Cookies

Cookies

While you may have heard that browser cookies track your actions on the Internet, that's not quite the case. Cookies are simply text files that are stored in Internet browsers, allowing publishers to better monetize their websites and offer users free content. They don't track everything that you're doing, they're not viruses, and they don't create spam. In fact, they can be quite useful to you. This page walks you through how cookies work in desktop browsers and mobile applications.

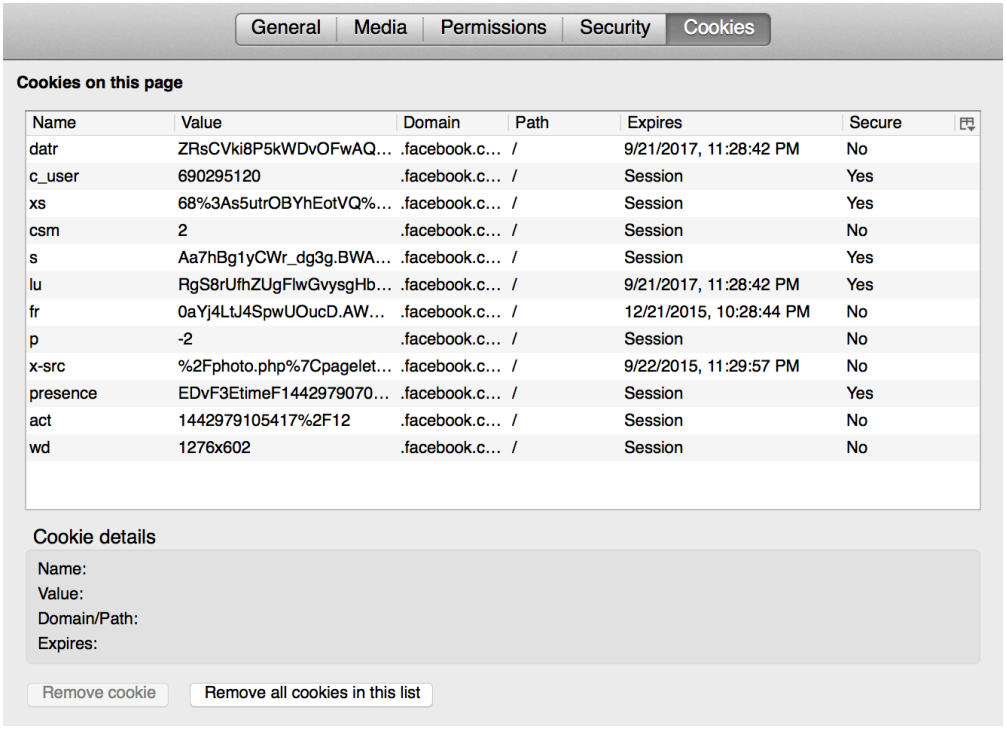
On This Page

How do Cookies Work for Desktop Browsers?

When your client sends an HTTP request to a server, the server has no way of distinguishing which client sends which requests. You may be wondering why servers don't use your IP address to keep track of who is sending which requests. IP addresses are not unique to one single user; they can be shared by multiple users. This often happens in offices. Because of IP sharing, cookies were developed to identify individual users and their actions.

Websites need to know that it's you (or your user account) every time that you send an HTTP request for data from a specified source. For example, when you sign into Facebook, you're assigned multiple cookies. One of these cookies is to keep you logged in to Facebook as yourself. If Facebook didn't know that it was you sending the HTTP GET request to see your friend's profile, it would not know whether or not you should be given access, and you would have to log in on every single page.

The screenshot below shows an example of what you'll see when you use the Firefox add-on, View Cookies. Within the list, one of the "session" cookies is what's most likely keeping the Facebook account logged in.



How are Cookies Set?

When you log in to Facebook, there are many HTTP requests and responses set in the short time it takes for you to navigate to your home page. The example below uses Tamper Data, another Firefox Add-On, to view the HTTP GET and POST calls sent when signed into Facebook. (HTTP GET calls requests data from a specified source, whereas POST calls submits data to be processed to a specified source). This is the first POST request received by the browser.

Response Header Name	Response Header Value
Status	OK - 200
content-security-policy	default-src * data: blob:;script-src *.facebook.com *.fbcdn.net *.facebook.net *.google-analytics.com *.virtualearth.net *.google.com…
public-key-pins-report-only	max-age=500; pin-pin-sha256="WoiWRyIOVNa9ihaBciRSC7XHjliYS9VwUGOIud4PB18="; pin-pin-sha256="r/mIkG3eEpVdm+u/k…
Cache-Control	private, no-cache, no-store, must-revalidate
Content-Encoding	gzip
Content-Type	application/x-javascript; charset=utf-8
Date	Mon, 28 Sep 2015 15:25:08 GMT
Expires	Sat, 01 Jan 2000 00:00:00 GMT
Pragma	no-cache
Strict-Transport-Security	max-age=15552000; preload
Vary	Accept-Encoding
x-content-type-options	nosniff
x-fb-debug	arTr16RHzZciRHFU37X7UF7yPMBqjhT3LatQVyA9rANuqH6FQMxokS8IV3QWFGz+SfNiBA0EDgDyYk4A1nGGCQ==
x-frame-options	DENY
X-XSS-Protection	0
X-Firefox-Spdy	3.1

Below, you can see the very next "PUT" request the browser sent, including the cookie.

Request Header Name	Request Header Value
Host	www.facebook.com
User-Agent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10.10; rv:33.0) Gecko/20100101 Firefox/33.0
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language	en-US,en;q=0.5
Accept-Encoding	gzip, deflate
Content-Type	application/x-www-form-urlencoded; charset=UTF-8
Referer	https://www.facebook.com/
Content-Length	288
Cookie	fr=0SfsUtDlqmVKGr6LG.AWVdT0z75N5OWsWpOyWV3u2xPCw.BWCVte.oI.FYJ.0.AWWOsTfW; lu=Rg7bzKTZEThmF7-F0mep6L5Q; s=Aa5kfrTG8qqnYyyx.…
Connection	keep-alive
Pragma	no-cache
Cache-Control	no-cache
POSTDATA	qe_name=test_flexible_targeting_new_ui_text&__user=690295120&__a=1&__dyn=aKTyBW8BgBlyi1p2uucKlGAy4y6zECQHyUmyVbGAGGi8VpdFLRGFoO8…

Your browser client can store hundreds of thousands of cookies. Each website has separate cookies and your browser client only sends a server the browser cookies that were generated by that server. For example, if you're on nytimes.com, nytimes.com doesn't have access to the Facebook.com browser cookie. And when you're on Facebook.com, Facebook.com doesn't have access to your nytimes.com cookie.

Cookies and Mobile Applications

A common misconception regarding mobile devices and the Internet is that cookies don't work on mobile devices. That's not exactly the case. However, it is true that cookies don't work in *all* mobile environments. There are limitations in functionality between mobile web and native applications. If you're accessing a desktop version of a website from a mobile device, then cookies work just fine. However, most often, you'll be accessing a mobile application on your phone or tablet, in which case, a different method is used to capture information. Device identifiers are leveraged, often designated specifically for advertising, to provide you relevant advertising across different native mobile applications. The most common ones are:

- IDFA (for iOS)
- AAID (for android)
- Windows advertising ID (for Windows)

Advertising SDKs are able to access device identifier information. However, because of the different tracking methodology for mobile web and mobile applications, each mobile environment of a single device will often have different user IDs.

For more information about targeting on mobile, see Introduction to Mobile Advertising.