Cross Site Scripting (XSS)

COMPUTER NETWORKING

Time: 60 mins

Introduction

In this class, the student(s) will understand the basics of cross site scripting(XSS) attacks, underlying vulnerabilities, and preventive measures.

New Commands Introduced

- alert('Hello!'): Displays a pop-up alert on the browser.
- document.createElement('script'); Creates a new script element in the HTML document.
- document.head.appendChild(script); Appends the created script element to the head of the HTML document.
- navigator.userAgent: Retrieves the user agent string, providing information about the user's browser.
- getBrowserName(): Returns the name of the user's browser.
- getBrowserType(): Returns the type or category of the user's browser.
- getBrowserVersion(): Returns the version number of the user's browser.
- getCookies(): Retrieves and returns the cookies associated with the current document.
- geolocationResult:: Returns the geographical location of that device, including GPS data
- getCurrentTime(): Returns the current time.
- getScreenSize(),: Returns the size of the user's screen
- getOperatingSystem(): Returns the operating system of the user's device.

Vocabulary

- **Cross Site scripting(XSS)** is a type of security vulnerability involving injection of malicious client-side scripts into trusted web pages viewed by other users.

- **JavaScript injection** is a subtype of cross site scripting (XSS) that involves the ability to inject arbitrary JavaScript code that is executed by the application inside the victim's browser.

Learning Objectives

Student/s should be able to:

- ***Demonstrate*** how to find out the possibility of a cross site scripting attack on a web page.
- ***Explain*** how an XSS attack is implemented.
- ***Describe*** preventive measures to avoid XSS attacks.

Activities

Class Narrative: *(3 mins)*

- Brief the student(s) about XSS attacks and how these exploit vulnerabilities in web pages by injecting malicious scripts through input fields(like feedback form, contact form), URLs, or even cookies to steal user information.

Concept Introduction Activity: *(4 mins)*

- Let the student/s observe that the script submitted in the feedback form, takes user info like user location, screen info, browser data, and cookies and sends it to the hacker dashboard.
- Explain how Cross-Site Scripting (XSS) attacks are performed, and common types of XSS(JavaScript injection) attacks.
- Using the slides, explain that the student/s will learn:
    - to show the alert on the browser.
    - to get the user data on the dashboard.
    - to protect against XSS attacks.

Activity 1: Show the Alert on the Browser *(16 mins)*

Teacher Activity: *(8 mins)*

- Introduce most Common XSS attacks, which can occur by injecting malicious scripts through input fields, URLs, or even cookies.
- Demonstrate how the XSS attack is possible on web pages.

Student Activity: *(8 mins)*

- Guide the student/s to find out how the XSS attack is possible on web pages.

Activity 2: Get the User Data on the Dashboard *(10 mins)*

- Explain limitations to submitting some characters in the text input box, in such case, we can host the script somewhere, and then inject an external script from the specified URL into the feedback form.
- Explain how we will create a hacker dashboard to host the script and collect the data.
- Demonstrate how to collect the browser data like browser name, type, version, cookies, screen sizes, and location.

  Student Activity: *(10 mins)*

- Guide the student/s to collect the browser data and display it on the hacker dashboard.

Activity 3: Protect Against XSS *(12 mins)*

- Explain several strategies for preventing XSS attacks and utilise the two approaches listed below to avoid XSS assaults:
  - Using innerText instead of innerhtml
  - Using Jinja to render and display data as text:
- Explain how we will prevent XSS attacks by treating user input as plain text.

  Student Activity: *(6 mins)*

- Guide the students to prevent the XSS attacks by treating user input as plain text.

Introduce the Post class project: *(2 min)*

- Perform the XSS attack to show an advertisement on the webpage.

Test and Summarize the class learnings: *(5 mins)*

- Check for understanding through quizzes and summarize learning after respective activities.
- Summarize the overall class learning towards the end of the class.

Additional activities:

- Encourage the student/s to gain a deeper understanding of XSS attacks by practicing an attack on a dummy website.
- Encourage the student/s to use and promote preventive measures against XSS attacks on the websites they create.

State the Next Class Objective: *(1 min)*

- In the next course, you will be introduced to the Internet of Things (IoT) and its applications.

U.S. Standards:

CSTA: 2-AP-11, 2-AP-12, 2-AP-13, 2-AP-14, 2-AP-19

Links Table		
Activity	Activity Name	Link
Class Presentation	Cross Site Scripting	https://s3-whjr-curriculum-uploads.whjr.online/0fcbcb27-9d55-4c49-9a32-4447b0f71fd3.html
Explore Activity	Cross Site Scripting (XSS)	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS-BP
Teacher Activity 1	Show the Alert on the Browser	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-TAS-BP
Teacher Reference: Teacher Activity 1 Solution	Show the Alert on the Browser	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-TAS
Student Activity 1	Show the Alert on the Browser	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS-BP
Teacher Reference: Student Activity 1 Solution	Show the Alert on the Browser	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS
Teacher Activity 2	Get the User Data on the Dashboard	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-TAS-BP
Teacher Reference: Teacher Activity 2 Solution	Get the User Data on the Dashboard	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-TAS
Student Activity 2	Get the User Data on the Dashboard	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS-BP
Teacher Reference: Student Activity 2 Solution	Get the User Data on the Dashboard	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS
Teacher Activity 3	Protect Against XSS	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-TAS-BP
Teacher Reference: Teacher Activity 3 Solution	Protect Against XSS	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-TAS
Student Activity 3	Protect Against XSS	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS-BP
Teacher Reference: Student Activity 3 Solution	Protect Against XSS	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS

Student's Additional Activity 1	Check the Website for Vulnerability	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS-BP
Teacher Reference: Student's Additional Activity 1 Solution	Check the Website for Vulnerability	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS
Student's Additional Activity 2	Protect Against XSS	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS-BP
Teacher Reference: Student's Additional Activity 2 Solution	Protect Against XSS	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-SAS
Post Class Project	XSS Advertisment	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-PCP-BP
Teacher Reference: Post Class Project Solution	XSS Advertisment	https://github.com/Tynker-Computer-Networks/TNK-M16-C128-PCP