SEMINAR

ABSTRACT

TOPIC : OWASP Zed Attack Proxy(ZAP)

Submitted by,
Karthika Suresh Babu
S10,INMCA

Zed Attack Proxy(ZAP) is a free,open-source penetration testing tool. It is maintained under the umbrella of Open Web Application Security Project(OWASP). ZAP is designed specifically for testing web applications and is both flexible and extensible.

At its core, ZAP is what is known as a "man-in-the-middle proxy." It stands between the tester's browser and the web application so that it can intercept and inspect messages sent between browser and web application, modify the contents if needed, and then forward those packets on to the destination. It can be used as a stand-alone application, and as a daemon process.

ZAP improves the security of software and web applications.The web security vulnerabilities are prioritized depending on exploitability, detectability and impact on software.

Some of the vulnerabilities as per OWASP are SQL Injection,Cross Site Scripting,Broken Authentication and Session Management,Insecure Direct Object References,Cross Site Request Forgery,Security Misconfiguration,Insecure Cryptographic Storage,Failure to restrict URL Access,Insufficient Transport Layer Protection,Unvalidated Redirects and Forwards.

- ➔ SQL Injection : Injection occurs when the user input is sent to an interpreter as part of command or query and tricks the interpreter into executing unintended commands and gives access to unauthorized data.
- ➔ Cross Site Scripting : Attackers can use XSS to execute malicious scripts on the users in this case victim browsers. Since the browser cannot know if the script is trusty or not, the script will be executed, and the attacker can hijack session cookies, deface websites, or redirect the user to unwanted and malicious websites.
- ➔ Broken Authentication and Session Management : The websites usually create a session cookie and session ID for each valid session, and these cookies contain sensitive data like username, password, etc. When the session is ended either by logout or browser closed abruptly, these cookies should be invalidated i.e. for each session there should be a new cookie.If the cookies are not invalidated, the sensitive data will exist in the system. For example, a user using a public computer (Cyber Cafe), the cookies of the vulnerable site sits on the system and is exposed to an attacker. An attacker uses the same public computer after some time, the sensitive data is compromised.
- ➔ Insecure Direct Object References : It occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, or database key as in URL or as a FORM parameter. The attacker can use this

information to access other objects and can create a future attack to access the unauthorized data.

- ➔ Cross Site Request Forgery : CSRF attack is an attack that occurs when a malicious website, email, or program causes a user's browser to perform an unwanted action on a trusted site for which the user is currently authenticated.
- ➔ Security Misconfiguration : Security Configuration must be defined and deployed for the application, frameworks, application server, web server, database server, and platform. If these are properly configured, an attacker can have unauthorized access to sensitive data or functionality.
- ➔ Insecure Cryptographic Storage : Insecure Cryptographic storage is a common vulnerability which exists when the sensitive data is not stored securely. The user credentials, profile information, health details, credit card information, etc. come under sensitive data information on a website. This data will be stored on the application database. When this data are stored improperly by not using encryption or hashing*, it will be vulnerable to the attackers.
- ➔ Failure to restrict URL Access : Web applications check URL access rights before rendering protected links and buttons. Applications need to perform similar access control checks each time these pages are accessed.In most of the applications, the privileged pages, locations and resources are not presented to the privileged users. By an intelligent guess, an attacker can access privilege pages. An attacker can access sensitive pages, invoke functions and view confidential information.
- ➔ Insufficient Transport Layer Protection : Deals with information exchange between the user (client) and the server (application). Applications frequently transmit sensitive information like authentication details, credit card information, and session tokens over a network. Using weak algorithms or using expired or invalid certificates or not using SSL can allow the communication to be exposed to untrusted users, which may compromise a web application and or steal sensitive information.
- ➔ Unvalidated Redirects and Forwards : The web application uses few methods to redirect and forward users to other pages for an intended purpose. If there is no proper validation while redirecting to other pages, attackers can make use of this and can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.