối TCP/IP, hacker bắt đầu quá trình thiết lập một kết nối TPC/IP với mục tiêu muốn tấn công nhưng sẽ phá vỡ kết nối ngay sau khi quá trình SYN và SYN ACK hoàn tất, khiến cho mục tiêu rơi vào trạng thái chờ (đợi gói tin ACK từ phía yêu cầu thiết lập kết nối) và liên tục gửi gói tin SYN ACK để thiết lập kết nối . Một cách khác là giả mạo địa chỉ IP nguồn của gói tin yêu cầu thiết lập kết nối SYN và cũng như trường hợp trên, máy tính đích cũng rơi vào trạng thái chờ vì các gói tin SYN ACK không thể đi đến đích do địa chỉ IP nguồn là không có thật. Cách thức này có thể được các hacker áp dụng để tấn công một hệ thống mạng có băng thông lớn hơn hệ thống của hacker.

- Lợi dụng nguồn tài nguyên của chính nạn nhân để tấn công:

Tấn công kiểu Land Attack: cũng tương tự như SYN flood nhưng hacker sử dụng chính IP của mục tiêu cần tấn công để dùng làm địa chỉ IP nguồn trong gói tin, đẩy mục tiêu vào một vòng lặp vô tận khi cố gắng thiết lập kết nối với chính nó.

Tấn công kiểu UDP flood: hacker gửi gói tin UDP echo với địa chỉ IP nguồn là cổng loopback của chính mục tiêu cần tấn công hoặc của một máy tính trong cùng mạng với mục tiêu qua cổng UDP echo (port 7) để thiết lập việc gửi và nhận các gói tin echo trên 2 máy tính (hoặc giữa mục tiêu với chính nó nếu mục tiêu có cấu hình cổng loopback) khiến cho 2 máy tính này dần dần sử dụng hết băng thông của chúng và cản trở hoạt động chia sẻ tài nguyên mạng của các máy tính khác trong mạng.

-Sử dụng băng thông:

Tấn công kiểu DDoS (Distributed Denial of Service): đây là cách thức tấn công rất nguy hiểm. Hacker xâm nhập vào các hệ thống máy tính, cài đặt các chương trình điều kiển từ xa và sẽ kích hoạt đồng thời các chương trình này vào cùng một thời điểm để đồng loạt tấn công vào một mục tiêu. Cách thức này có thể huy động tới hàng trăm thậm chí hàng ngàn máy tính cùng tham gia tấn công một lúc (tùy vào sự chuẩn bị trước đó của hacher) và có thể ngốn hết băng thông của mục tiêu trong nháy mắt.

-Sử dụng các nguồn tài nguyên khác:

Kẻ tấn công lợi dụng các nguồn tài nguyên mà nạn nhân cần đến để tấn công. Những kẻ tấn công có thể thay đổi dữ liệu và tự sao chép dữ liệu mà nạn nhân cần lên nhiều lần làm CPU bị quá tải và các quá trình xử lý dữ liệu bị đình trệ.

- Tấn công kiểu Smurf Attack: kiểu tấn công này cần một hệ thống rất quan trọng, đó là mạng khuyếch đại. Hacker dùng địa chỉ của máy tính cần tấn công gửi broadcast gói tin ICMP echo cho toàn bộ mạng. Các máy tính trong mạng sẽ đồng loạt gửi gói tin ICMP reply cho máy tính mà hacker muốn tấn công. Kết quả là máy tính này sẽ không thể xử lý kịp thời một lượng lớn thông tin như vậy và rất dễ bị treo.

- Tấn công kiểu Tear Drop: trong mạng chuyển mạch gói, dữ liệu được chia nhỏ làm nhiều gói tin, mối gói tin có một giá trị offset riêng và có thể truyền đi theo nhiều đường để tới đích. Tại đích, nhờ vào giá trị offset của từng gói tin mà dữ liệu lại được kết hợp lại như ban đầu. Lợi dụng điều này, hacker có thể tạo ra nhiều gói tin có giá trị offset trùng lặp nhau gửi đến mục tiêu muốn tấn công. Kết quả là máy tính đích không thể sắp xếp được những gói tin này và có thể bị treo do đã dùng hết năng lực xử lý của hệ thống.

+.Phá hoại hoặc chỉnh sửa thông tin cấu hình.

Lợi dụng việc cấu hình thiếu an toàn (ví dụ như việc không xác thực thông tin trong việc gửi và nhận bản tin update của các router) mà kẻ tấn công sẽ thay đổi từ xa hoặc trực tiếp các thông tin quan trọng khiến cho những người dùng hợp pháp không thể sử dụng dịch vụ. Ví dụ: hacker có thể xâm nhập vào DNS để thay đôi thông tin, dẫn đến quá trình biên dịch domain name sang IP của DNS bị sai lệch. Kết quả là các yêu cầu của client đến một domain nào đó sẽ biến thành một domain khác.

+.Phá hoại hoặc chỉnh sửa vật lý phần cứng.

Lợi dụng quyền hạn của chính bản thân kẻ tấn công đối với các thiết bị trong hệ thống mạng để tiếp cận phá hoại (các router, switch…)

**2.2.1.3 Các cách phòng chống**

- DoS có thể làm tiêu tốn rất nhiều thời gian cũng như tiền bạc, vì vậy, cần phải có những biện pháp để phòng chống:

- Mô hình hệ thống phải được xây dựng hợp lý, tránh phụ thuộc lẫn nhau quá mức dễ dẫn đến một bộ phận gặp sự cố sẽ làm cả hệ thống bị trục trặc.

- Thiết lập password bảo vệ các thiết bị hay các nguồn tài nguyên quan trọng.

- Thiết lập các mức xác thực đối với người dùng cũng như các nguồn tin trên mạng (các thông tin cập nhật định tuyến giữa các router cũng nên thiết lập ở chế độ xác thực)

- Xây dựng hệ thống lọc thông tin trên router, firewall… và hệ thống bảo vệ chống lại SYN flood.

- Chỉ chấp nhận các dịch vụ cần thiết, tạm thời dừng các dịch vụ chưa có yêu cầu cung cấp hoặc không sử dụng.

- Xây dựng hệ thống định mức, giới hạn cho người sử dụng để ngăn ngừa trường hợp người dùng có ác ý muốn lợi dụng các tài nguyên trên server để tấn công chính server hay mạng, server khác.

- Liên tục cập nhật, nghiên cứu, kiểm tra để phát hiện các lỗ hổng bảo mật và có biện pháp khắc phục kịp thời.

- Sử dụng các biện pháp kiểm tra hoạt động của hệ thống một cách liên tục để phát hiện ngay những hành động bất bình thường.

- Xây dựng hệ thống dự phòng.

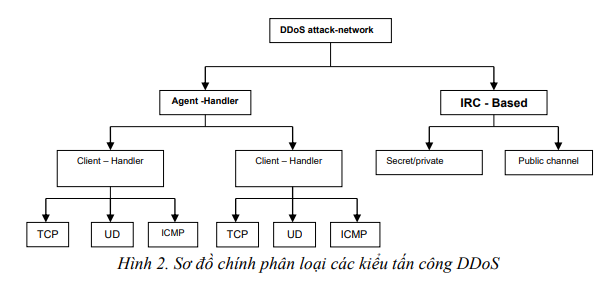
### **2.2.2. Ddos(Distributed Denial of Service)**

- Distributed Denial Of Service (DDoS) là kỹ thuật tấn công làm các ISP lo âu, giới hacker chính thống thì không công nhận DdoS là kỹ thuật tấn công chính thống. Thế nhưng Black hat đang có rất nhiều ưu thế khi triển khai tấn công bằng kỹ thuật DdoS. Việc phòng ngừa và ngăn chặn DdoS vẫn còn đang thực hiện ở mức độ khắc phục hậu quả và truy tìm thủ phạm.

**2.2.2.1. Kiến trúc tổng quan của DDoS attack-network**

Nhìn chung DDoS attack-network có hai mô hình chính:

* Mô hình Agent – Handler
* Mô hình IRC – Based



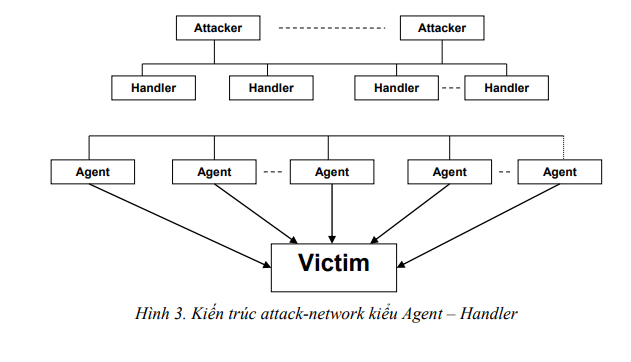
i. Mô hình Agent – Handler:

*Theo mô hình này, attack-network gồm 3 thành phần: Agent, Client và Handler*

- Client : là software cơ sở để hacker điều khiển mọi hoạt động của attack-network

- Handler : là một thành phần software trung gian giữa Agent và Client

- Agent : là thành phần software thực hiện sự tấn công mục tiêu, nhận điều khiển từClient thông qua các Handler.



- Attacker sẽ từ Client giao tiếp với cc1 Handler để xác định số lượng Agent đang online, điều chỉnh thời điểm tấn công và cập nhật các Agent. Tùy theo cách attacker cấu hình attack-network, các Agent sẽ chịu sự quản lý của một hay nhiều Handler.

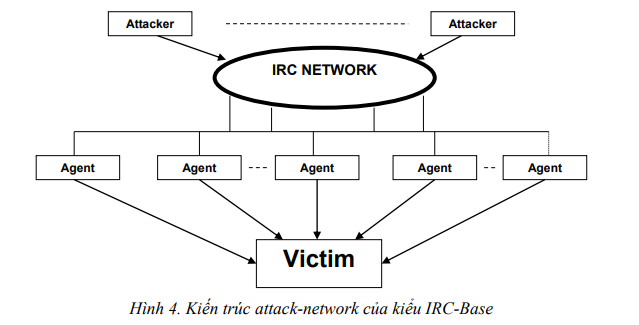
- Thông thường Attacker sẽ đặt Handler software trên một Router hay một server có lượng traffic lưu thông nhiều. Việc này nhằm làm cho các giao tiếp giữa Client, handler và Agent khó bị phát hiện. Các gia tiếp này thông thường xảy ra trên các protocol TCP, UDP hay ICMP. Chủ nhân thực sự của các Agent thông thường không hề hay biết họ bị lợi dụng vào cuộc tấn công kiểu DDoS, do họ không đủ kiến thức hoặc các chương trình Backdoor Agent chỉ sử dụng rất ít tài nguyên hệ thống làm cho hầu như không thể thấy ảnh hưởng gì đến hiệu năng của hệ thống.

ii. Mô hình IRC – Based:

- Internet Relay Chat (IRC) là một hệ thống online chat multiuser, IRC cho phép User tạo một kết nối đến multipoint đến nhiều user khác và chat thời gian thực. Kiến trúc củ IRC network bao gồm nhiều IRC server trên khắp internet, giao tiếp với nhau trên nhiều kênh (channel). IRC network cho phép user tạo ba loại channel: public, private và serect.

• Public channel: Cho phép user của channel đó thấy IRC name và nhận được message của mọi user khác trên cùn• Private channel: được thiết kế để giao tiếp với các đối tượng cho phép. Không cho phép các user không cùng channel thấy IRC name và message trên channel. Tuy nhiên, nếu user ngoài channel dùng một số lệnh channel locator thì có thể biết được sự tồn tại của private channel đó.

• Secrect channel : tương tự private channel nhưng không thể xác định bằng channel locator.



- IRC – Based net work cũng tương tự như Agent – Handler network nhưng mô hình này sử dụng các kênh giao tiếp IRC làm phương tiện giao tiếp giữa Client và Agent (không sử dụng Handler). Sử dụng mô hình này, attacker còn có thêm một số lợi thế khác như:

+ Các giao tiếp dưới dạng chat message làm cho việc phát hiện chúng là vô cùng khó khan

IRC traffic có thể di chuyển trên mạng với số lượng lớn mà không bị nghi ngờ

+ Không cần phải duy trì danh sách các Agent, hacker chỉ cần logon vào IRC server là

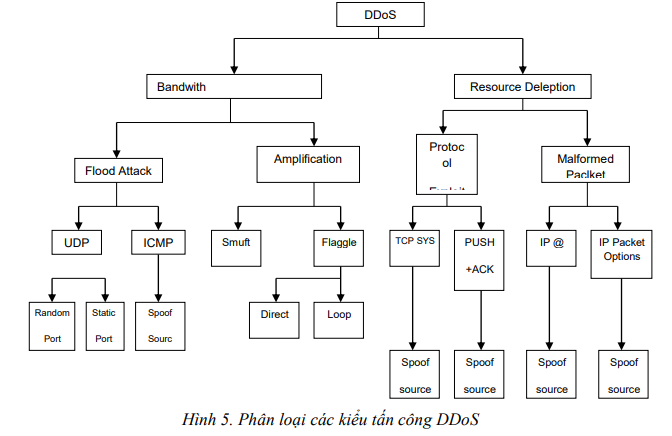
đã có thể nhận được report về trạng thái các Agent do các channel gửi về.

+ Sau cùng: IRC cũng là một môi trường file sharing tạo điều kiện phát tán các Agent

code lên nhiều máy khác.

**2.2.2.3. Phân loại tấn công kiểu DDOS**

- Nhìn chung, có rất nhiều biến thể của kỹ thuật tấn công DDoS nhưng nếu nhìn dưới góc độ chuyên môn thì có thể chia các biến thề này thành hai loại dựa trên mụch đích tấn công: Làm cạn kiệt băng thông và làm cạn kiệt tài nguyên hệ thống



1. *Những kiểu tấn công làm cạn kiệt băng thông của mạng (BandWith Depletion Attack*)

- BandWith Depletion Attack được thiết kế nhằm làm tràng ngập mạng mục tiêu với những traffic không cần thiết, với mục địch làm giảm tối thiểu khả năng của các traffic hợp lệ đến được hệ thống cung cấp dịch vụ của mục tiêu.

- Có hai loại BandWith Depletion Attack:

+ Flood attack: Điều khiển các Agent gởi một lượng lớn traffic đến hệ thống dịch vụ của mục tiêu, làm dịch vụ này bị hết khả năng về băng thông.

+ Amplification attack: Điều khiển các agent hay Client tự gửi message đến một địa chỉ IP broadcast, làm cho tất cả các máy trong subnet này gửi message đến hệ thống dịch vụ của mục tiêu. Phương pháp này làm gia tăng traffic không cần thiết, làm suy giảm băng thông của mục tiêu.Flood attack:

Trong phương pháp này, các Agent sẽ gửi một lượng lớn IP traffic làm hệ thống dịch vụ của mục tiêu bị chậm lại, hệ thống bị treo hay đạt đến trạng thái hoạt động bão hòa.

Làm cho các User thực sự của hệ thống không sử dụng được dịch vụ.

Ta có thể chia Flood Attack thành hai loại:

+ UDP Flood Attack: do tính chất connectionless của UDP, hệ thống nhận UDP message chỉ đơn giản nhận vào tất cả các packet mình cần phải xử lý. Một lượng lớn các UDP packet được gởi đến hệ thống dịch vụ của mục tiêu sẽ đẩy toàn bộ hệ thống đến ngưỡng tới hạn.

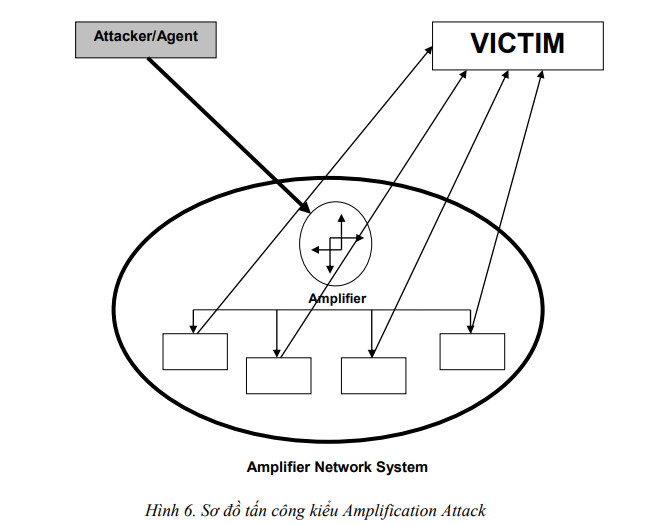
+ Các UDP packet này có thể được gửi đến nhiều port tùy ý hay chỉ duy nhất một port. Thông thường là sẽ gửi đến nhiều port làm cho hệ thống mục tiêu phải căng ra để xử lý phân hướng cho các packet này. Nếu port bị tấn công không sẵn sàng thì hệ thống mục tiêu sẽ gửi ra một ICMP packet loại “destination port unreachable”. Thông thường các Agent software sẽ dùng địa chỉ IP giả để che giấu hành tung, cho nên các message trả về do không có port xử lý sẽ dẫn đến một đại chỉ IP khác. UDP Flood attack cũng có thể làm ảnh hưởng đến các kết nối xung quanh mục tiêu do sự hội tụ của packet diễn ra rất mạnh.

+ ICMP Flood Attack: được thiết kế nhằm mục đích quản lý mạng cũng như định vị thiết bị mạng. Khi các Agent gởi một lượng lớn ICMP\_ECHO\_REPLY đến hệ thống mục tiêu thì hệ thống này phải reply một lượng tương ứng Packet để trả lời, sẽ dẫn đến nghẽn đường truyền. Tương tự trường hợp trên, địa chỉ IP của cá Agent có thể bị giả mạo.

+ Amplification Attack:

- Amplification Attack nhắm đến việc sử dụng các chức năng hỗ trợ địa chỉ IP broadcast của các router nhằm khuyếch đại và hồi chuyển cuộc tấn công. Chức năng này cho phép bên gửi chỉ định một địa chỉ IP broadcast cho toàn subnet bên nhận thay vì nhiều địa chỉ. Router sẽ có nhiệm vụ gửi đến tất cả địa chỉ IP trong subnet đó packet broadcast mà nó nhận được.

- Attacker có thể gửi broadcast message trực tiếp hay thông qua một số Agent nhằm làm gia tăng cường độ của cuộc tấn công. Nếu attacker trực tiếp gửi message, thì có thể lợi dụng các hệ thống bên trong broadcast network như một Agent.



Có thể chia amplification attack thành hai loại, Smuft va Fraggle attack:

+ Smuft attack: trong kiểu tấn công này attacker gởi packet đến network amplifier (router hay thiết bị mạng khác hỗ trợ broadcast), với địa chỉ của nạn nhân. Thông thường những packet được dùng là ICMP ECHO REQUEST, các packet này yêu cầu yêu cầu bên nhận phải trả lời bằng một ICMP ECHO REPLY packet. Network amplifier sẽ gửi đến ICMP ECHO REQUEST packet đến tất cả các hệ thống thuộc địa chỉ broadcast và tất cả các hệ thống này sẽ REPLY packet về địa chỉ IP của mục tiêu tấn công Smuft Attack.

+ Fraggle Attack: tương tự như Smuft attack nhưng thay vì dùng ICMP ECHO REQUEST packet thì sẽ dùng UDP ECHO packet gởi đếm mục tiêu. Thật ra còn một biến thể khác của Fraggle attack sẽ gửi đến UDP ECHO packet đến chargen port (port19/UNIX) của mục tiêu, với địa chỉ bên gửi là echo port (port 7/UNIX) của mục tiêu, tạo nên một vòng lặp vô hạn. Attacker phát động cuộc tấn công bằng một ECHO REQUEST với địa chỉ bên nhận là một địa chỉ broadcast, toàn bộ hệ thống thuộc địa chỉ này lập tức gửi REPLY đến port echo của nạn nhân, sau đó từ nạn nhân một ECHO REPLY lại gửi trở về địa chỉ broadcast, quá trình cứ thế tiếp diễn. Đây chính là nguyên nhân Flaggle Attack nguy hiểm hơn Smuft Attack rất nhiều.

*ii. Những kiểu tấn công làm cạn kiệt tài nguyên: (Resource Deleption Attack)*

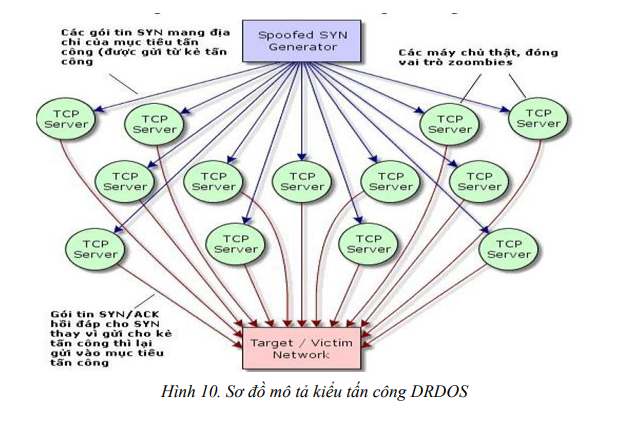
- Theo định nghĩa: Resource Deleption Attack là kiểu tấn công trong đó Attacker gởi những packet dùng các protocol sai chức năng thiết kế, hay gửi những packet với dụng ý làm tắt nghẽn tài nguyên mạng làm cho các tài nguyên này không phục vụ user thông thường khác được.

### **2.2.3. Tấn công từ chối dịch vụ phản xạ nhiều vùng DRDoS (Distributed Reflection Denial of Service)**

- Xuất hiện vào đầu năm 2002, là kiểu tấn công mới nhất, mạnh nhất trong họ DoS. Nếu được thực hiện bởi kẻ tấn công có tay nghề thì có thể hạ gục bất cứ hệ thống phút chốc

- Mục tiêu chính của DDDoS là chiếm đoạt toàn bộ băng thông của máy chủ, tức là làm tắc ngẽn hoàn toàn đường kết nối từ máy chủ vào xương sống của Internet và tiêu hao tài nguyên máy chủ. Trong suốt quá trình máy chủ bị tấn công bằng DrDoS, không một máy khách nào ch thể kết nối được vào máy chủ đó. Tất cả các dịch vụ chạy trên nền TCP/IP như DNS, HTTP, FTP, POP3, ... đều bị vô hiệu hóa.

- Về cơ bản, DrDoS là sự phối hợp giữa hai kiểu DoS và DDoS. Nó có kiểu tấn công SYN với một máy tính đơn, vừa có sự kết hợp giữa nhiều máy tính để chiếm dụng băng thông như kiểu DDoS. Kẻ tấn công thực hiện bằng cách giả mạo địa chỉ của server mục tiêu rồi gửi yêu cầu SYN đến các server lớn như Yahoo,Micorosoft…,để các server này gửi các gói tin SYN/ACK đến server mục tiêu. Các server lớn, đường truyền mạnh đã vô tình đóng vai trò zoombies cho kẻ tấn công như trong DdoS



- Quá trình gửi cứ lặp lại liên tục với nhiều địa chỉ IP giảp từ kẻ tấn công, với nhiều server lớn tham gia nên server mục tiêu nhanh chóng bị quá tải, bandwidth bị chiếm dụng bởi server lớn. Tính nghệ thuật là ở chổ chỉ cần với một máy tính với modem 56kbps, một hacker lành nghề có thể đánh bại bất cứ máy chủ nào trong giây lát mà không cần chiếm đoạt bất cứ máy nào để làm phương tiện thực hiện tấn công.

# **2.3. SQL Injection**

## **2.3.1. Tấn công SQL injection**

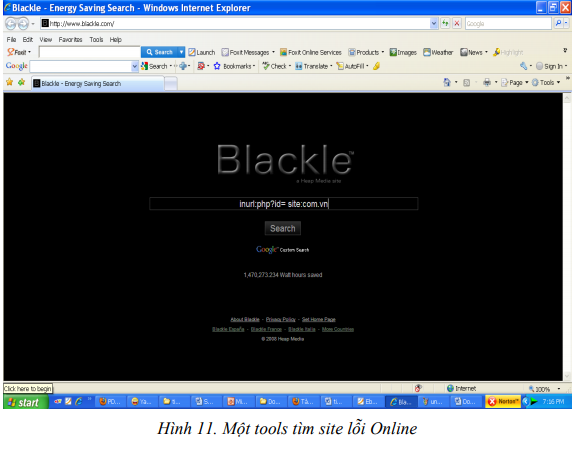
**2.3.1.1. SQL Injection là gì?**

- Khi triển khai các ứng dụng web trên Internet, nhiều người vẫn nghĩ rằng việc đảm bảo an toàn, bảo mật nhằm giảm thiểu tối đa khả năng bị tấn công từ các tin tặc chỉ đơn thuần tập trung vào các vấn đề như chọn hệ điều hành, hệ quản trị cơ sở dữ liệu, webserver sẽ chạy ứng dụng, ... mà quên mất ằng ngay cả bản thân ứng dụng chạy trên đó cũng tiềm ẩn một lỗ hổng bảo mật rất lớn. Một trong số các lỗ hổng này đó là SQL injection. Tại Việt Nam, đã qua thời kì các quản trị website lơ là việc quét virus, cập nhật các bản vá lỗi từ các phần mềm hệ thống, nhưng việc chăm sóc các lỗi của các ứng dụng lại rất ít được quan tâm. Đó là lí do tại sao trong thời gian vừa qua, không ít website tại Việt Nam bị tấn công và đa số đều là lỗi SQL injection. Vậy SQL injection là gì ?

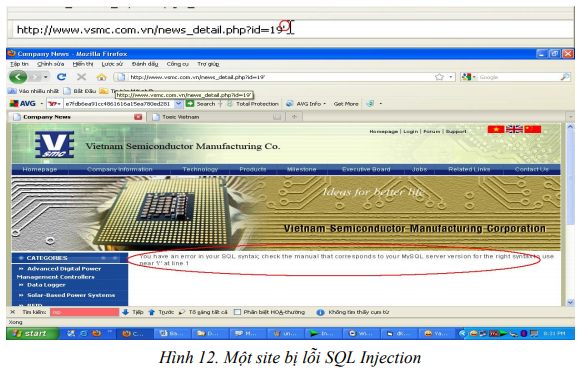
- SQL injection là một kĩ thuật cho phép những kẻ tấn công lợi dụng lỗ hổng trong việc kiểm tra dữ liệu nhập trong các ứng dụng web và các thông báo lỗi của hệ quản trị cơ sở dữ liệu để "tiêm vào" (inject) và thi hành các câu lệnh SQL bất hợp pháp (không được người phát triển ứng dụng lường trước). Hậu quả của nó rất tai hại vì nó cho phép những kẻ tấn công có thể thực hiện các thao tác xóa, hiệu chỉnh, … do có toàn quyền trên cơ sở dữ liệu của ứng dụng, thậm chí là server mà ứng dụng đó đang chạy. Lỗi này thường xảy ra trên các ứng dụng web có dữ liệu được quản lí bằng các hệ quản trị cơ sở dữ liệu như SQL Server, MySQL, Oracle, DB2, Sysbase.

**2.3.1.2. Các Dạng Tấn Công SQL Injection**

- Để biết các website bán hàng sử dụng CSDL SQL ta sử dụng các soft hoặc các công cụ tìm lỗi.Hoặc các công cụ tìm kiếm như Google.Và dùng các Dork tìm kiếm như : inurl : product.php?id= câu lện SELECT, sử dụng câu lệnh INSERT, sử dụng các stored-procedures.



- Để biết website nào dính lỗi SQL Injection ta thêm dấu “ ’ ” vào sau thanh địa chỉ. Ví dụ : <http://www.doanchuyenganh.com/product.php?id=123>’



1. *Dạng tấn công vượt qua kiểm tra đăng nhập*

- Với dạng tấn công này, tin tặc có thể dễ dàng vượt qua các trang đăng nhập nhờ vào lỗi khi dùng các câu lệnh SQL thao tác trên cơ sở dữ liệu của ứng dụng web.

Xét một ví dụ điển hình, thông thường để cho phép người dùng truy cập vào các trang web được bảo mật, hệ thống thường xây dựng trang đăng nhập để yêu cầu người dung nhập thông tin về tên đăng nhập và mật khẩu. Sau khi người dùng nhập thông tin vào, hệ thống sẽ kiểm tra tên đăng nhập và mật khẩu có hợp lệ hay không để quyết định cho phép hay từ chối thực hiện tiếp. Trong trường hợp này, người ta có thể dùng hai trang, một trang HTML để hiển thị form nhập liệu và một trang ASP dùng để xử lí thông tin nhập từ phía người dùng.

Ví dụ: login.html

<form action="ExecLogin.asp" method="post">

Username: <input type="text" name="fUSRNAME"><br>

Password: <input type="password" name="fPASSWORD"><br>

<input type="submit">

</form>

execlogin.asp

<%

Dim vUsrName, vPassword, objRS, strSQL

vUsrName = Request.Form("fUSRNAME")

vPassword = Request.Form("fPASSWORD")

strSQL = "SELECT \* FROM T\_USERS " & \_

"WHERE USR\_NAME=' " & vUsrName & \_

" ' and USR\_PASSWORD=' " & vPassword & " ' "

Set objRS = Server.CreateObject("ADODB.Recordset")

objRS.Open strSQL, "DSN=..."

If (objRS.EOF) Then

Response.Write "Invalid login."

Else

Response.Write "You are logged in as " & objRS("USR\_NAME")

End If

Set objRS = Nothing

%>

- Thoạt nhìn, đoạn mã trong trang execlogin.asp dường như không chứa bất cứ một lỗ hổng về an toàn nào. Người dùng không thể đăng nhập mà không có tên đăng nhập và mật khẩu hợp lệ. Tuy nhiên, đoạn mã này thực sự không an toàn và là tiền đề cho một lỗi SQL injection. Đặc biệt, chỗ sơ hở nằm ở chỗ dữ liệu nhập vào từ người dùng được dùng để xây dựng trực tiếp câu lệnh SQL. Chính điều này cho phép những kẻ tấn công có thể điều khiển câu truy vấn sẽ được thực hiện. Ví dụ, nếu người dùng nhập chuỗi sau vào trong cả 2 ô nhập liệu username/password của trang login.htm là: ' OR ' ' = ' '. Lúc này, câu truy vấn sẽ được gọi thực hiện là: SELECT \* FROM T\_USERS WHERE USR\_NAME ='' OR ''='' and USR\_PASSWORD= '' OR ''='' - Câu truy vấn này là hợp lệ và sẽ trả về tất cả các bản ghi của T\_USERS và đoạn mã tiếp theo xử lí người dùng đăng nhập bất hợp pháp này như là người dùng đăng nhập hợp lệ.

*ii. Dạng tấn công sử dụng câu lệnh SELECT*

- Dạng tấn công này phức tạp hơn. Để thực hiện được kiểu tấn công này, kẻ tấn công phải có khả năng hiểu và lợi dụng các sơ hở trong các thông báo lỗi từ hệ thống để dò tìm các điểm yếu khởi đầu cho việc tấn công. Xét một ví dụ rất thường gặp trong các website về tin tức. Thông thường, sẽ có một trang nhận ID của tin cần hiển thị rồi sau đó truy vấn nội dung của tin có ID này.

Ví dụ: http://www.doanchuyennganh.com/product.asp?ID=123 . Mã nguồn cho chức năng này thường được viết khá đơn giản theo dạng

<%

Dim vNewsID, objRS, strSQL

vNewsID = Request("ID")

strSQL = "SELECT \* FROM T\_NEWS WHERE NEWS\_ID =" & vNewsID

Set objRS = Server.CreateObject("ADODB.Recordset")

objRS.Open strSQL, "DSN=..."

Set objRS = Nothing

%>

- Trong các tình huống thông thường, đoạn mã này hiển thị nội dung của tin có ID trùng với ID đã chỉ định và hầu như không thấy có lỗi. Tuy nhiên, giống như ví dụ đăng nhập ở trước, đoạn mã này để lộ sơ hở cho một lỗi SQL injection khác. Kẻ tấn công có thể thay thế một ID hợp lệ bằng cách gán ID cho một giá trị khác, và từ đó, khởi đầu cho một cuộc tấn công bất hợp pháp, ví dụ như:

0 OR 1=1 (nghĩa là, http://www.doanchuyennganh.com/product.asp?ID=0 or 1=1).

- Câu truy vấn SQL lúc này sẽ trả về tất cả các article từ bảng dữ liệu vì nó sẽ thực hiện câu lệnh:

SELECT \* FROM T\_NEWS WHERE NEWS\_ID=0 or 1=1

- Một trường hợp khác, ví dụ như trang tìm kiếm. Trang này cho phép người dung nhập vào các thông tin tìm kiếm như Họ, Tên, … Đoạn mã thường gặp là:

<%

Dim vAuthorName, objRS, strSQL

vAuthorName = Request("fAUTHOR\_NAME")

strSQL = "SELECT \* FROM T\_AUTHORS WHERE AUTHOR\_NAME =' " & \_

vAuthorName & " ' "

Set objRS = Server.CreateObject("ADODB.Recordset")

objRS.Open strSQL, "DSN=..."

…

Set objRS = Nothing

%>

- Tương tự như trên, tin tặc có thể lợi dụng sơ hở trong câu truy vấn SQL để nhập vào trường tên tác giả bằng chuỗi giá trị:

' UNION SELECT ALL SELECT OtherField FROM OtherTable WHERE ' '='

- Lúc này, ngoài câu truy vấn đầu không thành công, chương trình sẽ thực hiện them lệnh tiếp theo sau từ khóa UNION nữa.

- Tất nhiên các ví dụ nói trên, dường như không có gì nguy hiểm, nhưng hãy thử tưởng tượng kẻ tấn công có thể xóa toàn bộ cơ sở dữ liệu bằng cách chèn vào các đoạn lệnh nguy hiểm như lệnh DROP TABLE.

Ví dụ như: ' DROP TABLE T\_AUTHORS –

- Chắc các bạn sẽ thắc mắc là làm sao biết được ứng dụng web bị lỗi dạng này được.

Rất đơn giản, hãy nhập vào chuỗi (\*) như trên, nếu hệ thống báo lỗi về cú pháp dạng:

Invalid object name “OtherTable”; ta có thể biết chắc là hệ thống đã thực hiện câu

SELECT sau từ khóa UNION, vì như vậy mới có thể trả về lỗi mà ta đã cố tình tạo ra trong câu lệnh SELECT.

- Cũng sẽ có thắc mắc là làm thế nào có thể biết được tên của các bảng dữ liệu mà thực hiện các thao tác phá hoại khi ứng dụng web bị lỗi SQL injection. Cũng rất đơn giản, bởi vì trong SQL Server, có hai đối tượng là sysobjects và syscolumns cho phép liệt kê tất cả các tên bảng và cột có trong hệ thống. Ta chỉ cần chỉnh lại câu lệnh SELECT, ví dụ như: ' UNION SELECT name FROM sysobjects WHERE xtype = 'U' là có thể liệt kê được tên tất cả các bảng dữ liệu.iii. Dạng tấn công sử dụng câu lệnh INSERT

- Thông thường các ứng dụng web cho phép người dùng đăng kí một tài khoản để tham gia. Chức năng không thể thiếu là sau khi đăng kí thành công, người dùng có thể xem và hiệu chỉnh thông tin của mình. SQL injection có thể được dùng khi hệ thống không kiểm tra tính hợp lệ của thông tin nhập vào.

- Ví dụ, một câu lệnh INSERT có thể có cú pháp dạng:

INSERT INTO TableName VALUES('Value One', 'Value Two', 'Value Three').

Nếu đoạn mã xây dựng câu lệnh SQL có dạng :

<%

strSQL = "INSERT INTO TableName VALUES(' " & strValueOne & " ', ' " \_

& strValueTwo & " ', ' " & strValueThree & " ') "

Set objRS = Server.CreateObject("ADODB.Recordset")

objRS.Open strSQL, "DSN=..."

…

Set objRS = Nothing

%>

- Thì chắc chắn sẽ bị lỗi SQL injection, bởi vì nếu ta nhập vào trường thứ nhất ví dụ như: ' + (SELECT TOP 1 FieldName FROM TableName) + '. Lúc này câu truy vấn sẽ là: INSERT INTO TableName VALUES(' ' + (SELECT TOP 1 FieldName FROM TableName) + ' ', 'abc', 'def'). Khi đó, lúc thực hiện lệnh xem thông tin, xem như bạn đã yêu cầu thực hiện thêm một lệnh nữa đó là: SELECT TOP 1 FieldName FROM TableNameiiii. Dạng tấn công sử dụng stored-procedures

- Việc tấn công bằng stored-procedures sẽ gây tác hại rất lớn nếu ứng dụng được thực thi với quyền quản trị hệ thống 'sa'.

- Ví dụ: nếu ta thay đoạn mã tiêm vào dạng: ' ; EXEC xp\_cmdshell ‘cmd.exe dir C: '.

- Lúc này hệ thống sẽ thực hiện lệnh liệt kê thư mục trên ổ đĩa C:\ cài đặt server.

- Việc phá hoại kiểu nào tuỳ thuộc vào câu lệnh đằng sau cmd.exe. Nếu cài SQL

Server ở chế độ mặc định thì SQL Server chạy trên nền SYSTEM, tương đương mức truy cập ở Windows. Có thể dùng master..xp\_cmdshell để thi hành lệnh từ xa:

-- exec master..xp\_cmdshell 'ping 10.10.1.2'--

Thử dùng dấu nháy đôi (") nếu dấu nháy đơn (') không làm việc.

Dưới đây là một số extended stored procedure mà hacker thường hay sử dụng để thực thi những câu lệnh xem nội dung thông tin trong máy nạn nhân:

Xp\_availablemedia: Hiển thị những ổ đĩa hiện hành trên máy

Xp\_dirtree: Hiển thị tất cả các thư mục kể cả thư mục con

Xp\_loginconfig: Lấy thông tin về chế độ bảo mật trên server

Xp\_makecab: Cho phép người sử dụng tạo các tập tin lưu trữ trên Server (hay bất cứ tập tin nào mà server có thể truy xuất

Xp\_ntsec\_enumdomain: liệt kê những domain mà server có thể truy vấn.

Xp\_terminate\_process: chấm dứt một tiến trình với tham số PID của nó.

### **2.3.2.Cách Phòng Tránh SQL Injection**

- Như vậy, có thể thấy lỗi SQL injection khai thác những bất cẩn của các lập trình viên phát triển ứng dụng web khi xử lí các dữ liệu nhập vào để xây dựng câu lệnh SQL. Tác hại từ lỗi SQL injection tùy thuộc vào môi trường và cách cấu hình hệ thống. Nếu ứng dụng sử dụng quyền dbo (quyền của người sở hữu cơ sở dữ liệu - owner) khi thao tác dữ liệu, nó có thể xóa toàn bộ các bảng dữ liệu, tạo các bảng dữ liệu mới, … Nếu ứng dụng sử dụng quyền sa (quyền quản trị hệ thống), nó có thể điều khiển toàn bộ hệ quản trị cơ sở dữ liệu và với quyền hạn rộng lớn như vậy nó có thể tạo ra các tài khoản người dùng bất hợp pháp để điều khiển hệ thống của bạn.

• Trong hầu hết trình duyệt, những kí tự nên được mã hoá trên địa chỉ URL trước khi được sử dụng.

• Việc tấn công theo SQL Injection dựa vào những câu thông báo lỗi do đó việc

phòng chống hay nhất vẫn là không cho hiển thị những thông điệp lỗi cho người dung bằng cách thay thế những lỗi thông báo bằng 1 trang do người phát triển thiết kế mỗi khi lỗi xảy ra trên ứng dụng.

• Kiểm tra kĩ giá trị nhập vào của người dùng, thay thế những kí tự như ‘ ; v..v.. Hãy loại bỏ các kí tự meta như “',",/,\,;“ và các kí tự extend như NULL, CR, LF, ... trong các string nhận được từ:

* Dữ liệu nhập do người dùng đệ trình
* Các tham số từ URL
* Các giá trị từ cookie

• Đối với các giá trị numeric, hãy chuyển nó sang integer trước khi thực hiện câu truy vấnSQL, hoặc dùng ISNUMERIC để chắc chắn nó là một số integer.

• Dùng thuật toán để mã hoá dữ liệu

1. *Kiểm tra dữ liệu*

- Kiểm tra tính đúng đắn của dữ liệu là 1 vấn đề phức tạp và thường chưa được quan tâm đúng mức trong các ứng dụng. Khuynh hướng của việc kiểm tra tính đúng đắn của dữ liệu không phải là chỉ cần thêm một số chức năng vào ứng dụng, mà phải kiểm tra một cách tổng quát nhanh chóng để đạt được mục đích.

- Những tóm tắt sau đây sẽ bàn về việc kiểm tra tính đúng đắn của dữ liệu, cùng với ví dụ mẫu để minh hoạ cho vấn đề này.

Có ba giải pháp tiếp cận vấn đề này:

1) Cố gắng kiểm tra và chỉnh sửa để làm cho dữ liệu hợp lệ.

2) Loại bỏ những dữ liệu bất hợp lệ.

3) Chỉ chấp nhận những dữ liệu hợp lệ

• Giải pháp 1: khó thực hiện

- Thứ nhất, người lập trình không cần thiết phải biết tất cả dữ liệu bất hợp lệ, bởi vì những dạng dữ liệu bất hợplệ rất đa dạng.

- Thứ hai, là vấn đề của trường hợp bị tấn công 2 tầng (second-oder SQL injection) trong việc lấy dữ liệu từ hệ thống ra.

• Giải pháp 2: bị vô hiệu trong các trường hợp như giải pháp 1 là do :

- Dữ liệu bất hợp lệ luôn luôn thay đổi và cùng với việc phát triển các kiểu tấn công mới.

• Giải pháp 3: tốt hơn hai giải pháp kia, nhưng sẽ gặp một số hạn chế khi cài đặt.

- Cách bảo mật tốt nhất là kết hợp cả giải pháp 2 và 3. Một ví dụ cho sự cần thiết

kết hợp 2-3 là dấu nối giữa họ và tên “Quentin Bassington-Bassington” phải cho phép dấu gạch ngang trong bộ định nghĩa dữ liệu hợp lệ, nhưng chuỗi kí tự “--“ là một chuỗi kí tự đặc biệt trong SQL server.

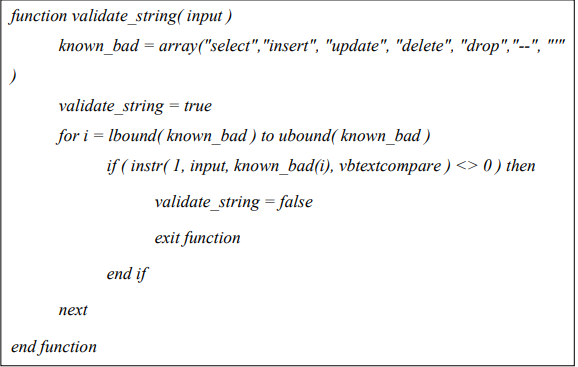
- Ví dụ nếu có bộ lọc để :

+ Lọc bỏ những dữ liệu bất hợp lệ như ‘--‘,’select’ và ‘union’

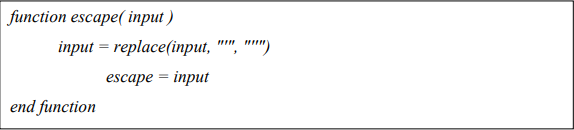
+ Một hàm kiểm soát để loại bỏ dấu nháy đơn thì có thể đối phó như sau : uni’on se’lect @@version-‘-

- Một số cách cài đặt các chức năng kiểm tra dữ liệu cơ bản

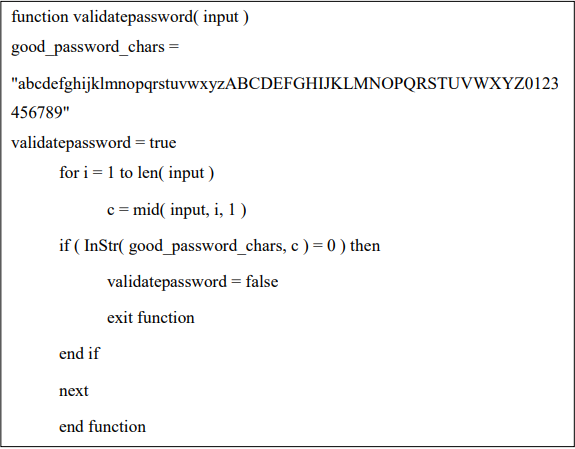
• Cách 1: Từ chối dữ liệu bất hợp lệ



• Cách 2: Thay thế dấu nháy đơn:



• Cách 3: Chỉ chấp nhận dữ liệu hợp lệ



1. *Khoá chặt SQL Server (SQL Server Lockdown)*

Đây là một danh sách các công việc cần làm để bảo vệ SQL server:

• Xác định các phương pháp kết nối đến server:

* Dùng tiện ích Network Utility để kiểm tra rằng chỉ có các thư viện mạng đang

dùng là hoat động.

• Kiểm tra tất cả các tài khoản có trong SQL Server

* Chỉ tạo tài khoản có quyền thấp cho các ứng dụng
* Loại bỏ những tài khoản không cần thiết
* Đảm bảo rằng tất cả tài khoản có một mật khẩu hợp lệ, …

• Kiểm tra các đối tượng tồn tại

* Nhiều extended stored procedure có thể được xoá bỏ một cách an toàn. Nếu điều này được thực hiện, thì cũng nên xem xét việc loại bỏ luôn những tập tin .dll chứa mã của các extended stored procedure
* Xoá bỏ tất cả cơ sở dữ liệu mẫu như “northwind” và “pubs”
* Xóa các stored procedure không dùng như: master..xp\_cmdshell,xp\_startmail,xp\_sendmail,sp\_makewebtk

• Kiểm tra những tài khoản nào có thể truy xuất đến những đối tượng nào

* Đối với những tài khoản của một ứng dụng nào đó dùng để truy xuất cơ sở dữ liệu thì chỉ được cấp những quyền hạn cần thiết tối thiểu để truy xuất đến những đối tượng nó cần dùng. Kiểm tra lớp sửa chữa của server
* Có một số cách tấn công như “buffer overflow”, “format string” thường chú ý đến lớp bảo vệ này.

• Kiểm tra các phiên làm việc trên server.

• Thay đổi "Startup và chạy SQL Server" ở mức người dùng quyền hạn thấp trong SQL Server Security.

- Nhận xét:

+ Qua phần tìm hiểu về SQL Injection,càng thấy rằng việc kiểm tra dữ liệu

trước khi xử lý là cần thiết.

+ Ứng dụng ngoài việckiểm tra tính đúng đắn của dữ liệu, cần mã hóa dữ liệu ngay bên trong cơ sở dữ liệu và không cho xuất trang Web lỗi, báo nội dung lỗi cú pháp SQL để hacker không thể thu thập thông tin cơ sở dữ liệu. Song song đó là công việc của người quản trị mạng.

*iii. Thiết lập cấu hình an toàn cho hệ quản trị cơ sở dữ liệu*

- Cần có cơ chế kiểm soát chặt chẽ và giới hạn quyền xử lí dữ liệu đến tài khoản người dùng mà ứng dụng web đang sử dụng. Các ứng dụng thông thường nên tránh dùng đến các quyền như dbo hay sa. Quyền càng bị hạn chế, thiệt hại càng ít. Ngoài ra để tránh các nguy cơ từ SQL Injection attack, nên chú ý loại bỏ bất kì thông tin kĩ thuật nào chứa trong thông điệp chuyển xuống cho người dùng khi ứng dụng có lỗi. Các thông báo lỗi thông thường tiết lộ các chi tiết kĩ thuật có thể cho phép kẻ tấn công biết được điểm yếu của hệ thống.

# **CHƯƠNG 3. DEMO, ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN ĐỀ TÀI**

## **3.1. Demo**

- Trước tiên ta sử dụng một thủ thuật tìm kiếm nhỏ trên google để có thể tìm kiếm site bị lỗi SQL Injecton. Ở đây dùng từ khóa: inurl:keywords

Ví dụ: inurl:sanpham.php?id=3

- Sử dụng từ khóa trên google.com tôi chọn được một website thiết kế sơ sài là http://nhanquynhphat.com/sanpham.php?id=3 ; dự đoán nó bị dính lỗi SQL Injetion và tiến hành khai thác lỗi.

- Tiến hành kiểm tra lỗi và thấy website này bị lỗi SQL Injection, tiếp tục lấy các thông tin về website như version MySQL để việc khai thác trở nên rõ ràng hơn. Ở đây website sử dụng version MySQL >=5 nên tôi có thể dễ dàng khai thác lỗi thông qua information\_shema.tables mà không cần phải đoán table của nó là gì.



- Bỏ qua các table không liên quan ta lấy được các table như sau: khuyenmai, lienhe, loaispcon, online, sanpham, tbl\_gioithieu, tbl\_lienhe, tbl\_lienket, tbl\_tintuc, thanhtoan, tintuc, user

- Sau đó tiến hành lấy thông tin column và data và kết quả như hình 14.



- Theo hình 14. dữ liệu lấy được đang ở dạng mã hóa. Việc khai thác SQL Injection đến đây còn 1 bước nữa là tìm đường dẫn đăng nhập quản trị và nếu mật khẩu nằm ở dạng mã hóa thì ta cần phải tiến hành giải mã.

## **3.2. Kết luận**

### **3.2.1. Các vấn đề đạt được**

- Theo yêu cầu đặt ra ban đầu thì cho đến thời điểm hiện tại, đồ án đã đạt được các nội dung sau:

• Tìm hiểu các kĩ thuật tấn công ứng dụng Web bao gồm các kĩ thuật

* Chèn mã lệnh thực thi trên trình khách Cross-site Scripting.
* Chèn câu truy vấn SQL và Tấn công SQL Injection nâng cao
* Tấn công Local Acttack.
* Từ chối dịch vụ .

• Các biện pháp bảo mật từ sự kết hợp giữa nhà quản trị mạng, nhà thiết kế ứng dụng Web và người dùng

* Kiểm tra một trang Web có khả năng bị tấn công bằng những kĩ thuật chèn câu lệnh SQL, thay đổi tham số hay không.
* Có thể phòng chống được các lỗi tấn công thông dụng hiện nay, như các vấn đề đã tìm hiểu ở trên.

**PHÂN CÔNG CÔNG VIỆC**

|  |  |  |
| --- | --- | --- |
| 1  **STT** | **Sinh Viên** | **Công việc phân công** |
| 21  1 | Phạm Văn Dưỡng – B15DCVT106 | Tìm hiểu, làm word và slide phần tấn công SQL injection |
| 22 | Đinh Duy Phong – B15DCVT298 | Tìm hiểu, làm word và slide phần tấn công từ chối dịch vụ - (Denial Of Service) |
| 3 3 | Nguyễn Tuấn Ngọc – B15DCVT285 | Tìm hiểu, làm word và slide phần tấn công local attack |