OWASP TOP 10

A01:2021-Broken Access Control

Broken Access Control, bir sistemdeki yetki veya izin kontrollerinin hatalı ya da eksik olması durumudur. Bu durum, kullanıcıların veya saldırganların, erişim hakları olmayan verilere veya işlevlere erişmesine, onları değiştirmesine veya silmesine olanak tanır. Basit bir örnekle açıklamak gerekirse, bir web uygulamasında bir kullanıcının yalnızca kendi profil bilgilerini görüntüleyebilmesi gerekirken, bozuk erişim kontrolü nedeniyle başka kullanıcıların profillerine de erişebilmesi bu güvenlik açığına bir örnektir.

Neden Kaynaklanır ?

Geliştiriciler, kullanıcıların belirli kaynaklara veya işlevlere erişim yetkilerini doğru şekilde kontrol etmeyi ihmal edebilir. Örneğin, bir API çağrısında, kullanıcının gerçekten o kaynağa erişme yetkisi olup olmadığını kontrol etmeyi unutmak.

Erişim kontrollerinin istemci tarafında (tarayıcıda) yapılması, saldırganların bu kontrolleri atlatabilmesine neden olabilir.

Yetkilendirme işlemlerinde kullanılan mantıksal hatalar, bir kullanıcının erişmemesi gereken bir kaynağa erişmesine yol açabilir. Bu, genellikle yeterince test edilmemiş veya karmaşık yetkilendirme mantıklarının sonucudur.

Nasıl Önlenir ?

Kullanıcılara yalnızca işlerini yapmaları için gerekli olan minimum izinleri verin. Gereksiz izinlerin verilmemesine özen gösterin.

Erişim kontrolünü uygulamak için merkezi bir yetkilendirme mekanizması kullanın. Bu, farklı modüller arasında tutarlı bir erişim kontrolü sağlamaya yardımcı olur ve yönetimi kolaylaştırır.

Erişim kontrol mekanizmalarının doğruluğunu test etmek için güvenlik testleri gerçekleştirin. Bu testler, kullanıcıların sadece izin verilen kaynaklara erişip erişemediğini kontrol etmeyi içermelidir.

A02:2021-Cryptographic Failures

Cryptographic Failures, bir sistemin gizlilik, bütünlük veya kimlik doğrulama gibi güvenlik özelliklerini sağlamak için kullanılan kriptografik yöntemlerin yanlış uygulanması veya yanlış seçilmesinden kaynaklanan güvenlik açıklarıdır. Bu tür hatalar, hassas bilgilerin yetkisiz kişiler tarafından ele geçirilmesine, değiştirilmesine veya kimlik doğrulamanın atlatılmasına yol açabilir.

Neden Kaynaklanır ?

Eski veya güvenliği kırılmış kriptografik algoritmaların (örneğin, MD5, SHA-1) kullanılması.

Yetersiz uzunlukta veya tahmin edilebilir şifreleme anahtarlarının kullanılması.

Anahtarların güvenli olmayan bir şekilde saklanması, iletilmesi veya yenilenmemesi.

Hassas verilerin (şifreler, kişisel bilgiler) şifrelenmeden saklanması veya iletilmesi.

SSL/TLS gibi güvenli iletişim protokollerinin yanlış yapılandırılması veya güvenli olmayan versiyonlarının kullanılması.

Nasıl Önlenir ?

AES, SHA-256 gibi modern ve güvenli algoritmalar kullanın. Eski ve zayıf algoritmalardan kaçının.

Şifreleme anahtarlarını güvenli bir şekilde saklayın, yönetin ve gerektiğinde düzenli olarak değiştirin. Anahtarların uzunluğu yeterli olmalı ve tahmin edilemez olmalıdır.

SSL/TLS gibi güvenli iletişim protokollerinin en güncel ve doğru yapılandırılmış versiyonlarını kullanın.

Hassas bilgileri saklamadan veya iletmeden önce mutlaka şifreleyin. Şifreleme yapmadan verileri iletmek veya saklamak büyük riskler taşır.

A03:2021 – Injection

Injection, bir saldırganın, uygulamaya kötü niyetli veriler (genellikle komutlar veya kod parçaları) enjekte ederek, arka uç sistemlerine yetkisiz komutlar çalıştırmasını sağlayan bir güvenlik açığıdır. Bu tür saldırılar, veritabanı sorguları, işletim sistemi komutları, LDAP sorguları veya diğer komutlar üzerinde gerçekleşebilir

Neden Kaynaklanır ?

.Enjeksiyon saldırıları, genellikle kullanıcı girdilerinin yeterince doğrulanmadığı veya temizlenmediği durumlarda ortaya çıkar.

Türleri Nelerdir ?

SQL Injection, saldırganın, bir uygulamaya gönderilen SQL sorgularına kötü niyetli SQL komutları enjekte etmesiyle ortaya çıkar. Bu durum, saldırganın veritabanına yetkisiz erişim sağlamasına, verileri görüntülemesine, değiştirmesine veya silmesine olanak tanır.

Command Injection, saldırganın, uygulamanın çalıştırdığı işletim sistemi komutlarına kötü niyetli komutlar enjekte etmesiyle oluşur. Bu tür bir saldırı, sistemdeki dosyalara erişim, sistem bilgilerini elde etme veya zararlı yazılımlar yükleme gibi sonuçlara yol açabilir.

LDAP Injection, saldırganın, bir uygulamanın LDAP (Lightweight Directory Access Protocol) sorgularına kötü niyetli girdiler enjekte ederek dizin hizmetlerine yetkisiz erişim sağlamasıdır. Bu saldırı, kullanıcı hesaplarını ele geçirme veya dizin hizmetindeki verileri değiştirme gibi sonuçlara yol açabilir.

XML Injection, saldırganın, XML verilerine kötü niyetli girdiler enjekte ederek XML tabanlı veri işlemeyi bozmasıdır. Bu saldırı, veri manipülasyonu, XML şemasının kırılması veya yetkisiz veri erişimine yol açabilir.

NoSQL Injection, NoSQL veritabanlarında yapılan sorgulara kötü niyetli girdiler enjekte edilerek, veri manipülasyonu veya yetkisiz erişim sağlanmasıdır.

Nasıl Önlenir ?

SQL, LDAP, NoSQL ve diğer sorgularda, kullanıcı girdilerini doğrudan sorguya eklemek yerine parametrik sorgular kullanarak, enjekte edilen girdilerin komut olarak değil veri olarak ele alınmasını sağlayın.

Kullanıcıdan gelen tüm verileri doğrulayın ve temizleyin. Girdilerin beklenen formatta olup olmadığını kontrol edin ve tehlikeli karakterleri (örneğin, ', ;, --) kaldırın veya etkisiz hale getirin.

Veritabanı ve sistem kullanıcılarına yalnızca gerekli olan minimum erişim izinlerini verin. Bu, saldırganın elde edebileceği zararları sınırlayabilir.

Uygulamanızı bir web uygulaması güvenlik duvarı (WAF) ile koruyarak, bilinen enjeksiyon saldırılarına karşı ek bir koruma katmanı ekleyin.

A04:2021 – Insecure Design

Insecure Design, bir sistemin veya uygulamanın, güvenlik tehditlerine karşı yeterli koruma sağlayamayacak şekilde tasarlanmış olması durumudur. Bu, güvenlik kontrollerinin eksik veya yetersiz olması, güvenlik tehditlerinin dikkate alınmaması veya kötü tasarım kararlarının alınması nedeniyle ortaya çıkar. Güvensiz tasarım, güvenlik açıklarının temel nedenidir ve saldırganların sistemi istismar etmesine olanak tanır.

Neden Kaynaklanır ?

Yazılım geliştirme sürecinde güvenlik gereksinimlerinin göz ardı edilmesi veya yetersiz planlanması. Geliştiricilerin ve tasarımcıların güvenliği bir öncelik olarak görmemesi.

Sistem veya uygulama geliştirilirken potansiyel tehditlerin ve risklerin doğru bir şekilde değerlendirilmemesi. Bu durum, güvenlik açıklarının belirlenmemesine ve uygun önlemlerin alınmamasına yol açar.

Sistemin karmaşık veya gereksiz derecede geniş bir saldırı yüzeyine sahip olması. Bu durum, saldırganların sistemi istismar etmesi için daha fazla fırsat yaratır.

Nasıl Önlenir ?

Yazılım geliştirme sürecinin başından itibaren güvenliği bir öncelik olarak belirleyin. Güvenlik gereksinimlerini belirleyin ve tüm tasarım kararlarında bu gereksinimleri göz önünde bulundurun.

Sistem veya uygulama geliştirilirken tehdit modelleme ve risk değerlendirmesi yapın. Potansiyel tehditleri ve riskleri belirleyin ve bunlara uygun güvenlik kontrolleri tasarlayın.

A05:2021 – Security Misconfiguration

Security Misconfiguration, bir sistemin, uygulamanın veya ağın güvenlik ayarlarının yanlış yapılandırılması sonucu ortaya çıkan güvenlik açıklarını ifade eder. Bu tür güvenlik açıkları, gereksiz hizmetlerin etkinleştirilmesi, güvenlik yamalarının uygulanmaması, varsayılan ayarların kullanılması gibi durumlar nedeniyle oluşabilir. Yanlış yapılandırmalar, saldırganların sisteminize kolayca erişim sağlamasına, verileri ele geçirmesine veya sistemi istismar etmesine olanak tanır.

Neden Kaynaklanır ?

Yazılım, uygulama veya sistemler varsayılan ayarlarla bırakıldığında, bu ayarlar genellikle güvenli olmayan konfigürasyonlar içerebilir.

İhtiyaç duyulmayan hizmetlerin, modüllerin veya özelliklerin etkinleştirilmiş olması, saldırganların bunları kullanarak sistemi istismar etmesine yol açabilir.

Dosya ve dizinlerin yanlış izinlerle yapılandırılması, yetkisiz kullanıcıların kritik verilere erişmesine izin verebilir.

Şifre politikaları, oturum yönetimi veya loglama gibi güvenlik politikalarının eksik veya yanlış uygulanması.

Nasıl Önlenir ?

Yazılım, uygulama veya sistemler için varsayılan ayarları güvenli hale getirin. Varsayılan kullanıcı adı ve şifreleri değiştirin ve gereksiz özellikleri devre dışı bırakın.

istem ve yazılım bileşenleri için güvenlik yamalarını ve güncellemeleri düzenli olarak kontrol edin ve zamanında uygulayın. Bu, bilinen güvenlik açıklarını kapatmanıza yardımcı olur.

İhtiyaç duyulmayan hizmetleri, modülleri veya özellikleri devre dışı bırakarak saldırı yüzeyini azaltın. Bu, saldırganların potansiyel istismar noktalarını en aza indirir.

Dosya ve dizinler için doğru izinleri yapılandırın. Yalnızca yetkili kullanıcıların kritik verilere erişmesine izin verin ve gereksiz erişim izinlerini kısıtlayın.

Güvenlik politikalarının (örneğin, güçlü şifre politikaları, oturum yönetimi, loglama) doğru ve eksiksiz bir şekilde uygulanmasını sağlayın.

A06:2021 – Vulnerable and Outdated Components

Vulnerable and Outdated Components (Güvenlik Açığı Bulunan ve Güncel Olmayan Bileşenler), bir yazılım veya sistemde kullanılan üçüncü taraf kütüphanelerin, framework'lerin, modüllerin veya diğer bileşenlerin eski veya güvenlik açıkları barındıran sürümlerini ifade eder. Bu tür bileşenler, sistemin güvenliğini tehlikeye atabilir ve saldırganların bilinen zafiyetlerden faydalanmasına neden olabilir.

Neden Kaynaklanır ?

Üçüncü taraf bileşenlerin güvenlik açıklarını kapatan güncellemelerin veya yamaların zamanında uygulanmaması. Bu, bilinen açıkların sistemde var olmaya devam etmesine neden olur.

Uygulamada kullanılan bağımlılıkların (dependency) ve kütüphanelerin versiyonlarının doğru bir şekilde yönetilmemesi. Eski veya güvenlik açıkları bulunan sürümlerin fark edilmeden kullanılmaya devam etmesi.

Uygulamada gereksiz bileşenlerin ve kütüphanelerin kullanılması. Gereksiz bileşenler, ek güvenlik riskleri oluşturabilir ve saldırı yüzeyini artırabilir.

Nasıl Önlenir ?

Kullandığınız tüm üçüncü taraf bileşenleri ve bağımlılıkları düzenli olarak güncelleyin. Güvenlik yamalarının ve yeni sürümlerin yayınlanıp yayınlanmadığını kontrol edin ve bunları zamanında uygulayın.

Bir yama yönetim sistemi uygulayarak, kullanılan bileşenlerdeki güvenlik açıklarını hızla tespit edin ve yamalayın. Otomatik araçlar, bu süreci hızlandırabilir ve insan hatasını azaltabilir.

İhtiyaç duyulmayan hizmetleri, modülleri veya özellikleri devre dışı bırakarak saldırı yüzeyini azaltın. Bu, saldırganların potansiyel istismar noktalarını en aza indirir.

Common Vulnerabilities and Exposures (CVE) gibi güvenlik açığı veritabanlarını düzenli olarak takip edin. Kullanılan bileşenlerle ilgili bilinen güvenlik açıklarını izleyin ve gerektiğinde güncellemeler yapın.

Üçüncü taraf bileşenlerin ve bağımlılıkların güvenlik testlerini yapın. Bu, bileşenlerin sisteme entegre edilmeden önce güvenli olduğunu doğrulamanıza yardımcı olur.

A07:2021 – Identification and Authentication Failures

Identification and Authentication Failures (Kimlik Tespiti ve Kimlik Doğrulama Hataları), bir sistemin veya uygulamanın kullanıcı kimliklerini doğrulama ve yetkilendirme süreçlerinde meydana gelen güvenlik açıklarını ifade eder. Bu tür hatalar, saldırganların yetkisiz erişim sağlamasına, kullanıcı hesaplarına izinsiz giriş yapmasına veya sistemdeki hassas verilere erişmesine neden olabilir. Kimlik tespiti ve kimlik doğrulama, kullanıcıların doğru bir şekilde tanımlanması ve sistem kaynaklarına erişimlerinin yönetilmesi açısından kritik öneme sahiptir.

Neden Kaynaklanır ?

Kullanıcıların şifrelerinin zayıf veya tahmin edilebilir olması. Örneğin, kısa, basit veya kolay tahmin edilebilen şifrelerin kullanılması.

Şifrelerin güvenli olmayan şekilde (örneğin, düz metin olarak) saklanması. Bu, veri ihlalleri durumunda şifrelerin kolayca ele geçirilmesine neden olabilir.

Tek faktörlü kimlik doğrulama (örneğin, sadece şifre) yerine, daha güvenli yöntemlerin kullanılmaması. Çift faktörlü kimlik doğrulama (2FA) gibi ek güvenlik katmanlarının eksikliği.

Oturum yönetiminde güvenlik açıkları. Örneğin, oturum kimlik doğrulama jetonlarının uygun şekilde saklanmaması veya oturumların süresiz olarak açık kalması.

Nasıl Önlenir ?

Kullanıcıların güçlü ve karmaşık şifreler oluşturmasını zorunlu kılın. Şifrelerin minimum uzunluk, özel karakterler ve sayı içermesi gibi gereksinimleri belirleyin.

Şifreleri güvenli bir şekilde saklamak için güçlü hash algoritmaları (örneğin, bcrypt, Argon2) kullanın. Şifreleri düz metin olarak saklamaktan kaçının.

Kullanıcıların kimliklerini doğrulamak için çift faktörlü kimlik doğrulama gibi ek güvenlik katmanları ekleyin. Bu, şifrenin çalınması durumunda ek bir koruma sağlar.

Güvenlik sorularının ve yanıtlarının güçlü ve tahmin edilmesi zor olmasını sağlayın. Ayrıca, güvenlik sorularının alternatif kimlik doğrulama yöntemleri ile desteklenmesini düşünün.

A08:2021 – Software and Data Integrity Failures

Software and Data Integrity Failures (Yazılım ve Veri Bütünlüğü Hataları), bir yazılımın veya verinin yetkisiz kişiler tarafından değiştirilmesi, bozulması ya da sahte veri veya yazılımın kullanılması sonucu ortaya çıkan güvenlik açıklarını ifade eder. Bu tür hatalar, özellikle yazılım güncellemelerinin, kritik verilerin veya CI/CD (Sürekli Entegrasyon ve Sürekli Dağıtım) süreçlerinin güvenilir olmayan kaynaklardan alınması ya da yeterli bütünlük kontrollerinin yapılmaması durumunda meydana gelir.

Neden Kaynaklanır ?

Yazılım güncellemelerinin güvenli olmayan kaynaklardan indirilmesi veya dijital imzaların doğrulanmaması. Bu durum, kötü niyetli yazılımların sisteme entegre olmasına neden olabilir.

Sürekli entegrasyon ve sürekli dağıtım (CI/CD) süreçlerinin güvenliksiz bir şekilde yönetilmesi. Bu, yazılımın dağıtım aşamalarında kötü amaçlı kod eklenmesine izin verebilir.

Üçüncü taraf kütüphanelerin veya yazılımların güvenilir kaynaklardan alınmaması veya bu bileşenlerin zafiyetler barındıran eski sürümlerinin kullanılması.

Nasıl Önlenir ?

Yazılım güncellemelerini yalnızca güvenilir kaynaklardan indirin ve dijital imzaların doğruluğunu kontrol edin. Bu, sahte yazılımların kullanılmasını önler.

Yazılım ve veri transferlerinde kriptografik hash algoritmaları kullanarak bütünlük kontrolleri yapın. Bu, verilerin veya yazılımların yetkisiz kişiler tarafından değiştirilip değiştirilmediğini kontrol etmenizi sağlar.

Üçüncü taraf yazılım bileşenlerinin güvenilir kaynaklardan geldiğinden emin olun. Bileşenlerin düzenli olarak güncellenmesini ve güvenlik testlerinin yapılmasını sağlayın.

A09:2021 – Security Logging and Monitoring Failures

Security Logging and Monitoring Failures (Güvenlik Kayıt ve İzleme Hataları), bir sistemin güvenlik olaylarını yeterince kaydedememesi veya izleyememesi sonucu ortaya çıkan güvenlik açıklarını ifade eder. Bu durum, saldırıların tespit edilememesine, güvenlik ihlallerinin zamanında fark edilememesine ve tehditlere karşı geç müdahale edilmesine yol açabilir. Güvenlik kayıtları ve izleme mekanizmaları, olayların incelenmesi, saldırıların tespiti ve önlenmesi için kritik öneme sahiptir.

Neden Kaynaklanır ?

Kritik olayların veya sistem etkinliklerinin yeterince kaydedilmemesi ya da logların belirli bir süre saklanmaması. Önemli güvenlik olaylarının loglanmaması, bu olayların analiz edilmesini imkânsız hale getirir.

Logların düzenli olarak analiz edilmemesi ya da izleme araçlarının kullanılmaması. Bu durum, potansiyel güvenlik tehditlerinin gözden kaçmasına neden olabilir.

Logların güvenli bir şekilde saklanmaması veya yetkisiz kişiler tarafından erişilebilir olması. Bu, logların manipüle edilmesine veya silinmesine neden olabilir.

Nasıl Önlenir ?

Kritik sistem olaylarının, erişimlerin, hata mesajlarının ve güvenlikle ilgili diğer aktivitelerin detaylı bir şekilde loglandığından emin olun. Logların belirli bir süre boyunca güvenli bir şekilde saklanmasını sağlayın.

Logları düzenli olarak gözden geçirmek ve analiz etmek için güvenlik analizi araçları ve süreçleri oluşturun. Anormal aktiviteleri ve potansiyel güvenlik tehditlerini tespit edebilecek araçlar kullanın.

Güvenlik olaylarını gerçek zamanlı olarak izleyebilecek ve kritik olaylar meydana geldiğinde otomatik olarak alarm verecek izleme sistemleri kurun. Bu, olaylara hızlı bir şekilde tepki vermenizi sağlar.

Logların yetkisiz erişimden korunmasını sağlamak için uygun erişim kontrol mekanizmalarını ve şifreleme yöntemlerini kullanın. Logların değiştirilemez veya silinemez şekilde saklanmasını sağlayın.

A10:2021 – Server-Side Request Forgery (SSRF)

Server-Side Request Forgery (SSRF), bir saldırganın bir sunucuyu, normalde erişememesi gereken kaynaklara istek göndermeye zorladığı bir güvenlik açığıdır. SSRF saldırıları, genellikle sunucunun bir URL veya kaynağa istek göndermesine izin veren işlevler üzerinden gerçekleştirilir. Saldırganlar, bu güvenlik açığını kullanarak dahili sistemlere, hizmetlere veya ağlara erişebilir ve hassas bilgileri ele geçirebilir.

Neden Kaynaklanır ?

Uygulamanın kullanıcıdan aldığı URL veya diğer dış kaynak taleplerini güvenli bir şekilde doğrulamadan işlemeye alması. Bu, saldırganların kötü niyetli URL'leri sunucuya göndermesine olanak tanır.

Uygulamanın sunucunun iç ağına erişimi olması durumunda, saldırganlar bu iç ağ kaynaklarına SSRF yoluyla erişebilir. Bu, özellikle sunucunun internete doğrudan bağlı olmadığı durumlarda daha risklidir.

Uygulamanın harici kaynaklara istek göndermesi gereken durumlarda, bu isteklerin güvenli bir şekilde yapılmaması. Saldırganlar, bu istekleri manipüle ederek sunucunun istemediği kaynaklara erişmesini sağlayabilir.

Nasıl Önlenir ?

Kullanıcı tarafından sağlanan URL veya diğer kaynak girdilerini sıkı bir şekilde doğrulayın ve filtreleyin. Bu girdilerin sunucu tarafından doğrudan işlenmesine izin vermeyin.

Sunucunun yalnızca belirli ve güvenilir dış kaynaklara istek göndermesine izin verin. Gerekirse, bir beyaz liste uygulayın ve iç ağdaki kaynaklara doğrudan erişimi engelleyin.

Sunucunun internete ve iç ağa erişimini sıkı bir şekilde kontrol eden güvenlik duvarları ve ağ izolasyonu uygulayın. Bu, sunucunun yalnızca güvenli ve belirli bir şekilde iletişim kurmasını sağlar.

Harici kaynaklara istek gönderirken bir proxy sunucusu kullanarak, istekleri güvenli bir şekilde yönlendirin ve doğrudan erişimi engelleyin. Proxy sunucusu, potansiyel kötü niyetli istekleri filtreleyebilir.