Cookie Based Authentication Vulnerabilities

- Cookie based session identification is a way to allow application to provide authentication mechanism for it's users.
- However, these days Token based authentication is being widely used over Cookie Based Authentication due to a large attack surface that Cookie Based Authentication leaves open.
- Everyone working in the domain of penetration testing or application security is familiar to cookies and how they are being used. Let's talk about some of the vulnerabilities that are often found in Cookie Based Authentication.

XSS in Cookies Parameter

- Assume that the value of the cookie parameter “Name” Is reflected in in the apllication

- Change the “Name” value to XSS payload, its may result into XSS

Sensitive data Stored in cookie

- Cheak if any PII other sensitive information stored cookies
- This information usually includes Email, Date of birth, Mobile Number, Address etc

Insuffcient Session Mangment

- Check does the session doesn't expire on logout
- Replace the old cookie with new cookie , if it working then its vuln...

Or

- Check for Session Doesn't Expire on Password Reset/change

IDOR in Cookie

- If the cookie are using some access defining parameter Such as “user-id”....

- Change the value of these parameter in order to check If you can access other user’s data

Authentication Bypass (Cookies are not validate)

- Try accesssing a protected by removing cookies

Privilege Escalation

a. Horizontal Privilege Escalation

i. Assume that the application uses Multi-Organization Model.

ii. Cookies are used to define which organization a user can access.

iii. Alter the Cookies in order to Access some other Organization.

b. Vertical Privilege Escalation

i. Assume that the cookies are used to determine the "Role" of the User.

ii. Alter the Cookies in order to Elevate the "Role" of the User.

Cookie Based SQL Injection

- Try to inject SQLI query and check for SQLI
- First try (‘ or “ or /) observe if there any error
- Check for blind SQLI (or ‘1’=’1--) or (‘ and 1=1--)
- Check for time based SQLI

All this you can check in Cookie parameter

Denial of Service

- Forcing the server to process cookies larger than the restricted cookie size defined by the server may cause Denial of Service Attack.

EXAMPLE -:

Set-Cookie:AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Or

Set-Cookie:000000000000000000000000000000000000000000000000000

Parameter Pollution

- Assume that the cookies utilize a parameter called "user_id=" to retrieve some data.
- However, the application is not vulnerable to IDOR and changing "user_id=" to victim value, doesn't help out.
- Attacker, add an additional "user_id=" parameter value to the cookie with victim's user ID. Like: "user_id=attacker&user_id=victim"
- Three things can happen here:

  - The application may retrieve data of Victim User.

  - The Application may retrieve data of both Attacker & Victim User.

  - The Application is not vulnerable and doesn't return anything.