OWASP Top 10

OWASP Top 10, web uygulamalarında en yaygın ve tehlikeli güvenlik açıklarını sıralayan bir rehber niteliğindedir. Bu liste, geliştiricilerin ve güvenlik uzmanlarının dikkat etmesi gereken en kritik zafiyetleri içeriyor. Liste her yıl değişmiyor, ancak 2017 yılında yayınlanan listede bazı değişiklikler yapıldı ve 2021 yılında güncellenmiş hali yayınlandı.

2021 yılı itibarıyla OWASP Top 10 listesi;

1. Broken Access Control:

Kullanıcıların yetkilendirilmemiş kaynaklara erişmesine izin veren bir zafiyettir. Erişim kontrollerinin yetersiz veya yanlış yapılandırılması, genellikle bu tür açıkların sebebidir.

Türleri; **Privilege Escalation:** Kullanıcıların yetki seviyelerini artırmaları.

Nasıl Önlenir?

- Erişim kontrollerini her seviyede uygulayın.
- Kullanıcı yetkilerini kontrol edin ve doğrulayın.

2. Cryptographic Failures

Şifreleme ve veri koruma ile ilgili hatalardır. Bu tür açıklar, hassas verilerin sızmasına veya kötüye kullanılmasına neden olabilir. Zayıf şifreleme algoritmaları, şifrelenmemiş veri iletilmesi veya kötü yönetilen anahtarlar bu tür açıkların kaynağıdır. Türleri;

Unencrypted Data: Şifrelenmemiş veri aktarımı veya depolama.
Weak Encryption: Güvenlik standartlarına uygun olmayan şifreleme yöntemleri.

Nasıl Önlenir?

- Hassas verileri her zaman şifreleyin.
- Güçlü ve güncel şifreleme algoritmaları kullanın.
- Veriyi güvenli bir şekilde depolayın ve iletin.

3. **Injection**

Kötü niyetli verilerin bir uygulamanın sorgu veya komutlarına dahil edilmesidir. Bu durum, veritabanı sorguları, komut satırı komutları veya diğer veri işleme mekanizmalarına zarar verebilir. Türleri;

SQL Injection: Veritabanı sorgularında kullanılan kodun manipüle edilmesi.
Command Injection: Sistem komutlarının kötüye kullanılması.
XML Injection: XML verilerinin manipüle edilmesi.

Nasıl Önlenir?

- WAF kullanımı
- Parametreli sorgular veya hazırlıklı ifadeler kullanarak, kullanıcı girişlerini sorgu yapısından ayırarak SQL enjeksiyonu gibi güvenlik açıklarını önleyin.

4. **Insecure Design**

Uygulama tasarımındaki güvenlik zayıflıklarıdır. Bu, uygulamanın temel tasarım prensiplerinde güvenlik önlemlerinin yeterince düşünülmemesinden kaynaklanır. Türleri;

Poor Security Design: Güvenlik standartlarına uymayan tasarımlar.
Lack of Threat Modeling: Tehdit modellemesi eksiklikleri.

Nasıl Önlenir?

- Güvenlik tasarımı prensiplerini uygulayın.
- Tehdit modellemesi ve risk değerlendirmeleri yapın.
- Güvenli kodlama standartlarına uyun.

5. **Security Misconfiguration**

Güvenlik yapılandırmalarında yapılan hatalardır. Yanlış yapılandırmalar, sistemin güvenliğini tehlikeye atabilir. Varsayılan ayarların kullanılmasından, hatalı yapılandırmalardan veya güncel olmayan yazılım bileşenlerinden kaynaklanabilir. Türleri;

Default Credentials: Varsayılan kimlik bilgileri kullanımı.
Unnecessary Features Enabled: Gereksiz özelliklerin etkinleştirilmesi.

Nasıl Önlenir?

- Güvenlik yapılandırmalarını düzenli olarak gözden geçirin.
- Varsayılan ayarları değiştirme ve gereksiz bileşenleri devre dışı bırakma.
- Yazılımları ve bileşenleri güncel tutma.

6. **Vulnerable and Outdated Components**

Güvenlik açıkları bulunan veya güncel olmayan yazılım bileşenleridir. Bu tür açıklar, sistemin çeşitli güvenlik tehditlerine maruz kalmasına neden olabilir. Eski yazılım sürümleri, güncellenmemiş bileşenler ve bilinen güvenlik açıklarına sahip kütüphaneler bu tür açıkların nedenidir. Türleri;

Outdated Libraries: Güncel olmayan kütüphaneler.

Known Vulnerabilities: Bilinen güvenlik açıklarına sahip bileşenler.

Nasıl Önlenir?

- Güncel sürümleri ve güvenlik güncellemelerini takip edin.

7. **Identification and Authentication Failures**

Kimlik doğrulama ve oturum yönetimi ile ilgili güvenlik açıklarıdır. Bu açıklar, kötü niyetli kişilerin kullanıcı hesaplarına yetkisiz erişim sağlamasına neden olabilir. Kimlik doğrulama ve oturum yönetiminin yetersiz uygulanması, zayıf parolalar veya oturum yönetim hataları bu tür açıkların kaynaklarıdır. Türleri;

Brute Force: Parolaların deneme yanılma yoluyla tahmin edilmesi.
Session Fixation: Kullanıcı oturumunun sabitlenmesi ve kötüye kullanılması.

Nasıl Önlenir?

- Güçlü şifre politikaları uygulayın.
- Oturum sürelerini ve kullanıcı kimlik doğrulama mekanizmalarını güvenli şekilde yönetin.
- Çok faktörlü kimlik doğrulama (MFA) kullanın.

8.Software and Data Integrity Failures

Yazılım ve veri bütünlüğü ile ilgili güvenlik açıklarıdır. Bu tür açıklar, verilerin veya yazılım bileşenlerinin bütünlüğünün bozulmasına neden olabilir.Veri ve yazılım bütünlüğünün sağlanmaması, kötü niyetli değişikliklere izin vermek veya güvenlik açıkları bu tür zafiyetlere yol açar. Türleri;

Insecure Software Updates: Güvenli olmayan yazılım güncellemeleri.

Compromised Software: Kötü niyetli yazılım bileşenleri.

Nasıl Önlenir?

- Veri bütünlüğünü korumak için imza ve doğrulama yöntemlerini kullanın.

9. Security Logging and Monitoring Failures

Güvenlik Kayıtları ve İzleme Hataları, bir sistemin güvenlik olaylarını yeterince kaydedememesi ve bu olayları izleyememesi durumunu ifade eder. Bu tür eksiklikler, saldırıların zamanında tespit edilememesine ve uygun yanıtların verilememesine neden olabilir. Örneğin, bir saldırganın sisteme erişim sağladığı tespit edilemeyebilir veya olay sonrasında yeterli kanıt toplanamayabilir.

Nasıl Önlenir:

- Kapsamlı bir güvenlik izleme ve olay yönetimi sistemi kurun.
- Tüm önemli güvenlik olaylarını kaydedin ve düzenli olarak izleyin.
- Anormal davranışları tespit etmek için otomatik izleme araçları kullanın.

10. Server-Side Request Forgery (SSRF)

Sunucu Taraflı İstek Sahteciliği (SSRF), saldırganın bir sunucunun güvenilmeyen kaynaklara istek göndermesini sağlamasıdır. SSRF açıkları, saldırganların sunucular üzerinden iç ağlara, diğer sistemlere veya kötü amaçlı kaynaklara erişim sağlamasına olanak tanır. Bu tür bir açık, güvenlik duvarlarının ve diğer güvenlik önlemlerinin aşılmasına neden olabilir.

Nasıl Önlenir:

- Güvenli URL işleme
- Sunucu ayarlarının kontrol edilmesi