

## Cookie Consent

### What is it for?

New in [Tiki10](#), to comply with "[EU Privacy and Electronic Communications Regulations](#)."

To allow a site to comply with the European regulation that a user must give positive (not passive) informed consent before a web site loads cookies onto their accessing device (PC, phone etc) a new optional Cookie Consent feature has been added to the Login set of admin screens. When set 'on' this enables the prevention of any cookies (except the session cookie) from being set and also lets the following be defined:

- the name of a new cookie that is loaded when consent has been given
- the duration of the consent cookie
- various parameters so that warning text and 'consent giving' dialogue can be displayed, and
- an Id can be defined so that the dialogue can be styled as required using css

The session cookie can be prevented for Anonymous users by configuring the 'Silent session' option in the 'Sessions' section of the 'Performance' admin screen - but as this means users will only have a session if they log in, this also means that anonymous users will lose the capacity to have things like Switch Language, since a session is needed for this. You therefore need to decide whether this additional cookie suppression make sense for you.

### Impact

It should be noted however that some actions by admins and users with edit permissions can bypass the cookie prevention e.g. setting Google Analytics or some social media functions with plugins, etc.; so admin/editor user education is required.

If some action like using Google Analytics is set by an admin in (say) the 'Customization' section of the 'Look & Feel' admin screen, then a check on the state of the custom consent cookie should be added as a wrap around the code used, like this:

```
{if $smarty.cookies.your_custom_cookie_consent_name eq 'y'}  
your code that might set a cookie  
{/if}
```

By contrast, the `{GOOGLEANALYTICS()}` plugin is Cookie consent aware. If Cookie Consent is active, the plugin will not log anything until the user accepts cookies. Do not be surprised if your Google Analytics figures go down and you miss the statistics of where users come from, etc.

Before cookies are accepted, there is also some change of behaviour in user modules (anyone who knows which changes, please update this part).

For any editor that might use some code in a Wiki page etc., that could set a cookie then they need to use a new 'wrap around' plugin called `COOKIECONSENT` which will prevent anything in the body of this new plugin from being executed if the user has

not set the consent cookie eg

```
{COOKIECONSENT()} any content that could set a cookie {COOKIECONSENT}
```

For more info check the [Plugin Site](#).

Note that the **Cookie consent disabled** feature (introduced in Tiki15) does not actually let the user refuse cookies but it shows the Cookie consent banner. This is for the cases where you wish to direct users to a third-party mechanism for cookie management, or if you wish to inform the user that the only way of avoiding cookies from your site is to not surf the site. Interpretations vary as to this option makes you still compliant with the EU directive, but anyway, people outside the EU may also want to inform users about their privacy policy.

## Improvements in Tiki24

New option in [Tiki24](#) to make the checkbox agree to third party cookies so users can agree to essential (Tiki) cookies without opting in to analytics or other "tracking" cookies from other providers (also in 23.1)

## More cookie control in custom look & feel, templates or modules

If you need to check for the newer "analytics" cookie to control what your "Custom code just before the closing

tag" pref for instance, in Smarty syntax, this should work (but as always, **test on a non-production site** as entering syntax errors in these preferences can break your site)

```
{if not empty($smarty.cookies[$prefs.cookie_consent_name|cat: '_analytics']) and  
$smarty.cookies[$prefs.cookie_consent_name|cat: '_analytics'] eq 'y'}{/if}
```

Alias names for this page:

[CookieConsent](#)