OWASP TOP TEN

A01:2021 – Broken Access Control

Nedir?

Web uygulamalarında her kullanıcı için belirlenmiş yetkiler vardır. Admin ile sıradan bir kullanıcının web uygulaması üzerindeki yetkileri aynı değildir. Bu zafiyetin bulunduğu durumlarda saldırgan kullanıcılar için belirlenen izinleri yok sayarak yetki sınırları dışında hareket edebilir. Erişimi olmayan bilgileri görüntüleme, düzenleme ya da silme gibi işlevleri yerine getirebilir.

Neden Kaynaklanır?

Erişim kontrol mekanizmasının yanlış ya da eksik yapılandırılmasından kaynaklanır. Örneğin; kullanıcıların belirli işlemlerden önce doğrulanması veya yetkilendirilmesi gerekir. Yetkiler kısıtlandırılmadıysa her kullanıcının erişebilmesi zafiyete yol açacaktır. Yetki yükseltme ya da URL manipülasyonlarına açık olması da nedenlerden bazılarıdır.

Nasıl Önlenir?

Erişim kontrol mekanizmaları yalnızca sunucu tarafı kodlarında ya da sunucusuz API'larda tutulmalı bu sayede erişim kontrol denetimlerinin değiştirilmesi engellenmelidir. Tüm erişim istekleri varsayılan seçenek olarak reddedilmelidir. Web köklerinde yedek dosyaların yer almadığından emin olunmalıdır.

A02:2021 – Cryptographic Failures

Nedir?

Verilerin aktarım ve depolanma süreçlerinde belirli güvenlik önlemleri alınmaktadır. Bu önlemlerden biri şifreleme yöntemidir. Kredi kartı bilgileri, parolalar, oturum hesapları gibi kritik düzeyde olan bilgiler şifreleme algoritmaları kullanılarak belirli formatlarda saklanır. Bu formatta tutulmasının sebebi olası bir veri hırsızlığında dahi şifrelenmiş bilgilerin anlaşılamamasını sağlamaktır. Bu şifreleme algoritmalarının çözümlenebilmesi ya da hiç var olmaması Cryptographic Failures zafiyetini doğurur.

Neden Kaynaklanır?

Şifreleme için kullanılan key'lerin zayıf olması, ya da şifrelemenin hiç yapılmaması bu zafiyeti doğurabilir. Bunun yanı sıra kullandığımız protokollerin güvenli olmaması, örneğin HTTP başlığının eksik olması gibi durumlarda da bu açık görülebilir. Güncel olmayan şifreleme yöntemleri ve dijital sertifikasyonun düzgün doğrulanmaması gibi çeşitli başka nedenleri de vardır.

Türleri Nelerdir?

- Sensitive Data Exposure

Nedir?

Hassas verilerle ilgili olan bu zafiyet her ne kadar cryptographic failures ile benzer güvenlik konularını ele alsa da farklı olan yönleri söz konusudur. Bu zafiyet, şifrelenmiş verilerden ziyade şifrelenmemiş verilerle ilgilidir. Daha geniş bir kapsamı vardır ancak aynı şekilde saldırganın hassas verileri kullanması ya da değiştirmesiyle bağlantılıdır.

Neden Kaynaklanır?

Yetkisiz erişim, güvenliksiz erişim kontrolleri, şifrelenmemiş veriler gibi nedenlerden kaynaklanabilir. Bilgi aktarımı yapılırken şifreleme algoritmalarının doğru kullanılmaması durumunda paket iletiminde sniffer ile şifrelenmemiş olan bu bilgilere ulaşılabilir. Clear text formatında, default şifreleme algoritması ya da önceden açığa çıkmış şifrelemeler kullanılıyorsa saldırının gerçekleşme olasılığı yüksektir.

Nasıl Önlenir?

HTTP, FTP, SMTP gibi güvensiz protokoller yerine HTTPS, FTPS, SMTPS gibi şifreli protokoller tercih edilmeli, kullanılmayan veriler tutulmadan silinmelidir. Şifrelemeler default seçeneklerde tutulmamalı ve daha öncesinde veri sızıntısında ortaya çıkmış şifreleme algoritmaları tekrar kullanılmamalıdır. Daima güncel tutulmalı, veri transferlerinde ve tutulan kayıtlarda şifreleme yapılmalıdır.

Nasıl Önlenir?

Kullanılmayacak hassas bilgiler tutulmadan silinmelidir. Kullanılan şifreleme yöntemlerinin güncel olduğundan emin olunmalıdır. Verileri taşımak için FTP ve SMPT gibi güvenliği olmayan protokollerin kullanımından uzak durulmalıdır. Kimlik doğrulamalı ve kolay tahmin edilemeyecek şifreleme yöntemlerinin kullanılması sağlanmalıdır.

A03:2021- Injection

Nedir?

Web sunucusunda kullanıcıların istenmeyen sistem komutlarını çalıştırılabilmesi ya da kendi dosyalarını farklı formatlarda siteye enjekte edebilmeleri sonucu ortaya çıkar. Bu zafiyet sonucunda kullanıcı sisteme sınırsız erişim elde edebilir ve hassas bilgilerden oturum yönetimine kadar pek çok alanı ele geçirebilir.

Neden Kaynaklanır?

Kullanıcıdan alınan verilerin doğru bir şekilde filtrelenmemesi ya da kontrol edilmemesi sonucunda ortaya çıkabilir. Saldırgan bu verileri zararlı komutlar ya da sorgular şeklinde enjekte ettiğinde sistem erişimi sağlayabilir. Aynı nedenden dolayı girdiler kullanıcıdan doğrudan alınmamalıdır. Bazı arama parametrelerinin filtrelenmemesi, yazılımın zayıf yapılandırılması da bu zafiyete yol açabilecek nedenlerdendir.

Türleri Nelerdir?

1) **SQL Injection:** Saldırganın veri tabanına erişimi olmamasına rağmen çalıştırdığı SQL sorguları ile veri tabanındaki kritik verilere ulaşması, değiştirmesi ya da veri çalmasından kaynaklanır.
2) **NoSQL Injection:** NoSQL veri tabanına yetkisiz erişim sağlanmasından kaynaklanır. Hassas verilerin çalınması ya da değiştirilmesi söz konusudur.
3) **OS Command Injection:** Saldırganın yetkisiz işlemler yapabilmesi adına sistem komutlarını çalıştırabilmesi sonucu ortaya çıkan bir zafiyettir. Bu komutlar ile sistem bilgilerini sızdırabilir ve hassas dosyaları görüntüleyebilir.
4) **LDAP Injection:** Dosya dizinleri arasında yetkisiz bir şekilde dolaşım sağlanabilmesi sonucunda oluşur. Dizin üzerinden dosyaların görüntülenmesi ve manipülasyonu söz konusudur.
5) **XML Injection:** XML dosyalarının ya da komutlarının sistem üzerinde çalıştırılarak çeşitli yetkiler elde edilebilmesi nedeniyle oluşur.

Nasıl Önlenir?

Kullanıcıdan alınan girdilerin kontrolü dinamik olarak sağlanmalıdır. Olası komutlara karşı filtrelemelerin doğru yapılandırılması gerekmektedir. Veri tablolarında isimlendirme

konularında dikkatli olunmalı, yüklenecek olan dosya formatları doğru belirlenmelidir. Bunun yanı sıra güvenli API'lerin kullanılması da önerilmektedir.

A04:2021-Insecure Design

Nedir?

Kısaca "Eksik veya etkisiz kontrol tasarımı." olarak ifade edilir. Güvensiz uygulama yazılımından ayrı bir kategoride değerlendirmemiz gerekmektedir. Güvenli tasarımın olduğu noktalarda ise başka güvenlik açıkları bulunabilir ancak güvensiz bir tasarım sonucunda sonradan alınan önlemlerin bir önemi olmayacaktır.

Neden Kaynaklanır?

Tehdit modellemesine dikkat edilmemesinden ve web uygulaması geliştirme aşamasında tasarım kısmına yeterli zaman ile önemin verilmemesinden kaynaklanır. Hızlı bir şekilde projenin bitirilmesi hedeflendiğinde tasarım güvenliği bölümünün önemsenmemesi de bu zafiyete yol açabilir.

Nasıl Önlenir?

Bu işin uzmanlarıyla beraber güvenli bir gelişim yaşam döngüsü oluşturulmalı, bu tehdit modelleme yöntemlerine sadık kalınmalıdır. Uygulamanın her katmanında çeşitli senaryolar göz önünde bulundurulmalı ve buna uygun şekilde önlemler alınmalıdır. Kimlik doğrulama, erişim kontrolü, anahtar akışlar için tehdit modellemesi yapılmalıdır.

A05:2021- Security Misconfiguration

Nedir?

Web uygulamalarında güvenlik kontrollerinin ya da bulut hizmetlerinin yanlış yapılandırılması nedeniyle ortaya çıkan zafiyettir. Saldırgan bu açıkları kullanarak sunucunun kontrolünü ele geçirebilir.

Neden Kaynaklanır?

Genel anlamda hatalı yapılandırmadan kaynaklanır. Sistemin ya da yazılımın güncel olmaması, güvenlik önlemlerinin devre dışı bırakılması da bu zafiyete yol açabilir. Aşırı derecede bilgilendirici hata mesajları da bu açıklığa neden olan durumlardandır.

Nasıl Önlenir?

Yazılım ve sistemin güncel olmasına dikkat edilmeli, zafiyetli kodlamalardan uzak durulmalıdır. Sisteme gereksiz özellikler yüklenmemeli ya da etkinleştirilmemelidir. Kullanıcı uygulama içerisinde gerektiği kadar bilgilendirilmelidir.

A06:2021- Vulnerable and Outdated Components

Nedir?

Yazılımın güncel olmayan ya da zafiyetli halinin kullanılmasından kaynaklı ortaya çıkan bir zafiyettir. Hem istemci hem de sunucu tarafından kullanılabilir, yani zafiyet yalnızca sunucu tarafındaki bileşenlerle sınırlı değil aynı zamanda istemci tarafını da kapsamaktadır. Saldırgan bu açıkları kullanarak sisteme yetkisiz erişim sağlayabilir, verileri çalabilir ya da sistem kontrolünü ele geçirebilir.

Neden Kaynaklanır?

Yazılım, iletim sistemi, database gibi kullanılan sistemlerin, güncel olmayan ya da zafiyetli versiyonlarının kullanılmasından kaynaklanır. Düzenli zafiyet taramalarının yapılmaması, güncellense bile bileşenlerin kendi aralarında uyumluluğunun test edilmemesi başlı başına nedenlerdendir.

Nasıl Önlenir?

Kullanılmayan özellikler ve bağlamlar kaldırılmalıdır. Olası güvenlik açıklarını gözlemleyebilmek için sürekli olarak CVE (Common Vulnerability and Exposures) ve NVD (National Vulnerability Database) gibi kaynaklar incelenmeli, açıkların yakalandığı durumlarda geri dönüş alınması sağlanmalıdır. Sistem güncel tutulmalı ve yamaları yapılmalıdır.

A07:2021 – Identification and Authentication Failures

Nedir?

Bir web uygulamasındaki kullanıcı kimlik doğrulama ya da oturum açma süreçlerinde ortaya çıkan açıkları içeren zafiyettir. İzinsiz erişim ya da oturum bilgilerinin çalınması gibi durumlara neden olabilir.

Neden Kaynaklanır?

Zayıf kimlik doğrulama mekanizmaları, örneğin kolay tahmin edilebilir parolalara izin verilmesi ya da varsayılan parolaların değiştirilmemesi gibi nedenlerden kaynaklanabilir. Otomatik saldırıların engellenmemesi, URL'de oturum tanımlayıcılarının görünmesi, zayıf kimlik doğrulama süreçleri de bu açıklığa neden olan durumlardandır.

Nasıl Önlenir?

Zayıf şifreleme yöntemlerinden uzak durulmalı, kullanıcıların seçecekleri parolaların güçlü olmasına dikkat edilmelidir. Çok faktörlü kimlik doğrulama sistemleri kullanılmalıdır. Özellikle admin kullanıcıları için varsayılan oturum bilgilerinin kullanılmasından kaçınılmalıdır. Olası saldırılara karşı parola deneme sayıları düşük tutulmalı, güvenli oturum yönetimine dikkat edilmelidir.

A08:2021 – Software and Data Integrity Failures

Nedir?

Yazılım ve veri bütünlüğünde oluşan hatalardan kaynaklanan bir zafiyettir. Saldırgan bu zafiyeti kullanarak sistemi istismar edebilir ya da manipüle edebilir.

Neden Kaynaklanır?

Kod ve altyapının, veri ve yazılım bütünlüğünü koruyamadığı durumlarda ortaya çıkar. Güvenlik kontrolleri eksik bir şekilde yapılmış olabilir. Doğrulanmamış kaynaklardan gelen bileşenler kullanıldığında zafiyetli sistemler oluşabilir. Otomatik yazılım güncellemeleri kontrol edilmeden yapıldığında kötü amaçlı yazılımlar sisteme enjekte edilebilir ve manipülasyona açık hale gelir.

Nasıl Önlenir?

Yazılım bileşenleri sadece onaylı, güvenilir kaynaklar tarafından temin edilmelidir. Yetkisiz erişim ve kötü amaçlı güncellemelere karşı önlem alınmalıdır. Güncellemelerin güvenilir olduğundan emin olunmalıdır.

A09:2021 – Security Logging and Monitoring Failures

Nedir?

Güvenlik loglama ve izleme süreci sistemde oluşan olağandışı hareketlerin gözlemlenmesi için kullanılan bir yöntemdir. Bu zafiyet sistemde meydana gelen kötü amaçlı aktivitelerin önceden belirlenememesi ve bunun sonucunda zamanında önlem alınamaması durumunda ortaya çıkar.

Neden Kaynaklanır?

Logların izlenmemesi ya da izlenen log mesajlarının yeterince ayrıntılı ve anlaşılır olmaması sonucunda bu açık ortaya çıkar. Penetrasyon testi ya da güvenlik araçlarının saldırı tespit edememesi ya da bilgi sızıntıları da bu açığa neden olacak durumlardandır.

Nasıl Önlenir?

Logların düzenli olarak incelenmesi ve yapılan loglama işlemlerinin ayrıntılı bir şekilde yapılması bu zafiyetin önüne geçmeyi kolaylaştırır. Daha hızlı aksiyon almak için otomatik uyarı mekanizmaları oluşturulmalıdır.

A10:2021- Server-Side Request Forgery

Nedir?

Kullanıcıdan gelen URL ya da başka bir kaynağın doğrulanmadan istek göndermesi sonucunda ortaya çıkar. Bu açık nedeniyle saldırganlar başka bir sunucuya ya da dahili bir birime istek gönderebilir.

Neden Kaynaklanır?

Kullanıcı girdilerinin kontrol edilmemesi, URL'lerin herhangi bir kontrol ya da kısıtlama olmadan kullanılması bu açığa neden olabilir. Bunun yanı sıra karmaşık ağ yapıları birden fazla sunucuya istek gönderilebilmesi nedeniyle saldırılara açık olabilir.

Nasıl Önlenir?

Hem ağ katmanında hem de uygulama katmanında alınacak önlemler vardır. Ağ katmanında veri akışını kontrol ederek güvenlik duvarı politikaları uygulanmalı aynı zamanda uzaktan kaynak erişim işlemlerini kısıtlanmalıdır. Uygulama katmanında ise

kullanıcıdan gelen URL’ler dikkatle incelenmeli ve temizlenmelidir. Yalnızca belli şemalar ve portların kullanımına izin verilmeli, URL tutarlılığına dikkat edilmelidir.