Cross-Site Scripting (XSS)

Cross-Site Scripting (XSS), web uygulamalarında en yaygın görülen güvenlik açıklarından biridir. Bu saldırı türü, kötü niyetli kişilerin web uygulamalarına zararlı komut dosyaları (script) enjekte ederek kullanıcıların tarayıcılarında çalıştırmasına olanak tanır. XSS saldırıları, genellikle kullanıcıyı hedef alır ve veri hırsızlığı, oturum çalma, kimlik sahtekarlığı gibi sonuçlara yol açabilir.

1. XSS Saldırısının Tanımı

XSS (Cross-Site Scripting), bir web uygulamasının kullanıcıdan aldığı verileri yeterince filtrelemeden veya doğrulamadan doğrudan sayfaya yerleştirmesi sonucu ortaya çıkan bir güvenlik açığıdır. Bu açık sayesinde saldırgan, zararlı JavaScript kodlarını uygulamaya enjekte edebilir ve bu kodlar, diğer kullanıcıların tarayıcılarında çalıştırılabilir.

2. XSS Saldırısının Ortaya Çıkışı

XSS zafiyeti genellikle şu durumlarda ortaya çıkar:

Kullanıcıdan alınan verilerin HTML çıktısına doğrudan yerleştirilmesi

Giriş verilerinin yeterince filtrelenmemesi veya kaçırılmaması

Dinamik içerik üretiminde güvenlik kontrollerinin eksik olması

JavaScript, HTML veya URL parametrelerinin kontrolsüz biçimde kullanılması

Örneğin, bir yorum formuna `<script>alert('XSS')</script>` gibi bir kod girildiğinde ve bu içerik doğrudan sayfada gösterildiğinde, XSS açığı oluşur.

3. XSS Türleri

XSS saldırıları üç ana kategoriye ayrılır:

a. Saklı (Stored) XSS

Zararlı kod, veritabanına veya sunucuya kalıcı olarak kaydedilir.

Diğer kullanıcılar bu içeriği görüntülediğinde kod çalıştırılır.

Forumlar, yorum sistemleri gibi kullanıcı içeriklerinin saklandığı alanlarda yaygındır.

b. Yansıtılan (Reflected) XSS

Zararlı kod, URL parametresi veya form verisi aracılığıyla gönderilir.

Sunucu bu veriyi doğrudan sayfaya yansıtır.

Genellikle sosyal mühendislik ile kullanıcıya özel hazırlanmış bağlantılar gönderilir.

c. DOM Tabanlı XSS

Zararlı kod, istemci tarafında (tarayıcıda) çalışan JavaScript tarafından işlenir.

Sunucu tarafında herhangi bir işlem yapılmaz.

JavaScript ile dinamik olarak oluşturulan içeriklerde görülür.

4. XSS Saldırılarının Etkileri

XSS saldırıları şu sonuçlara yol açabilir:

Oturum Çalma: Kullanıcının oturum bilgileri çalınabilir.

Kimlik Sahtekarlığı: Kullanıcı adına işlem yapılabilir.

Veri Hırsızlığı: Form verileri, çerezler ve diğer hassas bilgiler ele geçirilebilir.

Kötü Amaçlı Yönlendirme: Kullanıcı zararlı sitelere yönlendirilebilir.

Web Sitesinin İtibar Kaybı: Güvenlik açıkları kullanıcı güvenini zedeler.

5. XSS Açığının Tespiti

XSS açıkları genellikle şu yöntemlerle tespit edilir:

Güvenlik tarama araçları (OWASP ZAP, Burp Suite, Acunetix)

Manuel testler ve kod incelemeleri

Otomatikleştirilmiş penetrasyon testleri

6. XSS'e Karşı Korunma Yöntemleri

Web uygulamalarını XSS saldırılarına karşı korumak için şu önlemler alınmalıdır:

Girdi Doğrulama (Input Validation): Kullanıcıdan gelen veriler kontrol edilmelidir.

Çıktı Kaçırma (Output Encoding): HTML, JavaScript ve URL çıktıları uygun biçimde kodlanmalıdır.

Güvenli Kodlama Prensipleri: Dinamik içerik üretiminde güvenlik öncelikli olmalıdır.

İçerik Güvenlik Politikası (CSP): Tarayıcıya hangi kaynaklardan içerik yüklenebileceğini belirten politika uygulanmalıdır.

HTTPOnly ve Secure Çerezler: Çerezlerin JavaScript erişimine kapalı ve sadece HTTPS üzerinden iletilmesi sağlanmalıdır.

7. XSS ve OWASP

OWASP (Open Web Application Security Project), XSS'i yıllardır en tehlikeli web uygulama açıkları arasında göstermektedir. OWASP Top 10 listesinde XSS, genellikle ilk sıralarda yer alır ve geliştiricilere bu açıkla ilgili farkındalık kazandırmayı amaçlar.

8. Gerçek Hayattan Örnekler

MySpace XSS Saldırısı (2005): Samy Kamkar tarafından yapılan saldırıda, bir XSS açığı kullanılarak milyonlarca kullanıcı profili etkilenmiştir.

Twitter XSS Açığı (2010): Kullanıcıların tweetlerine zararlı kod eklenerek otomatik yayılma sağlanmıştır.