April, 2020

SESSION HIJACKING

LIFARS
your digital world, **secured**

Contents

244 Fifth Avenue, Suite 2035, New York, NY 10001
LIFARS.com (212) 222-7061 info@lifars.com

OVERVIEW

LIFARS frequently conducts penetration tests to ensure the effectiveness of our client's security implementations and to evaluate whether their systems can hold up to real world incident scenarios and stay resilient. Our cyber resiliency experts deliver calculated attacks against systems the same way black hat hackers.

In April, our client requested LIFARS Pen Testing Team to perform an external black box penetration test as a part of a due diligence exercise. The client understands the risks they are daily facing as well as the importance of meeting compliance standards. Therefore, this client asked for an external black box penetration test on their website.

The intent of this engagement was to identify weaknesses in the company's website and to detail how these vulnerabilities could impact the organization.

Therefore, the team used Session Hijacking as a main target for mounting other attacks. This security testing effort was conducted with emphasis on the actual state of the systems examined and no documentation to the client was provided.

Note: All information in this case study has been modified to maintain confidentiality of our client

SESSION HIJACKING

Session hijacking is a technique used to take control of another user's session and gain unauthorized access to data or resources.

Because http communication uses many different TCP connections, the web server needs a method to recognize every user's connections. The most useful method depends on a token that the Web Server sends to the client browser after a successful client authentication. A session token is normally composed of a string of variable width and it could be used in different ways, like in the URL, in the header of the http requisition as a cookie, in other parts of the header of the http request, or yet in the body of the http requisition.

DISCOVER

Our first step was doing detailed enumeration and analysis on client's website. We were spidering directories and files using Burp Suite, dirbuster and dirb. After we had done this phase, we were scraping files (mostly JavaScript) to uncover additional URLs. Focused mainly on possible post-authentication URLs, we have found some of them.

244 Fifth Avenue, Suite 2035, New York, NY 10001
LIFARS.com (212) 222-7061 info@lifars.com

We started to replay found URLs and found some interesting behavior.

```
GET /welcome HTTP/1.1
Host: .com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0)
Accept: text/html,application/xhtml+xml,application/xml
Connection: keep-alive
Referer: /welcome/index.html
Cookie:
Upgrade-Insecure-Requests: 1
```

Figure 1 request

```
HTTP/1.1 302 Redirect
Location: /welcome/index.html
Connection: Keep-Alive
Content-Length: 139
Cache-Control: no-store, no-cache, must-revalidate
X-Frame-Options: DENY
Content-Security-Policy: script-src 'self'
X-XSS-Protection: 1; mode=block
Strict-Transport-Security: max-age=31536000; includeSubDomains
Set-Cookie: username=
Set-Cookie: role
Set-Cookie: AN nav
```

Figure 2 Response

In this case, the server itself disclosed the full session information of another authenticated user.

During testing, we have successfully obtained username and session cookies of logged in users and gained access without knowing the credentials. Sessions of multiple users could be highjacked by requesting the information at different times.

IMPACT

Attacker could connect to the service as a random authenticated user. During the active session, they could perform any actions that the original user is authorized to do, like accessing protected services.

One particular danger for larger organizations is that cookies can also be used to identify authenticated users in single sign-on systems (SSO). This means that a successful session hijack can give the attacker SSO access to multiple web applications, from financial systems and customer records to line-of-business systems potentially containing valuable intellectual property.

For individual users, similar risks also exist when using external services to log into applications, but due to additional safeguards when you log in using your Facebook or Google account, hijacking the session cookie generally won't be enough to hijack the session.

REFERENCES

[1] OWASP, "Session hijacking attack" Available: https://owasp.org/www-community/attacks/Session_hijacking_attack

244 Fifth Avenue, Suite 2035, New York, NY 10001
LIFARS.com (212) 222-7061 info@lifars.com