



## **Consent Mode V2:**

## An all-in-one course



## **How to implement Consent Mode V2**

## How to implement Consent Mode v2

The most important aspect of formulating a strategy is having clear parameters, which is what this unit is all about. In it, you'll learn about the famed Consent Mode parameters and the implementation of default settings. You'll also find out how to activate v2 in Usercentrics and Cookiebot, and how to test that it is working properly and seamlessly. Do you feel prepared to master the inner workings of Consent Mode v2? You absolutely should.

### **Module 1: The Consent Mode parameters**

- Consent Mode collects consents for specific categories or purposes of data processing. Each category has a "consent parameter" which are:
  - ad\_storage: To enable or disable cookies or other identifiers for advertising.
  - **analytics\_storage:** To enable or disable cookies or other identifiers for analytics.
  - ad\_user\_data: Sets consent for sending user data to Google for advertising purposes. "User data" means information like email addresses, phone numbers, etc. Introduced with v2.
  - ad\_personalization: Sets consent for personalized advertising. Introduced with v2.
- Each parameter is either set to "granted", meaning enabled or activated, or "denied", meaning disabled or deactivated.

## **Module 2: Implementation of default status**

• Before activating Consent Mode V2, you'll want to implement a default status. Consider it your back-up plan if no consent is given, especially when implementing "Advanced Mode":

```
<script>
window.dataLayer = window.dataLayer || [];
function gtag() {
    dataLayer.push(arguments);
}
gtag("consent", "default", {
    ad_personalization: "denied",
    ad_storage: "denied",
    ad_user_data: "denied",
    analytics_storage: "denied",
    wait_for_update: 2000
```



- Note: The script needs to be loaded before the CMP, Google Tag Manager, and other tracking scripts.
- For the Cookiebot CMP, it's also important to add this parameter to the script tag: datacookieconsent="ignore"
- If you have implemented your CMP through Google Tag Manager, you can also set these default values via your CookiebotTM or Usercentrics Tags:

#### Default Consent State ③ Statistics (analytics\_storage) Marketing (ad\_storage) Action Region (leave blank to Preferences (functionality\_storage and apply globally) personalization\_storage) Denied Denied Denied Edit Delete US-CA Granted Granted Granted Edit Delete 7.4 전에는 바람이 가슴이 Add region

- Set the default state for each parameter as needed. You can also apply a different default status for users from a specific geographical region by specifying the region code together with the default status.
- If you do this implementation through Google Tag Manager, you don't need to add the code block from the beginning of this video.

# Module 3: Activating Consent Mode V2 in Usercentrics and Cookiebot™

- For Usercentrics CMP:
  - Login to the Usercentrics Admin Portal
  - Open your Setting-ID
  - Click on "Configuration" and "CMP Settings"
  - Enable the "Google Consent Mode" Toggle:

#### **Google Consent Mode**

If enabled, the consent mode allows you to adjust how your Google tags behave based on the consent status of your users. For more details, visit the official documentation.

- Publish the change by clicking the "Publish" button
- For Cookiebot CMP:
  - If you have implemented the Cookiebot script directly on your website, Consent Mode should be enabled by default.
  - If you have implemented Cookiebot<sup>™</sup> via Google Tag Manager, make sure that this checkbox is selected.
  - For both CMPs, the Consent Mode signal will be automatically sent as soon as the Google Services have been added. For Cookiebot, it's automatically after the website scan, for Usercentrics, it's via the DPS database. In the Usercentrics CMP, you can also use "Custom Services" instead of the default ones to send Consent Mode signals by adding a small script to your website. Custom Services are data processing services with your own texts that are not from the Usercentrics database.

Tag Configuration	
Tag Type	
Cookiebot CMP cybotcorp	GALLERY
• Tag permissions	4 permissions >
Cookiebot ID ③	
Language ③	
Default (auto-detect)	
Add Geo Region(s) ⑦ Enable Google Consent Mode ⑦	
> TCF Framework	
> Advanced Settings	



• You don't need to make any changes to your tags in Google Tag Manager, but make sure that they are set to the "Not set" or "No additional consent required" default.

# Module 4: How to check if Consent Mode V2 is working properly

• First option is to use the Google Tag Manager Preview. Select the event that is used to trigger your tag in the sidebar and check the "Consent" tab.

	Su	mmary
•	<b></b>	Full data control thro
	22	consent_status
	21	Set
	20	user_engagement
	19	Consent
	18	Set
	17	Link Click
	16	Scroll Depth
	15	Scroll Depth
	14	Window Loaded 👩
	13	DOM Ready 🖸

ent: consent_sta	atus				
API Call					
dataLayer.push({ev	ent: "consent_s	tatus",})			
Output of GTM-MH	HMTWLV ⑦				
Tags	Variables	Data L	ayer	Consent	Errors
Event Consent State	?				
Туре	On-page Default	On-page Update	Current State		
ad_storage	Denied	Granted	Granted		
analytics_storage	Denied	Granted	Granted		
ad_user_data	Denied	Granted	Granted		
ad_personalization	Denied	Granted	Granted		

 Second option is to check the Data Layer. To do this, open your browser's Developer Tools and click on "Console." Enter "dataLayer" into the console and press the Enter key. You should see several "consent" events, one for the default status and at least one more for the update after the CMP interaction:

dataLayer			
<pre>&lt;· (21) [4 ents(3)</pre>	Arguments(3), {}, Arguments(3), Arguments(3), Arguments(3), {}, {}, {}, {}, {}, {}, {}, {}, {}, {}, {}, {}, Arguments(3), Argum ), Arguments(3), {}, {}, push: f] i	m	
<b>▼ 0:</b> Ar	rguments(3)		
0:	"consent"		
1:	"default"		
▶ 2:	{ad user data: 'denied', ad personalization: 'denied', ad storage: 'denied', analytics storage: 'denied', wait for update: 1000}		
▶ cal	<pre>llee: f gtag()</pre>		
gtm	n.uniqueEventId: 1		
len	ngth: 3		
► Sym	<pre>mbol(Symbol.iterator): f values()</pre>		
▶ [[P	Prototype]]: Object		
▶ <b>1</b> : {g	gtm.start: 1711448294700, event: 'gtm.js', gtm.uniqueEventId: 2}		
▶ <mark>2</mark> : Ar	rguments(3) ['set', 'developer_id.dOThhZD', true, callee: (), gtm.uniqueEventId: 115, Symbol(Symbol.iterator): f]		
▶ <b>3:</b> Ar	rguments(3) ['consent', 'update', {…}, callee: (), gtm.uniqueEventId: 130, Symbol(Symbol.iterator): <i>f</i> ]		
▶ <b>4:</b> Ar	rguments(3) ['set', 'ads_data_redaction', true, callee: (), gtm.uniqueEventId: 131, Symbol(Symbol.iterator): f]		
▶ <b>5:</b> {a	action: 'onInitialPageLoad', event: 'consent_status', type: 'explicit', ucCategory: {}, Cloudflare: true,}		
► <b>6</b> : {e	event: 'Google Analytics EXPLICIT_DENY', gtm.uniqueEventId: 184}		
▶ 7: {∈	event: 'Matomo EXPLICIT_DENY', gtm.uniqueEventId: 222}		
▶ 8: {∈	event: 'Hotjar EXPLICIT_DENY', gtm.uniqueEventId: 260}		
▶ 9: {∈	event: 'Google Ads Conversion Tracking EXPLICIT_DENY', gtm.uniqueEventId: 298}		

▶ 10: {event: 'HubSpot EXPLICIT_DENY', gtm.uniqueEventId: 336}	
11: {event: 'gtm.dom', gtm.uniqueEventId: 374}	
12: {event: 'gtm.load', gtm.uniqueEventId: 412}	
13: {event: 'gtm.scrollDepth', gtm.scrollThreshold: 25, gtm.scrollUnits: 'percent', gtm.scrollDirection: 'vertical', gtm.triggers: '524	404671
14: {event: 'gtm.scrollDepth', gtm.scrollThreshold: 50, gtm.scrollUnits: 'percent', gtm.scrollDirection: 'vertical', gtm.triggers: '524	404671
15: {event: 'gtm.linkClick', gtm.element: a, gtm.elementClasses: '', gtm.elementId: '', gtm.elementTarget: '',}	
16: Arguments(3) ['set', 'developer_id.d0ThhZD', true, callee: (), gtm.uniqueEventId: 576, Symbol(Symbol.iterator): f]	
▼ 17: Arguments(3)	6
0: "consent"	
1: "update"	
2: {ad_personalization: 'granted', ad_storage: 'granted', ad_user_data: 'granted', analytics_storage: 'granted'}	
callee: ()	
gtm.uniqueEventId: 591	
length: 3	
<pre>Symbol(Symbol.iterator): f values()</pre>	
<pre>&gt; get callee: f ()</pre>	
<pre>set callee: f ()</pre>	
[[Prototype]]: Object	
18: Arguments(3) ['set', 'ads_data_redaction', false, callee: (), gtm.uniqueEventId: 629, Symbol(Symbol.iterator): f]	
19: {action: 'onAcceptAllServices', event: 'consent_status', type: 'explicit', ucCategory: {}, Cloudflare: true,}	
20: {event: 'gtm.scrollDepth', gtm.scrollThreshold: 90, gtm.scrollUnits: 'percent', gtm.scrollDirection: 'vertical', gtm.triggers: '524	404671
▶ push: <i>f</i> ()	

• You should also always check that no cookies are set without user consent!





# Looking for extra support?

Navigating marketing privacy compliance practices for your business doesn't have to be a solo adventure. Explore the wealth of resources available at Usercentrics and Cookiebot<sup>™</sup> websites to master the inner workings of Google Consent Mode V2 and future-proof your consent management strategies. Remember: with the right tools, compliance isn't just a requirement—it's an opportunity.

Learn more on our website

