7. Analisis Forense - ¿Qué es Wireshark?

**Práctica: Análisis de Tráfico HTTP con Wireshark en una Web Vulnerable**

*Caso práctico: testphp.vulnweb.com*

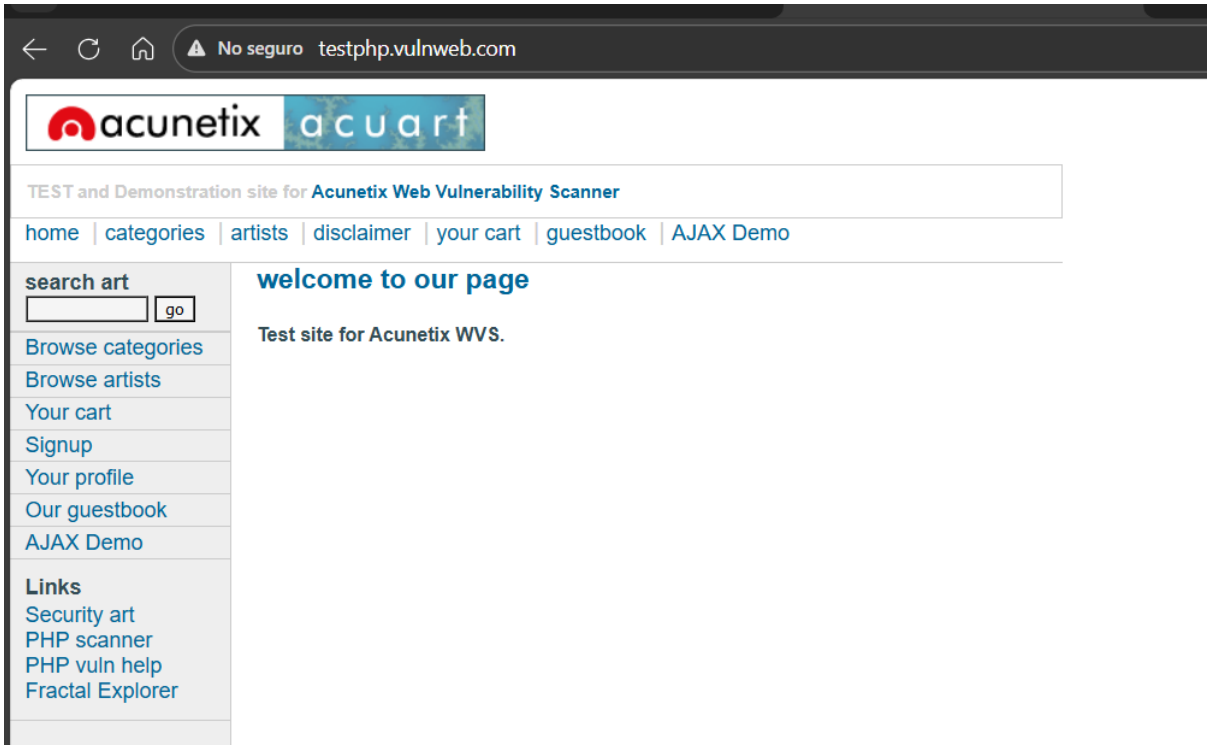
# 1. Introducción

En esta práctica aprenderás a utilizar Wireshark para analizar el tráfico HTTP generado al navegar por una aplicación web vulnerable. El objetivo es comprender cómo viajan las peticiones y respuestas en una web sin cifrado, identificar parámetros, observar cabeceras, analizar formularios y cookies de sesión, y entender los riesgos de seguridad asociados.

La web utilizada es:

http://testphp.vulnweb.com/

Este sitio es un entorno público y autorizado para prácticas de ciberseguridad ofrecido por Acunetix, por lo que su uso es completamente legal con fines académicos.




# 2. Objetivos de la práctica

Al finalizar la actividad, deberás ser capaz de:

- Capturar tráfico HTTP real con Wireshark.
- Aplicar filtros para aislar información concreta.
- Analizar peticiones GET y POST.
- Identificar parámetros, rutas internas y patrones de navegación.
- Localizar cookies y sesiones enviadas en texto claro.
- Comprender los riesgos de enviar credenciales por HTTP.

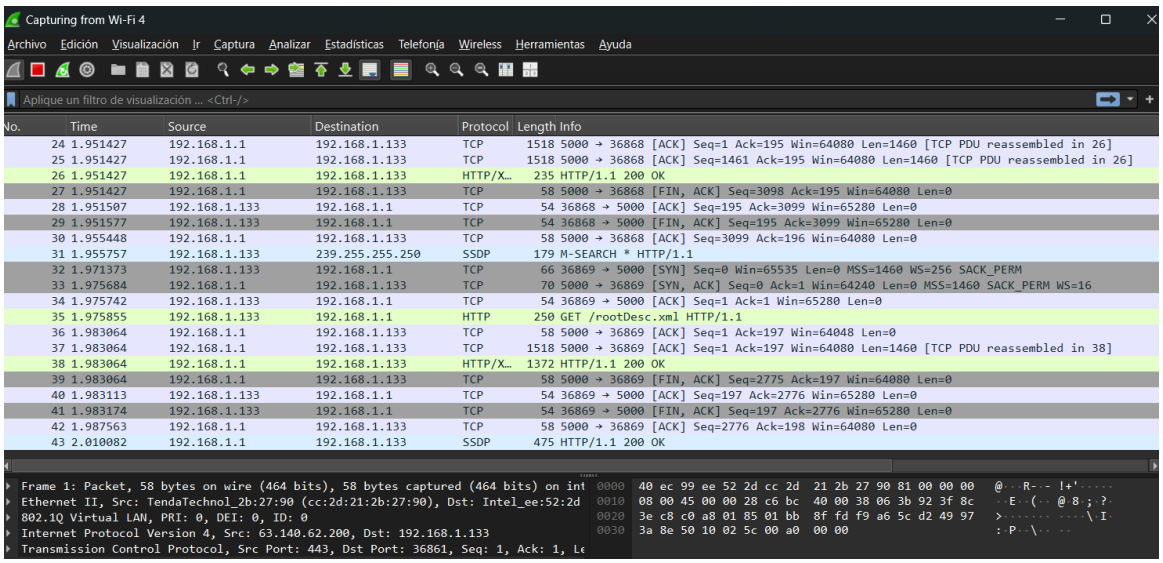
# 3. Procedimiento

## 3.1. Preparación

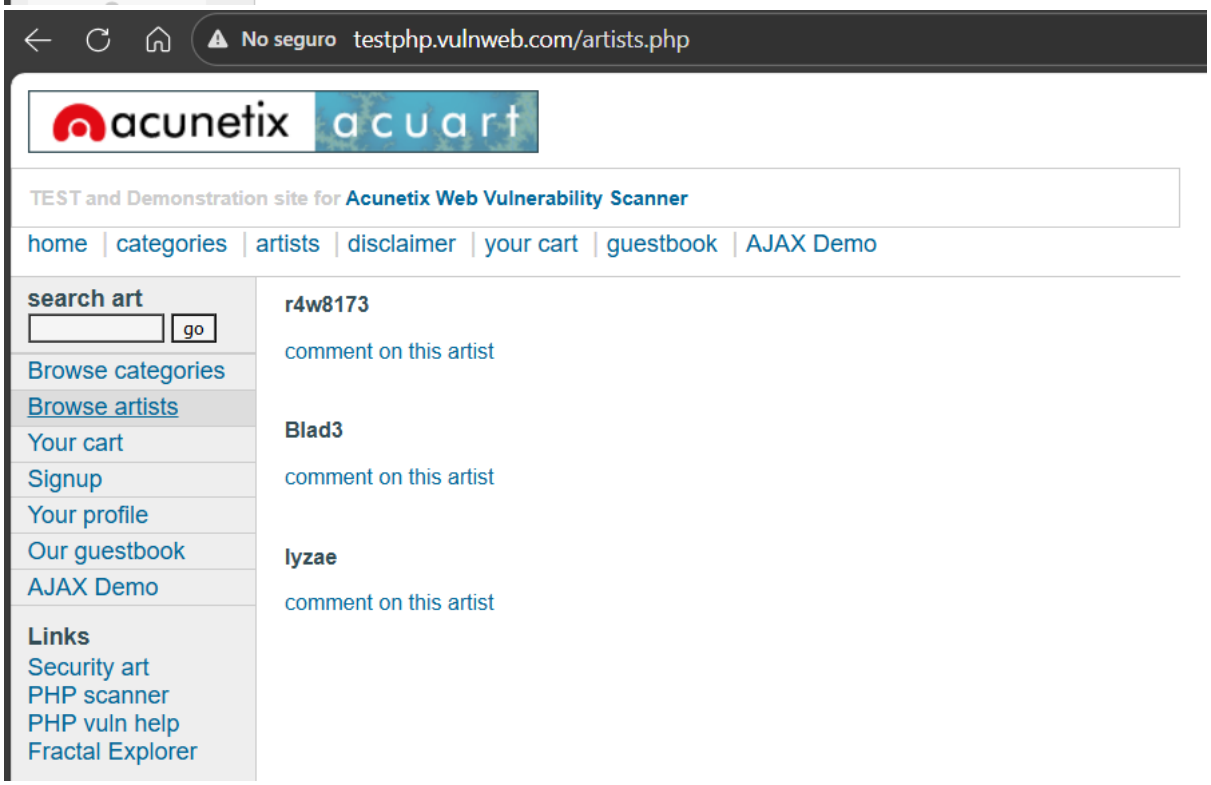
1. Abre **Wireshark**.
2. Selecciona la interfaz de red que utilices para navegar por Internet.
3. Inicia la captura de paquetes.
4. Abre el navegador y visita:

http://testphp.vulnweb.com/

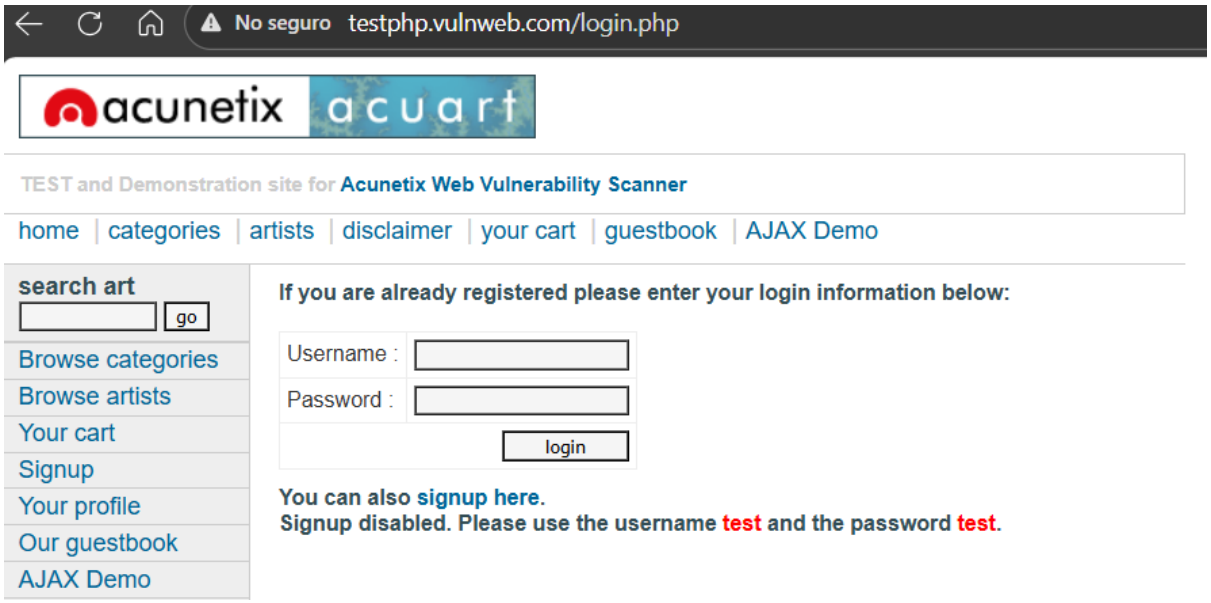
5. Navega por distintas secciones: categorías, productos, artista, carrito...
6. Accede al formulario de login e introduce cualquier usuario y contraseña (fallará siempre, es parte del diseño).
7. Realiza varias acciones para generar tráfico suficiente.
8. Regresa a Wireshark y detén la captura.

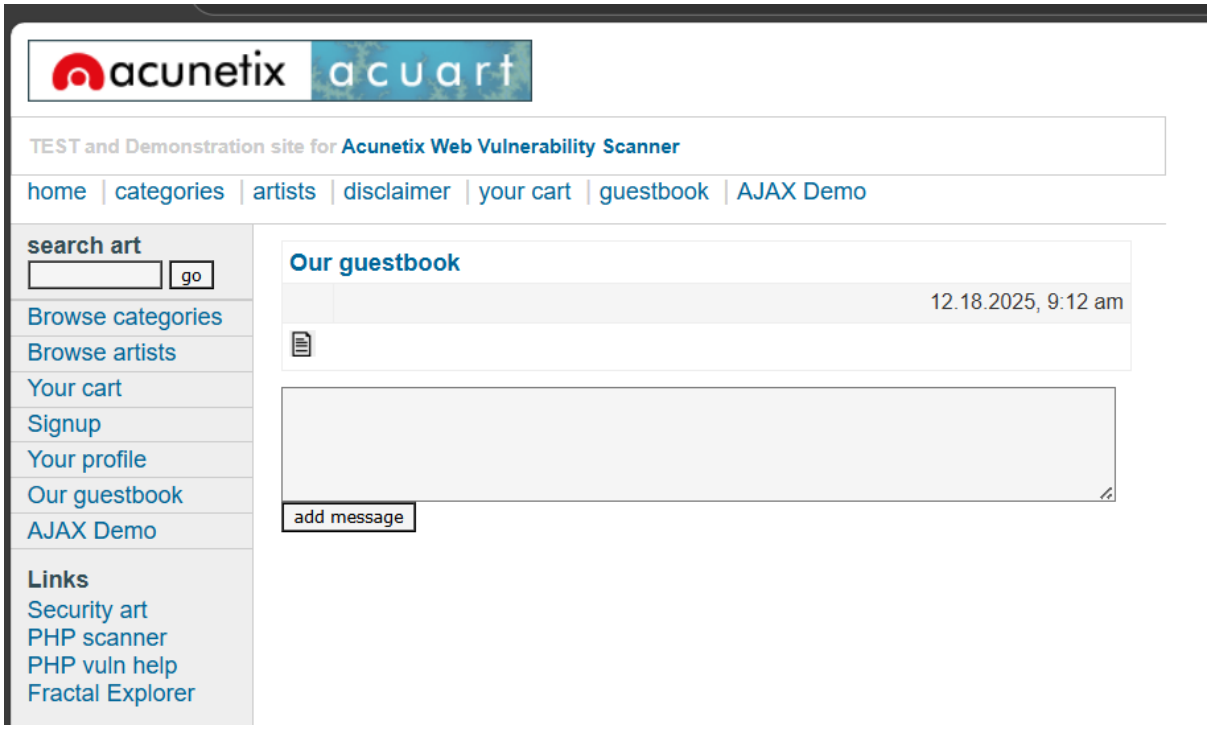



No seguro testphp.vulnweb.com/categories.php
acunetix acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
categories
Posters
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin. Vestibulum condimentum facilisis nulla. In hac habitasse platea dictumst. Nulla nonummy. Cras quis libero. Cras venenati
Paintings
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin. Vestibulum condimentum facilisis nulla. In hac habitasse platea dictumst. Nulla nonummy. Cras quis libero. Cras venenati
Stickers
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin. Vestibulum condimentum facilisis nulla. In hac habitasse platea dictumst. Nulla nonummy. Cras quis libero. Cras venenati
Graffity
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin. Vestibulum condimentum facilisis nulla. In hac habitasse platea dictumst. Nulla nonummy. Cras quis libero. Cras venenati


No seguro testphp.vulnweb.com/artists.php
acunetix acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
r4w8173
comment on this artist
Blad3
comment on this artist
lyzae
comment on this artist


acunetix acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
Error
You are not logged on. To log on please visit our login page


No seguro testphp.vulnweb.com/login.php
acunetix acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
If you are already registered please enter your login information below:
Username :
Password :
login
You can also signup here.
Signup disabled. Please use the username test and the password test.




# 4. Análisis guiado en Wireshark

A continuación aplicarás varios filtros y analizarás lo que ocurre en cada caso.

## 4.1. Visualizar únicamente tráfico HTTP

Filtro:

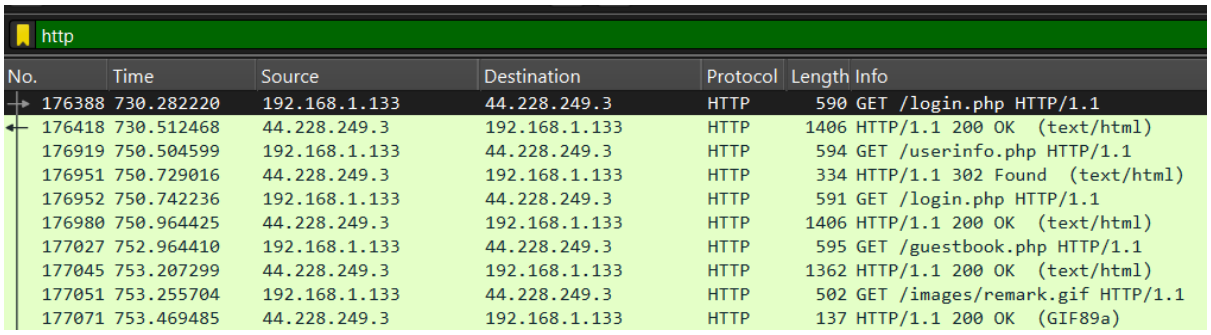
```
http
```

Qué debes observar:

- Todas las peticiones realizadas al servidor.
- Rutas como `/index.php`, `/listproducts.php`, `/product.php`, etc.
- Imágenes, scripts y otros recursos solicitados.

Ejemplo típico que deberías encontrar:

```
GET /listproducts.php?cat=1 HTTP/1.1
Host: testphp.vulnweb.com
```




## 4.2. Listar únicamente peticiones GET

Filtro:

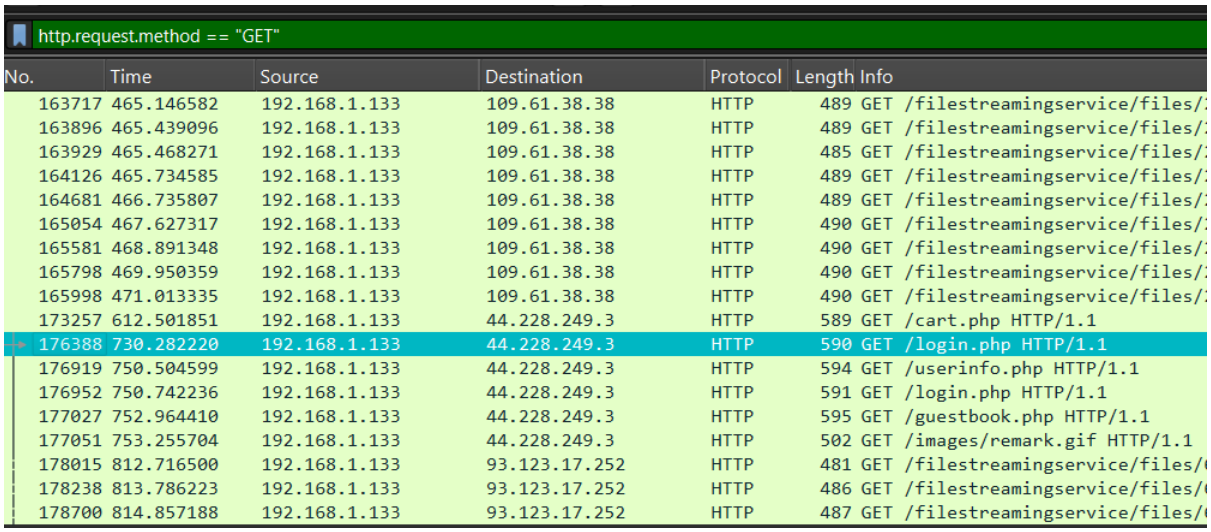
```
http.request.method == "GET"
```

Observa cómo la web utiliza parámetros en la URL. Ejemplos que verás:

```
GET /artist.php?artist=4
GET /product.php?pic=3
```

```
GET /listproducts.php?cat=2
```

Esto permite mapear la estructura del sitio.



http.request.method == "GET"

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 163717 | 465.146582 | 192.168.1.133 | 109.61.38.38 | HTTP | 489 | GET /filestreamingservice/files/ |
| 163896 | 465.439096 | 192.168.1.133 | 109.61.38.38 | HTTP | 489 | GET /filestreamingservice/files/ |
| 163929 | 465.468271 | 192.168.1.133 | 109.61.38.38 | HTTP | 485 | GET /filestreamingservice/files/ |
| 164126 | 465.734585 | 192.168.1.133 | 109.61.38.38 | HTTP | 489 | GET /filestreamingservice/files/ |
| 164681 | 466.735807 | 192.168.1.133 | 109.61.38.38 | HTTP | 489 | GET /filestreamingservice/files/ |
| 165054 | 467.627317 | 192.168.1.133 | 109.61.38.38 | HTTP | 490 | GET /filestreamingservice/files/ |
| 165581 | 468.891348 | 192.168.1.133 | 109.61.38.38 | HTTP | 490 | GET /filestreamingservice/files/ |
| 165798 | 469.950359 | 192.168.1.133 | 109.61.38.38 | HTTP | 490 | GET /filestreamingservice/files/ |
| 165998 | 471.013335 | 192.168.1.133 | 109.61.38.38 | HTTP | 490 | GET /filestreamingservice/files/ |
| 173257 | 612.501851 | 192.168.1.133 | 44.228.249.3 | HTTP | 589 | GET /cart.php HTTP/1.1 |
| 176388 | 730.282220 | 192.168.1.133 | 44.228.249.3 | HTTP | 590 | GET /login.php HTTP/1.1 |
| 176919 | 750.504599 | 192.168.1.133 | 44.228.249.3 | HTTP | 594 | GET /userinfo.php HTTP/1.1 |
| 176952 | 750.742236 | 192.168.1.133 | 44.228.249.3 | HTTP | 591 | GET /login.php HTTP/1.1 |
| 177027 | 752.964410 | 192.168.1.133 | 44.228.249.3 | HTTP | 595 | GET /guestbook.php HTTP/1.1 |
| 177051 | 753.255704 | 192.168.1.133 | 44.228.249.3 | HTTP | 502 | GET /images/remark.gif HTTP/1.1 |
| 178015 | 812.716500 | 192.168.1.133 | 93.123.17.252 | HTTP | 481 | GET /filestreamingservice/files/ |
| 178238 | 813.786223 | 192.168.1.133 | 93.123.17.252 | HTTP | 486 | GET /filestreamingservice/files/ |
| 178700 | 814.857188 | 192.168.1.133 | 93.123.17.252 | HTTP | 487 | GET /filestreamingservice/files/ |

## 4.3. Detectar peticiones con parámetros

Filtro:

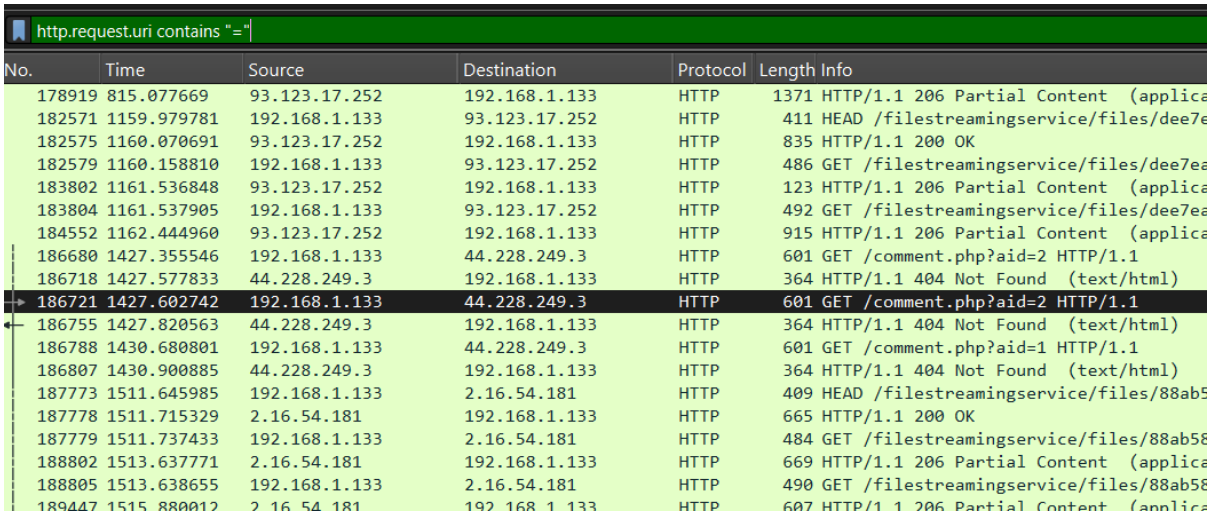
```
http.request.uri contains "="
```

Este filtro muestra exclusivamente peticiones con parámetros GET.

Ejemplos esperados:

```
GET /shoppingcart.php?add=2
GET /login.php?test=1
```

Fíjate en cómo la información viaja en texto plano.




# 4.4. Analizar el login (peticiones POST)

En la web, introduce un usuario y contraseña en el formulario.

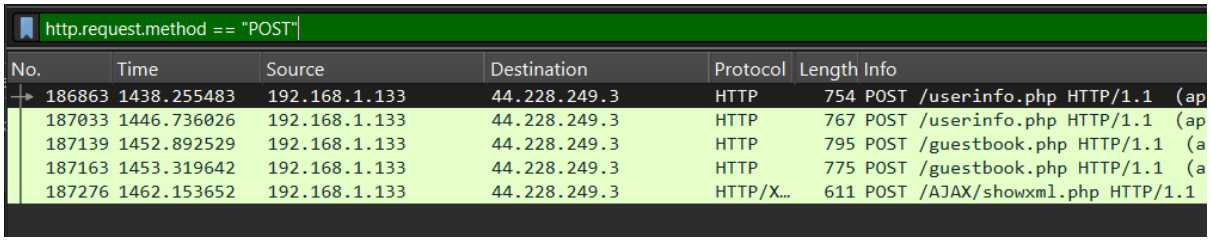
Filtro:

```
http.request.method == "POST"
```

Debes encontrar una petición similar a:

```
POST /userinfo.php HTTP/1.1
Content-Type: application/x-www-form-urlencoded

uname=prueba&pass=1234
```



Reflexiona sobre el riesgo de enviar credenciales por HTTP sin cifrar.

## 4.5. Visualizar cookies y sesiones

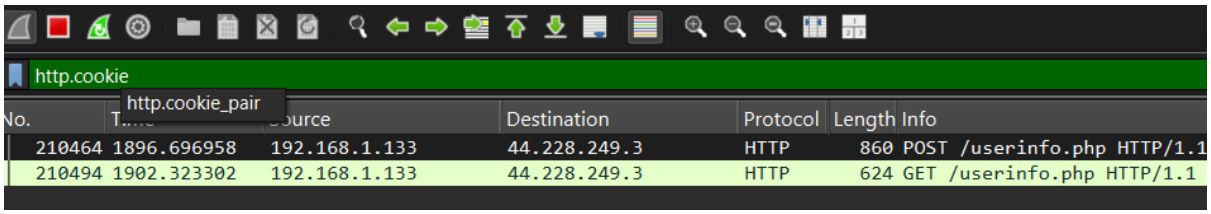
Filtro:

```
http.cookie
```

Deberías ver algo como:

```
Cookie: PHPSESSID=8d6v93h3k8a...
```

Esto confirma que la sesión también viaja sin protección alguna.




## 4.6. Encontrar errores en la web

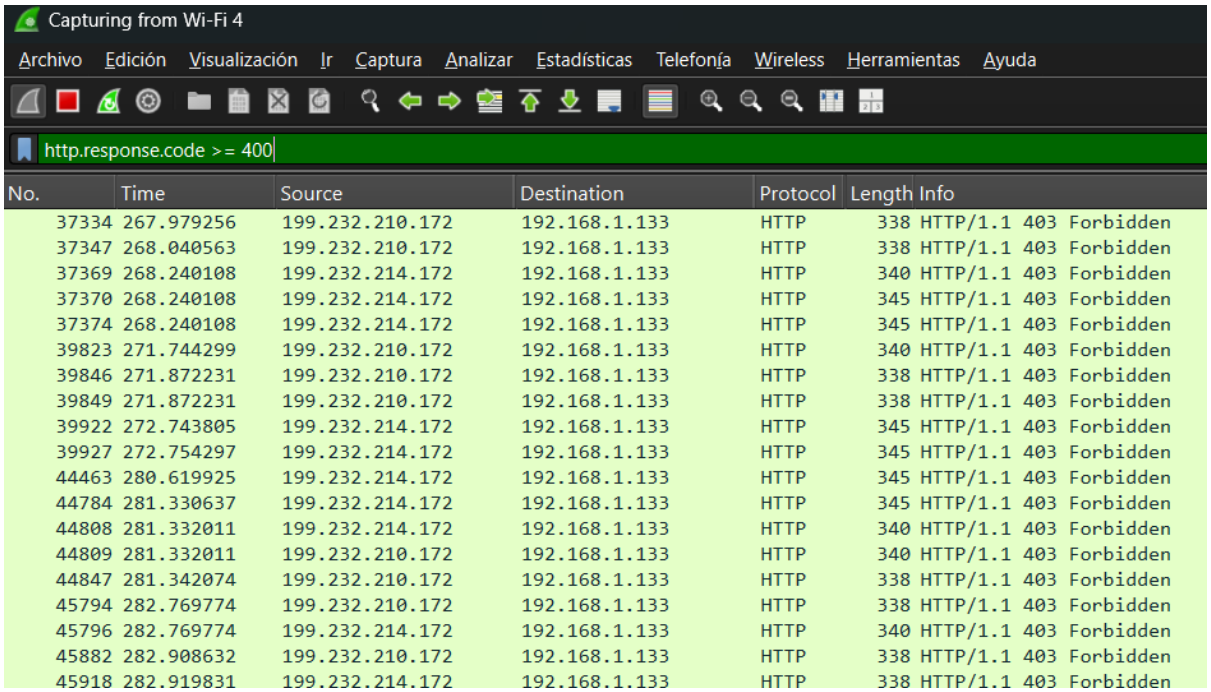
Filtro:

```
http.response.code >= 400
```

Este filtro permite localizar:

- Páginas no encontradas (404)
- Errores internos (500)
- Accesos no permitidos (403)

Esto ayuda a entender la estructura interna del sitio.

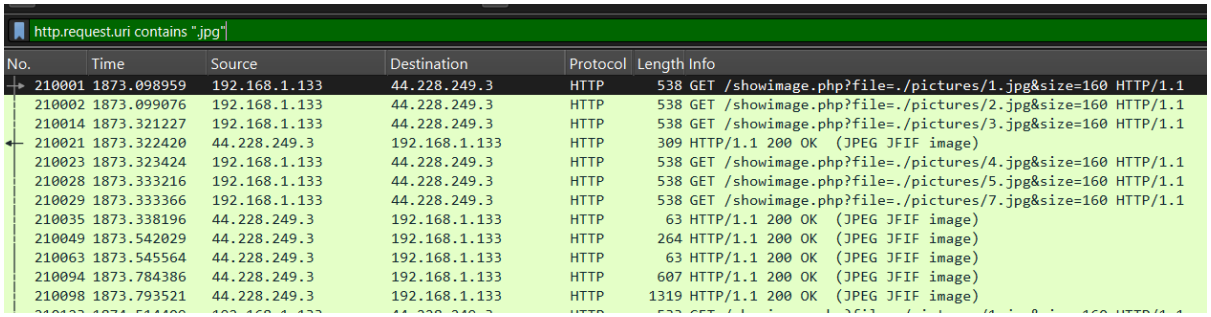



## 4.7. Localizar recursos concretos: imágenes, JS y CSS

Para analizar recursos específicos:

**Imágenes (JPG):**

```
http.request.uri contains ".jpg"
```

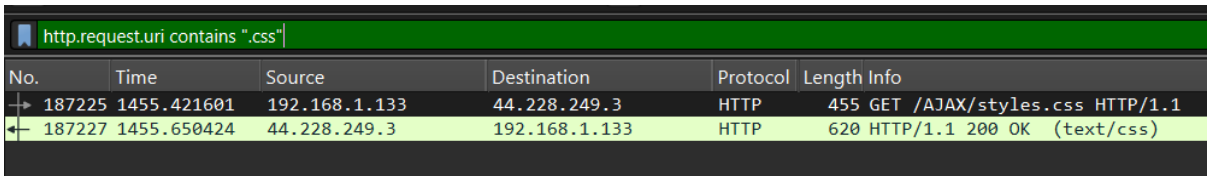



**Scripts:**

```
http.request.uri contains ".js"
```

**Hojas de estilo:**

```
http.request.uri contains ".css"
```




Permite reconstruir exactamente todo lo que carga el navegador.

# 4.8. Seguir una conversación completa

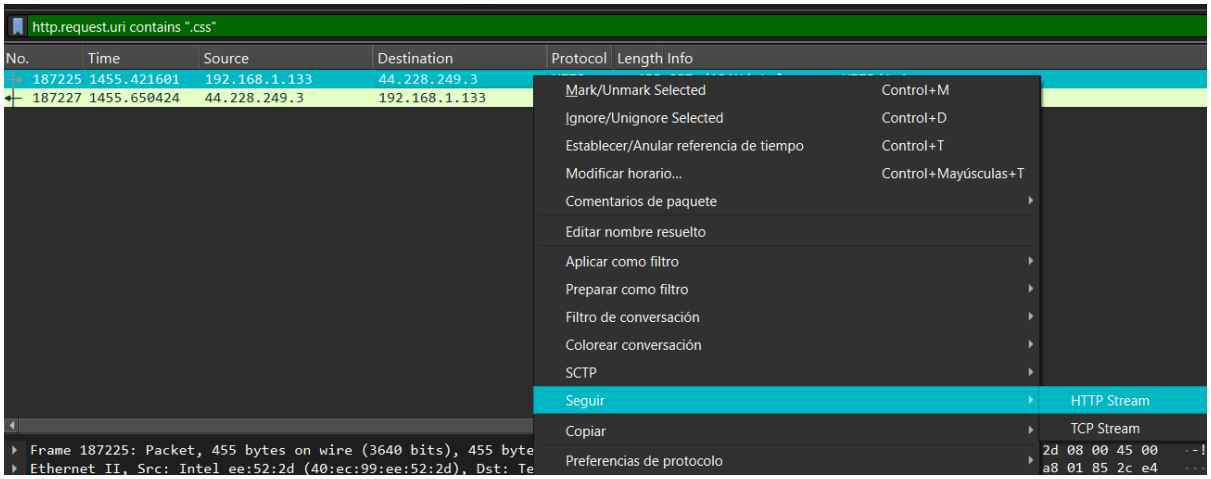
Selecciona cualquier paquete HTTP → clic derecho → **Follow → HTTP Stream**

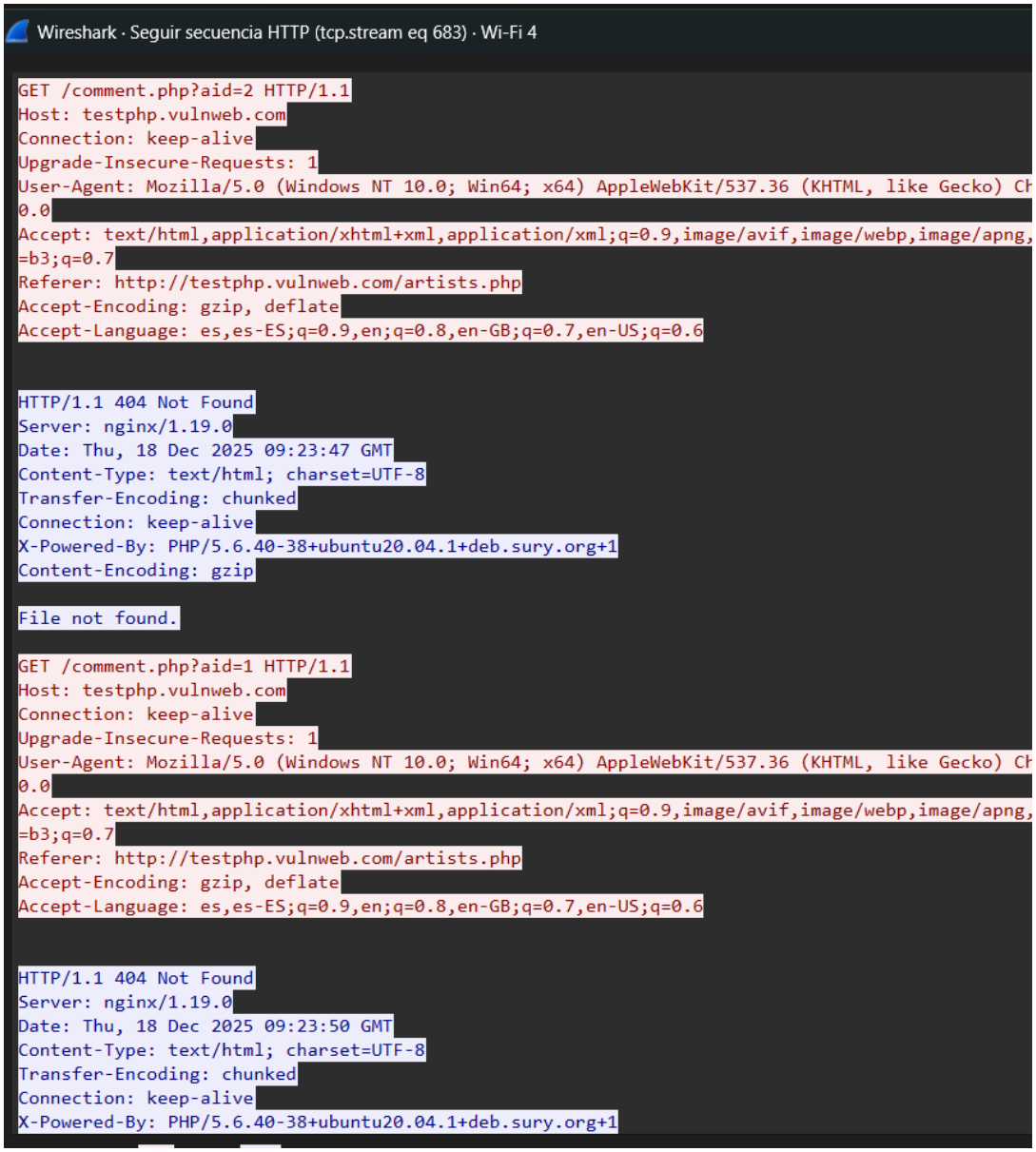
Verás la conversación completa entre cliente y servidor:

- Petición completa

- Respuesta completa
- HTML enviado

Perfecto para reconstruir acciones exactas del usuario.




Wireshark · Seguir secuencia HTTP (tcp.stream eq 683) · Wi-Fi 4
GET /comment.php?aid=2 HTTP/1.1
Host: testphp.vulnweb.com
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Ch
0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,
=b3;q=0.7
Referer: http://testphp.vulnweb.com/artists.php
Accept-Encoding: gzip, deflate
Accept-Language: es,es-ES;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
HTTP/1.1 404 Not Found
Server: nginx/1.19.0
Date: Thu, 18 Dec 2025 09:23:47 GMT
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
Content-Encoding: gzip
File not found.
GET /comment.php?aid=1 HTTP/1.1
Host: testphp.vulnweb.com
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Ch
0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,
=b3;q=0.7
Referer: http://testphp.vulnweb.com/artists.php
Accept-Encoding: gzip, deflate
Accept-Language: es,es-ES;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
HTTP/1.1 404 Not Found
Server: nginx/1.19.0
Date: Thu, 18 Dec 2025 09:23:50 GMT
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1