**React**

**Day 17th**

**JSX**

**What is XSS attack??**

Cross – site scripting (also known as XSS) is a web security vulnerability that allows an attacker to compromise the interactions that users have with a vulnerable application. It allows an attacker to circumvent the same origin policy, which is designed to segregate different websites from each other. Cross – site scripting vulnerabilities normally allow an attacker to masquerade as a victim user, to carry out any actions that the user is able to perform, and to access any of the user’s data. If the victim user has privileged access within the application, then the attacker might be able to gain full control over all of the application’s functionality and data.

**How does it work??**

Cross – site scripting works by manipulating a vulnerable webs site so that it returns malicious JS to users. When the malicious code executes inside a victim’s browser, the attacker can fully compromise their interaction with the application.

📝 JSX prevents this so we don’t need to worry about it. It doesn’t allow it to inject any malicious data.

Our react based app safer from XSS (Cross site Scripting attacks). React DOM escapes any value before rendering them on the screen which means that whatever value can be malicious code or attackers XSS attacks give in the input field will be converted into spring by JSX. This is the way how it prevents XSS attacks.

**Naming Conventions**

Always JSX elements or react elements should be in lower case like tagname. React engine easily distinguish which tag is an html tag or custom component or node otherwise some weird things can happen in react application. If it is in uppercase then it will say it is not defined. In HTML you can use because that’s a markup language and here JSX is not a markup language and it is case sensitive.

Built in HTML tags in JSX represents in lower & case user defined custom components in capital case.

Always use PascalCase or TitleCase means first letter or character must be a capital letter for react components or lowercase for HTML.

Whenever we are calling a component it needs to be enclosed in angular braces or we can go like opening tag and closing tag. Call the component with the same name as you defined.

Inside JSX expression you can’t use loops or if else statements instead of loops you can go with higher order function you can use `**map, filter**` and for condition you can use ternary conditional operator.

Inside JSX expression you can’t use only Boolean value, null value, undefined value.

