One of the most common ways of hacking someone's data is by impersonation. Cross-Site Scripting or XSS attack is a way in which the attacker steals the session cookies. Thus, they start impersonating their victim and get access to their information. Not only this, but XSS attacks can also result in malware, network worms, and deface the website. Attackers sometimes also use cross-site scripting and social engineering hack side by side which incurs more damage.

There are three major categories of XSS attacks:

## Stored XSS

Stored XSS or persistent XSS is one of the most dangerous attacks. The attackers inject malicious content into the system of their victim such as JavaScript codes. To worsen the issue, if your system does not have an input validation check then the attacker's code will permanently reside in it. When the victim opens this malicious webpage or application, they will end up running it in their software. XSS payload serves the victim just like a regular HTML code would do. Thus, they are unaware of their functionality.

## Reflected XSS

Reflected XSS or non-persistent XSS is the most common form of XSS attacks. As the name suggests, this cross-site scripting attack works through the mechanism of reflection. Unlike stored XSS, non-persistent XSS works by sending the attacker's payload in the webserver request. Afterwards, the HTTP response is reflected in such a way that it carries an image of the HTTP request protocol. Social engineering techniques come in handy here as they help to lure the victim into requesting the server. Sometimes attackers also use phishing emails or malicious links to lure the victim. As it is a non-persistent attack so the attacker sends a separate payload to each victim.

## DOM Based XSS

An advance form of XSS attack is the DOM (Document Object Model) based XSS. This attack is held on those servers which use DOM to store data provided by the users. This data is frequently read and fed to the browser. However, the incorrect handling of data can result in the injection of the payload. Thus, the payload resides in the DOM and executes every time data is read from the latter. It is a client-side attack which is why it is difficult to detect it by web application firewalls.

Cross-site scripting is nothing new but the increasing vulnerabilities of XSS attacks are dangerous. If you would like to know more about how you can protect yourself, get in touch with Aardwolf Security today.