

SQL Injection/Basic SQL Injection

Herkese selamlar bu yazımda Hackviser üzerinde bulunan Web security lablarından biri olan **SQL Injection/Basic SQL Injection labının çözümünü anlatacağım.**

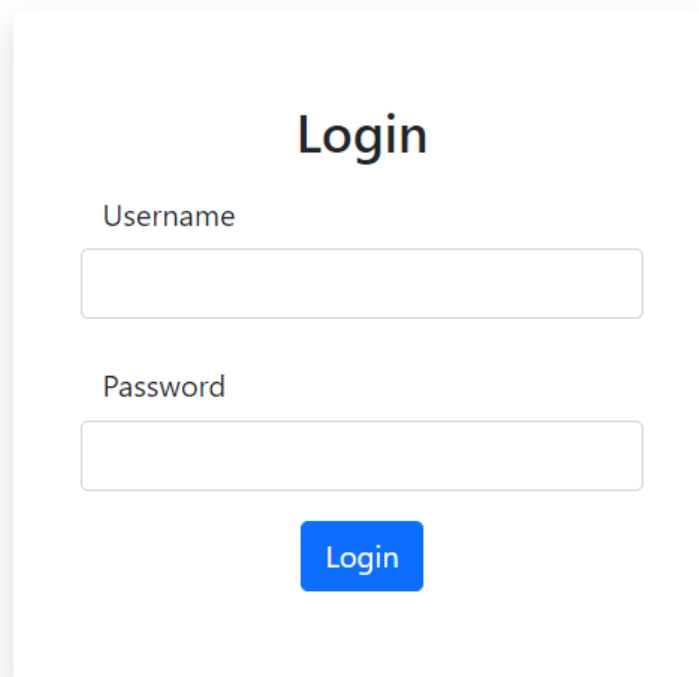
Öncelikle SQL Injection (SQLi) açığının ne olduğunu kısaca bir hatırlayalım:

SQL Injection açığı, web uygulamalarının kullanıcı girdilerini veritabanı sorgularına dahil ederken yeterli güvenlik önlemlerini almamasından kaynaklanır. Saldırgan bu açıktan faydalanarak SQL komutlarını manipüle edebilir ve veritabanına istenmeyen komutlar çalıştırabilir. Açığın türüne göre, saldırgan veritabanından hassas bilgileri çekebilir, veri ekleyebilir, silebilir veya güncelleyebilir.

SQL Injection açığını nerelerde bulabileceğimizi düşünüyorsak; uygulamaya veri girişi yapacağımız herhangi bir yer (örneğin giriş ekranları, arama kutuları, kayıt formları, URL parametreleri) SQL Injection için potansiyel bir hedef olabilir.

Sırada ise makinenin çözümü var.

Siteye ilk başta gittiğimizde bizi böyle bir login page karşılıyor. Bizden istenen ise SQL Injection saldırısı gerçekleştirerek oturum açma adımını atlamamız.



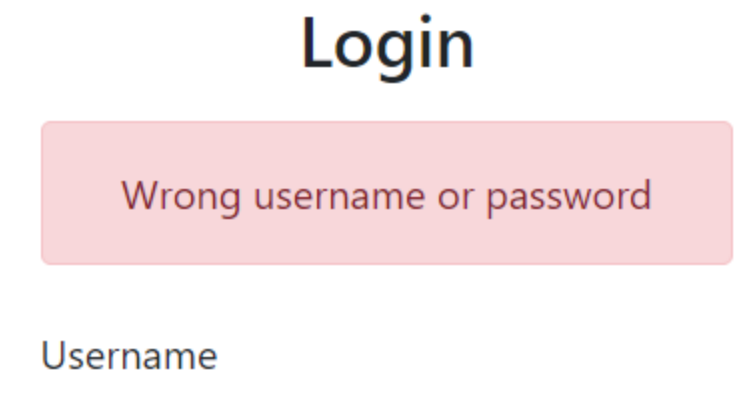
Login

Username

Password

Login

İlk başta " ' " koyup deniyorum ancak "Wrong username or password" uyarısını alıyorum

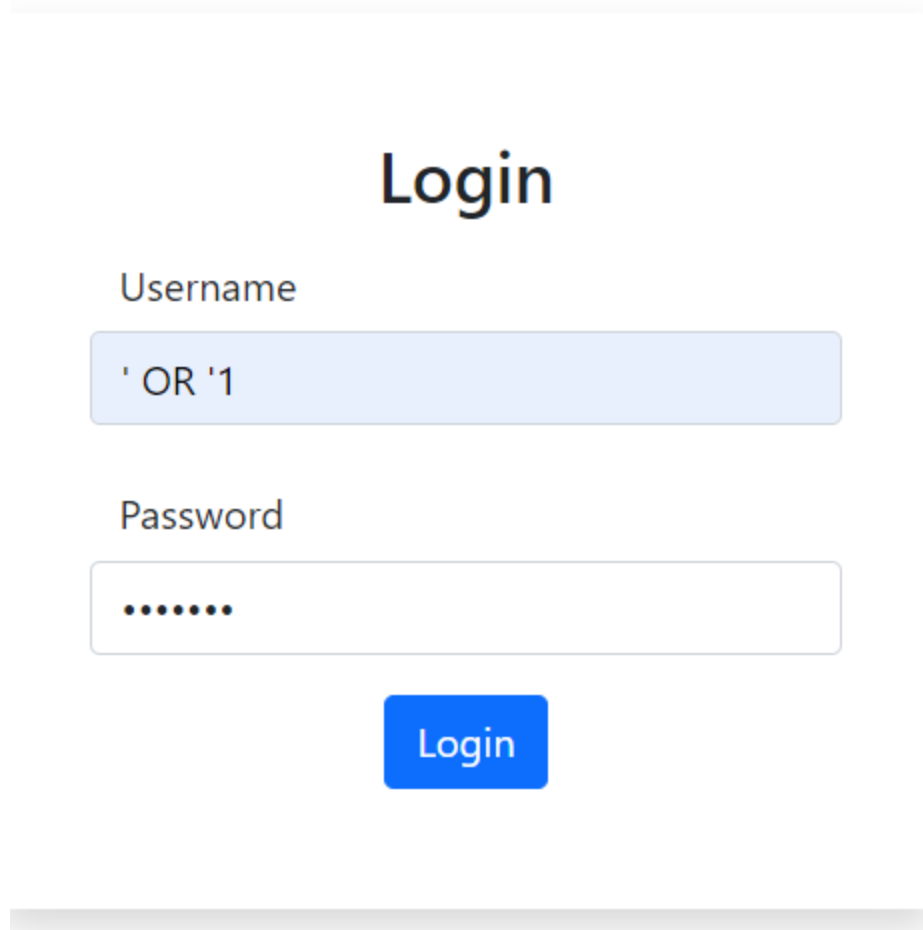


Login

Wrong username or password


Username

Ardından " ' OR '1 " payloadını deniyorum.



The image shows a login form with the title "Login" in a large, bold, black font. Below the title, there are two input fields. The first field is labeled "Username" and contains the text "' OR '1". The second field is labeled "Password" and contains seven dots, indicating a masked password. Below the password field is a blue button with the text "Login" in white. The entire form is enclosed in a light gray border with rounded corners.

Ve aşağıdaki gibi direkt olarak profil sekmesine giriş yapabiliyorum.



Sky Raincin
sraincin0@moonfruit.hv

Logout

Profile Settings

Name

Sky

Surname

Raincin

Mobile Number

172-496-3430

Address

33887 Raven Terrace

Postcode

57990

Email

sraincin0@moonfruit.hv

Country

Malaysia

State/Region

Coventry

Save Profile

Peki, bu açığı kullanarak neler yapabileceğimize bakalım. SQL Injection açığını kullanarak veritabanındaki verilere erişebilir, hassas bilgileri çalabilir, veritabanına yeni kayıtlar ekleyebilir, mevcut kayıtları silebilir veya güncelleyebilirim. Ayrıca, sistemin kontrolünü ele geçirebilir, admin yetkilerine sahip olabilir ve uygulamanın işleyişini bozabilirim... Liste uzar gider.

Şimdi de bu açığı nasıl kapatabileceğimize bakalım:

- Kullanıcıdan alınan veriler mutlaka filtrelenmeli ve doğrulanmalıdır.** Örneğin, bir formdan ya da URL'den gelen verilerin belirli formatta olup olmadığı kontrol edilerek güvenli hale getirilmelidir.
- Hazır parametrelili sorgular (prepared statements) ve PDO kullanılması** SQL Injection riskini büyük ölçüde azaltacaktır. Bu sayede kullanıcıdan gelen veriler doğrudan SQL sorgusuna eklenmez, ayrı bir parametre olarak işlenir.
- Veritabanı kullanıcılarının yetkileri sınırlandırılmalıdır.** Uygulamanın, veritabanında sadece ihtiyaç duyduğu yetkilere sahip olması sağlanarak hasar potansiyeli azaltılabilir.

Bunlar çözümlerden birkaçı. Bu tip zafiyetlerle karşılaşmamak için planlı ve güvenli kod geliştirme yöntemleri kullanarak, zafiyetlerin büyük oranda önüne geçebiliriz.

Umarım bu yazıyı sıkılmadan okumuşsunuzdur. Keyifli çalışmalar.