**DAY 24**

XSS – Cross Site Scripting

Google dorks: To find website which might have xss vulnerability

Client-Side:

* Html
* JS
* CSS

Sever-Side:

* PHP
* aspx
* jsp

google dorks:

inurl:search.php

inurl:/search.php

inurl:search.jsp

inurl:/search.jsp

inurl:search.aspx

inurl:/search.aspx

Search for dynamic pages:

i.e. “custom input” in html code

Server validate the user input:

i.e. < , > symbols are eliminated

URL encoding to mitigate XSS

Note:

To try for XSS following rules:

1. The given input should be reflected back

a) can be in the same input field

b) can be anywhere in the website

c) can be in the URL

🡪”><script>alert(123)</script>

Preventive measures:

1. Filter input on arrival

2. Using URL encoding

3. Using HTML encoding

4. Using appropriate response headers

5. Using content security policy

X-XSS-PROTECTION:1 ; mode=block in HTML response header

Solution:

NOSCRIPT: This is browser extension which will restrict the execution of javascript functions in the website you visit

Noscript.net

**SQL INJECTION:**

Structured Query Language

Few database languages are SQL, TSQL, PostgreSQL, MySQL

Userid= ‘=’‘OR’

password= ‘=’‘OR’

Google dorks: inurl:login.php or jsp or aspx