Özet

- Yazılım güvenliği çözülmemiş bir sorun olmaya devam etmektedir; iyi bilinen yazılım güvenlik zafiyetleri bile hala tehdit olmaya devam etmektedir. Güvenlik terminolojisi önemli anlamlar ifade ederken bu terimlerin doğru kullanılması önemlidir. Bilgi güvenliğinin temel özellikleri gizlilik, bütünlük ve kullanılabilirliktir. Saldırganlar, yazılımlardan yararlanmak için koddaki güvenlik açıklarından yararlanırlar ve bu güvenlik açıklarını istismar ederler. Bütün yazılım sistemleri (ve tüm varlıklar) birçok tehditle karşı karşıyadır. Yazılım güvenliğinin zafiyetlerinden kaynaklanacak, yetersiz güvenlik yüksek maliyetlere sebep olabilir. Çünkü bir web uygulamasının saldırıya uğraması özellikle kurumların mahrem bilgilerinin ifşa edilmesine sebep olabilir. Saldırı için kullanılan komutlar, güvenilmeyen girdiler dinamik olarak sentezlendiğinde, enjeksiyon saldırıları her zaman gerçekleşebilir. Noktalama işaretleri, (',*, <,>, vb) işaretleri ve meta karakterler tipik olarak saldırı aracı olarak kullanılabilecekleri için girdi doğrulamak veya potansiyel olarak tehlikeli dize girişlerini güvenli olanlara dönüştürmek amacıyla geçici kod yazmak hem çok zahmetlidir hem de tam olarak doğru olma ihtimali düşüktür. SQL enjeksiyonları, komut enjeksiyonları, kod enjeksiyonları veya XML enjeksiyonları en yaygın saldırı türlerindendir. Günümüzde yazılımları en çok istismar eden saldırı türleri arasında SQL enjeksiyon, XSS enjeksiyonu, XML enjeksiyonu, kaba kuvvet saldırıları bulunmaktadır. SQL enjeksiyonu veri tabanına sahip yazılımlarda bulunan SQL cümleleri kullanılarak gerçekleştirilmektedir. Yazılımın veri tabanındaki verilerin ele geçirilmesi, silinmesi veya değiştirilmesi gibi işlemlerin gerçekleştirilmesi için yapılır. XSS enjeksiyonu ise javascript veya ajax gibi betik diller ile kullanıcıların çerez bilgilerinin ele geçirilmesi ve yazılımlara yetkisiz erişimlerin gerçekleştirilmesine sebep olur. Kaba kuvvet saldırıları, kullanıcı adı ve şifre girişi için kullanılan web formlarına karşı yapılır. Web form alanlarında kullanıcı adı ve şifre bilgilerini tahmin etmek için sürekli deneme yapılır. Eğer şifreler kolay tahmin edilebilir olursa, kaba kuvvet saldırıları da başarılı olur. Kaba kuvvet saldırıları robot yazılımlar kullanılarak gerçekleştirilir. Kaba kuvvet saldırılarını engellemek için CAPTCHA doğrulama özellikleri kullanılır. Web uygulamalarındaki bir diğer zafiyet türü ise kabuk (shell) komutlarında, XML'de oluşturulmuş, özelleştirilmiş komut dosyalarında ortaya çıkabilir.
- Güvenli yazılım geliştirme yöntemleri kullanılarak bu tür saldırılar engellenebilir. Web yazılımları geliştirilirken dikkat edilmesi gereken en önemli güvenlik ilkeleri girdi doğrulama, kimlik doğrulama, yetkilendirme, parola yönetimi ve log yönetimi işlemleridir. Yazılım geliştirilirken bu ilkelere dikkat edilirse, geliştirilen yazılımlar daha güvenli olabilir. Web yazılımlarında bulunan zafiyetler yazılım geliştirme sürecinde, yazılım tasarımından veya yazılım geliştirme uzmanından kaynaklanabilir. Güvenli yazılım geliştirilirken, yazılım geliştirme yaşam döngüsü aşamalarının tamamında güvenlik ilkelerine uyulmalıdır. Yazılım geliştirme uzmanlarının da güvenli yazılım geliştirme bilgisine sahip olması gereklidir ve bilgi güvenliği farkındalığına sahip olmalıdırlar. Böylece yazılımlarda istenmeyen kötücül komutların yürütülmesine izin vermeyen, meta karakterlerin (noktalama işaretleri) uygun olmayan şekilde kullanılmasını engelleyen, kullanıcı girdilerinin doğrulanarak işlendiği güvenli yazılımlar geliştirilir.

DEĞERLENDİRME SORULARI

1. Yazılım geliştirme süreçlerinde güvenlik ilkelerine uymanın avantajları arasında aşağıdakilerden sayılamaz?
   a) Güvenli yazılım geliştirme ilkelerini benimseyerek güvenlik riskini azaltmak.
   b) Yazılım tasarım aşamasından itibaren güvenlik ilkelerine uyma
   c) Geliştirme maliyetlerinin azaltılması.
   d) Güvenlik metodolojisi kullanarak tamamlama süresini planlama.
   e) Yazılımın müşteriye teslimi aşamasında güvenlik önemlerini almak.

2. Statik kaynak analiz araçları için aşağıdakilerden hangisi doğrudur?
   a) Uygun bir yazılım için testçi ve geliştiriciler değil denetçiler de olmalıdır.
   b) Doğrudan erişim ile kaynak kod üzerinde gerçekleştirilir.
   c) Beyaz kutu olarak adlandırılır.
   d) Yerine kullanılabilirlik için tasarım, mimari ve dağıtım süreçlerinden oluşur.
   e) Kullanıcılara En Az Yetki verilmesi ilkesidir.

3. CAPTCHA kullanımının amacı aşağıdakilerden hangisinde doğru verilmiştir?
   a) Girdilerin robot yazılımlardan geldiğini doğrulamak.
   b) Girdilerin robot yazılımlardan değil, gerçek kişi tarafından girildiğini doğrulamak.
   c) Gerçek kişi veya robot tarafından hiçbir veri girilmediği doğrulamak.
   d) Girdilerin bir insandan ve bir robottan girildiğinden emin olmak.
   e) Girdilerin noktalama işareti veya meta karakter doğrulamasını yapmak.

4. "Kara kutu" testinin amacı aşağıdaki seçeneklerden hangisinde doğru verilmiştir?
   a) Üretim uygulamalarındaki zafiyetleri belirler.
   b) Yararlanma potansiyelini ölçmek için zafiyetleri belirler.
   c) Kodlama hataları için kaynak kodunu denetler.
   d) Bilinen bütün tehditleri belirler.
   e) Kötüye kullanımı ifade eder.

5. Yazılım güvenliğini en iyi ne zaman değerlendirmek gerekir?
   a) Test sırasında.
   b) Geliştirme sırasında.
   c) Tasarım sırasında.
   d) Bakım sırasında.
   e) Uygulama geliştirmenin tüm aşamalarında

6. Aşağıdaki seçeneklerin hangisinde beyaz kutu testi ile kara kutu testi arasındaki fark en doğru şekilde açıklanmıştır?
   a) Beyaz kutu testi, bağımsız bir programcı ekibi tarafından gerçekleştirilir.
   b) Kara kutu testi aşağıdan yukarıya yaklaşımı kullanır.
   c) Kara kutu testi, iş birimlerini içerir.
   d) Beyaz kutu testi, yazılımın dâhili mantıksal yapılarını inceler.
   e) Kara kutu testi belli olmayan yöntemlere karşı yapılır.

Tanım: Yazılımlarda kişisel gizliliğin ve özel bilgilerin korunmasına yönelik araçlar da dâhil olmak üzere, bilgi erişimi ve ifşasına ilişkin kısıtlamaların oluşturulmasıdır.

7. Yukarıdaki tanım, aşağıdaki seçeneklerin hangisine aittir?
   a) Kullanılabilirlik.
   b) Bütünlük.
   c) Gizlilik.
   d) Kimlik Doğrulama.
   e) Girdi Doğrulama.

8. Ağ güvenlik zafiyeti değerlendirmesi ile sızma testi arasındaki fark aşağıdaki seçeneklerin hangisinde doğru verilmiştir?
   a) Sızma testi, kaynakları sıralar ve güvenlik açığı değerlendirmesi, güvenlik açıklarını sıralar.
   b) Ağ güvenlik zafiyet değerlendirmesi ve sızma testi aynıdır.
   c) Sızma testi, çalışan hizmetleri tanımlar ve güvenlik açığı değerlendirmeleri, güvenlik açıklarının daha derinlemesine anlaşılmasını sağlar.
   d) Sızma testi, güvenlik açıklarından yararlanır ve güvenlik açığı değerlendirmesi, güvenlik açıklarını bulur.
   e) Sızma testleri yazılım uygulamaları için, ağ güvenlik testi değerlendirmeleri ise ağ alt yapıları için yapılır.

9. Aşağıdakilerden hangisi güvenli yazılım geliştirme yöntemlerine dikkat edilmeden geliştirilen Web uygulamalarını hedef alacak bir saldırı türü değildir?
   a) SQL Enjeksiyonu
   b) Kaba Kuvvet Saldırıları
   c) XML Enjeksiyonu
   d) XSS Enjeksiyonu
   e) Hizmet Aksatma (DOS, DDOS)

10. Yönetici kimlik doğrulaması gereksinimleri aşağıdakilerden hangisinde doğru tanımlanmıştır?
    a) Normal kullanıcılar için eşdeğer seviyede olmalıdır
    b) Tüm yöneticiler daha güvenilir olduğu için normal kullanıcılardan daha az doğrulama katmanına sahip olabilirler.
    c) Yöneticiler için kimlik doğrulama yapılmasına gerek kalmayabilir.
    d) Normal kullanıcılardan daha fazla yetki ile doğrulama yapılmalıdır.
    e) Doğrulama işlemlerinde bütün kullanıcılar gibi aynı seviyede doğrulama yapılmalıdır.

Cevap Anahtarı
1.e, 2.d, 3.b, 4.d, 5.e., 6.d, 7.c, 8.d, 9.d, 10.d