**OWASP TOP 10**

**1-Broken Acces Control**

Broken Acces Control, yetkili olmayan bir kullanıcının sisteme veya uygulamaya erişim hakkı olmadığı halde, bu kaynaklara erişmesine veya bunları değiştirmesine neden olabilir. Oluşma sebebi  uygulamalarda yapılan yanlış veya yetersiz erişim kontrolü ayarları sonucu, yetkisiz kullanıcıların uygulama içerisinde yer alan kaynaklara erişmesine olanak sağlanmasıdır.

Broken Access Control açığından korunmak için alınması gereken önlemler:

· Erişim kontrolleri düzgün yapılandırılmalıdır,

· Token-based authentication kullanılmalıdır,

· İşlemlerin logları tutulmalı

**2-Cryptographic Failures**

Cryptographic Failures, kriptografi kullanılarak korunan verilerin, yanlış bir şekilde şifrelendiği veya şifresinin çözüldüğü durumlardır. Uygulamalarda kullanılan şifreleme algoritmalarının yanlış seçilmesi veya uygulanması, anahtar yönetimi hataları veya rastgele sayı üretimindeki eksiklikler nedeniyle oluşur.

Cryptographic Failures açığından korunmak için alınması gereken önlemler:

· Doğru şifreleme algoritmaları kullanılmalıdır

· Anahtar yönetimi hatalarından kaçınılmalıdır

· Rastgele sayı üretimi doğru yapılmalıdır

· Kriptografik anahtarların düzenli olarak yenilenmesi gerekmektedir

**3-Injection**

Uygulamalarda kullanılan veri girişleri yoluyla, kötü niyetli kullanıcıların uygulama tarafından yürütülen veritabanı sorgularına, komutlara veya diğer işlevlere veri girişlerinde yapılan hatalı denetlemeler veya filtrelemeler nedeniyle zararlı kod enjekte etmesini sağlar. Uygulamalarda veya yapılan yazılımlarda verinin kontrol edilmeden direkt veri tabanı sorgusunda kullanılması ile ortaya çıkar.

Injection açığından korunmak için alınması gereken önlemler:

· Parametrelerin Doğrulanması

· SQL Parametreleştirme

· Kodlama Standartları

· WAF Kullanımı

**4-Insecure Design**

Insecure Design (Güvensiz Tasarım), bir web uygulamasının tasarımında yapılan hatalar veya eksiklikler nedeniyle ortaya çıkan bir güvenlik açığıdır. Bu, uygulamanın tasarımında yapılan hataların, uygulamanın tüm yaşam döngüsü boyunca devam etmesine ve güvenliğini olumsuz etkilemesine neden olabilir. Güvensiz tasarım, uygulamanın özellikle kimlik doğrulama, yetkilendirme, veri gizliliği ve bütünlüğü gibi önemli güvenlik konularında hatalar içermesiyle ortaya çıkabilir.

Insecure Design açığından korunmak için alınması gereken önlemler:

· Güvenli tasarım ilkeleri uygulanmalıdır

· Uygulama geliştirme ekibi, güvenlik odaklı bir yaklaşım benimsemelidir

· Güvenlik açıkları düzeltilmelidir

· Uygulama güvenliği için en iyi uygulamalar kullanılmalıdır

# 5-Security Misconfiguration

Security Misconfiguration, bir uygulamanın ya da sistemlerin yanlış yapılandırılması veya yapılandırılmamış olması sonucu oluşan bir güvenlik açığıdır. Bu açık, sistemin, uygulamanın veya sunucuların güvenliğini sağlamak için gereken en iyi uygulamaların uygulanmamış veya eksik uygulanmış olmasından kaynaklanabilir.

Security Misconfiguration açığından korunmak için alınması gereken önlemler:

· En iyi uygulamaları takip etmek

· Yazılımın güvenlik düzeyini kontrol etmek

· Güvenlik yamalarını ve güncellemeleri takip etmek

· Sistem yapılandırmasını güncellemek

**6-Vulnerable and Outdated Components**

Bir uygulamada kullanılan üçüncü taraf bileşenlerin güncellenmemesi veya bilinen güvenlik açıklarına sahip olması nedeniyle oluşan bir güvenlik açığıdır.  Bu bileşenler, genellikle web uygulama çerçeveleri, veritabanı yönetim sistemleri, açık kaynak kütüphaneler, sunucu yazılımı ve diğer araçlar gibi yazılım bileşenleri olabilir. Bu bileşenlerde güvenlik açıkları keşfedilmesi halinde, uygulama açığa karşı savunmasız hale gelebilir.

Vulnerable and Outdated Components açığından korunmak için alınması gereken önlemler:

· Bileşenleri izlemek

· Güncelleme politikaları oluşturmak

· Bileşenleri doğrulamak

**7-Identification and Authentication Failures**

Bir kullanıcının kimliğinin doğrulanması veya yetkilendirilmesi sırasında yaşanan sorunlar nedeniyle oluşan bir güvenlik açığıdır. Kimlik doğrulama ya da oturum yönetimi ile ilgili fonksiyonların yanlış uygulanması sonucunda ortaya çıkar. Saldırganlar parolaları, session token ları ile ele geçirebilir.

Identification and Authentication Failures açığından korunmak için alınması gereken önlemler:

· Güçlü kimlik doğrulama yöntemleri kullanmak

· Kimlik doğrulama işlemlerini izlemek

· Kimlik bilgilerini şifrelemek

· Düzenli olarak kimlik doğrulama politikalarını kontrol etmek

· Çok faktörlü kimlik doğrulama yöntemlerini kullanmak

**8-Software and Data Integrity Failures**

Yazılım ve veri bütünlüğü hataları, bir yazılım veya veri sisteminin beklenmeyen şekilde değiştirilmesi bozulması sonucu meydana gelen bir güvenlik açığıdır. Yazılım ve veri bütünlüğü hataları, bir yazılım güncellemesi sırasında, yazılımın kötü amaçlı bir saldırgan tarafından değiştirilmesi veya bir saldırganın bir veri depolama ortamına kötü amaçlı yazılım yerleştirmesi gibi birçok farklı şekilde ortaya çıkabilir. Bu tür saldırılar sonucunda, saldırgan verileri çalabilir, verileri bozabilir veya sistemi kontrol etmeye başlayabilir.

Software and Data Integrity Failures açığından korunmak için alınması gereken önlemler:

· Yazılımın veya verilerin beklenen kaynaktan olduğunu ve değiştirilmediğini doğrulamak için dijital imzalar veya benzer mekanizmaların kullanımı

· Npm veya Maven gibi kitaplıkların ve bağımlılıkların güvenilir depoları kullandığından emin olmak

· Kod ve yapılandırma değişiklikleri için bir inceleme süreci oluşturmak

· Yazılım güncellemelerini düzenli olarak yüklemek

· Güçlü erişim kontrolü uygulamak

· Güvenli yazılım geliştirme yöntemlerini kullanmak

· Yazılım ve veri yedeklemeleri oluşturmak

**9- Security Logging and Monitoring Failures**

Güvenlik olaylarının izlenmesi ve kaydedilmesi işlevlerinin yetersizliği veya hatalı yapılandırılması sonucu ortaya çıkan bir güvenlik açığıdır. Bu tür bir açık, kötü amaçlı aktivitelerin tespit edilememesi veya güvenlik olaylarına yanıt verilememesi gibi sonuçlara neden olabilir.

Security Logging and Monitoring Failures açığından korunmak için alınması gereken önlemler:

· Güvenlik olaylarının izlenmesi ve kaydedilmesi için uygun araçlar kullanmak

· Günlük kayıtlarının düzenli olarak incelenmesi

· Uyarı ve alarm sistemleri kullanmak

· Güvenlik politikalarının ve prosedürlerinin düzenli olarak gözden geçirilmesi

**10- Server-Side Request Forgery**

Sunucu Taraflı İstek Sahtekarlığı (SSRF), bir saldırganın hedef sunucuda bir URL’ye istek gönderirken sahte bir kaynak IP adresi ve alan adı sağlayarak sunucunun kendi iç ağlarına veya diğer harici kaynaklara erişmesine izin veren bir web güvenliği açığıdır. Bir SSRF saldırısı, bir uygulamanın doğrulama sürecini atlayarak veya bir URL’de bir sunucu adresi veya IP adresi gibi güvenliği kontrol etmeyen bir parametre kullanarak gerçekleştirilebilir.

Server-Side Request Forgery açığından korunmak için alınması gereken önlemler:

· Giriş doğrulaması

· Güvenlik duvarı kurulumu

· Güvenli URL işleme

· Sunucu ayarlarının kontrol edilmesi

Son olarak tüm zafiyetlerde korunmak için uygulanabilecek ortak yöntemler:

· Profesyoneller tarafından güvenlik testlerinin yapılması,

· Logların tutulup profesyonel ekipler tarafından faaliyetlerin izlenmesi,

· Yazılım geliştirici ve kullanıcılara farkındalık artırma faaliyetleri yapılmasıdır.