In The Name Of God

بسمه تعالی

Vulnerability Exploitation Guide

راهنمای اکسپلویت آسیب پذیری ها

English Sections

بخش فارسی

==========

IRODay.Today Bax

Telegram : t.me/LearnExploit

Author : XOP4SH4

==========

List of vulnerabilities that have been investigated:

لیست آسیب پذیری هایی که مورد بررسی قرار گرفته است :

1. path traversal
2. business logic
3. single sign-on (SSO) Misconfiguration
4. Insecure Deserialization

English Section

1. Path Traversal – Detailed Analysis and Exploit

1.1 - Overview:

Path Traversal occurs when user input is not properly validated, allowing unauthorized access to sensitive files using directory traversal characters (../).

1.2 - Vulnerable PHP Code:

```
<?php
$file = $_GET['file'];
include("uploads/" . $file);
?>
```

1.3 - Exploit Example:

```
https://example.com/download?file=../../../../etc/passwd
```

1.4 - Potential Impact:

- Access to sensitive server files
- Disclosure of database credentials
- Leak of API keys

1.5 – Fix :

```
<?php
$file = basename($_GET[‘file’]);
if (3trops($file, ‘..’) === false) {
     include(“uploads/” . $file);
} else {
     echo “Access Denied!”;
}
?>
```

---

2. Business Logic Flaw – Detailed Analysis and Exploit

2.1 - Overview:

Business logic flaws occur when attackers can bypass restrictions by manipulating logical conditions in the application.

2.2 - Vulnerable PHP Code:

```
<?php
$user_id = $_POST['user_id'];
$discount_code = $_POST['discount_code'];
if($discount_code == "WELCOME50") {
     $discount = 50;
}
?>
```

2.3 - Exploit Example:

```
POST /checkout
{
     "user_id": 9999,
     "discount_code": "WELCOME50"
}
```

2.4 – Fix:

```
 <?php
 $user_id = $_POST[‘user_id’];
 $discount_code = $_POST[‘discount_code’];
 $used_codes = getUsedCodesFromDB($user_id);
 if($discount_code == “WELCOME50” && !in_array($discount_code, $used_codes)) {
     $discount = 50;
 } else {
     echo “Invalid Code!”;
 }
?>
```

---

3. SSO Misconfiguration – Detailed Analysis and Exploit

3.1 - Overview:

JWT (JSON Web Token) misconfiguration allows attackers to manipulate the token and gain unauthorized access.

3.2 - Malicious JWT:

```
eyJhbGciOiJub25lIiwidXNlciI6ImFkbWluIn0=
```

3.3 - Exploit:

```
{
   "alg": "none",
   "user": "admin"
}
```

3.4 – Fix:

- Use secure algorithms like RS256
- Validate the token signature
- Implement proper key management

---

4. Insecure Deserialization – Detailed Analysis and Exploit

4.1 - Overview:

Insecure deserialization occurs when untrusted serialized data is processed without validation, allowing remote code execution.

4.2 - Vulnerable PHP Code:

```
<?php
$data = $_GET['data'];
$user = unserialize($data);
if ($user->role == "admin") {
     echo "Welcome Admin!";
}
?>
```

4.3 - Exploit (Malicious Object):

```
O:4:"User":1:{s:4:"role";s:5:"admin";}
```

4.4 - Fix:

```
$data = json_decode($_GET['data'], true);
if ($data['role'] == "admin") {
    echo "Welcome Admin!";
}
```

بخش فارسی

1. Path Traversal

۱/۱ - شرح کامل آسیب‌پذیری:

Path Traversal زمانی رخ می‌دهد که برنامه ورودی کاربر را برای انتخاب مسیر فایل به درستی اعتبارسنجی نمی‌کند و مهاجم می‌تواند با استفاده از کاراکتر ../ به مسیرهای خارج از دایرکتوری مجاز دسترسی پیدا کند.

۱/۲ - نمونه کد PHP آسیب‌پذیر:

```
1. <?php
2. $file = $_GET['file'];
3. include("uploads/" . $file);
4. ?>
```

۱/۳ - اکسپلویت:

```
https://example.com/download?file=../../../../etc/passwd
```

۱/۴ - اثرات احتمالی:

- دسترسی به فایل‌های حساس سرور
- استخراج فایل‌های کانفیگ دیتابیس
- افشای کلیدهایAPI

۱/۵ - روش رفع باگ:

```
<?php
$file = basename($_GET['file']);
if (strpos($file, '..') === false) {
          include("uploads/" . $file);
          } else {
                    echo "Access Denied!";
          }
?>
```

2. Business Logic Flaw

۱/۲ - شرح کامل آسیب‌پذیری:

این نقص زمانی رخ می‌دهد که مهاجم بتواند با دستکاری منطق کسب‌وکار، محدودیت‌هایی مثل تخفیف یکبار مصرف یا احراز هویت را دور بزند.

۲/۲ - نمونه کد PHP آسیب‌پذیر:

```php
<?php
$user_id = $_POST['user_id'];
$discount_code = $_POST['discount_code'];
if($discount_code == "WELCOME50") {
    $discount = 50;
}
?>
```

۳/۲ - اکسپلویت:

```
POST /checkout
{
  "user_id": 9999,
  "discount_code": "WELCOME50"
}
```

۴/۲ - روش رفع باگ (اصلاح کد سمت سرور):

```php
<?php
$user_id = $_POST['user_id'];
$discount_code = $_POST['discount_code'];
$used_codes = getUsedCodesFromDB($user_id);
if($discount_code == "WELCOME50" && !in_array($discount_code, $used_codes)) {
    $discount = 50;
} else {
    echo "Invalid Code!";
}
?>
```

3. SSO Misconfiguration (JWT Exploit)

۱٫۳ - شرح کامل آسیب‌پذیری:

JWT (JSON Web Token)توکن‌هایی هستند که شامل اطلاعات احراز هویت و مجوزهای کاربر هستند. ضعف در اعتبارسنجی امضای دیجیتال منجر به جعل توکن می‌شود.

۲٫۳ - JWTساختگی:

```
eyJhbGciOiJub25lIiwidXNlciI6ImFkbWluIn0=
```

۳٫۳ - اکسپلویت (تغییر alg به none)

```
{
  "alg": "none",
  "user": "admin"
}
```

۴٫۳ - روش رفع باگ:

- استفاده از الگوریتم‌های امن مانندRS256
- اعتبارسنجی توکن در سمت سرور
- بررسی کامل امضا با کلید عمومی/خصوصی

4. Insecure Deserialization

۱/۴ - شرح کامل آسیب‌پذیری:

وقتی که ورودی‌های سریال‌شده بدون اعتبارسنجی دی‌سریال‌شده و اجرا می‌شوند، مهاجم می‌تواند کد مخرب اجرا کند.

۲/۴ - نمونه کد PHP آسیب‌پذیر:

```php
<?php
$data = $_GET['data'];
$user = unserialize($data);
if ($user->role == "admin") {
    echo "Welcome Admin!";
}
?>
```

۳/۴ - اکسپلویت (شیء مخرب):

```
O:4:"User":1:{s:4:"role";s:5:"admin";}
```

۴/۴ - روش رفع باگ:

```php
$data = json_decode($_GET['data'], true);
if ($data['role'] == "admin") {
    echo "Welcome Admin!";
}
```


IRODay.Today Bax || Telegram : t.me/LearnExploit || Author : XOP4SH4