[How do PHP sessions work?](http://stackoverflow.com/questions/1535697/how-do-php-sessions-work-not-how-are-they-used)

|  |  |  |
| --- | --- | --- |
|  | In the general situation :   * the session id is sent to the user when his session is created. * it is stored in a cookie (called, by default, PHPSESSID) * that cookie is sent by the browser to the server with each request * the server (PHP) uses that cookie, containing the session\_id, to know which file corresponds to that user.   The data in the sessions files is the content of $\_SESSION, serialized *(ie, represented as a string -- with a function such as*[*serialize*](http://php.net/serialize)*)* ; and is un-serialized when the file is loaded by PHP, to populate the $\_SESSION array.  Sometimes, the session id is not stored in a cookie, but sent in URLs, too -- but that's quite rare, nowadays.  Can sessions work without cookies?  Sessions in PHP normally do use cookies to function. But, PHP sessions can also work ***without*** cookies in case cookies are disabled or rejected by the browser that the PHP server is trying to communicate with.  How PHP sessions work without cookies  PHP does two things in order to work without cookies:  1. For every HTML form that PHP finds in your HTML code (which of course can be part of a PHP file), PHP will automatically add a hidden input tag with the name PHPSESSID right after the <form> tag. The value of that hidden input tag would be whatever value PHP assigns your session ID. So, for example, the hidden input could look something like this:  <form>  <input type="hidden" name="PHPSESSID" value="12345678" >  </form>   |  | | --- | |  |   This way, when the form is submitted to the server, PHP will be able to retrieve the session identifier from the form and will know who it is communicating with on the other end, and will also know which session to associate the form parameters with if it is adding the form parameters to the PHP session.  2. PHP will find all the links in your HTML code, and will modify those links so that they have a GET parameter appended to the link itself. That GET parameter will also have the name of PHPSESSID, and the value will of course be the unique session identifier – so the PHP session ID will basically be a part of the URL query string.  So, for example, if your code has a link that originally looks like this:  <a href="http://www.example.com">Go to this link><a/>  When modified by PHP to include the session ID, it could look something like this:  <a href="http://www.example.com?PHPSESSID=72aa95axyz6cd67d82ba0f809277326dd">Go to this link</>  PHPSESSID can have its name changed in php ini file  Note that we said PHPSESSID is the name that will be used to hold the PHP session value. The name PHPSESSID can actually be changed to whatever you want if you modify the session.name value in the php.ini file.  What is a disadvantage of using PHP sessions without cookies enabled?  A disadvantage is that using PHP sessions without cookies is the fact that if you share a URL that has the PHP session ID appended to it with someone else, then they could potentially use the same exact session that you were using. It also opens you up to session hijacking – where a user’s session is deliberately stolen so that a hacker can impersonate you and do some damage. |