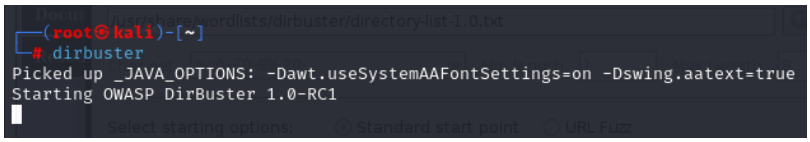
EJERCICIOS INTRODUCCIÓN WEB

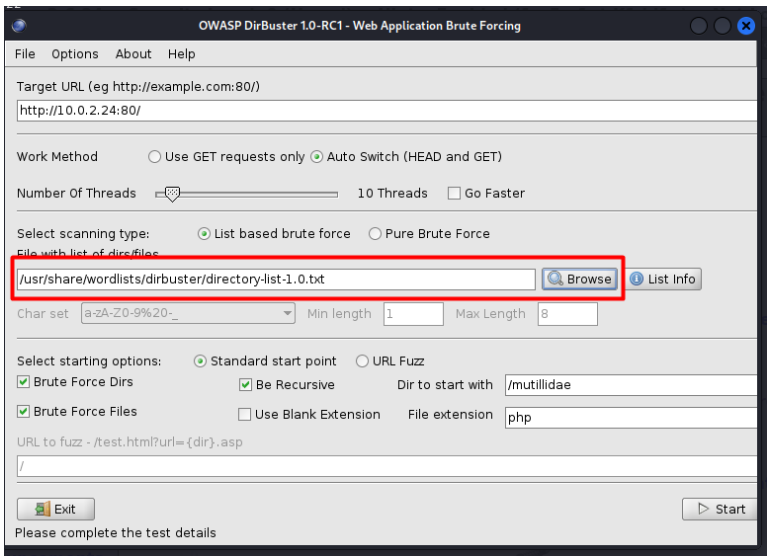
Prerrequisitos

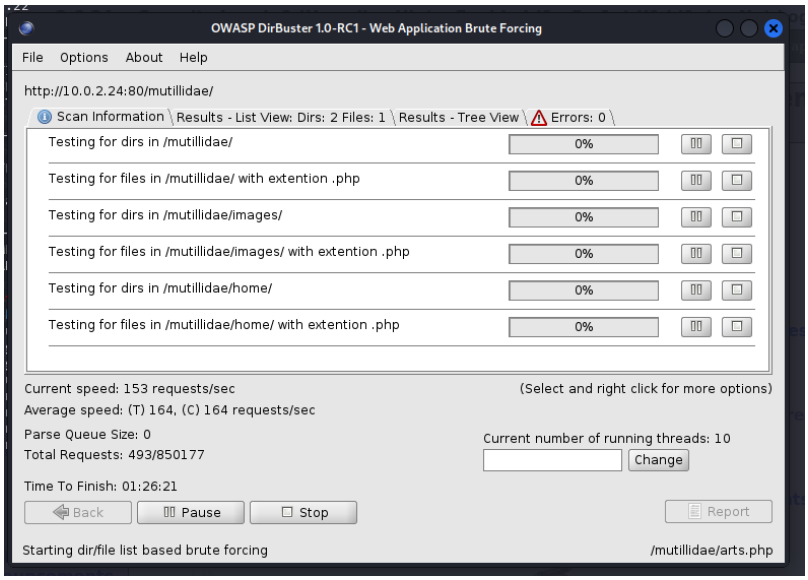
Kali Linux
OWASP BWA

Ejercicio 1 - Dirbuster

Realizar enumeración de la aplicación web Mutillidae II utilizando Dirbuster y un diccionario de directorios de tamaño medio







El tiempo para finalizar era de un día y por tanto lo he cancelado

Ejercicio 2 - Nikto, Nessus y OWASP Zap

Realizar un análisis de vulnerabilidades con Nikto a la aplicación web Mutillidae II volcando los resultados en un documento ".txt"

```
┌──(root㉿kali)-[~]
└─# nikto -host http://10.0.2.24/mutillidae/ > AnálisisVulnerabilidadesNikto.txt
```

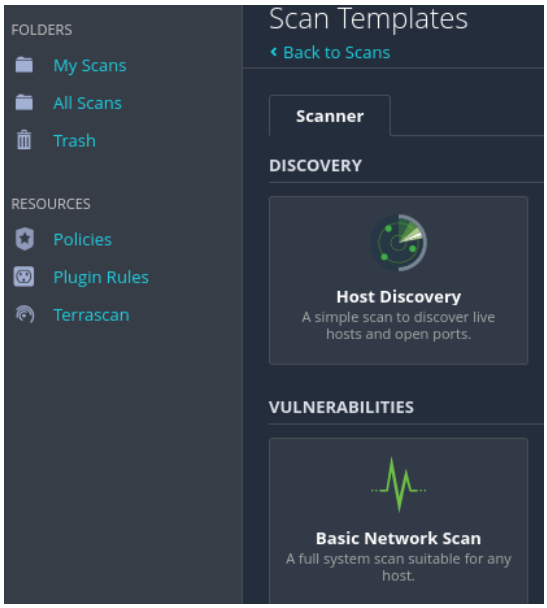
```
┌──(root㉿kali)-[~]
└─# cat AnálisisVulnerabilidadesNikto.txt
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.0.2.24
+ Target Hostname:    10.0.2.24
+ Target Port:        80
+ Start Time:         2023-10-16 15:15:47 (GMT2)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_pyt
hon/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
```

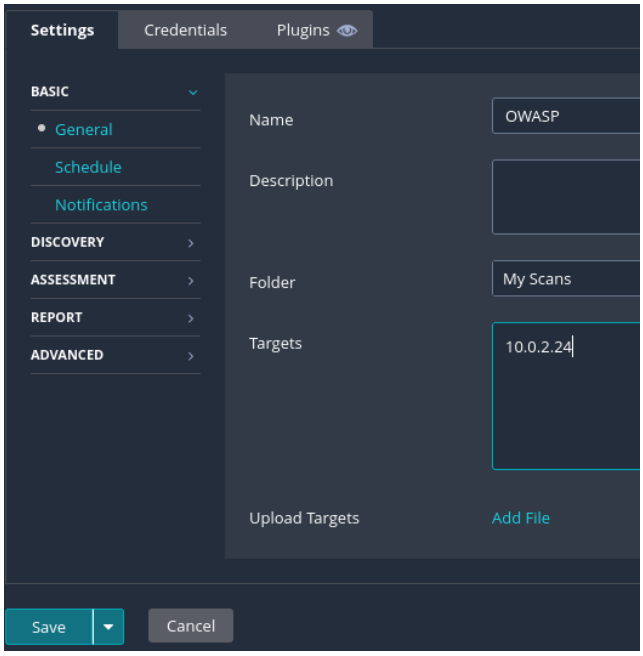
```
+ 8112 requests: 0 error(s) and 174 item(s) reported on remote host
+ End Time:           2023-10-16 15:16:43 (GMT2) (56 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

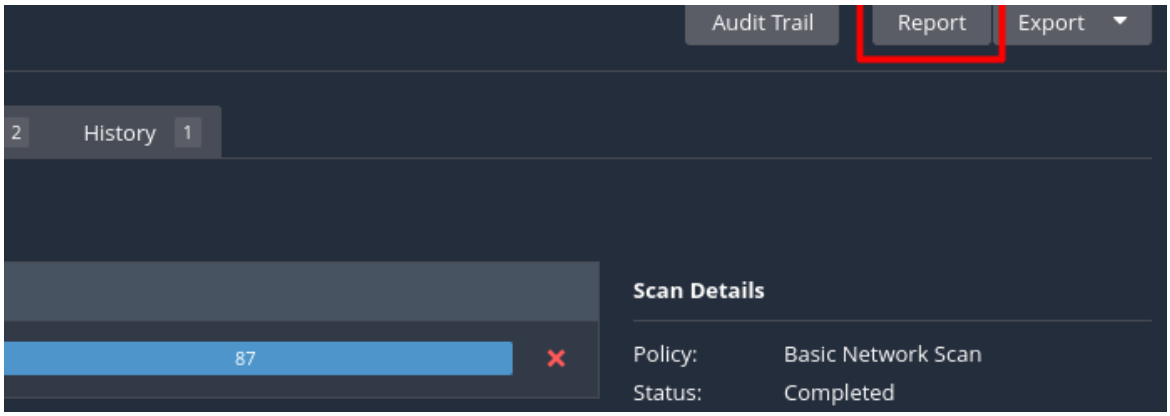
Realizar un análisis de vulnerabilidades con Nessus al servidor web OWASP BWA

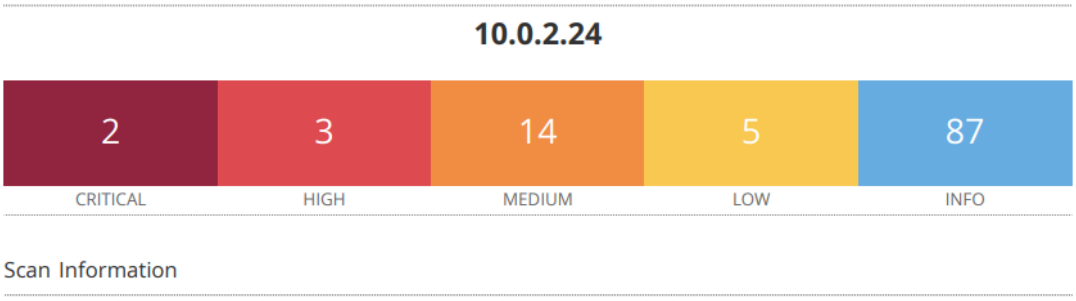
Iniciamos Nessus y entramos en nuestra cuenta para poder realizar un escaneo básico

```
┌──(root㉿kali)-[~]
└─# systemctl start nessusd
```

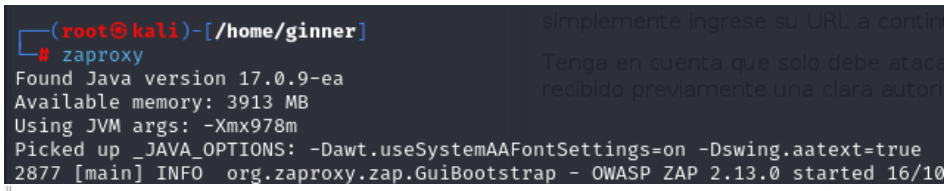


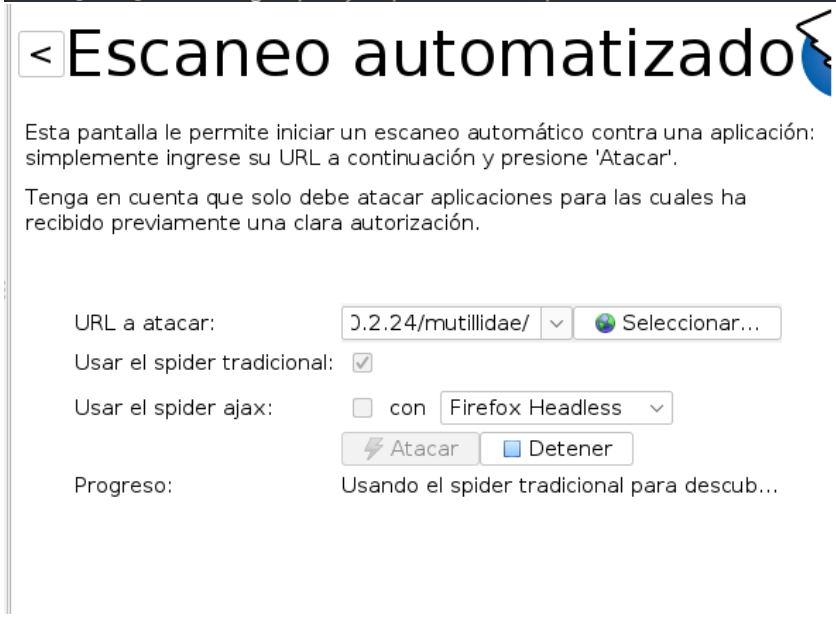


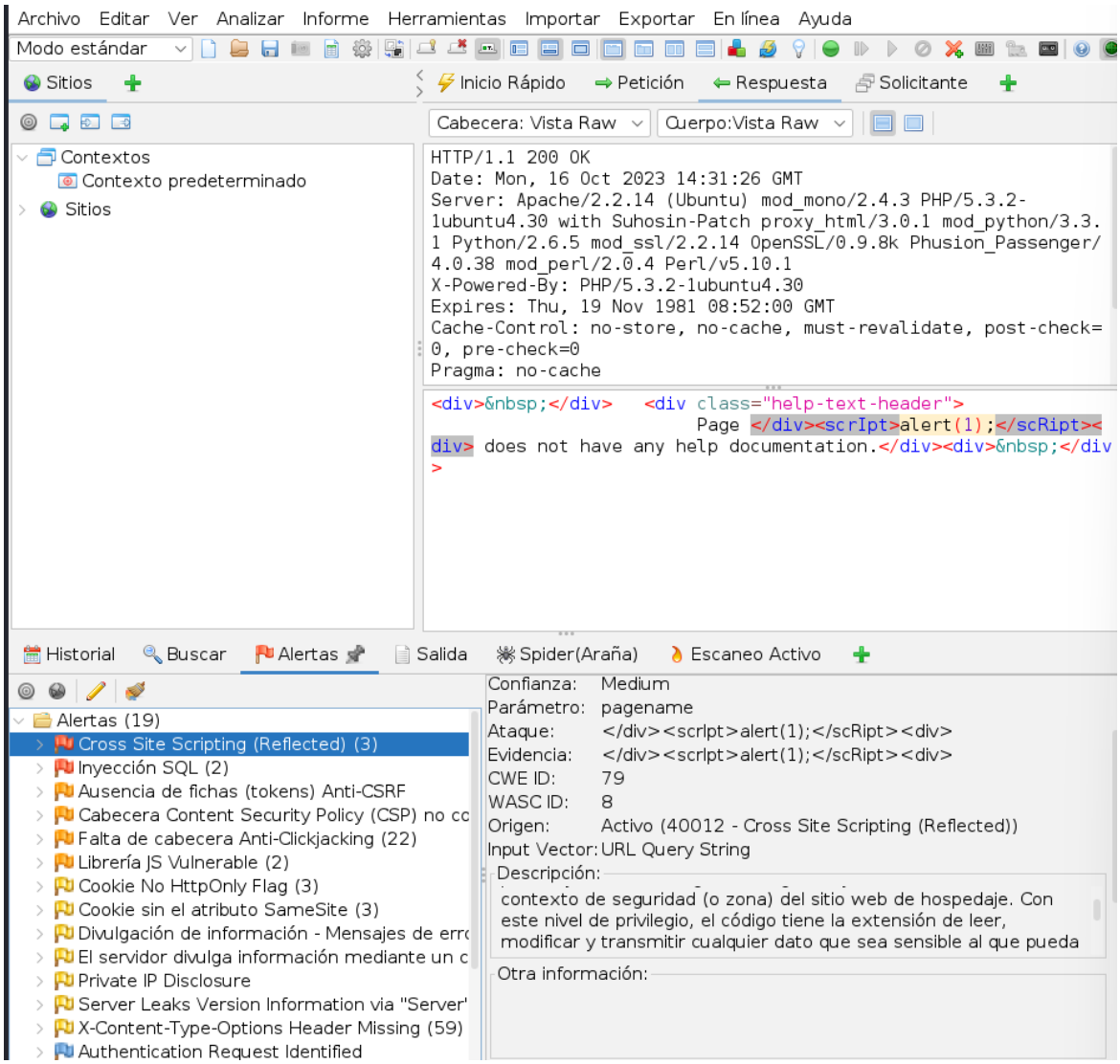




Realizar un análisis de vulnerabilidades con OWASP Zap a la aplicación web Mutillidae II






Archivo Editar Ver Analizar Informe Herramientas Importar Exportar En línea Ayuda
Modo estándar
Sitios
Inicio Rápido
Petición
Respuesta
Solicitante
Cabecera: Vista Raw
Cuerpo:Vista Raw
Contextos
Contexto predeterminado
Sitios
HTTP/1.1 200 OK
Date: Mon, 16 Oct 2023 14:31:26 GMT
Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
X-Powered-By: PHP/5.3.2-1ubuntu4.30
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache

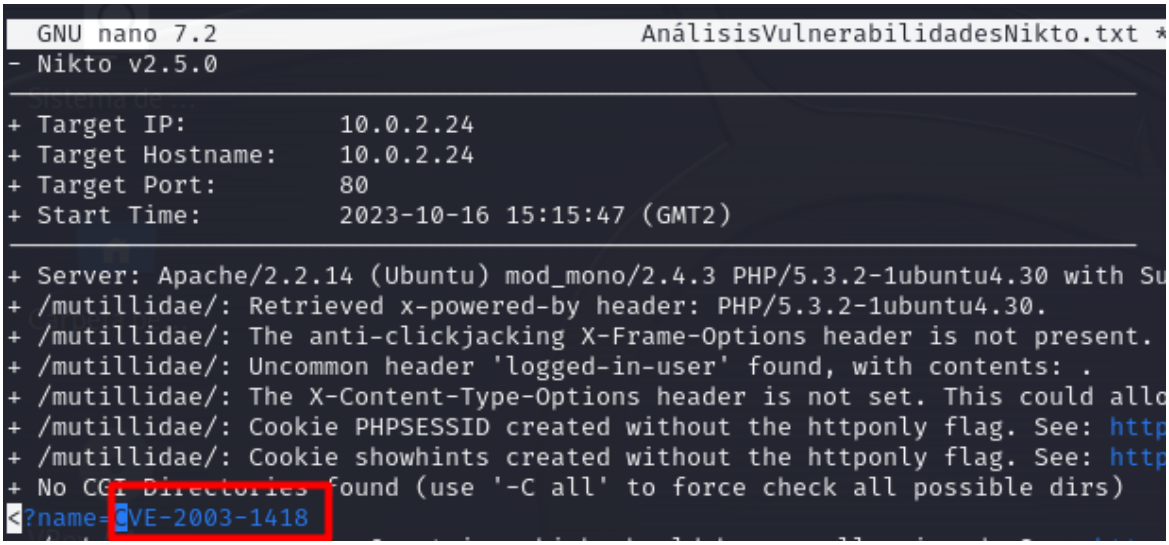
Page <scRipt>alert(1);</scRipt><
div> does not have any help documentation. </div
>
Historial
Buscar
Alertas
Salida
Spider(Araña)
Escaneo Activo
Alertas (19)
Cross Site Scripting (Reflected) (3)
Inyección SQL (2)
Ausencia de fichas (tokens) Anti-CSRF
Cabecera Content Security Policy (CSP) no cc
Falta de cabecera Anti-Clickjacking (22)
Librería JS Vulnerable (2)
Cookie No HttpOnly Flag (3)
Cookie sin el atributo SameSite (3)
Divulgación de información - Mensajes de erro
El servidor divulga información mediante un c
Private IP Disclosure
Server Leaks Version Information via "Server"
X-Content-Type-Options Header Missing (59)
Authentication Request Identified
Confianza: Medium
Parámetro: pagename
Ataque: <scrIpt>alert(1);</scRipt>
Evidencia: <scrIpt>alert(1);</scRipt>
CWE ID: 79
WASC ID: 8
Origen: Activo (40012 - Cross Site Scripting (Reflected))
Input Vector: URL Query String
Descripción:
contexto de seguridad (o zona) del sitio web de hospedaje. Con este nivel de privilegio, el código tiene la extensión de leer, modificar y transmitir cualquier dato que sea sensible al que pueda
Otra información:

Ejercicio 3 - Nikto, Nessus y OWASP Zap

- Evaluar los resultados del escaneo con Nikto y explicar en detalle alguna de las vulnerabilidades encontradas

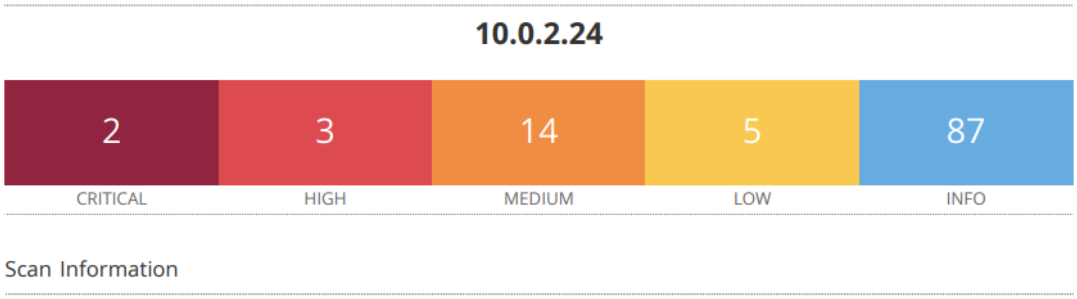
```
┌──(root㉿kali)-[~]
└─# cat AnálisisVulnerabilidadesNikto.txt
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.0.2.24
+ Target Hostname:    10.0.2.24
+ Target Port:        80
+ Start Time:         2023-10-16 15:15:47 (GMT2)
---------------------------------------------------------------------------
+ Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_pyt
hon/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1
+ /mutillidae/: Retrieved x-powered-by header: PHP/5.3.2-1ubuntu4.30.
+ /mutillidae/: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/e
n-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /mutillidae/: Uncommon header 'logged-in-user' found, with contents: .
+ /mutillidae/: The X-Content-Type-Options header is not set. This could allow the user agent to render the conte
nt of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner
/vulnerabilities/missing-content-type-header/
+ /mutillidae/: Cookie PHPSESSID created without the httponly flag. See: https://developer.mozilla.org/en-US/docs
/Web/HTTP/Cookies
+ /mutillidae/: Cookie showhints created without the httponly flag. See: https://developer.mozilla.org/en-US/docs
/Web/HTTP/Cookies
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /mutillidae/robots.txt: Server may leak inodes via ETags, header found with file /mutillidae/robots.txt, inode:
 389642, size: 190, mtime: Fri Sep 27 04:47:08 2013. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-
1418
+ /robots.txt: contains 8 entries which should be manually viewed. See: https://developer.mozilla.org/en-US/docs/
Glossary/Robots.txt
+ /mutillidae/index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file na
mes. The following alternatives for 'index' were found: index.php. See: http://www.wisec.it/sectou.php?id=4698ebd
c59d15,https://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ /images: The web server may reveal its internal or real IP in the Location header via a request to with HTTP/1.
0. The value is "127.0.1.1". See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0649
+ OpenSSL/0.9.8k appears to be outdated (current is at least 3.0.7). OpenSSL 1.1.1s is current for the 1.x branch
 and will be supported until Nov 11 2023.
+ Python/2.6.5 appears to be outdated (current is at least 3.9.6).
+ Phusion_Passenger/4.0.38 appears to be outdated (current is at least 6.0.7).
+ mod_mono/2.4.3 appears to be outdated (current is at least 3.12).
+ Apache/2.2.14 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x
branch.
+ mod_python/3.3.1 appears to be outdated (current is at least 3.5.0).
+ mod_ssl/2.2.14 appears to be outdated (current is at least 2.9.6) (may depend on server version).
+ Perl/v5.10.1 appears to be outdated (current is at least v5.32.1).
+ mod_perl/2.0.4 appears to be outdated (current is at least 2.0.11).
+ proxy_html/3.0.1 appears to be outdated (current is at least 3.1.2).
+ PHP/5.3.2-1ubuntu4.30 appears to be outdated (current is at least 8.1.5), PHP 7.4.28 for the 7.4 branch.
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE .
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-communi
ty/attacks/Cross_Site_Tracing
+ mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 mod_perl/2.0.4 Perl/v5.10.1 - mod_ssl 2.8.7 and lower ar
e vulnerable to a remote buffer overflow which may allow a remote shell.
+ PHP/5.3 - PHP 3/4/5 and 7.0 are End of Life products without support.
+ /mutillidae/index.php?page=../../../../../../../../../../../etc/passwd: The PHP-Nuke Rocket add-in is vulnerable t
o file traversal, allowing an attacker to view any file on the host. (probably Rocket, but could be any index.php
).
+ /mutillidae/phpinfo.php: Output from the phpinfo() function was found.
+ /mutillidae/?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certai
n HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /mutillidae/?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certai
n HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /mutillidae/?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certai
n HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /mutillidae/?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certai
n HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /mutillidae/data/: Directory indexing found.
+ /mutillidae/data/: This might be interesting.
+ /mutillidae/includes/: Directory indexing found.
+ /mutillidae/includes/: This might be interesting.
+ /mutillidae/passwords/: Directory indexing found.
+ /mutillidae/passwords/: This might be interesting.
+ /mutillidae/phpmyadmin/ChangeLog: phpMyAdmin is for managing MySQL databases, and should be protected or limite
d to authorized hosts.
+ /mutillidae/test/: Directory indexing found.
+ /mutillidae/test/: This might be interesting.
+ /mutillidae/phpinfo.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot o
f system information. See: CWE-552
+ /mutillidae/index.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of
system information. See: CWE-552
+ /mutillidae/images/: Directory indexing found.
+ /mutillidae/styles/: Directory indexing found.
+ /mutillidae/?_CONFIG[files][functions_page]=http://blog.cirt.net/rfiinc.txt: Remote File Inclusion (RFI) from R
Snake's RFI list. See: https://gist.github.com/mubix/5d269c686584875015a2
+ /mutillidae/?npage=-1&content_dir=http://blog.cirt.net/rfiinc.txt%00&cmd=ls: Remote File Inclusion (RFI) from R
```

Hacemos un nano para facilitar la búsqueda y encontramos CVE-2003-1418



- **CVE-2003-1418**
- **Fecha de publicación:** 12/31/2003
- **Código CWE:** CWE-200
- **CVSS 3.x:** N/A
- **CVSS 2.x:**
  - Base Score: 4.3 MEDIUM
  - Vector: (AV:N/AC:M/Au:N/C:P/I:N/A:N)
  - Vector de Acceso: Network
    - Complejidad del acceso: Media
    - Autenticación: No
    - Confidencialidad: Parcial
    - Integridad: No
    - Disponibilidad: No
    - Información adicional: Permite revelación de información no autorizada
- **URL del CVE:** https://nvd.nist.gov/vuln/detail/CVE-2003-1418
- **Referencias:**
  - http://www.openbsd.org/errata32.html
  - http://www.oracle.com/technetwork/security-advisory/cpuoct2017-3236626.html
  - http://www.securityfocus.com/bid/6939
  - http://www.securityfocus.com/bid/6943 Patch
  - https://exchange.xforce.ibmcloud.com/vulnerabilities/11438
- **Descripción de la vulnerabilidad:** Apache HTTP Server v1.3.22 a 1.3.27 en OpenBSD permite a atacantes remotos obtener información confidencial a través de (1) el encabezado ETag, que revela el número de inodo, o (2) el límite MIME multiparte, que revela los ID de procesos secundarios (PID).

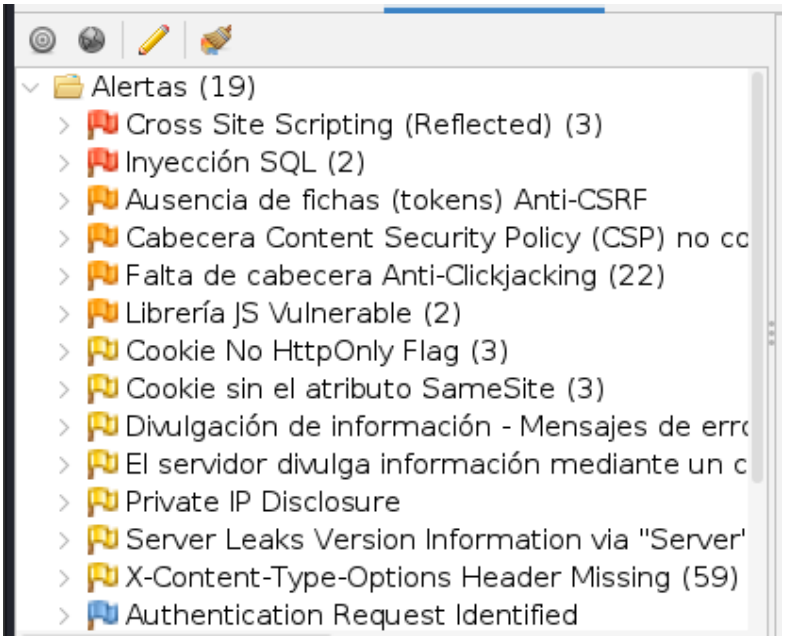
- Evaluar los resultados del escaneo con Nessus y explicar en detalle la vulnerabilidad más crítica según su CVSS

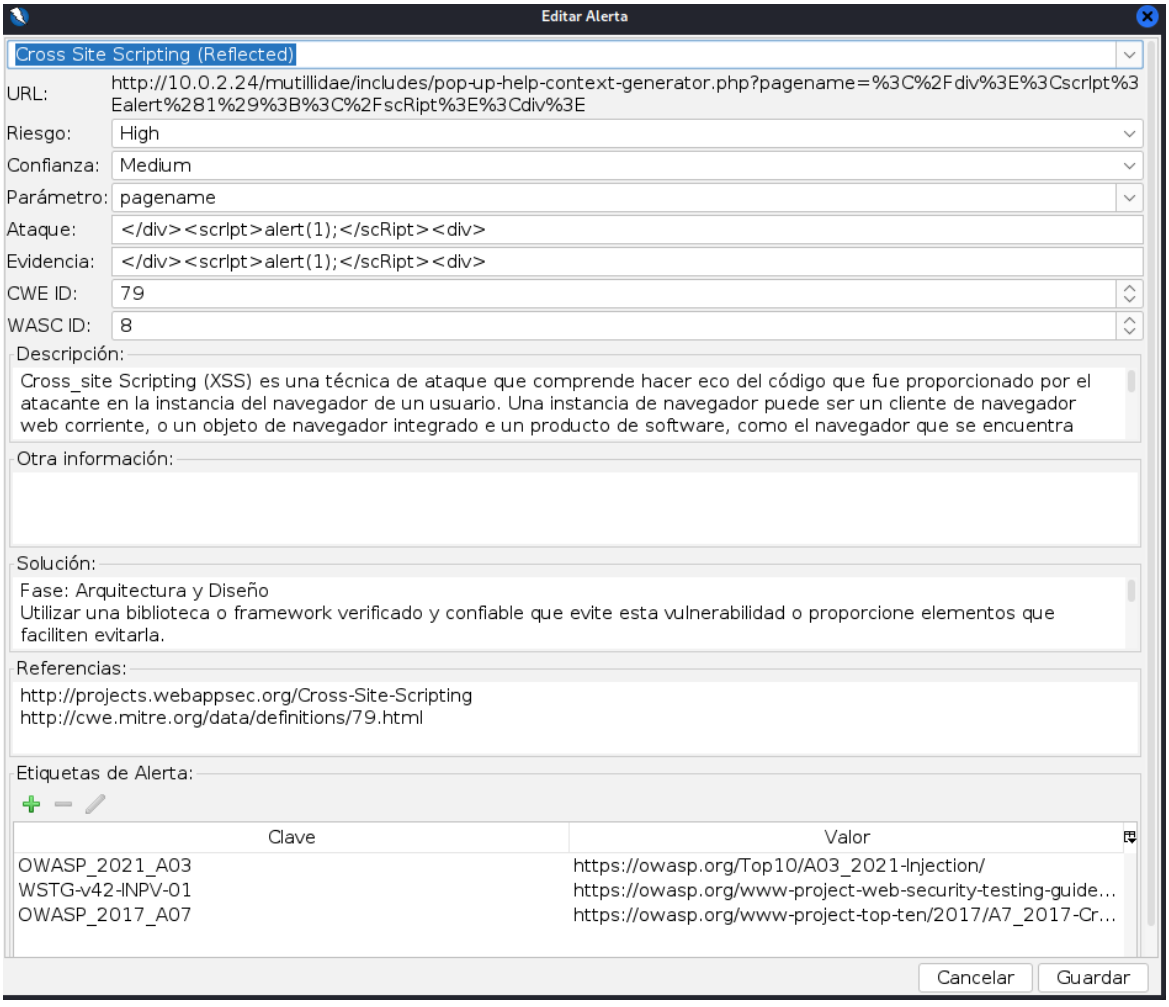


- Vulnerabilidad en algoritmo MD5 Message-Digest (CVE-2004-2761)
- **Fecha de publicación:** 2009/05/01
- **Código CWE:** CWE-300
- **CVSS 2.x:**
  - Base Score: 5.0 MEDIUM
  - Vector: AV:N/AC:L/Au:N/C:N/I:P/A:N
    - Vector de Acceso: Network
    - Complejidad del acceso: Baja
    - Autenticación: No
    - Confidencialidad: Parcial
    - Integridad: No
    - Disponibilidad: No
- **Referencias:**
  - http://www.phreedom.org/research/rogue-ca/ (Origen:MISC)
  - http://www.microsoft.com/technet/security/advisory/961509.mspx (Origen:MISC)
  - http://blog.mozilla.com/security/2008/12/30/md5-weaknesses-could-lead-to-certificate-forgery/ (Origen:MISC)
  - http://www.doxpara.com/research/md5/md5_someday.pdf (Origen:MISC)
  - http://www.securityfocus.com/bid/33065 (Origen:BID)
  - http://www.win.tue.nl/hashclash/SoftIntCodeSign/ (Origen:MISC)

    - http://www.win.tue.nl/hashclash/rogue-ca/ (Origen:MISC)
    - http://blogs.technet.com/swi/archive/2008/12/30/information-regarding-md5-collisions-problem.aspx (Origen:MISC)
- **Descripción de la vulnerabilidad**: El algoritmo MD5 Message-Digest no resistente a colisión, el cual hace más fácil para atacantes dependientes de contexto, llevar a cabo ataques de suplantación, como lo demuestran los ataques de utilización de MD5 en la firma del algoritmo de un certificado X.509.

- Evaluar los resultados del escaneo con OWASP Zap y explicar en detalle la vulnerabilidad que más alertas haya suscitado

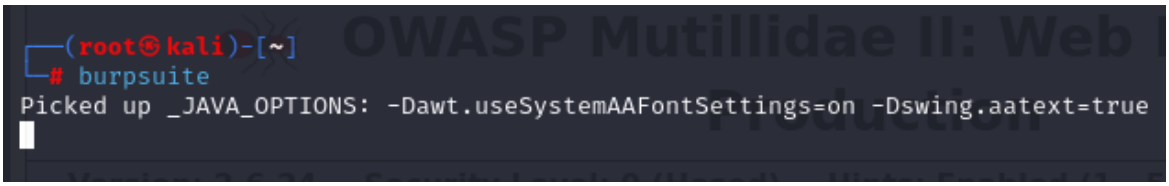


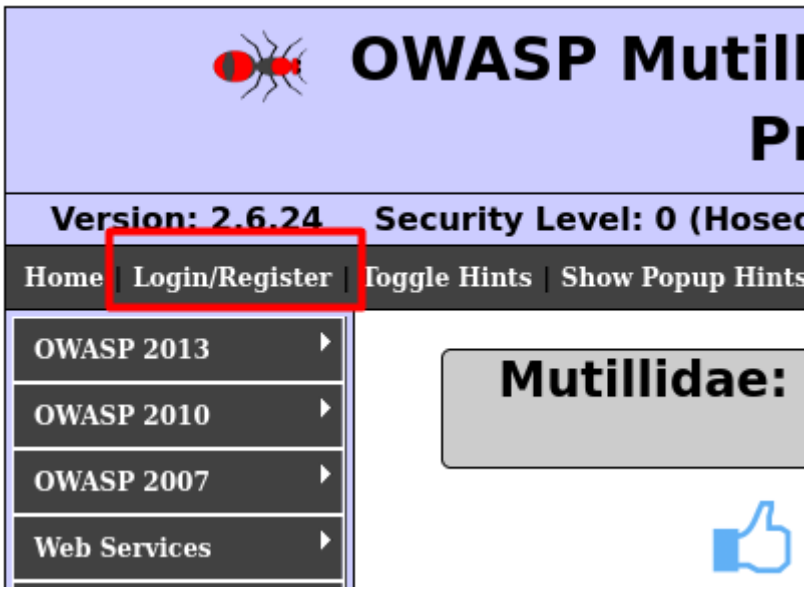


- La vulnerabilidad que mas alertas ha creado es "Cross Site Scripting (Reflected)"
- Código CWE:
    - CWE-79
- Referencias:
    - http://projects.webappsec.org/Cross-Site-Scripting
    - http://cwe.mitre.org/data/definitions/79.html
- Descripción de la vulnerabilidad:
    - Cross_site Scripting (XSS) es una técnica de ataque que comprende hacer eco del código que fue proporcionado por el atacante en la instancia del navegador de un usuario. Una instancia de navegador puede ser un cliente de navegador web corriente, o un objeto de navegador integrado e un producto de software, como el navegador que se encuentra dentro de WinAmp, un lector de RSS o un cliente de correos electrónicos. El código por sí mismo se encuentra escrito en HTML/JavaScript, pero también puede extenderse a VBScript, ActiveX, Jave, Flash o cualquier otra tecnología que sea compatible con el navegador. Cuando un atacante consigue el navegador de un usuario para poder ejecutar su código, el código se ejecutará dentro del contexto de seguridad (o zona) del sitio web que lo alberga. Con este nivel de privilegio, el código tiene la extensión de leer, modificar y transmitir cualquier dato que sea sensible al que pueda ingresar al navegador. Los ataques de scripting entre los sitios comprometen la relación de la confianza entre el usuario y el sitio web. Las aplicaciones que usan instancias de objetos del navegador que suben contenido desde el sistema de archivos puede activar el código bajo la zona de lam máquina, lo cual permite que el sistema se vea comprometido. Hay tres tipos de ataques diferentes de scripting entre los sitios: no persistentes, persistentes y basados en DOM.
- Solución:
    - Fase: Arquitectura y Diseño
    - Utilizar una biblioteca o framework verificado y confiable que evite esta vulnerabilidad o proporcione elementos que faciliten evitarla. Los ejemplos de las bibliotecas y marcos que facilitan el origen de resultados que son codificados de forma correcta incluyen la biblioteca Anti-XSS de Microsoft, el módulo de codificación OWASP ESAPI y Apache Wicket.
    - Fases: Implementación; Arquitectura y Diseño
    - Comprenda el contexto en el que se va a utilizar sus datos y la condificación que se va a esperar. Esto es fundamentalmente importante cuando se transmiten los datos entre diferentes componentes o cuando se generan las salidas que pueden comprender múltiples codificaciones al mismo tiempo, como paginas web o mensajes de correos de varias zonas. Estudie todos los protocolos de comunicacón y representaciones de los datos que son esperadas para poder determinar las estrategias de codificación que son necesarias. Por cualquier dato que se enviará a otra página web, en especial cualquier dato recibido de las entradas externas, utiice la codificación que sea conveniente en todos los caracteres que no sean alfanuméricos. Consulte la hoja de referencia de prevención de CSS para poder obtener más información detallada de los diferentes tipos de condificación y escape que se requieren.
    - Fase: Arquitectura y Diseño
    - Cualquier comprobación de seguridad que se vaya a realizar en el lado del cliente, asegúrese de que estas comprobaciones se encuentren duplicadas en el lado del servidor, para evitar el CWE-602. Los atacantes pueden eludir las comprobaciones del lado del cliente modificando los valores después de que se hayan realizado las comprobaciones, o cambiando al cliente para poder eliminar de forma completa las comprobaciones del lado del cliente. Después, estos valores que fueron modificados serán enviados al servidor. Si se encuentra disponible, utilice los mecanismos estructurados que apliquen de forma automática la separación entre los datos y códigos. Estos mecanismos pueden otorgar la cotización, codificación y validación relevantes de manera automática, en lugar de confiar en que el desarrollador proporcione esta capacidad en cada uno de los puntos donde se origina la salida.
    - Fase: Implementación
    - Para cada una de las páginas web que se origina, utilice y especifique una codificación de caracteres como ISO-8859 o UTF-8. Cuando no se puede especificar una condificación, el navegador web podría selaccionar una codificación distinta adivinando que codificiación está siendo utilizada en verdad por la página web. Esto puede permitir que el navegador web trate varias secuencias como especiales, abriendo al cliente a leves ataques XSS. Consulte CWE-116 para conseguir más mitigaciones con respecto a la codificación/escape. Para ayudar a mitigar los ataques XSS contra las cookies de la sesión del usuario, es necesario establecer que la cookie de la sesión sea HttpOnly. En navegadores que son compatibles con la característica HttpOnly (como las versiones más actualizadas de internet explorer y firefox), esta característica puede prevenir que la cookie de sesión del usuario sea accesible para las secuencias de comandos del lado del cliente malignas que utilizan document.cookie. Esta no es una solución muy completa, ya que HttpOnly no es compatible con todos los navegadores que hay. Más importante aún, XMLHTTPRequest y otras tecnologías poderosas de navegador otorgan acceso de lectura a los encabezados HTTP, incluido el encabezado Set-Cookie en el cual se establece el indicador HttpOnly. Asuma que toda la entrada es maliciosa. Utilizar una estrategia de validación de entradas de tipo "aceptar lo bueno conocido", es decir, utilizar una lista de entradas aceptables que se ajusten estrictamente a las especificaciones. Rechace cualquier entrada que no se adapte de forma estricta a las especificaciones, o cambielas por algo que sí lo haga. No confíe exclusivamente en la búsqueda de entradas maliciosas o malformadas (es decir, no confíe en una lista de denegación). Sin embargo, las listas de denegación pueden ser útiles para detectar posibles ataques o para determinar qué entradas están tan malformadas que deben ser rechazadas directamente. Al realizar la validación de entrada, usted debe considerar todas las propiedades potencialmente destacadas, incluida la longitud, el tipo de entrada, el rango completo de valores aceptables, las entradas faltantes o adicionales, la sintaxis, el sentido entre los campos que se encuentran relacionados y la conformidad con todas las reglas comerciales. Como ejemplo de lógica de regla de negocio, "barco" puede ser sintácticamente válido porque sólo contiene caracteres alfanuméricos, pero no es válido si se esperan colores como "rojo" o "azul". Asegúrese de realizar la validación de entradas en interfaces bien definidas dentro de la aplicación. Esto ayudará a cuidar la aplicación, incluso si un elemento se utiliza de nuevo o traslada a otro sitio

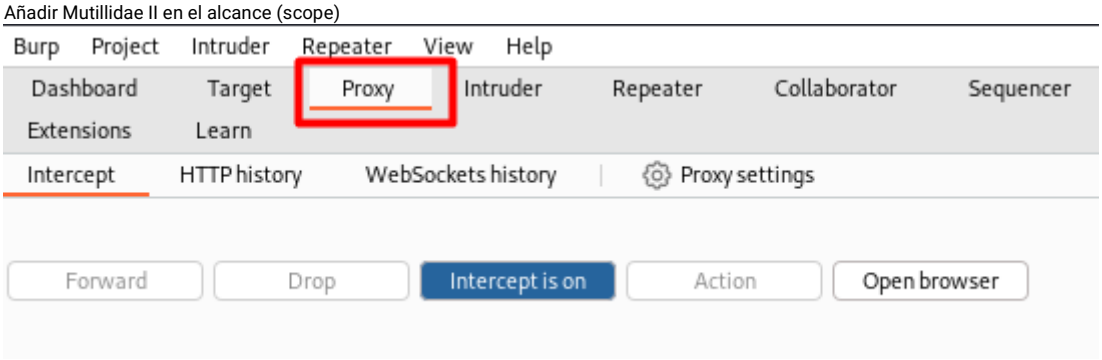
Ejercicio 4 - Burp Suite

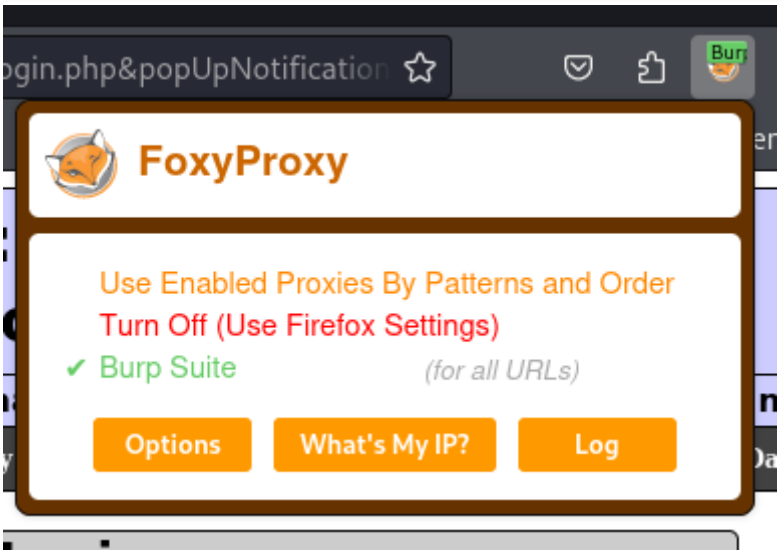
Utilizando Burp Suite y Firefox, cargar la web de Mutillidae II sección "Login/Register"





Añadir Mutillidae II en el alcance (scope)



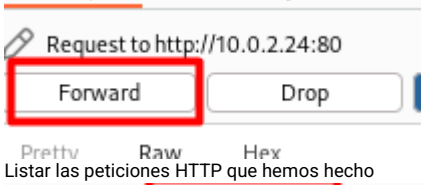


Interceptar la petición de login con las credenciales admin - admin

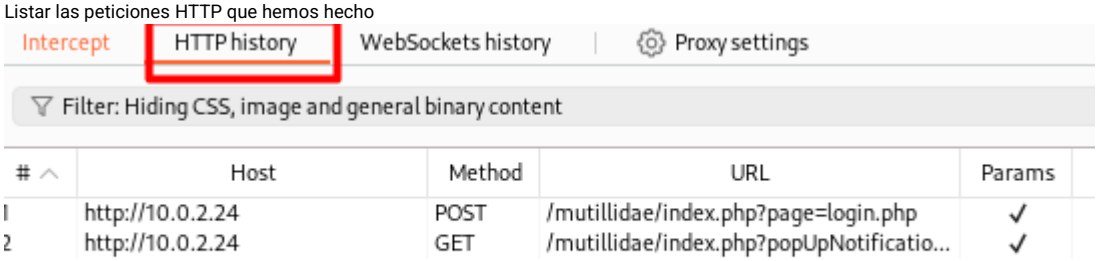
```
POST /mutillidae/index.php?page=login.php HTTP/1.1
Host: 10.0.2.24
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 59
Origin: http://10.0.2.24
Connection: close
Referer: http://10.0.2.24/mutillidae/index.php?page=login.php&popUpNotificationCode=LOU1
Cookie: showhints=1; PHPSESSID=uv4j5v7tit2ldo5s55pvo80926; acopendivids=
swingset,jotto,phpbb2,redmine; acgroupswithpersist=nada
Upgrade-Insecure-Requests: 1

username=admin&password=admin&login-php-submit-button=Login
```

Dejar de capturar peticiones para poder continuar y acceder



Listar las peticiones HTTP que hemos hecho



Filtrar y listar únicamente las peticiones de Mutillidae II → boton de filtro