What are cookies and sessions, and how do they relate to each other?

Ask Question

51

40

I am trying to understand cookies and sessions professionally. I know that when a browser connects to a server, the server "asks" the browser to "paste" a cookie with "phpsessid" in the client browser cookies folder.

Now that we have the "phpsessid", if the client enters the server the browser sends to the server the "phpsessid" and the server takes a look at the tmp folder and if we have a match it loads back every data the user has for this client, but I am kinda confused with the process.

I will be thankful if some one can help me understand those processes of creating a session and cookies - what is happening behind the scenes.

javascript php session cookies

edited Apr 17 at 16:05

Makoto

84.3k 18 133 184

asked Jun 21 '12 at 16:45

Blanktext

554 2 7 16

Home

PUBLIC

Stack Overflow

Tags

Users

Jobs

It seems you already know what's happening. Which specific part would you like to be enlightened on? The HTTP/cookie part, or how PHP loads the session store? – mario Jun 21 '12 at 16:52

The cookie that the browser is saving with the phpsessid, is this the phpsessid to identify the client for creating sessions and cookies? – Blanktext Jun 21 '12 at 16:54

The cookie the browser has contains the php session id, which tells the server "hey I know this chap", the server then grabs the session data (from /tmp/ for example) and restores the user $_SESSION – somedev Jun 21 '12 at 16:59

Yes i know that, but i am asking, is this "phpsessid" is using to identify the client before creating sessions and cookies, is this phpsessid is including inside the cookie data and the client computer and inside the session file at the server tmp folder? – Blanktext Jun 21 '12 at 17:06

1 Answer

127

Let's go through this:

Cookies and **sessions** are both ways to preserve the application's state between different requests the browser makes. It's thanks to them that, for instance, you don't need to log in every time you request a page on **StackOverflow**.

Cookies

Cookies are small bits of data, (maximum of 4KB long), which hold data in a key=value pairs:

```
name=value; name2=value2
```

These are set either by **JavaScript**, or via the server using an **HTTP header**.

Cookies have an expiry datetime set, example using HTTP headers:

```
Set-Cookie: name2=value2; Expires=Wed, 19 Jun 2021 10:18:1
```

Which would cause the browser to set a cookie named `name2` with a value of `value2`, which would expire in about 9 years.

Cookies are considered **highly insecure** because the user can easily manipulate their content. That's why you should **always validate cookie data**. Don't assume what you get from a cookie is necessarily what you expect.

Cookies are usually used to preserve login state, where a username and a special hash are sent from the browser, and the server checks them against the database to approve access.

Cookies are also often used in **sessions** creation.

Sessions

Sessions are slightly different. Each user gets a **session ID**, which is sent back to the server for validation either by **cookie** or by **GET variable**.

Sessions are usually short-lived, which makes them ideal in saving temporary state between applications. Sessions also expire once the user closes the browser.

Sessions are considered more secure than cookies because the variables themselves are kept on the **server**. Here's how it works:

1. Server opens a session (sets a cookie via HTTP header)
2. Server sets a session variable.
3. Client changes page
4. Client sends all cookies, along with the session ID from step 1.
5. Server reads session ID from cookie.
6. Server matches session ID from a list in a database (or memory etc).
7. Server finds a match, reads variables which are now available on `$_SESSION` superglobal.

If PHP does not find a match, it will start a new session, and repeat the steps from 1-7.

You can store sensitive information on a session because it is kept on the server, but be aware that the session ID can still be stolen if the user, let's say, logged in over an insecure WiFi. (An attacker can sniff the cookies, and set it as its own, he won't see the variables themselves, but the server will identify the attacker as the user).

That's the gist of it. You can learn more on the PHP manual on both subjects.

edited Sep 23 '18 at 11:37

Ashish Patel
142 2 14

answered Jun 21 '12 at 17:11

Madara Uchiha ♦
121k 44 217 266

Thank you so much i am reading now. – Blanktext Jun 21 '12 at 17:12

1 @Edmund: On most cases, yes. You can, however, set it so that session IDs are transmitted via GET variable (which would mean that the session_id would need to be appended to every page (`hello.php?sid=cbe709ac7bed98f7ecb89713`) – Madara Uchiha ♦ Jul 22 '13 at 19:51

3 Good explanation. *once the user closed her browse*, I am just wondering why her? – Mohammed Noureldin Dec 7 '17 at 22:27

2 @SurajJain It would, if it weren't for "remember me" cookies, that persist longer. – Madara Uchiha ♦ Dec 28 '17 at 8:48

1 @MohammedNoureldin *her* shows gender equality, being fair, instead of using him all the time. – CP3O May 24 '18 at 7:50