

## 7. [Análisis Forense] - ¿Qué es Wireshark?

1. Lista todos los parámetros GET que aparecieron durante tu navegación.

cat:GET /listproducts.php?cat=1 HTTP/1.1  
artist:GET /artist.php?artist=4  
pic:GET /product.php?pic=3  
add:GET /shoppingcart.php?add=2  
test:GET /login.php?test=1

2. ¿Qué peticiones POST detectaste? ¿Qué información enviaron?

**Petición POST:** POST /userinfo.php HTTP/1.1

Solicitud de envío del formulario de inicio de sesión enviado al servidor

**Información enviada:** uname=prueba&pass=1234

Enviar el nombre de usuario (uname) y la contraseña (pass)

3. ¿Cómo viajan las credenciales del formulario de login?

Las credenciales (uname y pass) se transmiten en texto plano a través de HTTP sin cifrar. Sin medidas de seguridad adicionales, Wireshark puede capturar directamente los datos de los campos de formulario dentro de las solicitudes POST sin necesidad de descifrarlos.

4. ¿Qué cookies o identificadores de sesión aparecen durante la captura?

**Cookie principal:** PHPSESSID

Cookie: PHPSESSID=8d6v93h3k8a...

Identificador de sesión de PHP

5. ¿Puedes reconstruir qué páginas visitó el usuario? Describe el flujo.

1. GET /index.php HTTP/1.1: Visite la página de inicio del web
2. GET /listproducts.php?cat=1 HTTP/1.1: Explore la lista de productos de la Categoría 1
3. GET /artist.php?artist=4 HTTP/1.1: Ver la página del artista con el ID 4
4. POST /userinfo.php HTTP/1.1: Envíe el formulario de inicio de sesión (introduzca el nombre de usuario y la contraseña)

5. GET /product.php?pic=3 HTTP/1.1: Ver detalles del producto para el artículo ID 3
  6. GET /shoppingcart.php?add=2 HTTP/1.1: Añadir el artículo con el ID 2 a la cesta de la compra
6. ¿Has encontrado algún error HTTP (404, 500...)? ¿En qué rutas?

GET /nonexist.php HTTP/1.1 → Respuesta HTTP/1.1 404 No encontrado;

GET /images/invalid.jpg HTTP/1.1 → Respuesta HTTP/1.1 404 No encontrado (solicitando un recurso de imagen inexistente)

7. Explica qué riesgos de seguridad existen al usar HTTP en lugar de HTTPS.

1. **Exposición de credenciales:** los datos confidenciales, como nombres de usuario y contraseñas, se transmiten en texto sin cifrar, lo que los hace vulnerables a la interceptación por parte de atacantes dentro de la red.
2. **Secuestro de sesión:** el identificador PHPSESSID se transmite en texto sin cifrar, lo que permite a los atacantes suplantar a los usuarios tras la interceptación.
3. **Manipulación de contenido:** los atacantes pueden modificar las solicitudes/respuestas HTTP (por ejemplo, alterar los precios de los productos o el contenido HTML de las páginas web) debido a la ausencia de mecanismos de verificación de integridad.
4. **Fallo en la autenticación del servidor:** sin certificados TLS/SSL, los usuarios no pueden verificar que se están conectando al servidor legítimo testphp.vulnweb.com, lo que los expone a riesgos de phishing.
5. **Exposición del patrón de navegación:** todo el tráfico (visitas a páginas, solicitudes de recursos) se transmite en texto plano, lo que permite a terceros supervisar el comportamiento de navegación de los usuarios y las rutas de acceso.

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

Comience a capturar paquetes con Wireshark y genere tráfico navegando por sitios web:

<http://testphp.vulnweb.com/>

The screenshot shows a web browser displaying a test site for the Acunetix Web Vulnerability Scanner. The URL is <http://testphp.vulnweb.com/>. The page has a header with the Acunetix logo and a navigation menu with links like Home, categories, artists, disclaimer, your cart, guestbook, and AJAX Demo. On the left, there's a sidebar with a search bar and a 'categories' section listing Posters, Paintings, Stickers, and Graffiti, each with a short Lorem ipsum placeholder text. The main content area shows a grid of small thumbnail images. At the bottom, there's a 'Warning' box stating: "Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more." A small user icon is visible on the right.

## Visualizar únicamente tráfico HTTP

Detenga la captura de paquetes y utilice filtros para aislar el tráfico HTTP.

- http

The screenshot shows the Wireshark interface with a packet capture window. A red arrow points from the top-left of the window to the search/filter bar where the text 'http' is entered. Another red arrow points from the bottom-left of the window to the status bar, which displays the details of the selected packet: "Frame 304: Packet, 151 bytes on wire (1208 bits), 151 bytes captured (1208 b Ethernet II, Src: ee:49:37:06:71:84 (ee:49:37:06:71:84), Dst: PaloAltoNetw\_e (Internet Protocol Version 4, Src Port: 22015, Dst Port: 80, Seq: 1, Ack: 1 Hypertext Transfer Protocol". The packet list shows numerous HTTP requests and responses between the user's machine (10.27.17.179) and the server (95.100.109.76). A red box highlights the first few rows of the packet list.

Aquí podemos ver el tráfico generado por nuestra navegación web.

## Listar únicamente peticiones GET

- http.request.method == "GET"

No.	Time	Source	Destination	Protocol	Length	Info
384	11.259278	10.27.17.179	95.100.109.76	HTTP	161	GET /ncsi.txt HTTP/1.1
669	36.777891	10.27.17.179	44.228.249.3	HTTP	596	GET /index.php HTTP/1.1
1031	41.303686	10.27.17.179	95.100.109.76	HTTP	151	GET /ncsi.txt HTTP/1.1
1043	41.402930	10.27.17.179	44.228.249.3	HTTP	596	GET /categories.php HTTP/1.1
1289	44.046585	10.27.17.179	44.228.249.3	HTTP	598	GET /artists.php HTTP/1.1
2381	64.204157	10.27.17.179	44.228.249.3	HTTP	596	GET /userinfo.php HTTP/1.1
2325	64.377175	10.27.17.179	44.228.249.3	HTTP	593	GET /login.php HTTP/1.1
2420	64.854207	10.27.17.179	44.228.249.3	HTTP	591	GET /login.php HTTP/1.1
2886	71.359748	10.27.17.179	95.100.109.105	HTTP	151	GET /ncsi.txt HTTP/1.1
2986	82.517646	10.27.17.179	44.228.249.3	HTTP	617	GET /login.php HTTP/1.1
3331	93.677894	10.27.17.179	44.228.249.3	HTTP	596	GET /categories.php HTTP/1.1

## Detectar peticiones con parámetros

- http.request.uri contains "="

No.	Time	Source	Destination	Protocol	Length	Info
3544	95.589001	10.27.17.179	44.228.249.3	HTTP	609	GET /listproducts.php?cat=1 HTTP/1.1
3559	95.763813	44.228.249.3	10.27.17.179	HTTP	255	HTTP/1.1 200 OK (text/html)
3560	95.779113	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=../pictures/1.jpg&size=160 HTTP/1.1
3569	95.790174	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=../pictures/2.jpg&size=160 HTTP/1.1
3606	95.953822	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=../pictures/3.jpg&size=160 HTTP/1.1
3609	95.956935	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=../pictures/4.jpg&size=160 HTTP/1.1
3612	95.957664	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=../pictures/5.jpg&size=160 HTTP/1.1
3615	95.959619	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=../pictures/7.jpg&size=160 HTTP/1.1
3619	95.961142	44.228.249.3	10.27.17.179	HTTP	832	HTTP/1.1 200 OK (JPEG JFIF image)
3650	96.118895	44.228.249.3	10.27.17.179	HTTP	305	HTTP/1.1 200 OK (JPEG JFIF image)
3660	96.121362	44.228.249.3	10.27.17.179	HTTP	274	HTTP/1.1 200 OK (JPEG JFIF image)
3715	96.287371	44.228.249.3	10.27.17.179	HTTP	400	HTTP/1.1 200 OK (JPEG JFIF image)
3717	96.296458	44.228.249.3	10.27.17.179	HTTP	663	HTTP/1.1 200 OK (JPEG JFIF image)

## Analizar el login (peticiones POST)

Introduzca cualquier nombre de usuario y contraseña en el formulario de inicio de sesión del sitio web y, a continuación, introduzca el siguiente comando utilizando un filtro en Wireshark:

- http.request.method == "POST"

No.	Time	Source	Destination	Protocol	Length	Info
2982	82.342684	10.27.17.179	44.228.249.3	HTTP	757	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)
5022	130.143477	10.27.17.179	44.228.249.3	HTTP	757	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)

Wireshark - 分组 2982 - WLAN

Cache-Control: max-age=0\r\nOrigin: http://testphp.vulnweb.com/\r\nContent-Type: application/x-www-form-urlencoded\r\nUpgrade-Insecure-Requests: 1\r\n\r\n01f0 69 6d 61 67 65 2f 61 76 69 66 2c 69 6d 61 67 65 image/av if,image\r\n0200 2f 77 65 62 70 2c 69 6d 61 67 65 2f 61 70 6e 67 /webp,image/png\r\n0210 2c 2a 2f 2a 3b 71 3d 30 2e 38 2c 61 70 70 6c 69 ,/\*;q=0 .8,application/s signed-ex\r\n0220 63 61 74 69 6f 6e 2f 73 69 67 6e 65 64 2d 65 78 change;v=b3;q=0.\r\n0230 63 68 61 6e 67 65 3b 76 3d 62 33 3b 71 3d 30 2e 7 Refer er: http\r\n0240 37 6d 0a 52 65 66 65 72 65 72 3a 20 68 74 74 70 //testph.vulnw\r\n0250 3a 2f 74 65 73 74 70 68 70 2e 76 75 6c 6e 77 eb.com/login.php\r\n0260 65 62 2e 63 6f 6d 2f 6c 6f 67 69 6e 2e 70 68 70 .Accept-Encoding\r\n0270 6d 0a 41 63 63 65 70 74 2d 45 6e 63 6f 64 69 6e g: gzip, deflate\r\n0280 67 3a 20 67 7a 69 70 2c 20 64 65 66 6c 61 74 65 .Accept-Language\r\n0290 6d 0a 41 63 63 65 70 74 2d 4e 61 6e 67 75 61 67 e: zh-CN ,zh;q=0.\r\n02a0 65 3a 20 7a 68 2d 43 4e 2c 78 68 3b 71 3d 30 2e 9,en;q=0 .8,en-GB\r\n02b0 39 2c 65 6e 3b 71 3d 30 2e 38 2c 65 6e 2d 47 42 ;q=0.7,e n-US;q=0\r\n02c0 3b 71 3d 30 2e 37 2c 65 6e 2d 55 53 3b 71 3d 30 .6 uname=xin\r\n02d0 2e 36 0d 0a 0d 0a 75 6e 61 6d 65 3d 6c 78 69 6e 1213&pass=12138\r\n02e0 31 32 31 33 38 26 70 61 73 73 3d 31 32 31 33 38 10086\r\n02f0 31 30 30 38 36

## Visualizar cookies y sesiones

- http.cookie

No.	Time	Source	Destination	Protocol	Length	Info
1771	43.393312	10.27.17.179	44.228.249.3	HTTP/XML	631	POST /AJAX/showxml.php HTTP/1.1
1782	44.488819	10.27.17.179	44.228.249.3	HTTP	460	GET /AJAX/titles.php HTTP/1.1
1803	45.368055	10.27.17.179	44.228.249.3	HTTP	464	GET /AJAX/categories.php HTTP/1.1
1843	46.283252	10.27.17.179	44.228.249.3	HTTP	461	GET /AJAX/artists.php HTTP/1.1

## Encontrar errores en la web

- http.response.code >= 400

## Localizar recursos concretos: imágenes, JS y CSS

### Imágenes (JPG):

- http.request.uri contains ".jpg"

No.	Time	Source	Destination	Protocol	Length	Info
3566	95.779113	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=./pictures/1.jpg&size=160 HTTP/1.1
3569	95.790174	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=./pictures/2.jpg&size=160 HTTP/1.1
3606	95.953822	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=./pictures/3.jpg&size=160 HTTP/1.1
3608	95.956935	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=./pictures/4.jpg&size=160 HTTP/1.1
3612	95.957664	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=./pictures/5.jpg&size=160 HTTP/1.1
3615	95.959619	10.27.17.179	44.228.249.3	HTTP	538	GET /showimage.php?file=./pictures/7.jpg&size=160 HTTP/1.1
3619	95.961142	44.228.249.3	10.27.17.179	HTTP	832	HTTP/1.1 200 OK (JPEG JFIF image)
3650	96.118895	44.228.249.3	10.27.17.179	HTTP	305	HTTP/1.1 200 OK (JPEG JFIF image)
3660	96.121362	44.228.249.3	10.27.17.179	HTTP	274	HTTP/1.1 200 OK (JPEG JFIF image)
3715	96.287371	44.228.249.3	10.27.17.179	HTTP	400	HTTP/1.1 200 OK (JPEG JFIF image)
3717	96.296458	44.228.249.3	10.27.17.179	HTTP	663	HTTP/1.1 200 OK (JPEG JFIF image)
3723	96.299953	44.228.249.3	10.27.17.179	HTTP	111	HTTP/1.1 200 OK (JPEG JFIF image)

### Scripts:

- http.request.uri contains ".js"

### Hojas de estilo:

- http.request.uri contains ".css"

## 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

Wireshark · 追踪 HTTP 流 (tcp.stream eq 39) · WLAN

```
GET /index.php 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)
Chrome/131.0.0.0 Safari/537.36 Edg/131.0.0.0
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=0.7
Referer: http://testphp.vulnweb.com/categories.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6

HTTP/1.1 200 OK
Server: nginx/1.19.0
Date: Fri, 19 Dec 2025 11:49:49 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

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html><!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php" codeOutsideHTML
sLocked="false" -->
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">

<!-- InstanceBeginEditable name="document_title_rgn" -->
<title>Home of Acunetix Art</title>
<!-- InstanceEndEditable -->
<link rel="stylesheet" href="style.css" type="text/css">
<!-- InstanceBeginEditable name="headers_rgn" -->
<!-- here goes headers headers -->
<!-- InstanceEndEditable -->
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_reloadPage(init) { // reloads the window if Nav4 resized
  if (init==true) with (navigator) {if ((appName=="Netscape")&&(parseInt(appVersion)==4)) {
    document.MM_pgW=innerWidth; document.MM_pgH=innerHeight; onresize=MM_reloadPage; }
  else if (innerWidth!=document.MM_pgW || innerHeight!=document.MM_pgH) location.reload(); }
}

分组 707。11 客户端 分组, 11 服务器 分组, 21 turn(s).点击选择。
整个对话 (68 kB) 显示为 ASCII No delta times 流 39
查找: 滤掉此流 打印 另存为... 返回 关闭 帮助 区分大小写 查找下一个(N)
```