Yavuzlar Web Güvenliği & Yazılım Takımı Task-1 Raporu

Görev Tarihi: 18.08.2024

Son Teslim Tarihi: 25.08.2024 17:00

Görev Açıklaması:

Takım üyelerinin OWASP Top 10 hakkında genel bir araştırma yapmaları ve her bir kategori hakkında kısa açıklamalar içeren bir rapor hazırlamaları istenmektedir.

Hazırlayan: Sabri KÜÇÜK

A. Zafiyet Nedir , Neden Kaynaklanır , Varsa Türeleri Nelerdir?

1) **A01:2021-Broken Access Control :** Web uygulamaları ve yazılım sistemlerinde kullanıcıların yetkisiz veya beklenmeyen şekilde kaynaklara erişmelerine olanak tanıyan bir güvenlik açığıdır. Bu, normalde kullanıcıların erişmemesi yada görmemesi gerektiği yerleri erişmesi ya da görmesi üzerine oluşan bir açık türüdür. Örneğin, bir kullanıcı diğer kullanıcıların verilerine erişebiliyor veya yönetici yetkilerine sahip olmadığı halde admin yetkisine sahip ayarları değiştirebiliyorsa, bu Broken Access Control zafiyeti anlamına gelir.

   Türleri :

   -IDOR, yetkisiz kullanıcıların doğrudan nesne referanslarını manipüle ederek başka kullanıcıların verilerine erişmesine olanak tanıyan bir güvenlik açığıdır.

2) **A02:2021-Cryptographic Failures :** Sistemlerde bazı veriler düz metin olarak tutuluyorken bazıları ise şifreli metin halinde tutuluyor veya iletiliyor. Bunu yapmaktaki amaç verilerin çalınması durumunda dahi şifreleme algoritmasından dolayı verilerin gerçekte ne anlam ifade ettiğinin öğrenilmesini önlemek. Ancak her zaman bu durum tahmin edildiği gibi olmuyor. Bazı durumlarda şifrelenip tutulması veya iletilmesi gereken verilerin düz metin olarak tutulması bazen de şifrelenip tutulduğu halde zayıf şifreleme algoritmalarının kullanılması veya keylerin güvenliğinin sağlanamaması gibi durumlarda bu zafiyet ortaya çıkabiliyor.

3) **A03:2021 – Injection** : Kötü niyetle hareket eden bireylerin hazırladığı kod blokları(kötü amaçlı script) web sayfalarına enjekte edilerek bu sayfaya tıklanılmasıyla scriptin çalıştırılması amaçlanır. Eğer web uygulamaları kullanıcı girdilerini herhangi bir şekilde ayıklamıyor ya da yasaklı sembol listesi kullanmıyorsa saldırının başarısı kaçınılmaz olur.Bu zafiyette tarayıcımız web uygulamasının güvenli olduğunu düşündüğü için scriptleri çalıştırır ve script her neyi amaçlıyorsa onu yapmak için işe koyulur. Bu bazen bir clickjacking bazense keylogging olabilir. Kullanıcının cihazından çerezleri çalınabilir veya zararlı sitelere yönlenmesini sağlayacak bağlantılar yollanabilir.

   Türleri :

   - DOM XSS sayfa içindeki nesnelerin özeliklerini değiştirmeye yönelik çalışır.

   - Reflected XSS scriptler doğrudan HTTP isteğine enjekte edilir. Buna bağlı

olarak bu istek kullanıcının tarayıcısına yansır.
- Stored XSS scriptler sunucu tarafında depolanır. Bundan dolayı bu script in görüntülendiği bir sayfa yüklendiğinde çalışır. Genellikle yorumlar sayfasında karşımıza çıkar.

4) **A04:2021-Insecure Design :** Geliştiricilerin güvenli olmayan tasarımları ve kod bloklarını kullanmaları sonucunda ortaya çıkan ve bu aşamadan sonra alınan güvenlik önlemlerinin işe yaramadığı bir zafiyet türüdür.

   Bazı durumlarda tasarımın güvenliği pek önemsenmez ve bunun sonucunda Insecure Design zafiyeti ortaya çıkar. Tasarım aşaması ilk adımlardan biri olduğu için bu aşamada önlem alınmadığı takdirde web sitesinin tamamı güvensiz hale gelir. Sonraki aşamalarda alınan güvenlik önlemleriyle Insecure Design zafiyeti kapatılamaz. Yalnızca kaynak kodu değiştirerek sorunu kökten çözülebilir.

5) **A05:2021-Security Misconfiguration :** Sunucu veya web uygulamalarında güvenlik kontrollerinin uygulanmak istendiği halde yanlış yapılandırmalardan dolayı bu kontrollerin uygulanamaması sonucunda ortaya çıkar.

6) **A06:2021-Vulnerable and Outdated Components :** Artık yeni sürümü çıkmayan, eski veya zafiyetli bileşenlerin kullanılmasıyla ortaya çıkar. Saldırganlar zafiyeti sömürmek için sistemlerde kullanılan bileşenleri ve bunların sürümlerini öğrenmeye çalışır. Sömürü işlemi genel olarak basit bir şekilde işler.

7) **A07:2021-Identification and Authentication Failures :** Oturum ve kimlik bilgisi yönetiminin düzgünce yapılmadığı durumlarda bu zafiyet oluşur. Zafiyetin sömürülmesi sırasında, saldırganlar kullanıcıların bilgilerini taklit etmeye çalışır ve eğer ki eksik veya yanlış yapılandırılmış denetimlere takılmazlar ise bir başkası adına kimliklerini tanımlar veya doğrularlar.

8) **A08:2021-Software and Data Integrity Failures :** Kullanılan altyapı ve kodlar bütünlük ihlallerine karşı koruma sağlanamadığında yazılım ve veri bütünlüğü hataları ortaya çıkar. Güvenilmeyen ya da yapımcısı bilinmeyen kaynaklardan alınan kodların kullanımı, zafiyetin oluşmasının önünü açan etkenlerden biridir. CI ve CD sistemindeki, yani yazılım geliştirmede mevcut tüm aşamaların otomatikleştirildiği sistemlerdeki zayıflıklar, zafiyetle doğrudan bağlantılıdır.

9) **A09:2021-Security Logging and Monitoring Failures :** Saldırganlar bilgi elde etme sürecinde pasif veya aktif yöntemler kullanır. Aktif yöntemlerde kurbanla etkileşime geçmek zorunda olduğundan kurban sistemde hareketleri Loglanabilir. Hedef sistemde loglama ve monitoring işlemleri düzgünce yapıldığında saldırganların bilgi toplama süreçleri veya saldırılar erken evrede fark edilip

önlenebilirler. Bunun için gerekli her şeyin loglandığından ve monitoring işlemlerinin yapıldığından emin olunmalıdır.

10) **A10:2021-Server-Side Request Forgery :** Zafiyetli bir web uygulaması üzerinden manipüle edilen istekler yollanan sunucuyu kendisi adına istek yapmaya zorlar. Böylece saldırgan sunucu üzerinden normalde erişemeyeceği verilere erişebilir. Bu istekler üzerinde yapılan oynamalarla isteklerin varış noktası değiştirilebilir. İç ağa sızılabilir ve burada pek çok saldırı gerçekleştirilebilir. SSRF, web sunucusunun uzak kaynakları çağırmasına izin verilen domain veya protokolleri denetlenmediğinde ortaya çıkar.

B. Nasıl Önlenir?

1) **A01:2021-Broken Access Control :** Kullanıcıların yalnızca yetkili oldukları kaynaklara erişebilmelerini sağlamak için erişim kontrolleri dikkatlice yapılandırılmalıdır. Bu, kullanıcı rollerine ve izinlerine göre erişim seviyelerinin belirlenmesini içerir. Kullanıcıların kimlik doğrulamasını güvenli bir şekilde sağlamak için token tabanlı kimlik doğrulama yöntemleri kullanılmalıdır. Bu yöntem, her oturum için benzersiz bir token oluşturur ve bu token, kullanıcının kimliğini doğrulamak için kullanılır. Sistem üzerindeki tüm işlemler ve erişim denemeleri kaydedilmelidir. Bu loglar, olası güvenlik ihlallerini tespit etmek ve analiz etmek için kullanılabilir. Ayrıca, düzenli olarak gözden geçirilerek anormal aktiviteler tespit edilebilir.

2) **A02:2021-Cryptographic Failures :** Saklanan veriler hassasiyet derecelerine göre sınıflandırılmalıdır. Bu, hangi verilerin daha fazla korunması gerektiğini belirlemeye yardımcı olur. Gereksiz hassas veriler saklanmamalıdır. Bu, veri ihlali durumunda riskin azaltılmasına yardımcı olur. Hassas veriler, yetkisiz erişimlere karşı korunmak için şifrelenmelidir. Bu, verilerin güvenliğini artırır. Verilerin şifrelenmesi için güçlü ve güncel şifreleme algoritmaları kullanılmalıdır. Bu, şifreleme yöntemlerinin kırılmasını zorlaştırır. Şifreleme anahtarları güvenli bir şekilde saklanmalı ve yönetilmelidir. Anahtarların güvenliği, şifreleme sisteminin bütünlüğünü korur.

3) **A03:2021-Injection :** Web uygulamalarındaki tüm varlıklar düzenli olarak denetlenmelidir. Bu, potansiyel güvenlik açıklarının tespit edilmesine ve giderilmesine yardımcı olur. Kullanıcıların girdiği veriler taranmalı ve doğrulanmalıdır. Bu, zararlı verilerin sisteme girmesini engeller. Formlar aracılığıyla alınan veriler sınırlandırılmalı ve doğrulanmalıdır. Bu, saldırganların zararlı veriler göndermesini zorlaştırır. Kullanıcı oturumlarını yönetmek için güvenli tanımlama bilgileri (cookies) kullanılmalıdır. Bu, oturum güvenliğini artırır ve yetkisiz erişimleri engeller.

4) A04:2021-Insecure Design : Güvenli tasarım modelleri ve hazır çözümlerden oluşan bir kütüphane oluşturulabilir ve bu kütüphane kullanılarak güvenli tasarımlar yapılabilir. Tehdit modelleme yapılabilir. Uygulamanın potansiyel tehditlerini belirlemek ve bu tehditlere karşı önlemler almak için tehdit modelleme yapılabilir. Uygulamanın her katmanına uygunluk kontrolleri entegre edilerek güvenlik standartlarına uyum sağlanabilir. Kritik akışların tehdit modeline karşı dirençli olduğunu doğrulamak için birim ve entegrasyon testleri yazılabilir ve uygulanabilir. Sistem ve ağ katmanları birbirinden ayrılarak

güvenlik artırılabilir. Tüm katmanlardaki gruplar birbirinden ayrılabilir. Uygulamanın tüm katmanlarındaki gruplar birbirinden ayrılarak güvenlik artırılabilir. Kullanıcı veya hizmetlerin kaynak tüketimi ihtiyaç duyulan seviyede sınırlandırılarak sistem performansı ve güvenliği artırılabilir.

5) **A05:2021-Security Misconfiguration :** Varsayılan ayarlara güvenmek yerine, sistemin ihtiyaçları ve performans gereksinimleri dikkate alınarak en güvenli yapılandırmalar yapılmalıdır. Sistemler her zaman güncel tutulmalıdır. Sistem yöneticileri uzman kişilerden seçilmeli ve bilinçli hareket etmelidir. Kullanılmayan dosya ve dizinler sistemden kaldırılmalıdır. Beyaz liste oluşturulmalıdır. Tüm kullanıcılar yetkilerine göre gruplandırılmalı ve buna uygun erişim izinlerine sahip olmalıdır.

6) **A06:2021-Vulnerable and Outdated Components :** Kullanılmayan tüm bileşenler sistemden kaldırılmalıdır. Yama veya güncellemesi olan yazılımlar vakit kaybetmeden güncellenmelidir. Kullanılan bileşenler resmi kaynaklardan temin edilmelidir. Bileşenlerin sürüm ve bağımlılıklarının envanteri tutulmalıdır. Düzenli olarak zafiyet taraması yapılmalıdır. CVE takip edilerek buradaki açıkları barındıran bileşenlerin sistemde mevcut olup olmadığı kontrol edilmelidir.

7) **A07:2021-Identification and Authentication Failures :** Güçlü parola politikaları uygulanmalıdır. Çift aşamalı doğrulama ile ek bir güvenlik katmanı sağlanmalıdır. Kullanıcı verileri veri tabanlarında şifrelenmiş olarak saklanmalıdır. Düzenli sızma testleri yapılmalı ve tespit edilen zafiyetler hızla giderilmelidir. Oturumdan çıkış yapıldığında oturum kimliğini içeren çerezler silinmelidir. Parola sıfırlama veya değiştirme işlemlerinde kullanıcının e-posta adresine bağlantı gönderilmeli ve bu bağlantılar kısa süre içinde kullanılmadığında geçersiz hale gelmelidir. Giriş işlemlerinde hatalı e-posta veya şifre girildiğinde hangi bilginin yanlış olduğu belirtilmemelidir. Brute-force saldırılarını önlemek için CAPTCHA kullanılmalıdır. Kullanıcı hesabına giriş için gerekli bilgilerin tamamı tek bir sayfada değil, ardışık sayfalarda istenerek otomatik saldırılar zorlaştırılmalıdır.

8) **A08:2021-Software and Data Integrity Failures :** Kimlik doğrulama yöntemiyle verilerin güvenilir kaynaklardan geldiği doğrulanmalıdır. Sistemdeki bileşenlerin güvenilir kaynaklardan temin edildiğinden ve sınırlı erişime sahip olduğundan emin olunmalıdır. CI/CD süreçlerinde uygun ayırma, erişim denetimi ve yapılandırmalar sağlanmalıdır. Kodlar dağıtılmadan önce kapsamlı testlerden geçirilmeli ve bu testler her yeni konfigürasyon ve güncellemede tekrarlanmalıdır. Güncellemeler geciktirilmeden uygulanmalıdır. Veriler şifrelenmeli, bütünlük kontrolünden geçirilmeli ve dijital imza ile

yedeklenmelidir.

9) **A09:2021-Security Logging and Monitoring Failures :** Loglama ve izleme süreçleri düzgün bir şekilde yürütülmelidir. Potansiyel saldırıları tespit edip bildirebilen IDS/IPS sistemleri kullanılmalıdır. Düzenli olarak sızma testleri yapılmalıdır. Loglar yedeklenmelidir. Gerçek zamanlı alarm oluşturabilen izleme sistemleri kullanılmalıdır. Logların bütünlüğü güvence altına alınmalıdır.

10) **A10:2021-Server-Side Request Forgery :** Uygulamaların erişmesi gereken DNS adları ve IP adresleri beyaz listeye eklenmelidir. Gerekli durumlarda kara liste de oluşturulmalıdır. Kullanıcı girdileri mutlaka filtrelenmelidir. Kullanılmayan URL şemaları devre dışı bırakılmalıdır, örneğin dict://, file://. İç ağdaki servislerde kimlik doğrulaması yapılmalıdır. Sunucuların gönderdiği yanıtlar kontrol edilmelidir. Çeşitli zafiyet tarama araçlarıyla düzenli olarak güvenlik kontrolleri yapılmalıdır. Acunetix ve Netsparker gibi tarama araçları SSRF zafiyetlerini tespit edebilen programlardan bazılarıdır.