Istanbul Gelisim Univ J Soc Sci 2025:12(2);847-861
DOI: 10.17336/igusbd.1321489

DERLEME / REVIEW

Web Uygulamaları Güvenliği Alanında Güvenlik Açığı Çalışmalarından OWASP Top 10 İncelemesi

OWASP Top 10 Review of Vulnerability Studies in Web Application Security

[1]Çisem YAŞAR
[2]Tuğba SARAY ÇETİNKAYA
[3]Ediz ERTİM

[1]Öğr. Gör., İstanbul Gelişim Üniversitesi, İstanbul Gelişim Meslek Yüksekokulu, Bilgisayar Teknolojisi Bölümü, İstanbul, Türkiye
✉ cyasar@gelisim.edu.tr

[2]Öğr. Gör., İstanbul Gelişim Üniversitesi, İstanbul Gelişim Meslek Yüksekokulu, Bilgisayar Teknolojisi Bölümü, İstanbul, Türkiye
✉ tgbsry@gmail.com

[3]İstanbul Gelişim Üniversitesi, İstanbul Gelişim Meslek Yüksekokulu, Bilişim Güvenliği Teknolojisi, İstanbul, Türkiye
✉ edizertim1@gmail.com

Geliş/Received: 01.07.2023
Kabul/Accepted: 29.07.2025

Öz

Bilgi güvenliğini sağlamak için bilgi sistemini oluşturan kaynaklarda var olan güvenlik açıkları ve risklerini bulmak gerekmektedir. Bu açıkların büyük bölümü etkili iletişim ve bilgi alışverişi sağlamak amacıyla kullanılan ve neredeyse bütün sistemlerde yer alan web uygulamalarında bulunmaktadır. OWASP (Open Web Application Security Project), web uygulamalarına yönelik kritik güvenlik açıkları ve risklerini oluşturan güncel ve önemli kavramları belirlemektedir. Bu amaçla 10 maddeden oluşan bir liste yayınlamaktadır. Bu çalışmada, OWASP tarafından yayımlanan güvenlik açıkları ve önerileri ele alınmış olup, web uygulamalarında bilgi güvenliğinin sağlanmasına yönelik katkı sunması amaçlanmıştır. Her madde ayrı ayrı araştırılarak analiz edilmiştir. Literatür incelendiğinde karşılaşılan çalışmaların amaçları ve kullanılan yöntemleri üzerinde durulmuştur. Yayımlanan güncel listeye dördüncü sırada olan güvensiz tasarım, sekizinci sırada yer alan yazılım ve veri bütünlüğü arızaları ve onuncu sırada yer alan sunucu taraflı istek arızaları kategorileri yeni eklenmiştir. Eklenen bu 3 kategori konusunda çalışmaların yetersiz olduğu görülmüştür. Sonuç olarak yeni eklenen maddelerle birlikte literatürde yer alan eksikliklerin giderilmesine yönelik katkı sağlanmıştır.

Anahtar Kelimeler

OWASP, Web Uygulama Güvenliği, Güvenlik Açıkları, Web Uygulamaları, Bilgi Güvenliği

Abstract

In order to ensure information security, it is necessary to find the vulnerabilities and risks that exist in the resources that make up the information system. Most of these vulnerabilities are found in web applications, which are used for effective communication and information exchange and are found in almost all systems. OWASP (Open Web Application Security Project) identifies

Çisem YAŞAR, Tuğba SARAY ÇETİNKAYA, Ediz ERTİM "OWASP Top 10 Review of Vulnerability Studies in Web Application Security", Istanbul Gelisim University Journal of Social Sciences, 12 (2), October 2025, pp. 847-861.

current and important concepts that constitute critical vulnerabilities and risks for web applications. For this purpose, it publishes a list of 10 items. In this study, the vulnerabilities and recommendations published by OWASP are discussed and it is aimed to contribute to the provision of information security in web applications. Each item was investigated and analyzed separately. When the literature was examined, the aims and methods used in the studies encountered were emphasized. The fourth category of insecure design, the eighth category of software and data integrity failures, and the tenth category of server-side request failures have been added to the current list. It has been observed that studies on these 3 categories are insufficient. As a result, with the newly added items, a contribution has been made to fill the gaps in the literature.

Keywords

OWASP, Information Security, Web Application Security, Vulnerabilities, Web Application

Giriş

İnternet kavramı Amerika Birleşik Devletleri (ABD) ordusunun geliştirme kolu olan "İleri Araştırma Projeleri Ajansı" tarafından yürütülen ve desteklenen bilim projesi olarak 1960'ların sonlarına doğru hayatımıza girmiştir (Barlett, 2016). Günümüzde ise bireysel olarak kullanılan en büyük ağ haline gelmiştir. Kullanımının hızlı bir şekilde artmasıyla beraber bilgi sistemlerinde kullanılan uygulamalar da internet ağı odaklı olarak gelişmiştir. Bu durum beraberinde masaüstü uygulamaların kullanımının azalmasına ve web uygulamalarının daha çok kullanılmasına yol açmıştır (Manikanta ve Sardana, 2012; Scholte, Balzarotti ve Kirda, 2012).

Web uygulamaları, bilgi paylaşımı, bankacılık işlemleri, çevrimiçi sohbet, sosyal etkileşim ve iletişim kurma gibi kullanıcı ihtiyaçlarının çevrimiçi ortamda gerçekleşmesine olanak sağlamıştır. Web uygulamaları, sunduğu bu olanaklar sayesinde bireylerin ihtiyaçlarını karşılayabilmektedir. Bunun yanında devletler tarafından gerçekleştirilen kamu hizmetleri de web uygulamaları üzerinden sağlanmaktadır (Monga, Paleari ve Passerini, 2009). Bu bağlamda bakıldığında birçok işlemin yapılabilmesine olanak sağlayan internet, barındırdığı bilgiler yüzünden saldırganların hedefi haline gelmiştir. İnternet ortamının güvenliğinin sağlanabilmesi web uygulama güvenliği ile doğrudan ilişkilidir. Web uygulama güvenliği, verilerin gizlilik, bütünlük ve erişilebilirliğinin sağlanması amacıyla alınan önlemlerin tümünü ifade etmektedir (Aydoğdu ve Gündüz, 2016; Aydın, Barışkan ve Çetinkaya 2021). Saldırganlar web ortamında kullanılan uygulamalarda yer alan sistem açıklarını hedef haline getirmektedir. Bu açıkları kullanarak sisteme giriş yapabilmekte ve bu sayede elde edinilen bilgileri kendi amaçları doğrultusunda kullanabilmektedir (Kara, 2020). Bu şekilde gerçekleştirilen eylemler sadece bireylere, kurum ve kuruluşlara değil ülke ekonomisine de maddi ve manevi zarar vermektedir. Bilim, Teknoloji ve Yenilik Politikaları Kuruluna (BTYPK) sunulan "Türkiye Bilim, Teknoloji ve Yenilik Politikaları ve Stratejileri" raporuna göre dünyada siber güvenlik pazarının 2020 yılında 173 milyar dolar değerinde olduğu görülmektedir. Bu değerin 2026 yılında ise 270 milyar dolar büyüklüğüne ulaşması öngörülmektedir. 2021 yılında en hızlı büyüyen alanlar içerisinde web ve e-posta güvenliği uygulamalarının olduğu belirtilmiştir (TÜBİSAD-Deloitte, 2021). Türkiye'nin siber güvenlik pazarına bakıldığında ise 2020 yılında 172 milyon dolar, 2021 yılında 160 milyon dolar ve 2023 yılında ise 172 milyon dolarlık pazar büyüklüğü öngörülmektedir (Gartner, 2020).

Çalışmanın Metodolojisi

Bu çalışmada, OWASP tarafından 2021 yılında yayınlanan ve günümüzde en yaygın görülen web açıklıklarını içeren Top 10 listesinin detaylı bir şekilde analizi gerçekleştirilmiştir. Bu kapsamda nitel araştırma yöntemi benimsenerek her madde detaylı bir şekilde araştırılmış, amaç, yöntem ve önerilere yer verilmiştir. Bu çalışma kapsamında:

- Günümüzde en yaygın web açıklıkları nelerdir?
- Web uygulamalarında bulunan açıklıklar ile ilgili yapılan çalışmalar ve kullanılan yöntemler nelerdir?
- Yaygın web açıklıkları ile ilgili literatürde verilen öneriler nelerdir?
- sorularına yanıt aranmıştır. Her bir açık için detaylı açıklamaya bulgular bölümünde yer verilmiştir.

OWASP Güvenlik Açıklıklarına Yönelik Örnekleri

Web uygulamalarındaki güvenlik zafiyetlerinin hızlı artışı ile birlikte uygulama geliştiricilere destek olmak ve bilgi sunmak amacıyla 2001 yılında kar amacı gütmeyen bir organizasyon olan OWASP kurulmuştur (Owasp, 2023). Bu kuruluş sayesinde web uygulama güvenliğinde en zararlı ve en sık görülen açıklar belirlenip OWASP Top 10 adı altında listelenmeye başlanmıştır (Wichers, 2023). OWASP tarafından 2021 yılında yayınlanan web uygulama güvenliği zafiyetlerinin 10 tanesi Şekil 1 ile verilmiştir (OWASP Top 10:2021, 2023).

A01:2021 - Kırık Erişim Kontrolü (Broken Access Control)

A02:2021 - Kriptografik Hatalar (Cryptographic Failures)

A03:2021 - Enjeksiyon (Injection)

A04:2021 - Güvensiz Tasarım (Insecure Design)

A05:2021 - Yanlış Güvenlik Yapılandırması (Security Misconfiguration)

A06:2021 - Savunmasız ve Güncel Olmayan Bileşenler (Vulnerable and Outdated Components)

A07:2021 - Tanımlama ve Kimlik Doğrulama Hataları (Identification and Authentication Failures)

A08:2021 - Yazılım ve Veri Bütünlüğü Arızaları (Software and Data Integrity Failures)

A09:2021 - Güvenlik Kaydı ve İzleme Arızaları (Security Logging and Monitoring Failures)

A10:2021 - Sunucu Taraflı İstek Arızaları (Server-Side Request Forgery)

Şekil 1. OWASP tarafından 2021 yılında yayınlanan 10 madde (OWASP Top 10:2021, 2023)

Web uygulamalarının güvenlik zafiyetleri üzerine yapılan çalışmalarının yetersizliğinden dolayı OWASP Top 10 içerisindeki saldırı türleri analiz edilerek sınıflandırılmaktadır (Karacan ve Sevri, 2021). Bu sınıflandırmalar içerisinde Yapısal Sorgulama Dili (Structured Query Language – SQL Injection) (Hassan vd., 2018), siteler arası komut dosyası çalıştırma saldırısı (Cross-site scripting – XSS) (Fang, Li, Liu ve Huang, 2018) gibi çalışmalar bulunmaktadır. Ayrıca HTTP paketleri içerisindeki korelasyonu hesaplayabilmek için HTTP isteklerinin içerisindeki bilgilerin analizinde n-gram analiz yöntemini kullanarak 5 gram harf analizi ve 5'li harflerin birliktelik tekrarı incelenmiştir (Torrano-Gimenez, Nguyen, Alvarez, Petrovic ve Franke, 2011). Yapılan n-gram analizinden elde edilen veri setinin çok büyük olması sebebiyle Temel Bileşen Analizi (TBA) yöntemi ile öz nitelik indirgemesi çalışması yapılmıştır (Vartouni, Teshnehlab ve Kashi, 2019). Elde edilen veri setinde Tekrarlayan Yapay Sinir Ağı (Recurrent Neural Network – RNN) ve Uzun Kısa Süreli Bellek LTSM (Long Short-Term Memory – LTSM) yöntemlerini kullanarak TBA özellik seçiminden farklı bir şekilde verilerin sınıflandırılması üzerine çalışma yapılmıştır (Liang, Zhao ve Ye, 2017). Web uygulamalarında derin öğrenme yöntemlerini kullanarak anomali tabanlı bir web saldırısı algılama mimarisi üzerine çalışmalar yapılmıştır (Tekerek, 2021).

1.1. Kırık Erişim Kontrolü (Broken Access Control)

Erişim kontrolleri kullanıcıların belirli izinlerin dışında hareket edememeleri şeklinde tanımlanmaktadır. Erişimdeki başarısızlıklar bilgilerin yetkisiz olarak ifşa edilmesine, tüm verilerin değiştirilmesine veya imha edilmesine, kullanıcının sınırları dışında bir işlevinin gerçekleştirilmesine yol açmaktadır. Kırık erişim kontrolü, erişim kontrolü ayarlarının yapılandırılmaması, diğer kullanıcıların hesaplarını düzenlenmesi, ayrıcalıkların yükseltilmesi, kısıtlanmış API'lere yetkisiz erişime izin veren Kökenler arası kaynak paylaşımı (Cross-Origin Resource Sharing - CORS) yanlış yapılandırmaları ve JavaScript Nesne Notasyonu (JavaScript Object Notation - JSON) gibi erişim kontrol belirteçleri aracılığıyla meta veri manipülasyonu gibi farklı nedenlerden oluşmaktadır (Bach-Nutman, 2020, s. 1-4).

Gupta, Singh ve Mohapatra (2022, s. 1-2), OWASP tarafından 2021 yılında yayınlanan raporda yer alan 300.000'den fazla web sitesinde güvenlik ihlaline yol açan, kimliği doğrulanmamış ve yetkisiz

erişim konusu üzerine çalışmışlardır. Web uygulamalarının kimlik doğrulama sırasında yaşanan güvenlik açığını analiz etmek için, var olan güvenlik modellerinin davranışsal yönünü test eden bir çalışma gerçekleştirmişlerdir. Bu çalışmada güvenlik modelini sınıf diyagramı kullanarak test etmişlerdir. Web uygulamalarında kimliği doğrulanmamış ve yetkisiz erişim riskini en aza indirmek için birinci dereceden yüklem mantığının kullanılması önerilmiştir.

Poel (2022), web uygulamalarında gri kutu penetrasyon testi ile kırık erişim kontrol tarama metodolojisini ve uygulamasını geliştirmek için araştırma yapmıştır. Yapılan araştırma, kırık erişim kontrollerinin test sürecinin karmaşıklığı ve zaman alıcı olması nedeniyle, bu alanda pratik çözümler geliştirmek amacıyla gerçekleştirilmiştir. Zor olması ve zaman almasından dolayı geliştirme amaçlı gerçekleştirilmiştir. Araştırmacı kırık erişim kontrollerini kapsamlı bir şekilde inceleyebilmek için, modern bir metodoloji ve araç seti kullanmıştır. Dört aşamadan oluşan metodolojinin test sonucuna göre kırık erişim kontrol tarama metodolojisinin uygulanabilir bir strateji olduğu ve değerli bir araç olduğu gözlemlenmiştir. Konu hakkında incelemeyi kolaylaştırmak için karşılaşılan çalışmaların özeti Tablo 1 ile verilmiştir.

Tablo 1. Kırık Erişim Kontrolü

Kaynak	Amaç	Yöntem
Gupta vd., 2022	Kırık erişim kontrolü güvenlik modelleri analizinin gerçekleştirilmesi	Kırık erişim kontrolü modelinin sınıf diyagramı ile testi
Poel, 2022	Gri penetrasyon testi ile kırık erişim kontrolü metodolojisi geliştirilmesi	Modern bir metodoloji ve araç seti kullanımı

1.2. Kriptografik Hatalar (Cryptographic Failures)

Kriptografik hata, kullanılan şifreleme yöntemlerinin yetersiz veya güvenli olmamasından dolayı verilerin ele geçirilmesi şeklinde tanımlanmaktadır. Güçlü olmayan şifreleme yöntemleri kullanılması veya verilerin şifrelenmeden veritabanında tutulması bu hataya yol açabilmektedir (Karakaya, 2022).

COVID-19 döneminde virüsün yayılma hızını azaltmak amacıyla dijital öğrenme ortamlarına geçiş başlamıştır. Dijital öğrenme ortamlarına hızlı geçiş ile birlikte siber saldırı ve güvenlik sorunu çok önemli bir problem haline gelmiştir. Djeki, Degila, Bondiombouy ve Alhassan (2022), yaptıkları çalışmada en yaygın kullanılan öğrenme yönetim sistemleri (Learning Management System - LMS) ve video konferans araçlarındaki güvenlik açıkları ile OWASP ve Common Weakness Enumeration (CWE) tarafından yayımlanan web uygulaması güvenlik açıklarını karşılaştırmalı olarak analizi etmişlerdir. Saldırılara karşı en savunmasız platformun Moodle, öğrenme yönetim sistemleri içerisinde en güvenlisinin Blackboard ve video konferans araçlarından en güvenlisinin ise Zoom olduğu sonucuna varmışlardır. Yapılan çalışmada öneri olarak teknik hataların yanında kullanıcıların şifrelemede yaptığı hatalara da vurgu yapılmıştır.

Günümüzde web uygulamaları oluşturmak için en çok mikro hizmet mimarisi kullanılmaktadır. Mikro hizmet mimarisinde ise dikkat edilmesi gereken en önemli husus OWASP'ın ilk 10 güvenlik açığıdır. Mikro hizmet mimarisi ile kullanılan N-versiyon, API'ler ile bileşenlerine ayrılan klasik bir güvenlik / dayanıklılık ilkesidir. Espinoza, Wood, Forrest ve Tiwari (2022), yaptıkları çalışmada veri sızıntılarına karşı direnci artırmak için mikro hizmet mimarisinde N-versiyon kullanılmıştır. Güvenlik açığı bulunan mikro hizmet mimarisinde minimum kod değişikliği gerektiren ve düşük çoğaltma maliyeti olan N-versiyon uygulanmıştır. Sonuç olarak, mikro hizmet mimarisinin çeşitlilik ve yedeklilik özelliklerinin olmasının OWASP'ın ilk 5 güvenlik açığını azaltmada etkili olduğu sonucuna varmışlardır. Huang, Li ve Cai (2023), yaptıkları çalışmada metaverse'deki güvenlik ve gizlilik sorunları araştırılmıştır. Metaverse kavramının tanımını yapmak yerine sahip olması gereken dört özelliği liste haline getirmişlerdir. Bunlar; sosyalleşme, sürükleyici etkileşim, gerçek dünya inşası ve genişletilebilirliktir. Bu özellikler kişisel bilgi sızıntısı, gizli dinleme, yetkisiz erişim, kırık erişim kontrolü, güvenli olmayan tasarım ve kimlik avı gibi risklerden zarar görmesine neden olmaktadır. Ortadaki adam saldırısı (Man in the middle attack - MITM), kullanıcı ile sunucu arasındaki iletişimi gizlice dinleyerek, veri paketlerini değiştirebilir veya engelleyebilir. Bu durumsistemin gizliliğini, bütünlüğünü ve kullanılabilirliğinin tehlikeye girmesine sebep olabilmektedir. Bu saldırıyı önlemenin en etkili yolu, güçlü kimlik doğrulama ve kriptografik

protokoller kullanmaktır. Veri bütünlüğünü korumak için kimlik doğrulama algoritmaları ve veri kullanılabilirliğini sağlamak için de kriptografik protokollerin kullanılması gerektiğini vurgulamışlardır. Konu hakkında incelemeyi kolaylaştırmak için karşılaşılan çalışmaların özeti Tablo 2 ile verilmiştir.

Tablo 2. Kriptografik Hatalar

Kaynak	Amaç	Yöntem
Djeki vd., 2022	Dijital öğrenme ortamları üzerinde güvenlik açıkları ve siber saldırı etkilerinin analizi	Güvenlik açıklarının sınıflandırması
Espinoza vd., 2022	Veri sızıntılarına karşı direnci artırmak	Mikro hizmet mimarisinde N-sürümü uygulanması
Huang vd., 2023	Metaverse'deki güvenlik ve gizlilik sorunları araştırılması	MITM saldırısı

1.3. Enjeksiyon (Injection)

SQL enjeksiyonu saldırıları, saldırganlar tarafından veritabanındaki gizli bilgilere erişmek için kullanılmaktadır. Web uygulamalarının giriş alanlarına kötü amaçlı karakterler girilerek, SQL sorgusunun değiştirilmesi ile gerçekleştirilmektedir. SQL enjeksiyon saldırıları maddi kayba ve itibarın zarar görmesine neden olabilmektedir (Alenezi, Nadeem, ve Asif, 2021). SQL enjeksiyonu web uygulamalarını hedef alan en tehlikeli saldırı türlerinden bir tanesidir. (Jemal, Cheikhrouhou, Hamam ve Mahfoudhi, 2020). OWASP'a göre de SQL enjeksiyonu web uygulamalarının güvenliği söz konusu olduğunda en önemli güvenlik açıklarından biri olarak sınıflandırılmaktadır (Harefa, Prajena, Alexander, Dewa ve Yuliandry, 2021)

Latchoumi, Reddy ve Balamurugan (2020), güvenlik açıkları arasında olan SQL enjeksiyon saldırılarını durdurmak ve önlemek için bir teknik önermişlerdir. SQL enjeksiyon saldırıları için Destek Vektör Makinesi (DVM) algoritmasına sahip makine öğrenimi yöntemi üzerinde durulmuştur. Kullanıcı yeni sorgu verdiğinde, sorgunun herhangi bir kötü amaçlı ifade içerip içermediğini tahmin etmek için DVM uygulanmıştır. Bunun için DVM algoritması olası tüm kötü amaçlı ifadelerle eğitilmiş ve model oluşturulmuştur. DVM olası girişleri veri setinde bulunan minimum sayıda sözdizimi ile eşleştirerek kötü niyetli ifadeyi algılamaktadır. Gelecekteki araştırmalar için çeşitli SQL enjeksiyon saldırı türlerini tanımlamak ve gruplandırmak için çok sınıflı sınıflandırıcı kullanılmasını önermişlerdir.

Harefa vd. (2021), web uygulamaları güvenliğinde en yüksek riske sahip olan SQL enjeksiyon saldırısını engelleyebilmek amacıyla SEA WAF adlı bir web uygulaması güvenlik duvarı uygulaması sunmuşlardır. SEA WAF, yöneticinin verileri yönetmesine yardımcı olacak birkaç web sayfasına sahiptir. Sunulan yöntem kullanıcının oturum açmasının ardından otomatik olarak 5 farklı tür SQL enjeksiyon saldırısından korunmasına çözüm getirmiştir. Ayrıca yöneticilerin manuel olarak daha fazla türde saldırıya karşı güvenlik yöntemi eklemesi sağlanabilmektedir. Yapılacak araştırmalar için, önerilen çalışmanın kapsamının genişletilmesi tavsiye edilmiştir. Bu kapsamda, yönetici için bir bildirim sistemi ve raporlama gibi daha fazla özellik eklemesinin yanında NoSQL, OS (İşletim Sistemi) veya basit indeks erişim protokolü (Lightweight Directory Access Protocol - LDAP) enjeksiyonu gibi daha fazla çeşit enjeksiyon saldırısı için kapsamın genişletilmesi önerilmiştir. Erçin ve Yolaçan (2022), SQL enjeksiyonları (SQLi) ve siteler arası betik çalıştırma (Cross-Site Scripting - XSS) saldırılarının tespit edilmesi sırasında yaşanan yanlış alarm durumlarını önlemek, yanlış tespit oranınlarının azaltılmasına yönelik çalışma yapmışlardır. Çalışmada makine öğrenmesi ve yapay sinir ağı uygulamaları geliştirmişlerdir. Uygulamalarda kullanılan veri setinin ön işleme aşamasında kullanılacak bir yaklaşım üzerinde durulmuştur. Bu sayede kelimeler ile sembolize edilen verilerin yapılan saldırıların daha kolay fark edilmesini ve hesaplamada kolaylık sağlayarak başarıyı arttırdığı aktarılmıştır.

Priyawati, Rokhmah ve Utomo (2022), web uygulamalarında karşımıza çıkacak siber saldırılar veya bilgisayar korsanlarının yapacağı kötü niyetli girişimlere karşı güvenlik açığı düzeyini öğrenmek için test etmek gerekliliği üzerine durmuşlardır. Bir ağ üzerinden dağıtılan web siteleri için gri kutu sızma testi uygundur. Yapılan çalışmada OWASP ZAP aracını kullanarak gri kutu sızma testi tekniği uygulanmıştır. Uygulamada test hedefine uygun bilgilerinin toplanması, OWASP ZAP yardımıyla otomatik tarama yapılması, tarama sonuçlarının kullanılması ve raporlanması aşamaları gerçekleştirilmiştir. İncelenen çalışmaların özeti Tablo 3 ile verilmiştir.

Tablo 3. Enjeksiyon

Kaynak	Amaç	Yöntem
Latchoumi vd., 2020	SQL enjeksiyonunu tespit etmek ve önlemek	DVM algoritmasına sahip makine öğrenmesi
Harefa vd., 2021	SQL enjeksiyon saldırılarını tespit etmek ve önlemek	Web Uygulaması Güvenlik Duvarı (WAF)
Erçin ve Yolaçan, 2022	SQLi ve XSS saldırı tespitinde yanlış alarm durumunu önlemek ve tespit arttırmak	Makine öğrenmesi ve yapay sinir ağı algoritmaları
Priyawati vd., 2022	Web sitesindeki uygulama özelliklerinin güvenlik açığı düzeyini test etmek	OWASP ZAP yardımıyla otomatik gri kutu penetrasyon testi

1.4. Güvensiz Tasarım (Insecure Design)

Güvensiz tasarım, web sitelerinin tasarım sorunları ile ilgili risklere odaklanan ve OWASP tarafından 2021 yılında tanımlanmış yeni bir kategoridir. Güvensiz tasarım, tasarım ve mimari sorunları ile ilgili risklerle odaklanmanın yanında tehdit modelleme, güvenli tasarım kalıpları ve referans mimarilerin daha fazla kullanılması üzerinde duran kategoridir. Saldırganların hassas bilgiler içeren dosyalara ulaşabilmesini web sitesinde bulunan yolların görüntülenmesi ile sağlamaktadır. Katmanlı yapıda olan mimarilerde daha yüksek ayrıcalıklara sahip tarafa yönelik saldırılarda güvensiz tasarımdan yararlanılan yöntemler arasındadır (Aljabri vd., 2022). Güvensiz tasarım OWASP listesine 2021 yılında dahil olduğu için literatürde kısıtlı çalışmada yer almaktadır. Bu yüzden OWASP konusunda üzerinde çalışılmış güvensiz tasarım kavramını da içeren çalışmalar incelenmiştir.

Hidayat vd. (2022, s. 242-245), şehirlerde Nesnelerin İnterneti (Internet of Things - IoT) teknolojisinin uygulamasıyla gerçekleşen "Akıllı Şehir" kavramında cihazların güvenliği açısından incelemeler yapılmıştır. Bu sayede akıllı şehirde IoT oluşturmanın gerektirdiği altyapı, internet ağı, sensör düğümleri, veri depolamanın yanında güvenlik ve gizliliğin de önemi üzerinde durmuşlardır. Güvensiz tasarım kavramının da içinde bulunduğu OWASP'ın güvenlik açıkları ile ilgili çeşitli çalışmalar derlenmiş ve sonuçlar sistematik literatür tarama yöntemi ile sunulmuştur. Güvenlik ve altyapı konuları incelenerek, akıllı şehrin nasıl inşa edileceği konusunda yardımcı olması amaçlanmıştır.

Djeki vd. (2022), COVID-19 salgını dolayısıyla hızla artan uzaktan çalışma ve çevrimiçi öğrenme yöntemlerinin güvenlik riskleri üzerinde çalışmışlardır. Öğrenme yönetim sistemi ve video konferans araçları için dijital öğrenme alanlarının güvenlik konuları üzerinde durmuşlardır. Bu konular içerisinde güvensiz tasarım kavramı da yer almaktadır. Güvensiz tasarım sorunları sistem tasarımı sırasında güvenlik dikkate alınmadığı durumda ortaya çıkmaktadır. Bunun için yazılım geliştiricilere uygulamaları izlemeleri, test güdümlü geliştirme şeklini benimsemeleri, kullanıcı veya hizmet başına gerekli olan kaynak tüketimini kısıtlamaları, üretimde kötü kod dağıtımını önlemek için derinlemesine kod incelemeleri gerçekleştirmeleri ve bu konularda uygulama veya sistemler oluşturmaları önerilmiştir. Ayrıca, tasarım tamamlanmadan önce tehdit modellemesi ve sistem penetrasyon testi gerçekleştiren bir ekip kurulması ve dağıtımdan sonra da tespit edilen güvenlik sorunlarını azaltmak için düzenli olarak denetimler ve sızma testleri gerçekleştirilmesi gerektiği belirtilmiştir. İncelenen çalışmaların özeti Tablo 4 ile verilmiştir.

Tablo 4. Güvensiz Tasarım

Kaynak	Amaç	Yöntem
Hidayat vd., 2022	Akıllı şehir IoT uygulamalarında cihazların güvenliği	OWASP maddeleri konusunda sistematik literatür taraması
Djeki vd., 2022	Uzaktan çalışma ve çevrimiçi öğrenme yöntemlerinin güvenlik risklerini azaltma	OWASP maddelerini inceleyen literatür taraması

1.5. Yanlış Güvenlik Yapılandırması (Security Misconfiguration)

Web uygulamaları veya sunucularının yanlış yapılandırılması, güvenlik sorunlarına yol açtığı için web geliştiricileri açısından önemli bir risk oluşturmaktadır (Kumi, Lim, Lee, Oktian ve Witanto, 2021). Yanlış güvenlik yapılandırmaları, güvenlik ayarlarının düzgün bir şekilde tanımlanmadığı ve uygulanmadığı, varsayılan değerlerin korunmadığı durumlarda ortaya çıkmakta ve güvenlikle ilgili

ciddi sorunlara neden olmaktadır. Tecrübeli bir siber suçlu tarafından hatalı yapılandırılmış web sunucular ve istismar edilebilir hale gelen uygulamalar kolaylıkla tespit edilebilmektedir. Bu yüzden yanlış yapılandırılmış sistemler özellikle hedef haline gelmektedir (Loureiro, 2021, s. 13-16).

Martínez, Cosentino ve Cabot (2017), web uygulamalarının geliştirilmesi için sıklıkla tercih edilen Java EE (Java Enterprise Edition) çerçevesinde karşılaşılan yanlış yapılandırmaları tespit etmeye çalışmışlardır. Java EE erişim kontrol mekanizmaları sözdizimi ve semantiği konusunda uzman düzeyinde bilgi gerektiren karmaşık ve hataya açık bir uygulamadır. Bu yüzden, istenmeyen güvenlik ve/veya kullanılabilirlik sorunlarına yol açabilecek yanlış yapılandırmalar kolayca ortaya çıkabilmektedir. Çalışmada, tersine mühendislik yöntemi uygulanarak Java EE güvenlik yapılandırmalarında güvenlik özelliğini otomatik olarak değerlendiren ve anormalliklerin varlığını tespit etmeye yardımcı olan bir model yaklaşımı sunulmuştur. Oluşturulan model GitHub'dan elde edilen gerçek Java EE yazılımlarına uygulanarak etkililiği ve uygunluğu değerlendirilmiştir. Sonuç olarak gerçek projelerdeki güvenlik yapılandırmalarının tanımlanan özelliklerinin ihmal edildiği belirlenmiştir.

Kumi, Lim, Lee, Oktian ve Witanto (2021), web uygulamalarındaki yanlış güvenlik yapılandırmalarını tespit edecek bir araç önermişlerdir. Söz konusu araç bir web güvenlik şirketinde görev yapan güvenlik uzmanlarıyla işbirliği yapılarak geliştirilmiştir. BitScanner olarak isimlendirilen araç, platform ve teknolojiden bağımsız olarak tüm web uygulamalarındaki yanlış yapılandırma sorunlarını etkili şekilde tanımlamaktadır. Web geliştiricilerinin geliştirme senaryolarını gerçek hayatta uygulamadan önce herhangi bir yanlış yapılandırma sorununu fark etmesini ve düzeltmesini sağlamaktadır. Sonuç olarak önerilen aracın yüksek algılama kapasitesine sahip olduğu ve yanlış pozitifleri önlediği görülmüştür.

Smith (2022), gerçekleştirdiği doktora tezi çalışmasında Amerika Birleşik Devletleri'ndeki yanlış güvenlik yapılandırmaları riskini azaltmak için Bilgi Teknolojileri (BT) uzmanları tarafından kullanılan strateji temalarının belirlenmesini amaçlamıştır. Çalışmada, nitel araştırma yaklaşımı, tematik analiz ve kartopu örneklemesi yöntemleri uygulanmıştır. On dokuz katılımcıyla açık uçlu görüşmeler sağlanarak gerçekleştirilen çalışmada, sistem ve ağ cihazlarının yanlış güvenlik yapılandırmalarını önlemek için BT uzmanlarının operasyonel süreçlere bakış açıları araştırılmıştır. Çalışmanın sonucunda yöneticilerin güvenlik yapılandırmalarını uygulamadan önce kontrol etme ve doğrulama aşamalarında sorun yaşadıkları görülmüştür. Ayrıca, güvenlik yapılandırmalarında amaçlanan sonuçların dikkate alınmadığı, ilgili görevin çalışan motivasyonu ve iş performansı kavramları açısından yanlış kurgulandığı ve aynı kuruluştaki diğer ekiplerle bilgi paylaşımında eksikliklik olduğu görülmüştür.

Rahman, Shamim, Bose ve Pandita (2023), Kubernetes uygulamalarında büyük ölçekli güvenlik ihlallerine neden olan yanlış güvenlik yapılandırmaları konusunda deneysel bir çalışma yapmışlardır. Kubernetes, Go programlama dili (Golang) ile geliştirilmiş olan bir konteyner kümeleme aracıdır. Konteyner kümelemede, konteyner haline getirilen tüm uygulamalar, otomatik olarak harekete geçirilir. Konteyner sayılarının artırılması ve azaltılması işlemleri ile uygulamanın yönetilmesi sağlanır. Çalışmada deneyde yer alan uygulayıcıların Kubernetes bildirimlerinde meydana gelen yanlış güvenlik yapılandırmaları belirlenerek kümelerinin güvenliğini sağlamak amaçlanmıştır. Yanlış güvenlik yapılandırmalarını sistematik olarak karakterize etmek için 92 açık kaynaklı yazılım havuzundan çıkarılan 2.039 Kubernetes bildirimi kullanılmıştır. Toplamda, 11 yanlış güvenlik yapılandırması kategorisi tanımlanmıştır. Sonuç olarak 2.039 bildirimde 1.051 yanlış güvenlik yapılandırması belirlenmiştir. İncelenen çalışmaların özeti Tablo 5 ile verilmiştir.

Tablo 5. Yanlış Güvenlik Yapılandırmaları

Kaynak	Amaç	Yöntem
Martínez vd., 2017	Java EE çerçevesinde karşılaşılan yanlış yapılandırmaları tespit etmek	Tersine mühendislik uygulaması ile model oluşturma
Kumi vd., 2021	Web uygulamalarındaki yanlış güvenlik yapılandırmalarını tespit edecek bir araç tasarımı	Güvenlik uzmanlarıyla işbirliği yapılarak BitScanner aracı önerilmiş
Smith, 2022	Yanlış güvenlik yapılandırmaları riskini azaltmak için BT uzmanlarının kullandığı strateji temalarını anlamak	Nitel araştırma yaklaşımı, tematik analiz ve kartopu örneklemesi
Rahman vd., 2023	Kubernetes uygulamalarında yanlış güvenlik yapılandırma kaynaklı bildirileri belirleme	Deneysel çalışma

1.6. Savunmasız ve Güncel Olmayan Bileşenler (Vulnerable and Outdated Components)

OWASP'a göre bir sistemi oluşturan bileşenlerinin tamamının sürümleri bilinmediğinde, yazılımların desteklenmediği veya güncel olmadığı durumlarda, güvenlik açıkları düzenli olarak taranmadığında, yazılım geliştiricileri tarafından güncellenen, yükseltilen veya yama olarak uygulanan kütüphanelerin uyumluluğu test edilmediği durumlarda o sistem savunmasız halde olmaktadır. Belirtilen bu nedenlerle savunmasız olan ve güncel hale getirilmeyen bileşenler sistemleri siber suçluların hedefi haline getirmektedir.

Lathifah, Amri ve Rosidah (2022), Endonezya, Asya Pasifik bölgesinde Finansal Teknoloji (FinTech) hizmetlerinde oldukça yüksek pazar gelişimine sahip fonlama siteleri başta olmak üzere çeşitli alanlardaki web sitelerinin kullanımında karşılaşılan sorunlar üzerinde durmuşlardır. Gerçekleştirilen çalışmada, Zed Attack Proxy (ZAP) aracı kullanılarak OWASP yaklaşımı ile bir fonlama web sitesinin güvenlik açıklarının analiz edilmesi amaçlanmıştır. Çalışmanın sonucunda, fonlama amacıyla kullanılan web sitesindeki güvenlik açığı düzeyi belirlenmiştir. Çalışma kapsamındaki test sonucunda elde edilen veriler ile fonlama amacıyla kullanılan bir web sitesinde güvenlik açıkları ve riskleri bulunmuş ayrıca web sitesinde güvenliği artırmaya yönelik öneriler sunulmuştur.

Galvão (2022), yaptığı çalışmada günümüzde özellikle ticari uygulamaların da dahil olduğu yazılım projelerinde, açık kaynak kütüphanelerin ve çerçevelerinin yaygın olarak kullanıldığı konusu üzerinde durmuştur. Açık kaynak kütüphanelerinin uygulama maliyetini düşürmesi ve geliştirme hızının artırılması açısından faydalı olmasına rağmen güvenlik açısından maliyetli olduğu vurgulanmıştır. Açık kaynak paketleri üzerine yapılan çalışmalarda güvenlik açıklarının sayısının her yıl arttığı ve saldırganların genellikle belirli uygulamaların kodu yerine yazılım tedarik zincirini hedef aldığı belirtilmiştir. Savunmasız ve güncel olmayan bileşenler, yazılım geliştiricilerin çalıştıkları projelerde bağımlı bölümlerin güvenliğini yönetme ve sürdürme gerekliliğini ortaya koymaktadır. Literatürde bunun için açık kaynaklı olarak özellikle kod düzeyinde bilgi içeren güvenlik açığı veritabanlarına dayanan birkaç güvenlik açığı tarayıcısı olduğu belirtilmiştir. Project-KB, kod düzeyi bilgilerinin bulunduğu veritabanı tarayıcılarından biridir. Her giriş, güvenlik açığını gideren kodların bir listesini içermektedir. Yapılan çalışmada, farklı güvenlik açığı veritabanlarının özellikleri, boyutu, kapsamı ve içerdikleri bilgilerin güvenilirliği açısından objektif bir analizi sunulmuş ve bu kaynaklardan bazıları kullanılarak Project-KB tarayıcısının geliştirilmesine katkı sağlanması amaçlanmıştır.

Shahid vd. (2022), internet kullanımının artışına paralel olarak hızla artan web uygulamalarının güvenlik açıklarını analiz etmişlerdir. İşletmeler, endüstriler, finans, eğitim kurumları tarafından tercih edilen ve kişisel olarak kullanılan web uygulamalarını savunmasız hale getiren ve dolayısıyla ilgili bilgi sistemlerinin gizlilik, bütünlük ve kullanılabilirliğini etkileyen birçok güvenlik sorunu üzerinde durulmuştur. Web uygulaması güvenlik açığı tarayıcısı, otomatik penetrasyon testi tekniklerini kullanarak web uygulama güvenliğini değerlendiren bir bilgisayar programı oluşturulmuştur. Bu güvenlik tarayıcıları zaman ve maliyeti azaltarak penetrasyon testi için gerekli kaynak ve test mühendislerine olan bağımlılığı ortadan kaldırmaktadır ancak web uygulamaların tamamını taramamak ve yanlış test sonuçları oluşturmak gibi çeşitli dezavantajları bulunmaktadır. Yapılan çalışmada, kasıtlı olarak tanımlanmış savunmasız uygulamalar ve bu uygulamaların kesinlik ve doğruluk düzeyleri test edilerek web uygulaması güvenlik açığı tarayıcılarının performansı değerlendirilmiştir. Web uygulamalarındaki gerçek güvenlik açıklarını tespit etmek amacıyla ilgili tarayıcının yetenekleri değerlendirilmiş ve karşılaştırılmıştır. İncelenen çalışmaların özeti Tablo 6 ile verilmiştir.

Tablo 6. Savunmasız ve Güncel Olmayan Bileşenler

Kaynak	Amaç	Yöntem
Lathifah vd., 2022	Fonlama amacıyla kullanılan web sitesinde güvenlik açıkları ve risklerinin belirlenmesi	Zed Attack Proxy (ZAP) aracı
Galvão, 2022	Veritabanlarının analizi ve Project-KB tarayıcısının geliştirilmesi	Eclipse Steady analiz aracı
Shahid vd., 2022	Web uygulaması güvenlik açığı tarayıcılarının analizi	Otomatik penetrasyon testi teknikleri

1.7. Tanımlama ve Kimlik Doğrulama Hataları (Identification and Authentication Failures)

Tanımlama ve kimlik doğrulama hatalarında, oturum yönetimi ve kimlik doğrulama sık karşılaşılan kavramlar arasındadır. Web uygulamalarında kimlik doğrulama hatalarında en sık kaba kuvvet (brute force) saldırısı gerçekleşmektedir. Bu saldırıdan korunmak için, kullanıcı kimlik bilgilerinin şifrelenmesi, bilgilerin otomatik doldurulmasına izin verilmemesi ve güçlü parola kullanımı sayesinde saldırıdan korunulabilir. Oturum yönetiminde, oturumu ele geçirme, oturum kimliği URL'i yeniden yazma, kimlik bilgisi doldurma, parola püskürtme ve kimlik avı saldırıları gerçekleşmektedir. Tanımlama ve kimlik doğrulama hatalarına karşı gerçekleştirilen saldırıları önlemek için çok faktörlü kimlik doğrulama (MFA) yöntemi kullanılmaktadır (Alahmad, Alkandari ve Alawadhi, 2022).

Son yıllarda web tabanlı uygulamaların sık kullanılması veri sızıntılarına sebep olmaktadır. Sisteme manuel olarak saldırı gerçekleştirmek zor olduğu için genellikle saldırılar internet ağı üzerinden yapılmaktadır. Febriana (2022), çalışan sistem üzerinde güvenlik açıklarını bulmak için Sızma Testi Yürütme Standardı (Penetration Testing Execution Standard - PTES)'na uygun ve geliştiriciler tarafından uyarlanan penetrasyon uygulaması gerçekleştirmiştir. Sonuç olarak, tanımlama ve kimlik doğrulama hataları, güvensiz tasarım ve yanlış yapılandırma olmak üzere üç güvenlik açığı kategorilendirilmiştir. Web 2.0 da yaşanan hızlı gelişme, hizmetlerin web üzerinden sunulması ve sosyal ağlar aracılığıyla genişleyen bilgi paylaşımı web uygulamalarının sıklıkla saldırıya uğramasına sebep olmuştur. Nadar, Chatterjee ve Jacob (2018), yaptıkları çalışmada web uygulamalarının tasarımı, yazılımı, yazılım geliştirme yaşam döngüsü boyunca güvenlik analizleri ve kontrolleri yapılması gerekliliği üzerinde durmuşlardır. Sınırlı kurallar ile aynı anda yalnızca bir saldırı algılayan mevcut sistemlerin karşılaştırılması sonucunda, siteler arası istek sahteciliği, bozuk kimlik doğrulama ve oturum yönetimi saldırısını algılayabilen gelişmiş bir algılama modeli önermişler.

Lakh, Nyemkova, Piskozub ve Yanishevskyi (2021), güvenlik açıkları, kimlik doğrulama hataları ve farklı kimlik doğrulama tipolojileri üzerinde çalışmışlardır. Tanımlama ve kimlik doğrulama hatalarının yanı sıra kırık erişim kontrolü hataları da araştırılmıştır. Öncelikle özet ve temel kimlik doğrulama kullanan sunucu yapılandırılmıştır. RESTful web sunucusunda yapılandırılmış olan HTTP Temel Kimlik Doğrulama kullanan bir kaynak için özet kimlik doğrulama ile birlikte kaba kuvvet saldırıları modellenmiştir. Uygulama için, ayrı bir RESTful kaynağının kullanımından oluşan ve kullanıcıların yapmış oldukları işlemlere yönelik bir bot sınıfı yazmışlardır. Kullanıcılardan otomatik talepler gerçekleştirebilen www.reddit.com web hizmetinde yazmış oldukları botun uygulanması sağlanmış ve kullanılmasının gerekliliği vurgulanmıştır. Aynı zamanda web uygulamalarının güvenlik açıklarını tespit etmek için öncelikle WebGoat ve Buggy Web Application (BWAPP) gibi araçların kullanımı önerilmiştir.

Web uygulamalarında kimlik doğrulamasına yönelik gerçekleştirilen saldırılar, kullanıcının açmış olduğu oturumda aktif iken iletişiminin kesilmesi ya da yetkisiz bir kullanıcının sisteme erişmek için aynı oturumu seçmesi ile gerçekleşir. Bu olayın sonucu sadece kimlik hırsızlığı değil aynı zamanda kişiye ait gizli bilgilerin silinmesi ve değiştirilmesine de sebep olmaktadır. Hassan vd. (2018), Bangladeş'teki kamu ve özel sektöre ait 267 web sitesinde tanımlama ve kimlik doğrulama güvenlik açığı hatalarının etkileri analiz etmişlerdir. Çift kör test stratejisinin ardından manuel penetrasyon testi yöntemi kullanılarak yapılan incelemede, web sitelerinin %56'sı belirtilen zayıflıklara karşı savunmasız bulunmuştur. İncelenen çalışmaların özeti Tablo 7 ile verilmiştir.

Tablo 7. Tanımlama ve Kimlik Doğrulama Hataları

Kaynak	Amaç	Yöntem
Nadar vd., 2018	Siteler arası istek sahteciliği, bozuk kimlik doğrulama ve oturum yönetimi saldırısı için algılama modeli	Simülasyon ve etkili test ortamı
Alahmad vd., 2022	Tanımlama ve kimlik doğrulama hatalarını örnekleme	Geliştiriciler ve ağ yöneticilerin MFA cihazlarını dağıtması ve yapılandırması
Febriana, 2022	Çalışan sistem üzerinde güvenlik açığı bulmak için testler yapılması	PTES'e uygun sızma testi
Lakh vd., 2021	Brute force saldırılarına karşı kaynak modeli	Kaba kuvvet saldırıları modellenmesi
Hassan vd., 2018	Örneklem üzerinde analiz yapılması	Çift kör test stratejisini izleyen manuel penetrasyon testi yöntemi

1.8. Yazılım ve Veri Bütünlüğü Arızaları (Software and Data Integrity Failures)

Yazılım ve veri bütünlüğü arızaları, bütünlük saldırılarına karşı korumasız kodlar ve altyapı ile ilgilidir. Bu kavram OWASP listesine 2021 yılında dahil olduğu için literatürde kısıtlı çalışmada yer almaktadır. Kovalenkinaitė (2023), eğitim kurumlarındaki 100 web sitesinde otomatik tarayıcıların kullanımına ilişkin bir çalışma yapmıştır. Yapılan çalışmada etki alanları arasına javascript kaynak dosyası ekleyerek OWASP ZAP üzerinde analiz gerçekleştirilmiştir. Her tarama için bulunan URL sayısı, kullanılan düğümlerin miktarı ve web uygulamasında yüksek, orta ve düşük risklerin analizi yapılmıştır. Analiz sonucunda güvenlik açıklarının listesini, her bir güvenlik açığının risk seviyesini, web uygulaması içerisindeki tekrarlanma sıklığını ve savunmasız olan belirli URL'leri istek için kullanılan yönteme göre açıklamıştır.

Priambodo, Rifansyah ve Hasbi (2023), nüfus belgesi oluşturma hizmetleri, erişim kayıt hizmetleri ve oturum açma özelliği işlevlerine sahip olan "XYZ" web sitesi üzerinde uygulama yapmışlardır. Uygulamada güvenlik açığının seviyesini ölçmek için kara kutu sızma testi yöntemi kullanılmıştır. Güvenlik açığı değerlendirilmesi işlemini; planlama, bilgi toplama, tarama, analiz ve rapor şeklinde dört aşamada gerçekleştirmişlerdir. Güvenlik açığı taramasında geniş kapsam elde edebilmek için Vega ve OWASP ZAP uygulamalarından yararlanılmıştır. Sonuçta, web uygulaması geliştiricilerin özellikle hizmet kullanılabilirliği kaybı ve veri sızıntılarına karşı bütünlüğü koruyabilmek için yapılan çalışmayı referans almaları gerekliliğine vurgu yapılmıştır. Konu hakkında incelenen çalışmaların özeti Tablo 8 ile verilmiştir.

Tablo 8. Yazılım ve Veri Bütünlüğü Arızaları Hataları

Kaynak	Amaç	Yöntem
Kovalenkinaitė, 2023	OWASP ZAP uygulaması üzerinde URL analizi	Etki alanları arasına javascript kaynak dosyası ekleme
Priambodo vd., 2023	"XYZ" web uygulamasının analizi	Kara kutu sızma testi yöntemi

1.9. Güvenlik Kaydı ve İzleme Arızaları (Security Logging and Monitoring Failures)

Bir sistemde uyarılar ve hatalar yetersiz veya hiç kayıt edilmediğinde, uygulamaların ve API günlüklerinin olağan dışı hareketleri izlenmediğinde, günlükler yalnızca yerel olarak depolandığında, uyarı eşikleri ve yanıt yükseltme süreçleri uygun veya etkili olmadığında, sızma testi ve dinamik uygulama güvenlik testi (DAST) araçları tarafından yapılan taramaların uyarıları dikkate alınmadığında gerçek zamanlı veya etkin saldırılar algılanamaz. Bu durumlar ciddi zaafiyetlere neden olmaktadır. Güvenlik kayıtlarının düzenli tutulması ve uyarı olayları fark edilerek bilgi sızıntısına karşı savunma mekanizması kurulmalıdır. İzleme zamanında oluşabilecek sorunların önüne geçilmelidir.

Grammatikis vd. (2021), yazılım tanımlı ağ oluşturma (SDN) teknolojisini kullanarak, akıllı şebeke esnekliğini geliştirici yeni bir mimari sunmuşlardır. SDN-microSENSE mimarisi olarak tanımlanan bu mimari üç katmandan oluşmaktadır. Birinci katmanı tüm akıllı şebekelerin risk düzeyini dinamik olarak değerlendirilmesi oluşturmaktadır. İkinci katman, güvenlik olaylarını tespit etmekte ve sınıflandırma yapmaktadır. Üçüncü katman ise, akıllı şebekelerin doğru çalışmasını sağlayarak potansiyel oluşabilecek tehditleri azaltmaktadır. SDN-microSENSE mimarisinin tüm katmanları yetkisiz girişleri tespit etmek veya yaşanabilecek ihlalleri en aza indirebilmek amacıyla SDN denetleyicisi ile birlikte etkileşim halinde çalışmaktadır.

Song ve Valls (2022), çalışmalarında web güvenliği ile ilgili daha az önem verilen, kritik bir alan olan IoT alanına odaklanmışlardır. Gerçekleştirilen çalışmada sunucu tarafında, saldırıların sunucudan istemciye gönderim aşamasında oluşan güvenlik zafiyetleri ile istemciden korunmasız sunuculara sızma yöntemlerinin analizi yapılmıştır. Konu hakkında incelenen çalışmaların özeti Tablo 9 ile verilmiştir.

Tablo 9. Güvenlik Kaydı ve İzleme Arızaları

Kaynak	Amaç	Yöntem
Grammatikis vd., 2021	SDN teknolojisi kullanarak, akıllı şebeke esnekliğini geliştiren yeni bir mimari tasarımı	Çok katmanlı SDN-microSENSE mimarisi tasarımı ve SDN denetleyicisi kullanımı
Song ve Valls, 2022	Sunucu üzerine düşük yük eklenerek, güvenlik açıklarının hızlı bir şekilde tespit edilmesini sağlamak	IoT ile web uygulamalarının güvenlik açıklarını sunucu şablonlama motorları aracılığı ile izleme

1.10. Sunucu Taraflı İstek Arızaları (Server-Side Request Forgery)

Sunucu taraflı istek arızaları (SSRF), web uygulamalarında bulunan bir güvenlik zafiyetidir. Örneğin, saldırgan hizmetlere URL aracılığıyla erişmek istediğinde, erişim izni verilmeyen sunuculardaki hizmetler için bir URL sağlar veya değiştirir. Al-talak ve Abbass (2021), yaptıkları çalışmada çeşitli SSRF saldırı türlerini incelemişler ve web uygulamalarının güvenliğinin nasıl sağlanacağını ifade etmişlerdir. Saldırıları tespit etmek ve azaltmak için makine öğrenimi teknikleri kullanmışlardır. Saldırıları tespit edebilen akıllı bir model oluşturmak için derin öğrenme tekniklerinden LSTM ağları uygulanmıştır. Oluşturulan derin öğrenme modelinin, modelin gücü ve SSRF saldırılarını tespit etme kabiliyetini gösteren 0,969'luk doğruluk oranına sahip olduğu sonucuna ulaşmışlardır.

Günümüzde bulut bilişim kullanımının artması SSRF saldırıları tehdidine karşı alınacak önlemleri zorunlu hale getirmiştir. Jabiyev, Mirzaei, Kharraz ve Kirda (2021), yaptıkları çalışmada SSRF saldırılarından korunmak için yeni bir savunma yaklaşımı önermişlerdir. 60'tan fazla SSRF güvenlik açığı bulunan rapor üzerinde yapılan analiz sonucunda geliştiricilerin bu güvenlik açığı hakkındaki farkındalıklarının yeterli düzeyde olmadığı görülmüştür. SSRF saldırılarından korunmak için, ters proxy uygulamasının işlevselliğini genişleterek bir prototip geliştirmişler ve birkaç savunmasız web uygulamasını incelemeye almışlardır. İncelemeler sonucunda SSRF saldırılarını performans kaybı olmadan önlemeyi başarmışlardır. İncelenen çalışmaların özeti Tablo 10 ile verilmiştir.

Tablo 10. Sunucu Taraflı İstek Arızaları

Kaynak	Amaç	Yöntem
Al-talak vd., 2021	SSRF saldırıları türleri incelemesi	Makine öğrenimi teknikleri kullanımı
Jabiyev vd., 2021	SSRF saldırılarından korunmak için yeni bir savunma yaklaşımı önerisi	Ters proxy'nin işlevselliğini genişleterek yeni bir prototip oluşturma

Sonuç

Web uygulamalarının güvenliğinde bulunan zafiyetleri belirleyen OWASP tarafından 2021 yılında yayınlanan 10 madde göz önüne alınarak uygulamalar üzerinde önlemler alınmaktadır. Bu önlemler sayesinde web uygulamaları daha güvenli hale gelmektedir. Bu çalışma kapsamında OWASP tarafından yayınlanan listede bulunan web açıklıkları ayrı ayrı incelenmiş ve en güncel çalışmalar analiz edilmiştir. Bu bağlamda web uygulamalarında bulunan açıklıklara yönelik sunulan çözüm önerileri listelenmiştir. Sonuç olarak, 2021 yılında yayımlanan listede güvensiz tasarım, yazılım ve veri bütünlüğü arızaları, sunucu taraflı istek arızaları maddeleri ilk defa yer almıştır. Özellikle yeni eklenen kategoriler ile birlikte kısıtlı olan kaynaklara katkı sağlamak amacıyla yapılan çalışma sonucunda aşağıdaki sonuçlara ulaşılmıştır.

- Literatürde yapılan çalışmalarda kırık erişim kontrolünün sağlanabilmesi için erişim denetimlerinin kapsamlı bir şekilde yapılması vurgulanmıştır. Erişim kontrolü için güvenlik modellerini analiz edecek yöntemler ve metodolojiler geliştirilmiştir.
- Kriptografik hata maddesi kapsamında oluşabilecek sorunların etkisi üzerinde durulmuştur. Yapılan çalışmalarda web uygulamalarında veri bütünlüğünü korumak için kimlik doğrulama algoritmaları ve veri kullanılabilirliğini sağlamak için de kriptografik protokollerin kullanılması gerektiği vurgulanmıştır.
- SQL enjeksiyon saldırıları web uygulamaları için ciddi bir tehdit oluşturmaktadır. İncelenen çalışmalarda enjeksiyon saldırıların fark edilmesi ve önlenmesi için filtre ve araçların tasarımının yapıldığı görülmüştür.
- Web sitelerinin tasarım sorunlarına odaklanan ve riskleri azaltmayı amaçlayan güvensiz tasarım ile ilgili çalışmalar incelenmiştir. İncelenen çalışmalarda farklı alanlarda yapılan uygulamalarda tasarım sorunlarının önüne geçecek önerilere yer verilmiştir.
- Yanlış güvenlik yapılandırmaları, web uygulamalarında ve sunucu sistemlerinde ciddi zafiyetlere yol açabilmektedir. Bu tür yanlış yapılandırılmalar, kötü niyetli kişilerin güvenlik açıklarından yararlanmasına olanak tanımakta, çeşitli güvenlik ihlallerine neden olmakta ve uygulamanın tamamen ele geçirilmesine yol açabilmektedir. Literatürde hem web uygulamalarında hem de sunucu sistemlerinde karşılaşılan çalışmalar yer almaktadır.

- Sistem sürümleri güncel olmadığında ve düzenli olarak test edilmediğinde sistemler savunmasız hale gelmektedir. Savunmasız ve güncel olmayan bileşenler kavramına yönelik yapılan çalışmalara bakıldığında sistem açıklarını bulmaya yönelik araçların kullanımı üzerinde durulduğu görülmüştür.
- Oturum yönetimi ve kimlik doğrulama aşamalarında karşılaşılan güvenlik sorunlarına yönelik olarak tanımlama ve kimlik doğrulama hataları değerlendirilmektedir. Yapılan çalışmalarda oturum ve kimlik doğrulama sırasında yaşanacak zaafiyetlere karşı modelleme ve test yöntemleri sunulmuştur.
- Yazılım ve veri bütünlüğü arızaları, veri sızıntılarına karşı bütünlüğü koruyabilmek için savunmasız kodlar ve altyapıya odaklanmaktadır. Yapılan çalışmalarda sistemi oluşturan kodlar ve altyapıya yönelik analiz çalışmaları üzerinde durulmuştur.
- Güvenlik kaydı ve izleme arızaları, sistemde bulunan günlüklerdeki hatalar ve güvenlik tarama sonuçlarındaki sorunlar göz ardı edildiğinde gerçekleşmektedir. İncelenen çalışmalarda oluşan hata ve eksikliklerin fark edilmesi amacıyla yeni sistem tasarımları önerilmiştir.
- Sunucu taraflı istek arızaları web uygulamalarında kullanıcı tarafından hedef URL doğrulanmadığı durumlarda gerçekleşen saldırı türüdür. İncelenen çalışmalarda daha çok makine öğrenimi teknikleri kullanımı ve savunma yaklaşımları önerilmiştir.

2021 yılında listeye yeni eklenen güvensiz tasarım, yazılım ve veri bütünlüğü arızaları, sunucu taraflı istek arızaları maddeleri kapsamında kısıtlı olduğu görülen konularda çalışmaların yapılması ile literatüre katkı sağlanabilir. Web uygulamaları güvenliği için bu çalışmada incelenmiş olan kaynaklarda bulunan yöntemler uygulanmaktadır. Bu alanda yapılan teorik çalışmaların kapsamlı bir şekilde uygulanması ve yaygınlaştırılması için güvenlik eğitim programlarının arttırılması önerilmektedir.

KAYNAKÇA

ALAHMAD, M., ALKANDARI, A., & ALAWADHI, N. (2022). "Survey of Broken Authentication and Session Management of Web Application Vulnerability Attack.", *Journal of Engineering Science and Technology*, 17(2), 0874-0882.

ALENEZI, M., NADEEM, M., & ASIF, R. (2021). "SQL injection attacks countermeasures assessments.", *Indonesian Journal of Electrical Engineering and Computer Science*, 21(2), 1121-1131. doi: 10.11591/ijeecs.v21.i2.pp1121-1131

ALJABRI, M., ALDOSSARY, M., AL-HOMEED, N., ALHETELAH, B., ALTHUBIANY, M., ALOTAIBI, O., & ALSAQER, S. (2022, December). "Testing and Exploiting Tools to Improve OWASP Top Ten Security Vulnerabilities Detection.", In *2022 14th International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 797-803). IEEE. doi: 10.1109/CICN56167.2022.10008360

AL-TALAK, K., & ABBASS, O. (2021). "Detecting Server-Side Request Forgery (SSRF) Attack by using Deep Learning Techniques.", *International Journal of Advanced Computer Science and Applications*, *12*(12). 1-7. doi: 10.14569/IJACSA.2021.0121230.

AYDIN, H., BARIŞKAN, M. A. & ÇETİNAYA, A. (2021). "Siber Güvenlik Kapsamında Enerji Sistemleri Güvenliğinin Değerlendirilmesi.", *Güvenlik Bilimleri Dergisi*, 10 (1), 151-174. doi: 10.28956/gbd.941801

AYDOĞDU, D. & GÜNDÜZ, M. S. (2016). "Web uygulama güvenliği açıklıkları ve güvenlik çözümleri üzerine bir araştırma.", *Uluslararasi Bilgi Güvenliği Mühendisliği Dergisi,* 1–7. doi: https://doi.org/10.18640/ubgmd.56836.

BACH-NUTMAN, M. (2020). "Understanding the top 10 owasp vulnerabilities.", *arXiv preprint* arXiv:2012.09960, 1-4, doi: https://doi.org/10.48550/arXiv.2012.09960

BARLETT, J. (2016). "Dark Net: İnternetin Yer Altı Dünyası.", *Konyalı, Y.(çev.). İstanbul: Timaş Yayınları, İstanbul*

DJEKI, E., DEGILA, J., BONDIOMBOUY, C., & ALHASSAN, M. H. (2022, April). "Preventive Measures for Digital Learning Spaces' Security Issues.", In *2022 IEEE Technology and Engineering Management Conference(TEMSCONEUROPE)*(pp.48-55).IEEE.doi:10.1109/TEMSCONEUROPE54743.2022.9801945

ERÇİN, M. S., & YOLAÇAN, E. (2022). "SQLi ve XSS Saldırı Tespitinde Kullanılan Yeni Bir Özellik Çıkarma Yöntemi.", *Uluslararası Bilgi Güvenliği Mühendisliği Dergisi*, 8(1), 1-11.

ESPINOZA, A. M., WOOD, R., FORREST, S., & TIWARI, M. (2022). "Back to the future: N-Versioning of Microservices.", In *2022 52nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks DSN,* s. 415-427. IEEE. doi: 10.1109/DSN53405.2022.00049.

FANG, Y., LI, Y., LIU, L. & HUANG C. (2018). "DeepXSS: Cross Site Scripting Detection Based on Deep Learning.", *In Proceedings of the 2018 International Conference on Computing and Artificial Intelligence*, 47–51. doi: https://doi.org/10.1145/3194452.3194469.

FEBRIANA, R. (2022). "Blackbox Testing Sistem Informasi Absensi Pegawai Karawang Dengan Metode Top 10 Owasp Attack.", *Jurnal Ilmiah Wahana Pendidikan*, *8*(12), 327-334. doi: https://doi.org/10.5281/zenodo.6945632

GALVAO, P. L. (2022). "Analysis and Aggregation of Vulnerability Databases with Code-Level Data. (master thesis)", *Faculdade De Engenharia Da Universidade Do Porto,* Portugal

GARTNER, (2020). "Küresel Bilgi Güvenliği ve Risk Yönetimi Pazarı 2018-2024 Yılları Öngörü Raporu.", Erişim tarihi: 27 Ocak 2023, https://tubitak.gov.tr/sites/default/files/18842/btypk_siberguv_rapor_20211027.pdf

GRAMMATIKIS, P. R., SARIGIANNIDIS, P., DALAMAGKAS, C., SPYRIDIS, Y., LAGKAS, T., EFSTATHOPOULOS, G., & ARCE, A. (2021). "Sdn-based resilient smart grid: The sdn-microsense architecture.", *Digital*, 1(4), 173-187. doi: https://doi.org/10.3390/digital1040013.

GUPTA, C., SINGH, R. K., & MOHAPATRA, A. K. (2022). "An Approach for Verification of Secure Access Control Using Security Pattern.", *Wireless Communications and Mobile Computing*, 2022, 1-2, doi: https://doi.org/10.1155/2022/1657627

HAREFA, J., PRAJENA, G., ALEXANDER, A. M., DEWA, E. V. S., & YULIANDRY, S. (2021). "Sea waf: The prevention of sql injection attacks on web applications.", Advances in Science. *Technology and Engineering Systems*, 6, 405-411. doi: 10.25046/aj060247

HASSAN, M. M., NIPA, S. S., AKTER, M., HAQUE, R., DEEPA, F. N., RAHMAN, M., & SHARIF, M. H. (2018). "Broken authentication and session management vulnerability: a case study of web application.", *Int. J. Simul. Syst. Sci. Technol*, *19*(2), s.1-11.

HIDAYAT, M. F., QUTHNI, A. D., DEFRIN, J. T., GAPILI, G., MONIAGA, J. V., & JABAR, B. A. (2022, November). "Infrastructure and Security for Supporting Smart City: A Systematic Literature Review.", In *2022 2nd International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS)* (s. 242-245). IEEE. doi: 10.1109/ICE3IS56585.2022.10009974

HUANG, Y., LI, Y. J., & CAI, Z. (2023). "Security and Privacy in Metaverse: A Comprehensive Survey.", *Big Data Mining and Analytics*, *6*(2), 234-247. doi: 10.26599/BDMA.2022.9020047.

JABIYEV, B., MIRZAEI, O., KHARRAZ, A., & KIRDA, E. (2021). "Preventing server-side request forgery attacks.", In *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, s. 1626-1635. doi: https://dl.acm.org/doi/10.1145/3412841.3442036

JEMAL, I., CHEIKHROUHOU, O., HAMAM, H., & MAHFOUDHI, A. (2020). "Sql injection attack detection and prevention techniques using machine learning.", *International Journal of Applied Engineering Research*, 15(6), 569-580.

KARA, İ. (2020). "Web Hackleme (Hacking) Saldirilari.", *Ejovoc (Electronic Journal of Vocational Colleges)*, 10, 1-6.

KARACAN, H. & SEVRİ, M. (2021). "A Novel Data Augmentation Technique and Deep Learning Model for Web Application Security.*", IEEE Access*, 9, s. 150781-150797

KARAKAYA, M. (2022). "Kurumsal güvenlik için siber tehditlerin incelenmesi ve saldırı senaryoları", s.15-21.

KOVALENKINAITE, G. K. (2023). "*Vulnerability testing and analysis of educational institution websites within lithuania* (Doctoral dissertation, Vilniaus Universitetas).", s. 1-10.

KUMI, S., LIM, C., LEE, S. G., OKTIAN, Y. O., & WITANTO, E. N. (2021). "Automatic Detection of Security Misconfigurations in Web Applications.", *In Proceedings of International Conference on Smart Computing and Cyber Security: Strategic Foresight, Security Challenges and Innovation (SMARTCYBER 2020)* (pp. 91-99). Springer Singapore.

LAKH, Y., NYEMKOVA, E., PISKOZUB, A., & YANISHEVSKYI, V. (2021). "Investigation of the Broken Authentication Vulnerability in Web Applications.", In *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, Vol. 2, s. 928-931. IEEE. doi: 10.1109/IDAACS53288.2021.9660889.

LATCHOUMI, T. P., REDDY, M. S., & BALAMURUGAN, K. (2020). "Applied machine learning predictive analytics to SQL injection attack detection and prevention.", *European Journal of Molecular & Clinical Medicine*, 7(02), 1-11.

LATHIFAH, A., AMRI, F. B., & ROSIDAH, A. (2022, September). "Security Vulnerability Analysis of the Sharia Crowdfunding Website Using OWASP-ZAP.", In 2022 10th International Conference on Cyber and IT Service Management (CITSM) (pp. 1-5). IEEE. doi: 10.1109/CITSM56380.2022.9935837

LIANG, J., ZHAO, W. & YE, W. (2017). "Anomaly-based Web Attack Detection: A Deep Learning Approach", *In Proceedings of the 2017 VI International Conference on Network, Communication and Computing*, s.80-85.

LOUREIRO, S. (2021). "Security misconfigurations and how to prevent them. *Network Security"*, 2021(5), 13-16. doi: 10.1016/S1353-4858(21)00053-2

MANIKANTA, Y. V. N., & SARDANA, A. (2012, August). "Protecting web applications from SQL injection attacks by using framework and database firewall.", In *Proceedings of the International Conference on Advances in Computing, Communications and Informatics*, s. 609-613.

MARTINEZ, S., COSENTINO, V., & CABOT, J. (2017). "Model-based analysis of Java EE web security misconfigurations.", *Computer Languages, Systems & Structures*, 49, 36-61.

MONGA, M., PALEARI, R., & PASSERINI, E. (2009, May). "A hybrid analysis framework for detecting web application vulnerabilities.", In *2009 ICSE Workshop on Software Engineering for Secure Systems*, s. 25-32. IEEE.

NADAR, V. M., CHATTERJEE, M., & JACOB, L. (2018). "A defensive approach for CSRF and broken authentication and session management attack.", In *Ambient Communications and Computer Systems*, 577-588. Springer, Singapore. doi: https://doi.org/10.1007/978-981-10-7386-1_49.

PRIYAWATI, D., ROKHMAH, S., & UTOMA, I. C. (2022). "Website Vulnerability Testing and Analysis of Website Application Using OWASP.", *International Journal of Computer and Information System (IJCIS)*, 3(3), 142-147. doi: https://doi.org/10.29040/ijcis.v3i3.90

PRIAMBODO, D. F., RIFANSYAH, A. D., & HASBI, M. (2023). "Penetration Testing Web XYZ Berdasarkan OWASP Risk Rating.", *Teknika*, *12*(1), 33-46. doi: https://doi.org/10.34148/teknika.v12i1.571.

RAHMAN, A., SHAMIM, S. I., BOSE, D. B., & PANDITA, R. (2023). "Security Misconfigurations in Open Source Kubernetes Manifests: An Empirical Study.", *ACM Transactions on Software Engineering and Methodology*. doi: https://doi.org/10.1145/3579639

SAVUNMA SANAYİİ BAŞKANLIĞI, (2020). BTYPK tarafından 08.06.2020 tarihli Resmi Yazı ile talep edilen 2009-2020 Nisan sonu arasında destek kararı verilen projelere ilişkin görüş yazısı (Belirtilen projeler, Ar-Ge projesi, Teknoloji Kazanım Yükümlülüğü Projesi ve Sanayii Katılımı/Offset (SK/O) KATEGORİ-C ve Hizmet Projesi kapsamlarında desteklenmiştir.)

SCHOLTE, T., BALZAROTTI, D., & KIRDA, E. (2012). "Have things changed now? An empirical study on input validation vulnerabilities in web applications.", *Computers & Security*, *31*(3), s. 344-356.

SMITH, K. J. (2022). "Exploring Information Technology Professional's Perspectives on Controlling Security Misconfigurations in the United States: A Generic Qualitative Inquiry (Doctoral dissertation)", *Capella University, United States of America*. s. 15-22.

SHAHID, J., HAMEED, M. K., JAVED, I. T., QURESHI, K. N., ALI, M., & CRESPI, N. (2022). "A Comparative Study of Web Application Security Parameters: Current Trends and Future Directions.", *Applied Sciences*, 12(8), 4077. doi: https://doi.org/10.3390/app12084077

SONG, L., & GARCIA-VALLS, M. (2022). "Improving Security of Web Servers in Critical IoT Systems through Self-Monitoring of Vulnerabilities.", *Sensors*, 22(13), 5004. doi: https://doi.org/10.3390/s22239501.

OWASP. (2023). Erişim tarihi: 15 Ocak 2023, https://owasp.org/www-project-top-ten/. 2022

TEKEREK, A. (2021). "A Novel Architecture for Web-Based Attack Detection Using Convolutional Neural Network.", *Computers & Security*, 100, 102096.

OWASP Top 10:2021. (2021). Erişim Tarihi: 22 Şubat 2023, https://owasp.org/Top10/

TÜBİSAD-DELOITTE, 2021, Bilgi ve İletişim Teknolojileri Sektörü 2020 Pazar Verileri, Erişim: https://www.tubisad.org.tr/tr/images/pdf/tubisad_bit_2020_raporu_tr.pdf, Eylül, 2021.

TORRANO-GIMENEZ, C., NGUYEN, H. T., ALVAREZ, G., PETROVIC, S. & FRANKE, K. (2011). "Applying Feature Selection to Payload-Based Web Application Firewall.", *In 2011 Third International Workshop on Security and Communication Networks (IWSCN),* s. 75-81.

VAN DER POEL, L. (2022). "Towards automated discovery of access control vulnerabilities (master thesis)", Delft University of Technology, Sweden, s. 19-36.

VARTOUNI, A. M., TESHNEHLAB, M. & KASHI, S. S. (2019). "Leveraging Deep Neural Networks for Anomaly-Based Web Application Firewall.", *IET Information Security,* 13(4), s.352-361.

WICHERS, D. Owasp top-10 2013. OWASP Foundation, Erişim: https://owasp.org/www-project-top-ten Ocak, 2023.

Summary

Through web applications, needs such as information sharing, banking transactions, online chatting, sharing, socializing and communicating are realized online. Thanks to these opportunities it offers, it can meet the needs of individuals. In this context, the internet, which allows many transactions to be performed, has become the target of attackers because of the information it contains. Attackers target system vulnerabilities in applications used in the web environment. By using these vulnerabilities, they can access the system and use the information they obtain for their own purposes. With the rapid increase of security vulnerabilities in web applications, The Open Web Application Security Project (OWASP), a non-profit organization, was established in 2001 to provide support and information to application developers. Thanks to this organization, the most harmful and most common vulnerabilities in web application security were identified and listed under the name OWASP 10.

In this study, the top ten items published by OWASP in 2021, which includes the list of the most common web vulnerabilities today, were analyzed. In this context, qualitative research method was adopted and each item was investigated in detail, and the purpose, method and recommendations were included. Within the scope of the study; what are the most common web vulnerabilities today, what are the studies and methods used in web applications, and what are the recommendations given in the literature on common web vulnerabilities?

The top 10 web application security vulnerabilities in the current list published by OWASP in 2021 are as follows; broken access control, cryptographic errors, injection, insecure design, incorrect security configuration, vulnerable and outdated components, identification and authentication errors, software and data integrity failures, security logging and monitoring failures, server-side request failures. As a result, the list published in 2021 included insecure design, software and data integrity failures, and server-side request failures for the first time. As a result of the study conducted to contribute to the limited resources, especially with the newly added categories, the following results were reached. It is emphasized that access controls should be done in a comprehensive manner to ensure broken access control. Methods and methodologies have been developed to analyze security models for access control. In order to provide cryptographic error control, it is emphasized that authentication algorithms should be used to protect data integrity and cryptographic protocols should be used to ensure data availability in web applications. SQL injection attacks pose a serious threat to web applications. In the studies examined, it was seen that filters and tools were designed to detect and prevent injection attacks. Regarding the insecure design clause, studies in different field.