FILE UPLOAD PORTSWIGGER

İçindekiler

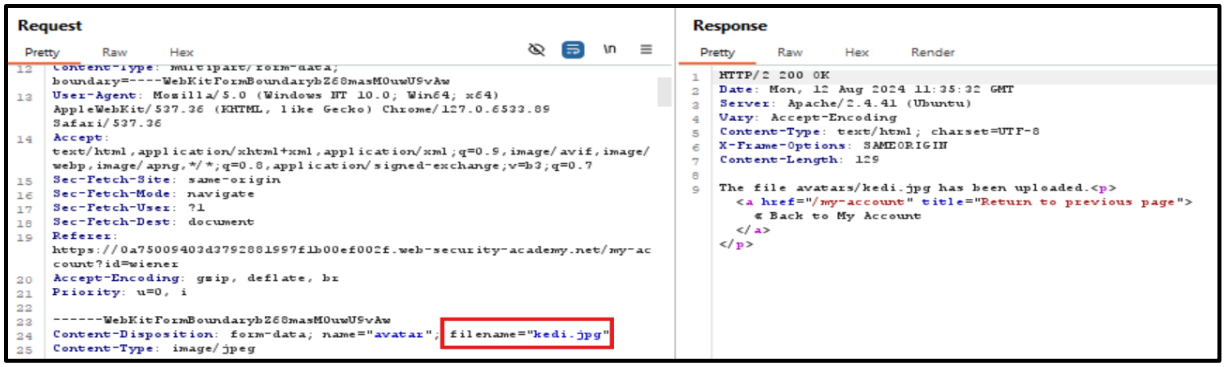
Lab: Remote code execution via web shell upload

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.



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.

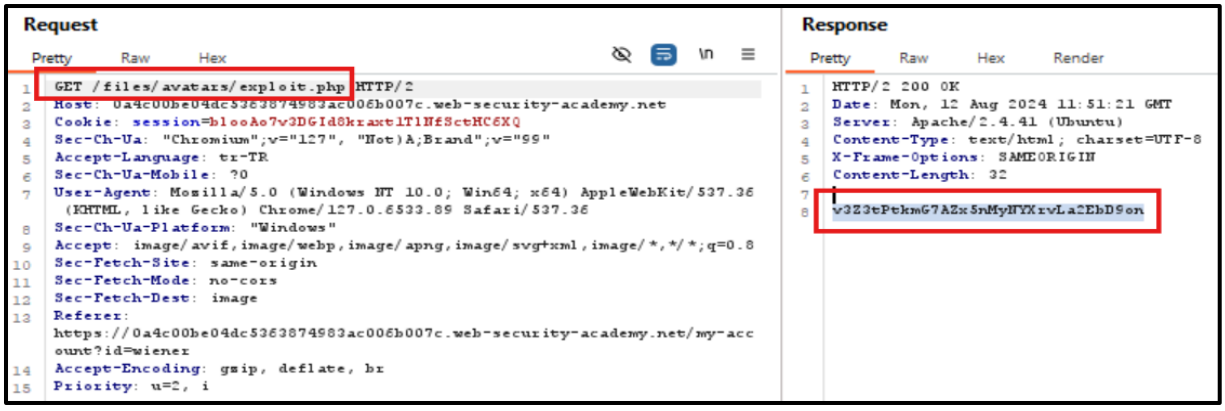
Lab: Web shell upload via Content-Type restriction bypass

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.



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 dizin geçişi ile dosyayı */../exploit.php* dizinine yüklemeye izin vermiştir.

Zafiyet Çözüm Önerisi:

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

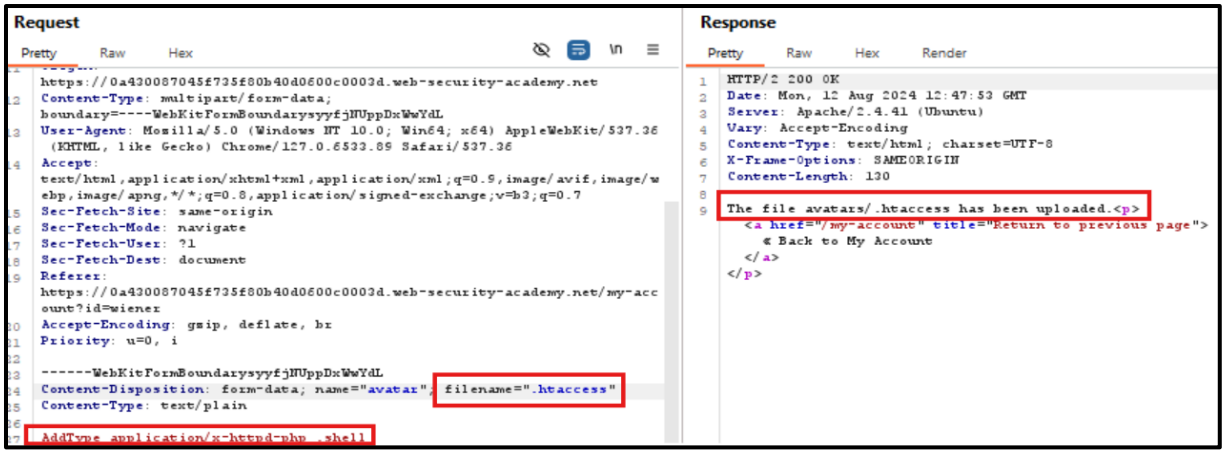
Lab: Web shell upload via extension blacklist bypass

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.



Zafiyet Çözüm Önerisi:

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

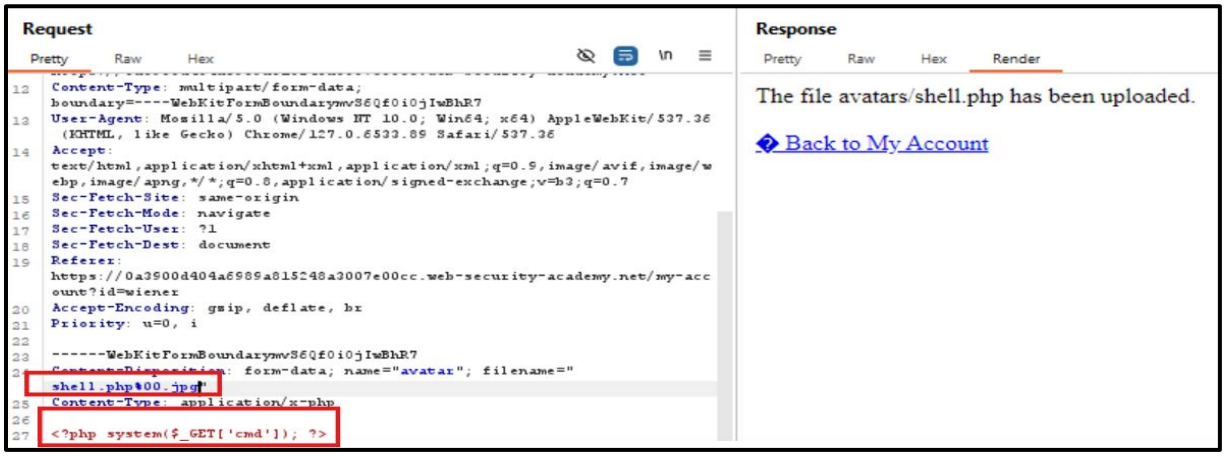
Lab: Web shell upload via obfuscated file extension

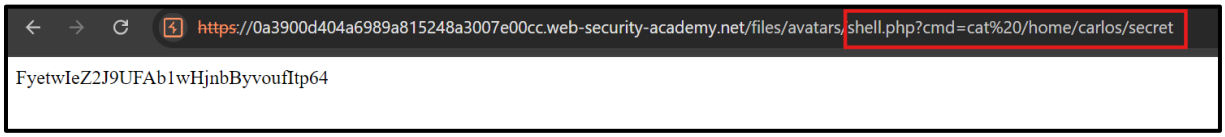
İçerik:

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.