**YAVUZLAR ODEV**

*(OWASP TOP 10)*

A01:2021-Broken Access Control

*Zafiyet nedir*:

Erişim denetimi politikası, belirli kullanıcılara belirli yetkiler ve izinler verilmesini sağlar. Yanlış yapılandırmalar sonucu bu zafiyet ortaya çıkar. Sıradan bir kullanıcının adminin yetkilerine sahip olmaması ve adminin yapabileceği şeyleri yapamaması buna örnek verilebilir.

*Neden kaynaklanır*:

Bu zafiyet hatalı yetkilendirme veya kullanılan politikanın olması gerektiği gibi hazırlanmaması sonucu ortaya çıkar.

Türleri;

IDOR (Insecure Direct Object References): Url parametrelerindeki spesifik kimlikleri değiştirerek o kimliklere izinsiz erişim sağlamak.

Yatay Yetki Atlaması: Kendisiyle aynı yetkilere sahip olan başka bir kullanıcının işlemlerini gerçekleştirebilmesi.

Dikey Yetki Atlaması: Kendisinden daha üst yetkilere sahip kullanıcının işlemlerini gerçekleştirebilmesi. ( Normal bir userın, admin yetkilerine erişebilmesi buna örnek olarak verilebilir.

Kısıtlı İşlemlere veya Sayfalara Erişim sağlanması: Yetkisiz erişim sağlamak. Örnek olarak admin paneline url üzerinden doğrudan kolaylıkla erişim sağlanması verilebilir.

Nasıl Önlenir:

-Erişimlerin kontrolleri olması gerektiği gibi yapılandırılmalıdır.

-İşlemlerin logları tutulabilir.

-Authentication türleri kullanılmalıdır.

A02:2021-Cryptographic Failures

Tanım:

Kriptografi veriyi gizleme ve şifreleme yöntemlerini içerir. Verilerin yalnızca yetkili kişiler tarafından anlaşılabilir ve okunabilir olmasını sağlar.

Kısacası amacı veriyi şifreleyerek verinin gizliliğini ve bütünlüğünü korumaktır.

Algoritmalardaki hatalar veya kullanılan prosedürün güncel olmaması sonucu bu zafiyet ortaya çıkar.

Neden kaynaklanır:

Kullanılan algoritmanın günümüz standartlarına göre güvenli olmayan eski tip şifreleme algoritmaları olması.

Algoritmanın yanlış, hatalı yapılandırılmasından da kaynaklı olabilir.

Nasıl Önlenir:

Güncel ve yeterince güvenli şifreleme algoritmaları kullanılabilir.

Güvenli hash fonksiyonları kullanılabilir ve Güvenlik protokollerinin doğru bir şekilde izlenmesi ile de önlenebilir.

Kriptografik anahtarların düzenli olarak yenilenmesi gerekmektedir.

A03:2021-Injection

Tanım:

Saldırganın uygulama sistemine yetkisiz bir şekilde komut veya veri enjekte ederek sistemi manipüle etmesidir. Uygulamanın zayıflıklarından yararlanılır. Özet olarak hatalı filtremeler sonucu saldırganın sistemi manipüle edebilmesidir.

Türleri:

SQLi( SQL Injection: Veritabanına gönderilen kötü niyetli SQL sorguları sayesinde gerçekleştirilir. Manipüle edilerek verilere erişim sağlanılabilir veya değiştirilebiir. Giriş formuna ‘admin veya ‘1’=’1’ gibi bir komutla giriş yapılabilinmesi bypass’a örnektir.

Command Injection: Kötü niyetli kullanıcıların uygulama komutlarına kötü niyetli komutlar enjekte etmesidir.

XML Injection: Kötü niyetli kullanıcının XML verilerine kötü amaçlı olarak kendi zararlı XML kodlarını eklemesinden kaynaklı meydana gelir.

Neden kaynaklanır:

Girdilerin yeteri kadar doğrulanmaması ve filtrelemenin doğru yapılmaması. Form alanlarına girilen verilerin düzgün filtrelenmemesi sonucu saldırgan sisteme istediği veriyi sokabilir.

Dinamik SQL sorguları: Kullanıcının girdiği verinin doğrudan SQL sorgusuna sokulması, SQL injectiona yol açar.

Hatalı ve eksik güvenlik önlemleri, kodlama hataları vb.

Nasıl önlenir:

Waf kullanımı, Güvenli API kullanımı, Girdinin doğrulanması ve filtrelenmesi önlenme yöntemlerinden bazılarıdır ve oldukça etkilidirler.

A04:2021-Insecure Design

Tanım: Uygulamanın tasarım kısmında yapılan güvenlik hatalarıdır ve tasarım güvenlik açısından yetersizdir.

Türleri:

Güvenlik ilkelerinin eksiklikleri, Veri doğrulama eksiklikleri, Yetersiz şifreleme vs.

Neden kaynaklanır:

Yazılım ve güvenlik kısmının koordine çalışmaması, Güvenlik uygulamalarının yeterince dikkate alınmaması veya göz ardı edilmesi.

Nasıl önlenir:

Güvenlik politikalarının dikkatli hazırlanması, Güvenlik testleri ile test edilmesi, Ekiplerin birbirleri ile iletişim içerisinde ilerlemesi ve hataların anlık olarak giderilmesi. Kısacası muhtemelen güvenlik açıklarının minimalize edilmesi hatta kapatılması.

A05:2021-Security Misconfiguration

Tanım:

Bir uygulamanın, sunucunun, veritabanının veya başka bir bileşenin güvenlik yapılandırmasının yanlış veya eksik yapılmasıdır.

Türleri:

Default ayarların kullanılması: Admin kullanıcı ve şifresinin admin olarak bırakılması örnek verilebilir.

Gereksiz hizmetlerin ve portların açık bırakılması, Güvenlik özelliklerinin kapatılmaması vb.

Neden kaynaklanır:

Yanlış veya hatalı yapılandırma, Güvenlik güncellemelerinin ihmal edilmesi veya yapılmaması.

Nasıl önlenir:

Güvenlik ilkelerinin düzenli ve doğru bir şekilde uygulanması.

Gereksiz hizmet ve portların açık bırakılmaması.

Varsayılan ayarların düzeltilmesi.

A06:2021-Vulnerable and Outdated Components

Tanım:

Uygulamada kullanılan bileşenlerin güncellenmemesi veya güvenlik açıklarına sahip olmasından kaynaklı meydana gelen bir güvenlik açığıdır.

Bu bileşenlere örnek oalrak veri tabanı yönetim sistemleri, açık kaynak kütüphaneler vb verilebilir.

Neden kaynaklanır:

Eski ve güncel olmayan güvenlik teknolojilerinin kullanılması, Bileşen yönetiminde yeterince tecrübeli olmamak ve hataların olmaması, güvenlik bilgisi eksikliği vb.

Nasıl Önlenir:

Uygulama üzerinde testler yapılabilir, Güvenlik policylerinin düzgün entegre edilmesi gerçekleştirilebilir.

Desteklenmeyen ve güncel olmayan bileşenlerin kullanılmasına son verilmesi vb.

Güncelleme policyleri oluşturulabilir.

A07:2021-Identification and Authentication Failures

Tanımı:

Sistemin kullanıcılarının kimliklerini tanımlama ve doğrulama süreçlerinde yaşadığı sorunları belirtir. Bu tür hatalar güvenlik kontrollerinin sağlanması süreçlerinin güvenliğini azaltır.

Türleri:

Zayıf parola poliycyleri ( 1234 gibi parolalara izin verilmesi), Parolaların sızması, Yetersiz doğrulama, Oturum sonlandırma sisteminin düzgün çalışmaması ( Çıkış yapan kullanıcının oturumunun hala açık tutulması) vb.

Neden Kaynaklanır:

Güvenlik policylerinin yetersiz olması ( kimlik doğrulama süreçlerindeki hatalar.),

Varsayılan ayarların kullanılması ( Şifrelerin hashlenmemesi gibi),

Güvenlik bilgisi eksikliği ( Doğrulama ve güvenlik mekanizmaları hakkında yeterli bilgi ve eğitime sahip olunmaması).

Nasıl Önlenir:

Güvenli parola policyleri: Güçlü ve karmaşık parola kullanımına teşvik edilmesi.

Şifreleme: Veri tabanındaki kayıtlı parolaların kaydedilirken şifrelenmesi ( hashlenmesi)

Çok faktörlü kimlik doğrulama (MFA): Kimlik doğrulaması için birden fazla doğrulama metodu kullanılması. Ekstradan doğrulama kodu istenmesi gibi.

A08:2021-Software and Data Integrity Failures

Tanım: Software and Data Integrity Failures, veriler ve yazılımların değiştirlemez, doğrulanamaz veya güvenli bir şekilde saklanamamasından kaynaklanan hatalar bütünüdür.

Bu tür hatalar yazılım bileşenlerindeki yetkisiz değişikliklere ve manipülasyonlara yol açabilir.

Türleri:

Yazılım Bütünlüğü Hataları (Güvenlik kontrolleri yapılmadan yazılım bileşenlerinin ve güncellemelerinin dağıtılması.)

Kaynak kod ve Üçüncü taraf bileşenlerin güvenliğinin doğrulanamaması ve sağlanamaması.

Neden Kaynaklanır:

Güvenlik testlerinin yapılmaması ( veri yapılandırmalarının veya yazılım bileşenlerinin güvenlik testlerinin sağlanmaması.)

Eksik veya Hatalı yapılandırmalar ( Yazılım ve veri bütünlüğü yapılandırmalarının eksik veya yanlış yapılması.

Nasıl Önlenir:

Düzenli olarak güvenli güncellemelerinin yapılması.

Paket bütünlüğünü doğrulamak için kriptografik yöntemlere başvurulması.

Düzenli güvenlik testleri ( Penetrasyon Testleri): Potansiyel risklerin ve açıkların belirlenmesi için yetkin ekipler ve kişiler tarafından düzenli olarak penetration test yapılması.

Verilerin hashlenmesi : Dijital imzalar ve hash’ler kullanarak bütün kontrollerin uygulanması.

A09:2021-Security Logging and Monitoring Failures

Tanım:

Sistemlerdeki güvenlik olayların, log kayıtlarının izlenmesi, analiz edilmesi ve kaydının tutulması süreçlerinde meydana gelen eksikliklerden kaynaklanır. Bu tarz hatalar saldırıların ve güvenlik ihlallerinin tespit edilip yanıtlanmasında zorluklar çıkarır.

Türleri:

Yetersiz kayıt tutulması ( Olay tespiti yaparken yalanılan zorluklar)

Yanlış yapılandırılmış log kayıt sistemi

Güvenlik günlüğü dosyalarının şifrelenmemesi ve yetkisiz erişimlere açık olması.

Neden kaynaklanır:

Hatalı yapılandırmalar: Günlükleme ve izleme sistemlerindeki kurul hataları ve yanlış yapılandırmalar.

Kaynak kısıtlamaları: Günlükleme ve izleme sistemleri için yeterli kaynağın ve sistemin ayrılmaması.

Nasıl önlenir:

Gerçek zamanlı izleme: Anlık izleme araçları vasıtasıyla sistemdeki etkinliklerin ve güvenlik olaylarının izlenilip anlık müdahale edilmesi.

Otomatik yanıtlama sistemi: Yazılan kurallara bağlı olarak olaylara hızlı ve etkili bir şekilde müdahale edilmesi.

Düzenli inceleme ve testler: Kayıtların düzenli olarak incelenmesi ve testlerin yapılması. Güvenlik açıklarının ve tehditlerin belirlenmesi için eylemler yapılması.

A10:2021-Server-Side Request Forgery

Tanım:

SSRF, saldırganın sunucudaki uygulamanın sahip olduğu yetkileri kullanarak kendi istediği istekleri göndermesine olanak tanır. Bu tarz bir açık, genellikle uygulamanın dış hizmetlere veya kaynaklara erişim sağlaması gerektiğinde ortaya çıkar. SSRF ile saldırgan, veritabanlarına, iç sistemlere, API’lara veya diğer kaynaklara erişim sağlayabilir.

Türleri:

Dış Url’lere İstek Gönderme ( Saldırgan url doğrulama aracılığıyla uygulama sunucusuna dış bir Url’ye istek göndermesini sağlar)

Illegal API kullanımı ( Uygulamanın yetkisiz API’lara ve hizmetlere erişim sağlaması)

Port tarama ve Hizmet Keşfi ( SSRF açığı aracılığıyla saldırganın iç ağda port taraması yapması.)

Neden Kaynaklanır:

Hatalı izinler: Yetkisiz bir şekilde dış hizmetlere istek gönderebilmesi.

Yetersiz Girdi Doğrulama: Kullanıcının veya Url’lerin diğer girişlerini yeterince doğrulamamak. Bundan kaynaklı saldırgan sunucuya zararlı istekler gönderebilir.

Nasıl Önlenir:

Girdi doğrulaması: Kullanıcıdan alınan URL’lerin sık sık doğrulanması.

URL Listeleme: Beyaz liste ( izinli urller) veya Siyah liste ( yasaklı listeler) yaparak policyler yapmak.

Güvenlik testleri ve Tarama: Penetration test ile açıkların tespit edilmesi ve kapatılması.  
API Güvenliği: API’lerin güvenli bir şekilde yapılandırılmaları ve yalnızda onaylanmış ve yetkili isteklerin işleme geçmesinin sağlanması.

**Alperen Kolenoğlu**