1. Broken Access Control

a. Zafiyet Nedir?

- **Kullanıcıların yetkisi olmayan fonksiyon yada kaynaklara erişebilmesi**

b. Neden Kaynaklanır?

- **Gerekli erişim(yetki) kontrollerinin düzgün bir şekilde yapılamamasından**

c. Türleri ve Kısa Açıklamaları

- **IDOR:** *bir id kullanılarak, yetkinin olmadığı bir kaynağa/fonksiyona doğrudan erişmek*
- **Privilege Escalation:** *Kullanıcının yetkisi olmayan bir fonksiyonu gerçekleştirebilmesi*

e. Nasıl Önlenir?

- **Gerekli yetki kontrollerinin tam olarak yapılması**
- **Least privilege prensibinin kullanılması**

2. Cryptographic Failures

a. Zafiyet Nedir?

- **Şifreli verilerin saldırganlar tarafından çözülmesi durumu**

b. Neden Kaynaklanır

- **kullanılan kriptografik yöntemi yanlış uygulama yada zayıf bir kriptografik yöntem kullanma durumlarından kaynaklanır**

c. Türleri ve Kısa Açıklamaları

- **Zayıf Kriptografik yöntemi kullanma:** *mesela MD5 hash algorithması*
- **Güvensiz Key Yönetimi:** *Şifreleme için kullanılan key'in zayıf olması, saldırganlar tarafından ele geçirilebilecek bir yerde saklanılması yada key'in yeniden kullanılabilmesi gibi durumları kapsar*
- **Hassas Verilerin Şifrelenmemesi:** *credi kartı bilgileri, şifre gibi hassas verilerin, veri aktarımı sırasında ve databasede plain text şeklinde gönderilmesi*

e. Nasıl Önlenir

- **Güçlü kriptografik yöntemler kullanılmalı**
- **Hassas verileri, veri aktarımı sırasında ve database'de şifrele**
- **Şifreleme için kullanılan key'lerin uzun ve random olarak oluştur, düzenli olarak key'leri yenile ve saldırganların ulaşabileceği bir yerde tutma**

3. Injection

a. Zafiyet Nedir?

- **Saldırganın uygulamaya kod enjekte edebilmesi**

b. Neden Kaynaklanır?

- **Yetersiz input sanitization'ı ve input'un enjeksiyona sebep olabilecek şekilde bir yerde kullanılması**

c. Türleri ve Kısa Açıklamaları

- **sql/nosql injection:** *Database'de kod çalıştırılabilmesi*
- **Command injection:** *Sistem shell'inde yada komut interpreter'inde kod çalıştırılabilmesi*
- **LDAP injection:** *LDAP query enjecte ederek, LDAP directory'den veri update/elde etme işlemlerinin yapılması*
- **XXE injection:** *xml input parse eden uygulama'ya, xml injecte etme*
- **xss:** *javascript injecte etme*
- **host header injection:** *host header kullanılarak oluşturulan herhangi bir link varsa, host header değiştirerek link manipüle edilebilir*

e. Nasıl Önlenir

- **User input'unun gerekli sanitization işlemlerinden geçirilmesi**
- **waf kullanımı**
- **parameterized query ve prepared statements kullanımı**
- **output encoding**

4. Insecure Design

a. Zafiyet Nedir?

- **uygulama tasarımından kaynaklı ortaya çıkan zafiyetler**

b. Neden Kaynaklanır?

- **Uygulama tasarımının yeterince güvenli yapılmaması**

c. Türleri ve Kısa Açıklamaları

- **Güvenli Tasarım patternlerinin yokluğu:** *kabul görmüş patternlerin aksini yapmak, mesela şifrelerin database'de plain text olarak tutulması gibi*

e. Nasıl Önlenir

- **Güvenli Tasarım patternlerini uygulayarak**

5. Security Misconfiguration

a. Zafiyet Nedir?

- default credential kullanımı, gereksiz özelliklerin aktif olması gibi konfigurasyonla ilgili açıkları kapsar.

b. Neden Kaynaklanır?

- default yada yanlış yapılmış konfigurasyonlardan kaynaklanır

c. Türleri ve Kısa Açıklamaları

- **Default Credentials:** *kullanılan software/hardware'ın, üretici tarafından koyulan default username-password'ü kullanmak*
- **Unnecessary Features Enabled:** *Gereksiz ve kullanılmayan; portların, servislerin yada özelliklerin açık bıraklması*
- **Improper File and Directory Permissions:** *file, directory permission'larının yanlış yapılması*

e. Nasıl Önlenir

- **Default credential'ları değiştirmek**
- **Gereksiz özellikleri kapatmak**
- **file ve directory permission'larını düzgün yapmak**

6. Vulnerable and Outdated Components

a. Zafiyet Nedir?

- Bilinen güvenlik açığı bulunan software patch'ini kullanmak

b. Neden Kaynaklanır?

- Software'ı güncel tutmamaktan dolayı

c. Türleri ve Kısa Açıklamaları

- **Güncel olmayan software'lar:** *bilinen açığı olan patch'i kullanmak*

e. Nasıl Önlenir

- **Software'ları güncel tutarak**

7. Identification and Authentication Failures

a. Zafiyet Nedir?

- saldırganın başkalarının kimliğini çalması

b. Neden Kaynaklanır?

- uygulamanın kimlik doğrulama konusunda zayıf olmasından kaynaklı

c. Türleri ve Kısa Açıklamaları

- **Weak Password Policies:** *user'a, yaygın ve tahmin edilebilir password kullanmasına göz yummak*
- **Brute Force Attacks:** *saldırganın; kısa sürede çok fazla password test edebilmesine neden olur*

e. Nasıl Önlenir

- **güçlü bir password policy uygulamak**
- **Brute force önlemek için rate limit koymak**
- **2fa(Two Factor Authentication) kullanmak**

8. Software and Data Integrity Failures

a. Zafiyet Nedir?

- **Software updates yada data iletimi sırasında, software yada verilerin değiştirilmesi**

b. Neden Kaynaklanır?

- **Yapılan software update'leri ve data iletiminin güvensiz ve doğrulama olmadan yapılması**

c. Türleri ve Kısa Açıklamaları

- **Insecure Software Update Mechanisms:** *Software update'lerinin doğrulama yapılmadan gerçekleştirilmesi, yetkisiz değişikliklere sebep olabilir*
- **Insecure Deserialization:** *Deserialization sırasında, herhangi bir doğrulama yapmadan user input'una güvenmek*
- **Untrusted Components:** *Güvenli olmayan 3.parti library ve bileşen kullanımı*

e. Nasıl Önlenir

- **Secure Update Mechanisms:** *HTTPS kullanmak ve softawre update'ini onaylamak için code-singin(veri imzalama) kullanmak*
- **Protect the Software Supply Chain:** *3.parti bileşen ve library'lere, sıkı kontroller ve denetimler koy*
- **Perform Integrity Checks:** *Data ve software bütünlüğünü korumak için; checksum, hash yada dijital imzalar kullan.*

9. Security Logging and Monitoring Failures

a. Zafiyet Nedir?

- **Güvenlik logging ve izleme yetersizliği**

b. Neden Kaynaklanır?

- **Güvenlik logging ve izleme için sistem kurmamak yada yetersiz sistem kurmak**

c. Türleri ve Kısa Açıklamaları

- **Insufficient Log Generation:** *Kritik fonksiyonlar için log oluşturmamak*
- **Lack of Real-Time Monitoring:** *Log'ları ve güvenlik olaylarını, gerçek zamanlı olarak kontrol etmemek*

e. Nasıl Önlenir

- **Log işlemi için düzgün bir sistem kurmak**
- **Log'ların gerçek zamanlı olarak incelenmesi**

10. SSRF

a. Zafiyet Nedir?

- **Server'a istek yaptırabilmek**

b. Neden Kaynaklanır?

- **kullanıcı input'unun güvensiz şekilde url isteğinde yada pdf generation tarzı bir işlemde kullanılması**

c. Türleri ve Kısa Açıklamaları

- **Basic SSRF:** *User input'u direk olarak bir url isteğinde kullanılır ve bu isteğin response'ı user'a döndürülür*
- **Blind SSRF:** *Basic ssrf ile aynı mantık, sadece isteğin response'ı user'a geri döndürülmez*

e. Nasıl Önlenir

- **User input'unu direk kullanmadan önce, url'in beklendik bir url olduğunu kontrol etmek**
-