Yılmaz ÜSTÜNTAŞ OWASP TOP 10 ÖDEVİ

1. Broken Acsess Control: Bu zafiyet, uygulamalarda yetersiz ya da hatalı erişim kontörlünden dolayı yetkisiz kullanıcıların uygulamada yetkisi olmayan yerlere erişmesi ve bunları değiştirmesine sebep olabilir.

1.1 IDOR (Insecure Direct Object Reference): Bir kullanıcının uygulamada ki parametreyi değiştirerek başka kullanıcıların verisine erişebilmesi.

Korunma Yolları;

-Erişim kontrolleri düzgün yapılandırılmalıdır.

-Logları tutulmalıdır.

-Token-Based Authentication kullanılmalıdır.

-Güncel olmak ve belli aralıklarla pentest yaptırılmalıdır.

2. Cryptographic Failures: Kriptografik hatalar ya da eksikliklerden kaynaklı ortaya çıkan zafiyettir. Bu zafiyet, yanlış şifreleme algoritmaları, zayıf şifreleme anahtarları veya rastgele sayı üretimindeki sorunlardan oluşabilir.

Korunma Yolları;

-Belirli zaman aralıklarında anahtar güncellemesi yapmak.

-Güçlü ve güvenli şifreleme algoritmaları kullanmak rsa gibi.

-Hassas verileri kredi kartı gibi verileri ekstra şifrelemek.

-Rastgele sayı üretimi doğru yapılmalıdır.

3. Injection: Bu zafiyet kategorisi, kullanıcıların zararlı kod veya sorgu çalıştırmasına olanak sağlamasıdır. Doğru şekilde veri girişinde doğrulama ve işleme olmaması ya da filtrelerin eksik olması nedeniyle zararlı kod enjekte edilebilir.

3.1 SQL Injection: İstenmeyen SQL sorgularını veritabanında çalıştırabilmek.

3.2 NoSQL Injection: NoSQL veritabanlarında sorguları manipüle etmek.

3.3 OS Command Injection: İşletim sistemi komutlarını çalıştırmasına olanak sağlayan zafiyettir.

3.4 Cross Site Scripting (XSS): Kullanıcının doğrulama olmadan zararlı javascript kodları çalıştırmasına olanak sağlayan zafiyettir.

3.5 Command Injection: Sistem komutları çalıştırmak.

3.6 XML Injection: XML verilerini kullanarak veri eklenmesi veya yapılandırılmasıdır.

Korunma Yolları;

-WAF kullanılması.

-Parametrik sorgular ve saklı prosedürler kullanmak.

-Parametrelerin doğrulanması.

-Kullanıcı girdilerini kısıtlamak filtrelemek.

4. Insecure Design: Güvensiz tasarım, bir web uygulamasının tasarımında hata ve eksiklikler olması nedeniyle ortaya çıkan bir zafiyettir.

Korunma Yolları;

-Yazılım güvenlik testleri yapılmalı ve hataların veya eksiklerin kapatılması.

-Güvenlik açıkları tespit edilip, düzeltilmelidir.

-Güvenli tasarım ilkeleri uygulanmalıdır.

-Tehdit modelleme yaplılmalıdır.

5. Security Misconfiguration: Bu zafiyet, uygulamanın güvenlik ayarlarının yanlış veya yetersiz yapılandırılması sebebiyle ortaya çıkan bir zafiyettir. Bu zafiyetin ortaya çıkmasında ki nedenler varsayılan ayarlar kullanılması, güncelleme eksikleri ve yanlış konfigürasyonlu firewall kullanımı gibi hatalardan kaynaklanabilir.

Korunma Yolları;

-Düzgün güvenlik konfigürasyonu yapılmalıdır.

-Güvenlik yamaları ve güncellemelerini zamanında yapmak.

-Uygulamanın güvenlik düzeyi düzenli test edilmeli.

-Varsayılan ayarlardan kaçınma.

-Kullanıcı yetkileri minimumda tutulmalı.

6. Vulnerable and Outdated Components: Güvenlik açıklarına sahip ya da güncellenmemiş bileşenler barındırması sebebiyle ortaya çıkan zafiyetlerdir.

Korunma Yolları;

-Düzenli güncelleme yapılmalı.

-Üçüncü parti bileşenlerinin takibini güzel yapmak.

-Güvenlik testleri yapılmalı.

-Düzgün güvenlik ve güncelleme adımları atılmalı.

7. Identification and Authentication Failures: Kimlik doğrulama ve tanımlama işleminde bir uygulamanın doğru bir şekilde kimlik doğrulaması ve tanımlama yapamasından kaynaklı ortaya çıkan güvenlik açığıdır.

Korunma Yolları;

-Güçlü kimlik doğrulama yöntemleri kullanılmalı.

-Kimlik bilgileri şifrelenerek korunmalı.

-Güvenli hashleme yapılmalı.

-2FA kullanılabilir.

-Düzenli yetki kontrolü ve düzenli olarak kimlik doğrulama politikalarını takip etmek.

8. Software and Data Integrity Failures: Yazılım ve veri bütünlüğü hataları, uygulamanın yazılımı ya da verilerinde beklenmeyen değişikliklere veya manipülasyonlara karşı savunmasız kalma durumunda ortaya çıkan güvenlik açığıdır.

Korunma Yolları;

-Yazılım güncellemelerini düzenli şekilde yapmak.

-Güçlü erişim kontrolü uygulamak.

-Güvenli yazılım geliştirmeleri uygulamak.

-Düzenli yedekleme ve kurtarma sistemi uygulamak.

-Dijital imzalar ve güvenilir sertifikalar kullanmak.

-Veri bütünlüğü ve güvenliğini sağlamak.

9. Security Logging and Monitoring Failures: Güvenlik olaylarının yeterince izlenememesi ve kaydedilmemesi durumunda ortaya çıkan güvenlik açığıdır. Bu açık, kaydedilme işlevlerinin yetersizliği veya hatalı olması nedeniyle saldırganların tespit edilememesi veya güvenlik sistemlerinin güvenlik olaylarına müdahale edememesine neden olabilir.

Korunma Yolları;

-Logların düzenli olarak incelenmesi.

-Alarm sistemleri kullanmak.

-Güncel güvenlik politikalarının ve prosedürlerinin düzenli olarak takip edilmesi.

-Güçlü izleme ve uyarı araçları kullanmak.

-Sürekli izleme.

10. Server-Side Request Forgery: Bir saldırganın sistemde ki bir sunucuyu kullanarak o sistemde ki iç ağlara sunuculara veya diğer harici kaynaklara istek yollamasına ve erişmesine sebep olan güvenlik açığıdır.

10.1 Classic SSRF: Saldırganın sunucuyu kullanarak başka bir sisteme istek atarak etkileşime geçmesi.

10.2 Blind SSRF: Saldırganın istek gönderdiği yeri ve isteğin sonucunu doğrudan göremediği ama yine de sunucunun başka bir sisteme istek gönderdiği saldırıdır.

10.3 Subdomain Takeover SSRF: Sunucunun belirli alt alan adlarını çözerek istek yapmasını sağlayan saldırı türüdür.

10.4 File Protocol SSRF: Saldırgan sunucunun dosya protokollerini kullanarak sisteme erişim sağlaması.

10.5 Port Scanning SSRF: Saldırgan sunucunun diğer sistemlerinin portlarını taramasına erişim sağlaması.

10.6 Cloud Metadata SSRF: Saldırganın bulut servislerinde ki API larına erişim sağlama amacıyla kullanılan saldırı türü.

Korunma Yolları;

-Güvenil duvarı kullanımı.

-Giriş doğrulaması.

-Erişim kontrolü sağlanması.

-Sunucu ayarlarının düzgün yapılandırılması.