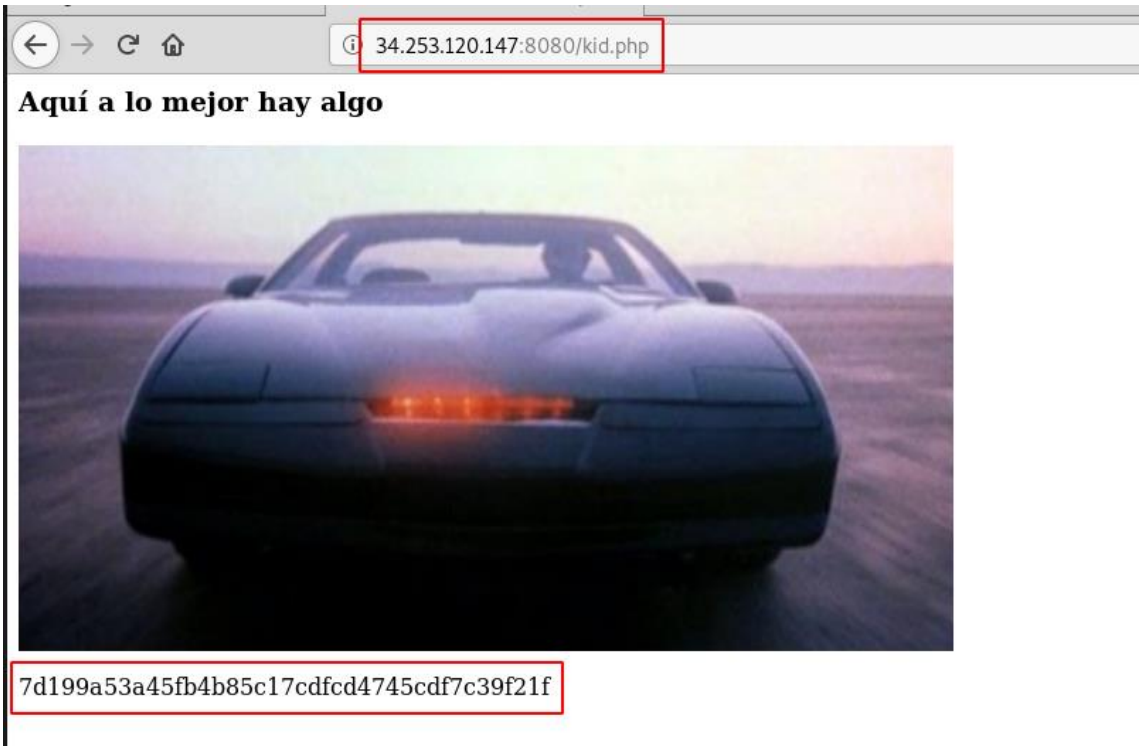
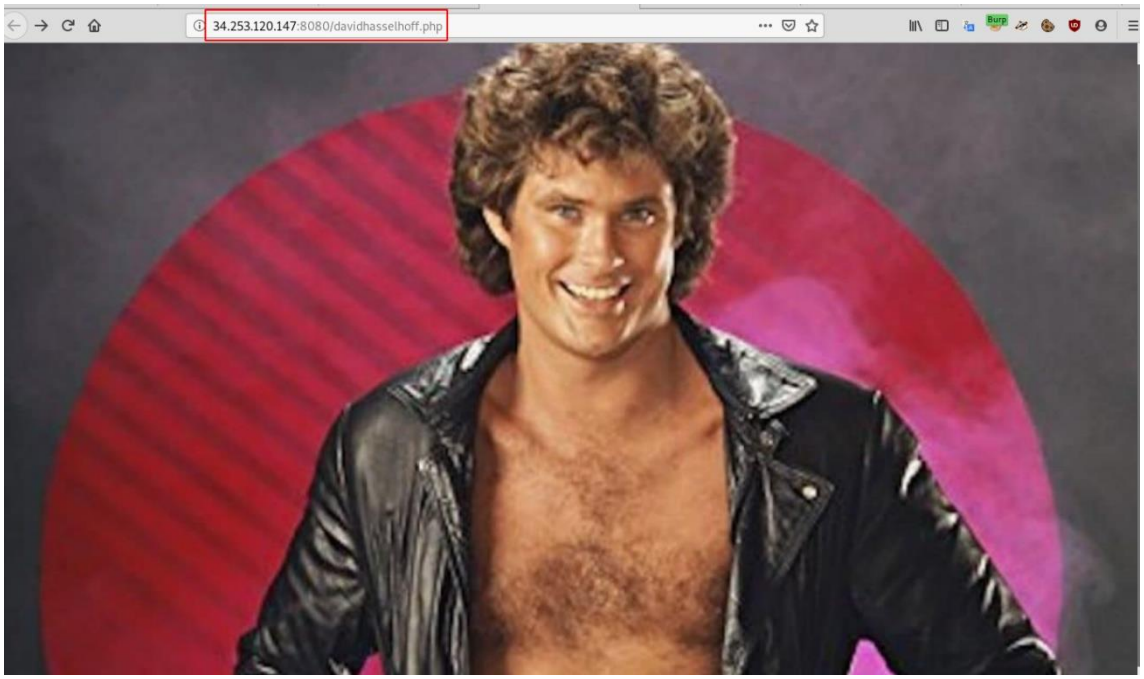
Llegó la hora de los fans de la serie “El coche fantástico”, tendremos la oportunidad de currar como mecánico de KITT.

Eso sí, preparen paracetamol que el reto cuenta con un porcentaje alto de troleo:

Vamos analizando cada uno de los ficheros PHP y encontramos un interesante hash en sha1 en el fichero “kid.php”:



También nos llama la atención esta imagen en el archivo “davidhasselhoff.php”.



En el código fuente del archivo nos encontramos con la siguiente pista:

view-source:http://34.253.120.147:8080/davidhasselhoff.php

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

    <meta charset="utf-8">
    <title></title>
<style media="screen">
  body{
      background-image: url('https://www.todorock.com/wp-content/uploads/2019/04/david-hasselhoff-metal.jpg');
      background-size: cover;
      text-align: center;
      height: 100vh;
      padding:0;
      margin:0;
  }
p {
      background-color: white;
}

</style>
  </head>
  <body>
<!-- me encanta la raza de perro de la famosa foto de David Hasselhoff... -->
</center>
</div>
<div class="center-post">
<pre>

    <br>
<br>
```

Si buscamos en Google podemos encontrar varias fotos de David Hasselhoff con perros de diferentes razas, pero está clarísimo que la más conocida es la foto del famoso ataque "David Hasselhoff":

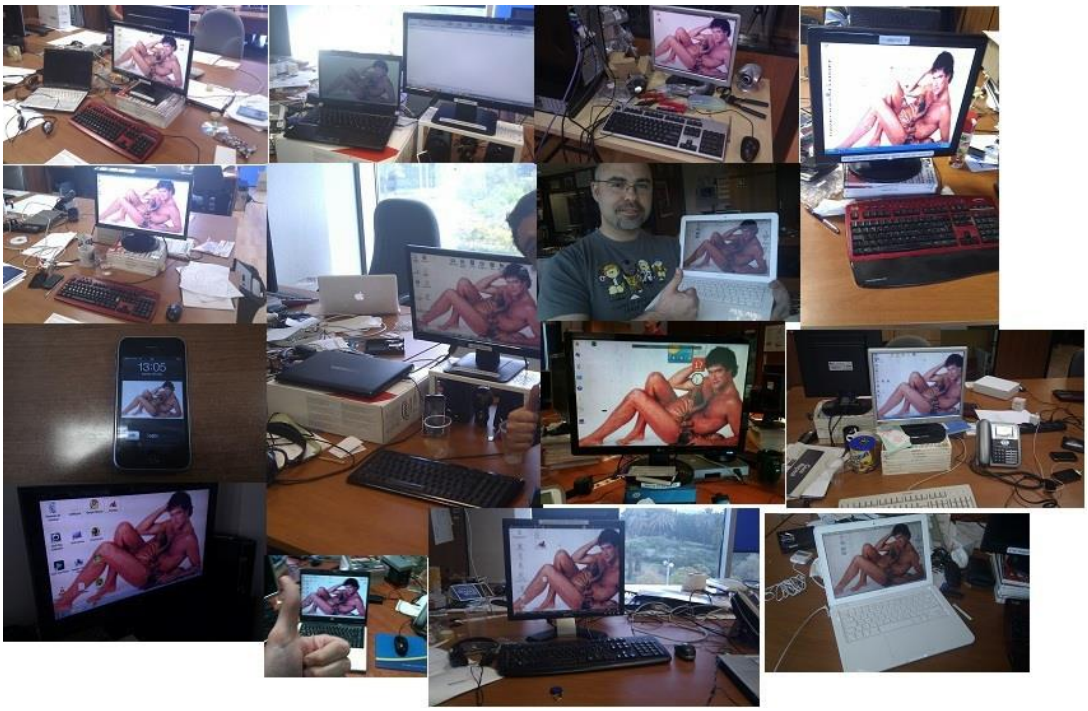
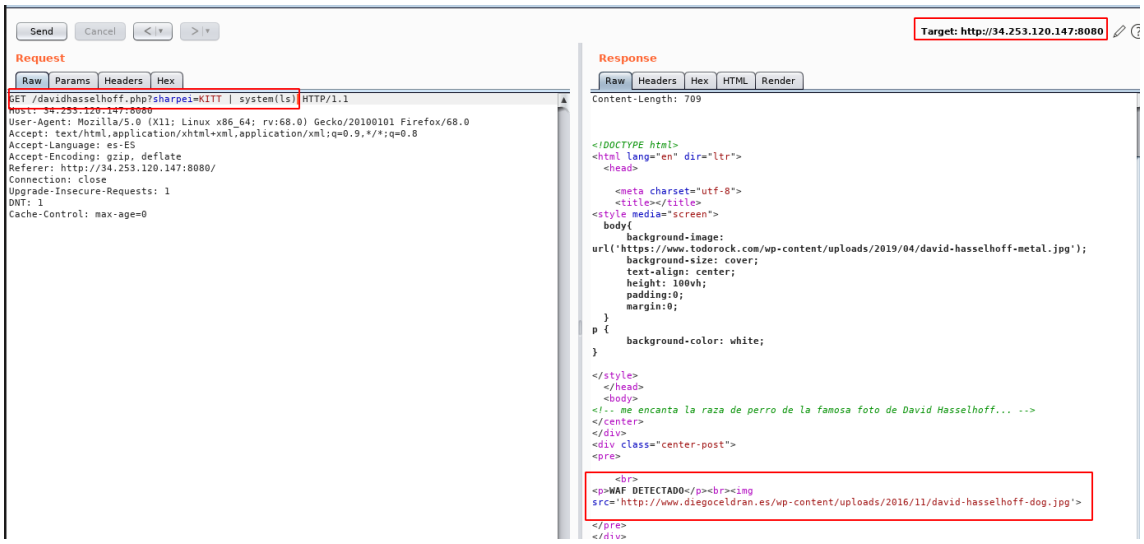


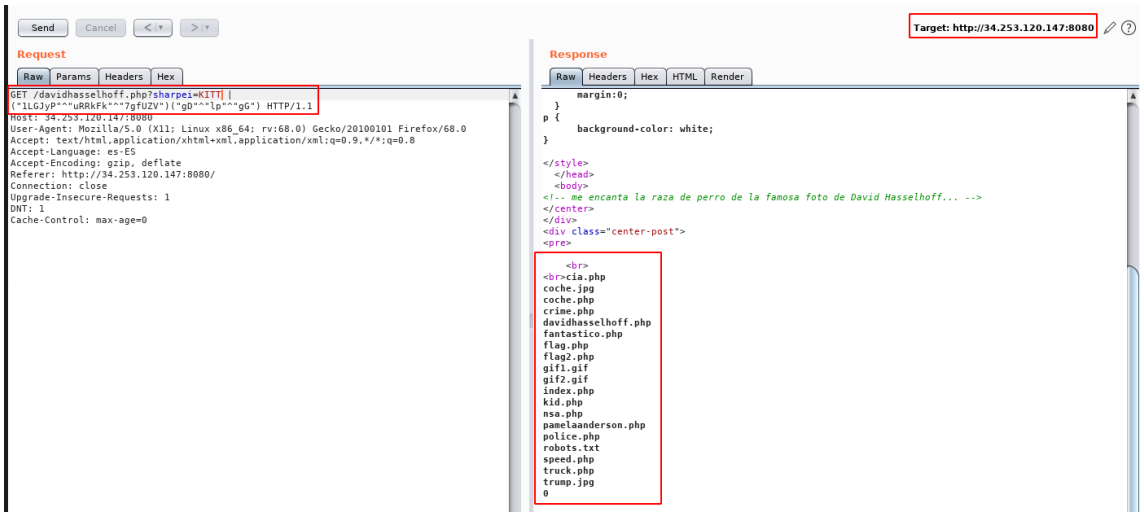
Ilustración 1 Evidencia del ataque David Hasselhoff – Imagen de Hispasec

Los cachorros que aparecen en la foto son de la raza "Shar Pei", tras la pista "*Leer Una al día*" lanzada por Telegram de Una al mes, y mucha prueba/error, utilizamos "sharpei" como variable donde se aprecia que el servidor cuenta con un WAF.

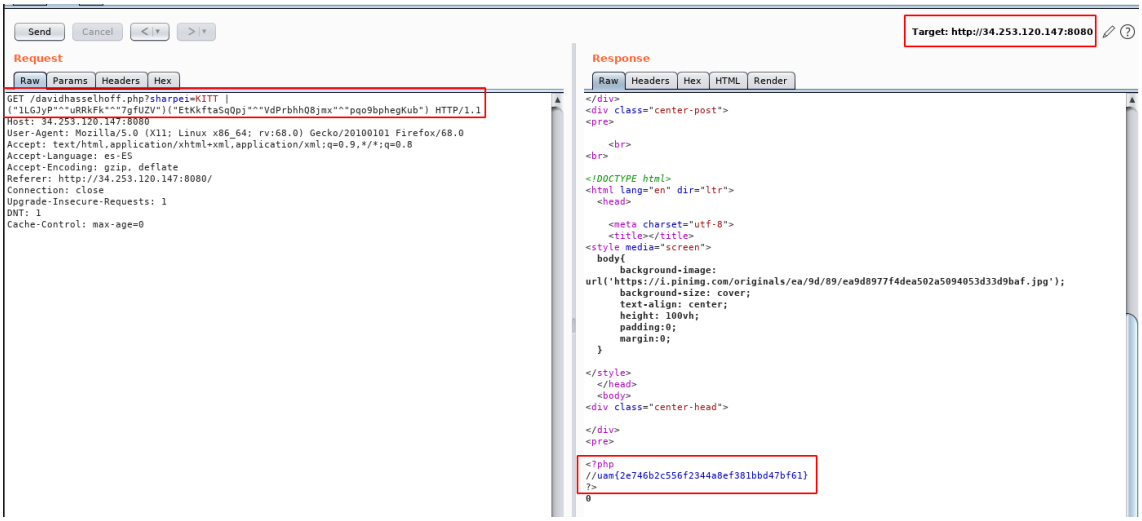


Si leemos las noticias de “Una al día”, podemos ver el [artículo de la tool de Daniel Púa @devploit](), donde nos muestra una alternativa muy cómoda para evadir filtros de WAF.

Hacemos uso de esta herramienta con una PoC para comprobar si el sitio es vulnerable, en la siguiente imagen vemos como es posible ejecutar comando remoto y poder listar todos los ficheros:

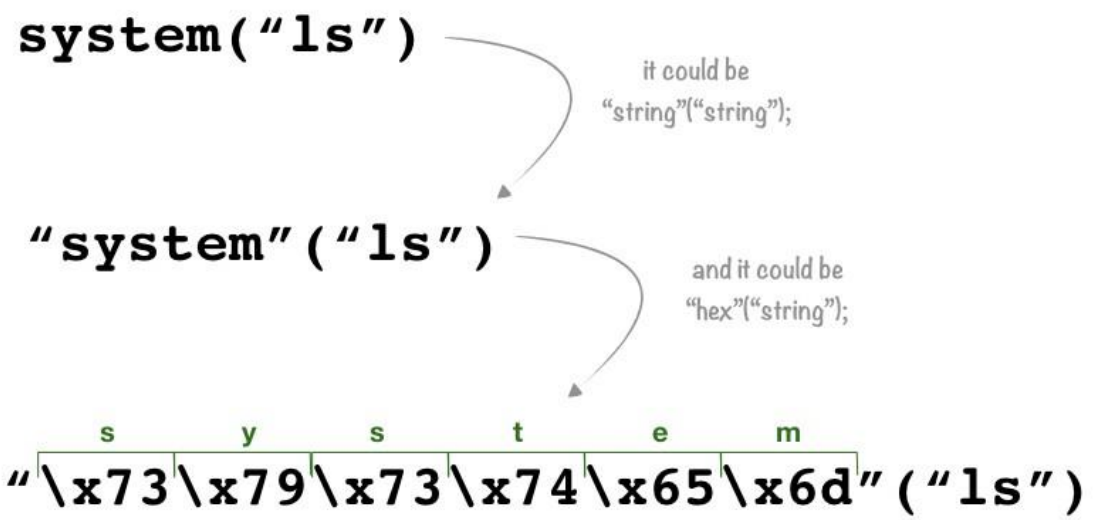


Ahora solo nos queda hacer un “cat flag.php” para poder leer el interior del fichero.

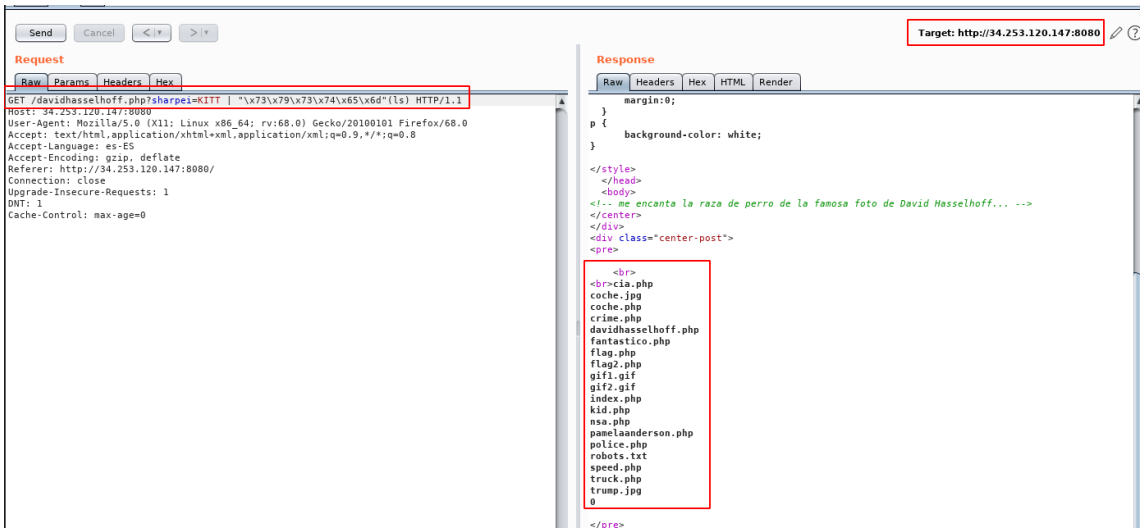


¡Y por fin nuestra flag!

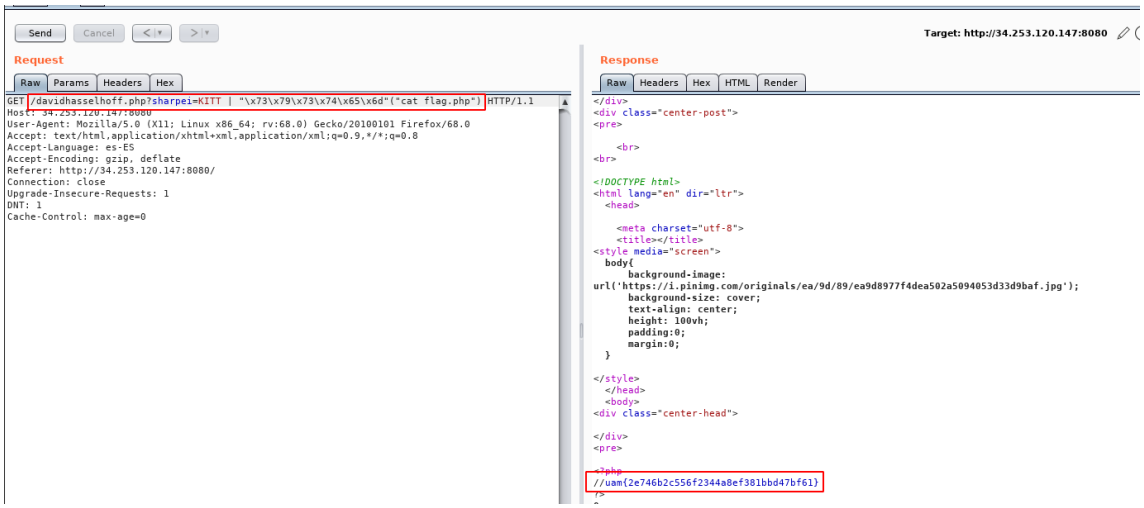
Otro método alternativo, es hacer uso de hexadecimal:



Cambiamos "system" por "\x73\x79\x73\x74\x65\x6d" y vemos que también es posible evadir el WAF:



Ahora realizamos la misma operación, pero con “cat flag.php” y obtenemos nuevamente la flag:



Backstage….