AJAX - Security

AJAX is the most commonly used web technique to send and receive data to and from the web server asynchronously without disturbing the functionality of the other components of the client-side application. Although AJAX itself does not provide any security vulnerabilities, still we have to keep some security measurements while implementing AJAX. The security measurements are –

Cross-Site Scripting(XSS) – AJAX applications should be vulnerable to XSS attacks. If proper input validation and output encoding are not implemented, then a hacker can easily inject malicious scripts inside the AJAX response. These malicious scripts are used to steal sensitive data from the system or can manipulate the content. So always create an AJAX application which is safe from this attack using proper validation and sanitization before displaying data on the web page.

Cross-Site Request Forgery(CSRF) – In this attack, the attacker tricks the browser by doing unwanted actions with the help of an authentication session. It can exploit the AJAX request and can perform unauthorized actions. So to prevent this attack we have to implement CSRF protection techniques like generation and validating random tokens Or can use the same origin policy.

Insecure Direct Object References(IDOR) – The request generally accesses the specified resource from the server with the help of a unique identifier. But if the attacker gets this identifier then it can easily manipulate or can access unauthorized resources. So to prevent this avoid exposing sensitive information. Also, check the user authorization for the specified resource of the developers, in the server side.

Content Security Policies(CSP) – It is a policy which helps users/developers to save themselves from malicious activities or unauthorized access. It provides a permitted source for secure scripts and other resources.

Server-Side validation – Server-side validation is very important because it ensures that the submitted data meets the specified criteria and it is safe for further process. We can not bypass or manipulate server-side validation but we can bypass client-side validation.

Secure Session Management – The AJAX application should properly maintain user sessions and session tokens to save the session from attacks. Always check that the session tokes are generated properly, and securely transmitted and can logout if the invalidation or session expiration happens.

Input Validation and Sanitization – Server should perform validation and sanitization of the data received from the client side to prevent attacks.

Regular Update and Security – As we know that AJAX uses external libraries or frameworks. So keeping them up to date is an important task. To avoid various vulnerabilities and improve the security of the application.

Conclusion

So while creating an AJAX application always remember these points for security purposes to save your application from attacks. Now in the next article, we will the major issues faced by AJAX.