Cookie Compliance Integration with Google Tag Manager

Apr 27, 2021•Article

Google Tag Manager (GTM) is a tool that allows you to manage and deploy tags (snippets of code or tracking pixels) on your website without having to modify your site code.

If you use Google Tag Manager (GTM) to inject cookies on your website and manage site content, you can set it up so that the scripts are controlled by the consent preferences choices selected by visitors.

This can be more efficient than using the standard helper methods described in the Client-Side Cookie Management section of this guide. However, there are some differences in behavior because of how GTM works.

This section of the guide explains how to set up GTM to take advantage of these changes.

> *Note*
>
> This is not intended to be a complete guide to Google Tag Manager. Consult official Google Tag Manager documentation for more detailed information about setting up and using GTM.

How it Works

To pass data to Google Tag Manager, the Cookie Compliance tool uses Variables. It supports JavaScript Variables and Data Layer Variables. We recommend using the GTM Data Layer and Data Layer Variables.

Cookie Compliance uses the existing `dataLayer` object or creates a new one if it doesn't already exist. If your site creates a `dataLayer` object, ensure that this does not overwrite the one Cookie Compliance creates.

It adds a key named **OnetrustActiveGroups** with a value of a comma delimited string of the current active category ids as selected by the visitor (or the default setup). This key is re-populated on every page load once the script is executed.

For example, the data contained in the value might be `,C0002,C0003,C0004`.

When a user updates their consent a `dataLayer` event **OneTrustGroupsUpdated** will be triggered. You will use this event when creating triggers to apply to your tags.

By creating a GTM Custom Variable and Triggers, you can make GTM tags only trigger when specific consent groups within OntrustActiveGroups are present.

Important Notes about Google Tag Manager Integration

- If a tag is blocked and then allowed, it'll fire without the need for page reload because of listening for **OneTrustGroupsUpdated**. If it had been previously allowed and then blocked, the tag would be blocked on subsequent page loads.
- If using a Custom HTML-type tag in Google Tag Manager, the tag content can be either of the following:
  - HTML / JavaScript
  - Making use of the Optanon helper methods to insert the HTML/JavaScript (i.e. `Optanon.InsertScript` or `Optanon.InsertHtml`)
- The Google Tag Manager rule operator and value fields can be customized if a different behavior is required than in the examples in this section.
- You can implement this integration in one of two ways: uploading and merging a provided container file or by manually establishing triggers for each tag.
- Tag blocking can also be performed using the first-party OptanonConsent cookie.
- **Auto-Blocking** is capable of blocking Google Tag Manager. If this occurs, use one of the follow methods to unblock Google Tag Manager:

```
j.setAttributeNode(d.createAttribute('data-ot-ignore'));
```

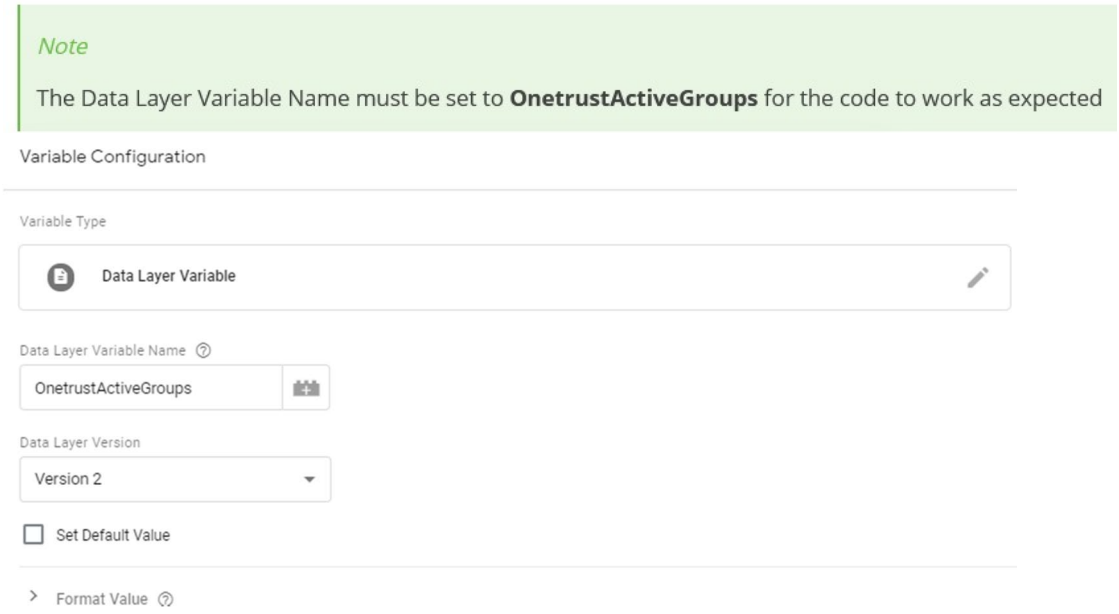
```
j.setAttribute('class','optanon-category-C0001');
```

To Create a New Variable

1. Open your container in Google Tag Manager.
2. Select the Variables tab from the main menu. The Variable screen appears.
3. Create a new **User-Defined Variable**.
4. Name the variable.

> **TIP**
>
> We suggest naming the variable **OnetrustActiveGroups** so you know what it refers to.

5. Set the Variable Type to **Data Layer Variable** under the Page Variables section.
6. Set the Data Layer Variable Name to **OnetrustActiveGroups**.



7. Press the Save button.

To Deploy the Banner Script in a Tag

1. Select the Tags tab from the main menu. The Tags screen appears.
2. Press New. The Tag modal appears.
3. Name the tag Cookie Banner Script
4. Under the Tag Configuration, press the edit button. The Choose tag type modal appears.
5. Select Custom HTML.
6. Paste the published script into the HTML editor and check the Support document.write box.

```
<!-- OneTrust Cookies Consent Notice start -->

<script src="https://cdn.cookielaw.org/opt-out/otCCPAiab.js"
type="text/javascript" charset="UTF-8" ccpa-opt-out-ids="{Category Id}"
ccpa-opt-out-geo="{geo}" ccpa-opt-out-lspa="{true or false}"></script>
<script type="text/javascript">
function OptanonWrapper(){}
</script>
      <!-- OneTrust Cookies Consent Notice end -->
```

7. Click the Triggering icon. The Choose a trigger modal appears.
8. Select All Pages.
9. Click Save.

Triggers

In GTM, triggers prompt tags to fire or not fire on certain criteria. You may already have a variety of different triggers applied to your tags.

As part of the Cookie Compliance integration with GTM you are going to create a trigger associated with each cookie category.

When you apply these triggers to your tags, this will prompt your tags to fire or not fire based on the Cookie Category being active or the consent given by the user.

You will need a separate trigger for each of the Cookie Categories that you will be blocking cookies under. For example, you may have a group called 'Performance Cookies', which has a category id of C0002 and contains the cookies set by your Google Analytics tag.

Each trigger needs to be in line with the category ids that are set in your cookie consent application. You can find the cookie category IDs in the cookie consent application under Categorizations.

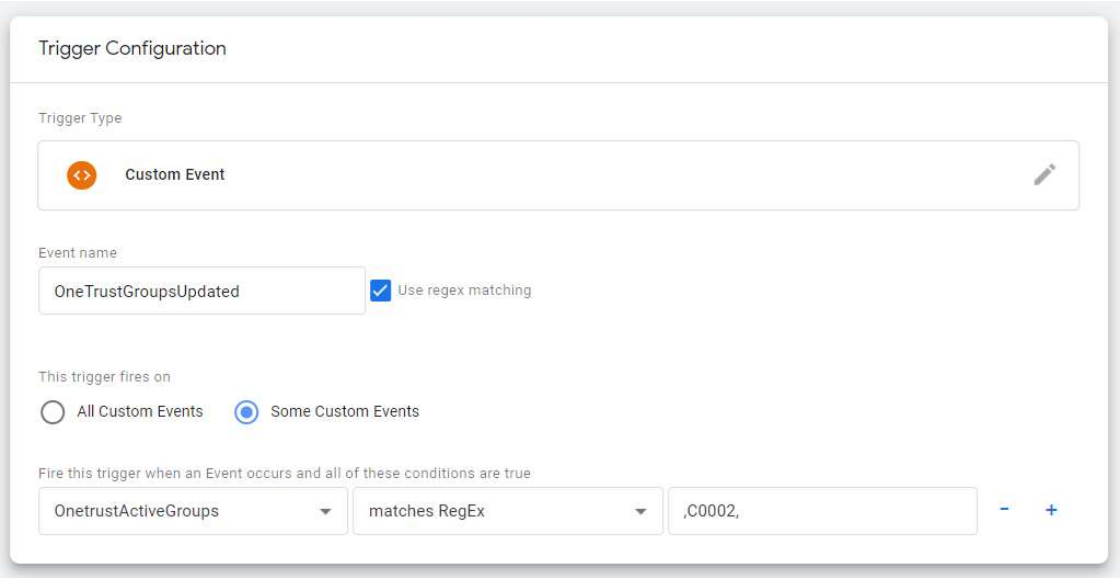
We want the OneTrust cookie category id triggers that you create or upload to be applied to your existing tags in such a way that the OneTrust trigger is the limiting factor to the tag firing. This can be done in several different ways. Three different ways are outlined below.

1. Creating a Firing Trigger
2. Using Firing Triggers on Existing Tags and Creating a Trigger Group
3. Creating Exception Triggers

To Create a Firing Trigger

1. Open your container in Google Tag Manager.
2. Select the Triggers tab from the main menu. The Triggers screen appears.
3. Press New. The Trigger Configuration screen appears.
4. Name it accordingly, e.g. **Performance Cookies Active**.
5. Press the Trigger Configuration and set the Trigger type to **Custom Event**.
6. Set the Event name to **OneTrustGroupsUpdated**. This event is embedded in the script.
7. Select Some Page Views and set it to fire when the following is true:

[OnetrustActiveGroups] [matches RegEx] [,C0002,]



8. Save the Trigger.
9. Repeat this process for the remaining Cookie Categories.
10. Apply the Triggers to Tags as a firing trigger.

To Create an Exception Trigger

You can also set up an exception trigger to fire the script if a category of cookies is not active.

You will only want to use an exception trigger if you already have a different firing trigger set up on your tag.

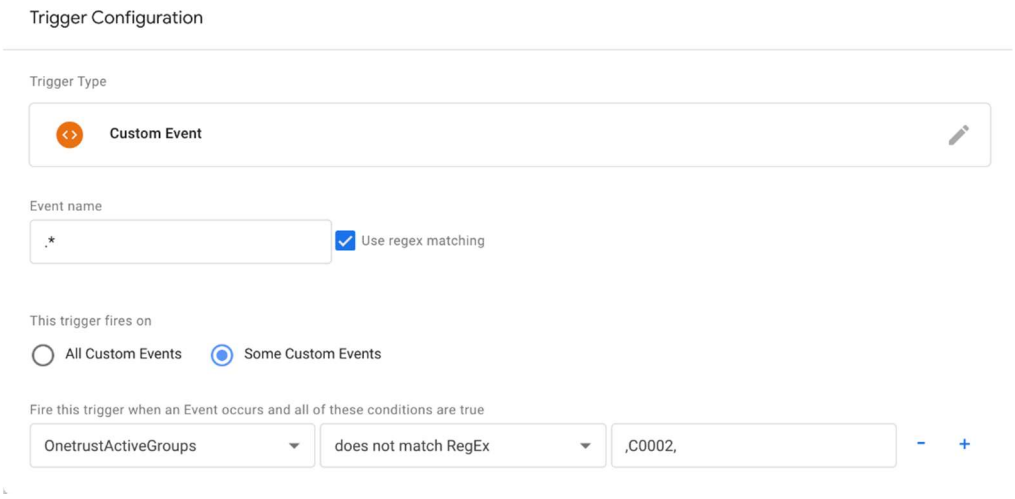
> *Note*
>
> In Google Tag Manager, Blocking triggers must fire on the same event as the Active triggers.

For example, set a trigger to fire when OneTrust Active Groups does not contain C0002 (where C0002 is the id for performance cookies). Apply this blocking trigger as an exception to tags in this group.

1. Select the Triggers tab from the main menu. The Triggers screen appears.
2. Press New. The Trigger Configuration screen appears.
3. Create a new trigger and name it accordingly, e.g. **Block Performance Cookies**.
4. Press the Trigger Configuration and set the Trigger type to **Custom Event**.
5. Set the Event name to **.***. This event applies to all events and will allow the exception trigger to override the event that is in the firing trigger.
6. Set the Trigger to fire on **Some Custom Events**.
7. Select Some Page Views and set the Trigger to fire when the following is true:

```
[OnetrustActiveGroups] [does not match RegEx] [,C0002,]
```

[OnetrustActiveGroups] [does not match RegEx] [,C0002,]



8. Save the Trigger.
9. Repeat this process for the remaining Cookie Categories.
10. Apply the Trigger to Tags as an Exception.

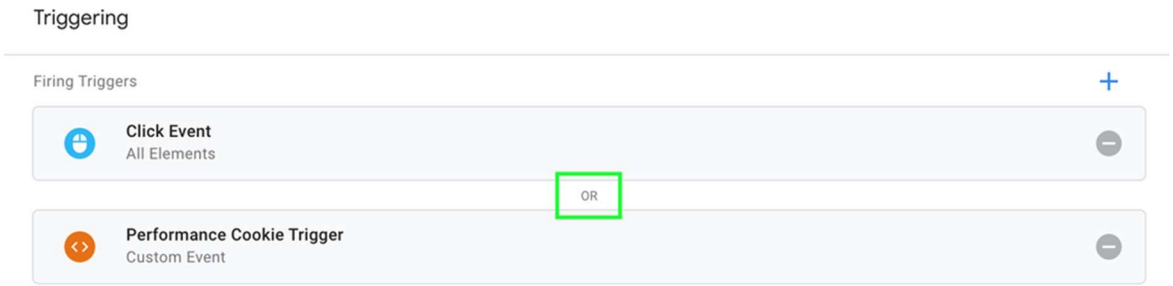
Using Firing Triggers on Existing Tags and Creating a Trigger Group

When using a firing trigger, you want to make the OneTrust trigger is the limiting factor to the tag firing.

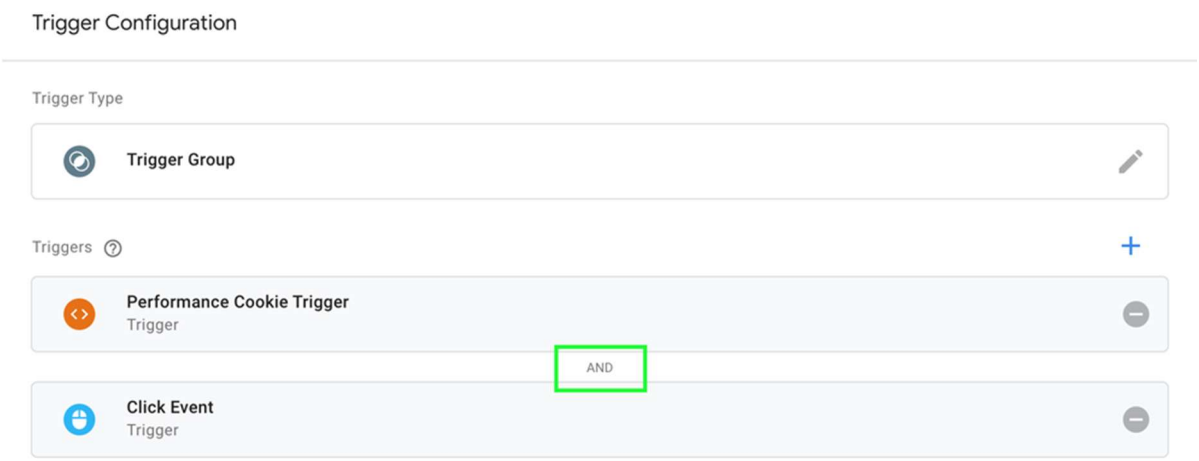
You will only want to use a trigger group if you already have a firing trigger set up on your tag.

Trigger groups utilize “AND” conditions, where as directly applying multiple triggers to a tag to be firing utilizes an “OR” condition.

Firing Triggers directly applied:



Within Trigger Group:



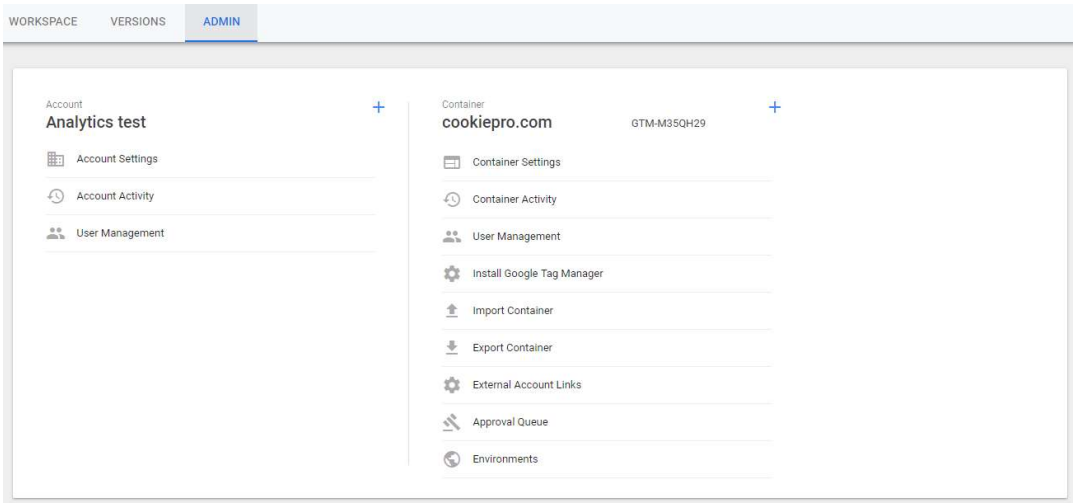
This is another method for setting up the OneTrust triggers and applying them to your tags such that the OneTrust triggers are the limiting factor to the tags firing.

1. Select the Triggers tab from the main menu. The Triggers screen appears.
2. Press New. The Trigger Configuration screen appears.
3. Create a new trigger and name it accordingly, e.g. **Example Trigger Group**.
4. Press the Trigger Configuration and set the Trigger type to Trigger Group.

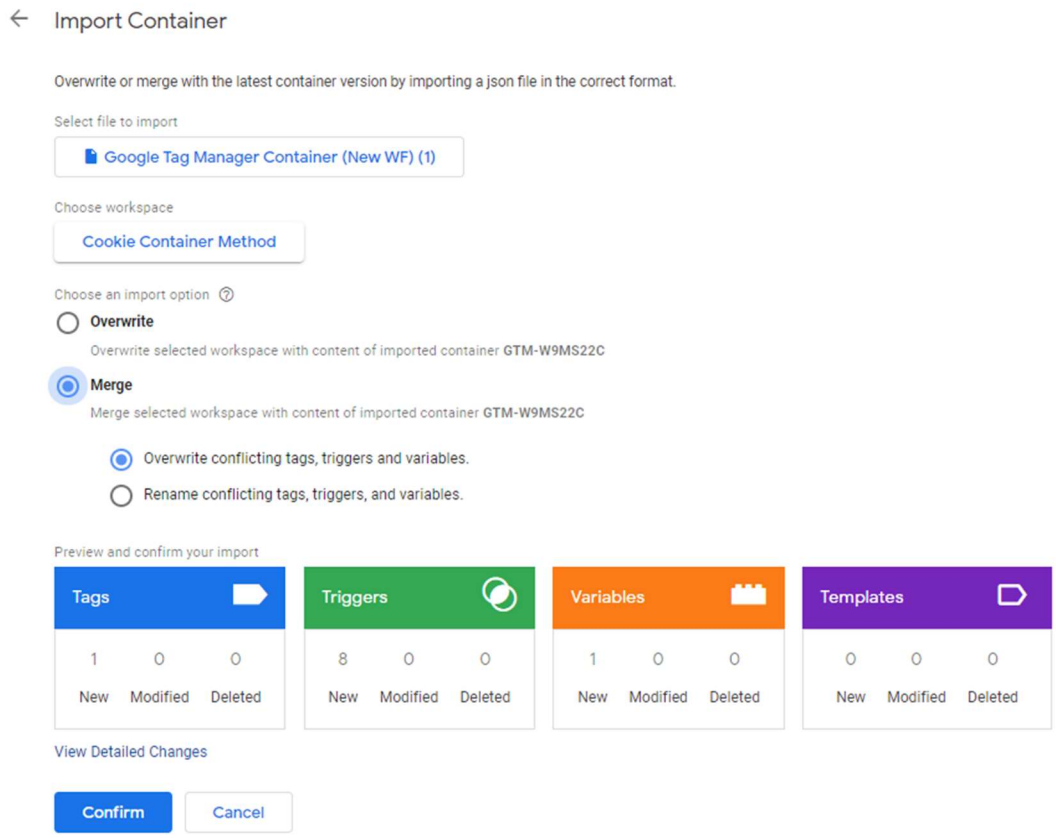
Using a Container to Integrate with Google Tag Manager

To Import a Container

1. Open your container in Google Tag Manager.
2. Go to the Admin tab.



3. Download the container from the link at the end of the article.
4. Click Import Container.



5. Select the downloaded container file.
6. Select Merge and then Overwrite conflicting tags, triggers and variables as the import option.
7. Click the Confirm button.

To Add the Triggers to Tags

1. On the main navigation menu in Google Tag Manager, select Tags.
2. Select a tag to apply the appropriate OneTrust Cookie Category Trigger to.
3. Apply the appropriate Firing Trigger, Trigger Group or Exception Trigger that you have created.
4. Click the Save button.

Blocking Tags Using the OptanonConsent Cookie

In addition to the `OneTrustActiveGroups` data layer variable, the OTSDKstub.js file also sets the first party cookie, `OptanonConsent`, to capture user consent. By decoding the value of the OptanonConsent cookie, the current consent can be read. For more information, see OneTrust Cookies .

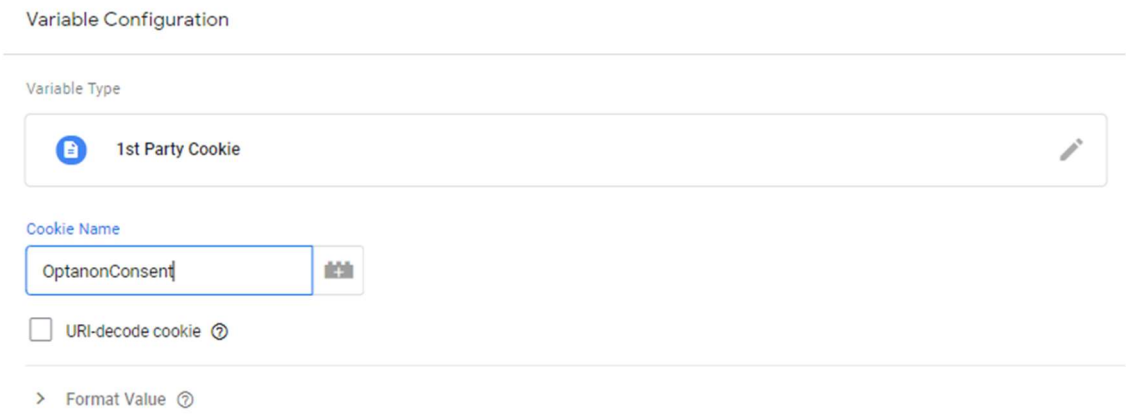
To Create a New Variable

1. Open your container in Google Tag Manager.
2. Select the Variables tab from the main menu. The Variable screen appears.
3. Create a new **User-Defined Variable**.
4. Name the variable.

> **TIP**
>
> We suggest naming the variable **OptanonConsent** so you know what it refers to.

5. Set the Variable Type to **1st Party Cookie** under the Page Variables section.
6. Set the Data Layer Variable Name to **OptanonConsent**.

> *Note*
>
> The Data Layer Variable Name must be set to **OptanonConsent** for the code to work as expected Additionally, the URI-decode cookie option must be disabled.



7. Press the Save button.

Triggers

Similar to the `OnetrustActiveGroups` data layer variable setup, active and blocking triggers will be created for each cookie category within the account. As the `OptanonConsent` cookie is set on page load, this allows you to trigger client tags before `OneTrustGroupsUpdated` event normally used by the data layer variable to block Cookie Categories.

For example, if you are using an e-commerce event that you have defined in the data layer, you can use the `OptanonConsent` cookie over the `OneTrustActiveGroups` data layer variable to trigger your analytics tags on the e-commerce event rather than on `OneTrustGroupsUpdated`.

To Create a Firing Trigger

1. Open your container in Google Tag Manager.
2. Select the Triggers tab from the main menu. The Triggers screen appears.
3. Press New. The Trigger Configuration screen appears.
4. Name it accordingly, e.g. **First Party Cookie - Active Performance - OT**.
5. Press the Trigger Configuration and set the Trigger type to the relevant use case.

> *Note*
>
> Adding Page View or DOM Ready triggers will likely the cause the triggers to not fire on the first page load.

6. Set the Event name to the event the on which affected tag will fire. This differs from the Data Layer variable `OneTrustActiveGroups` that must trigger on the event `OneTrustGroupsUpdated`.
7. Select the relevant trigger and set it to fire when the following is true:

`[OptanonConsent] [contains] [C0002:1]`

8. Save the Trigger.
9. Repeat this process for the remaining Cookie Categories.
10. Apply the Triggers to Tags as a firing trigger.

To Create an Exception Trigger

1. Open your container in Google Tag Manager.
2. Select the Triggers tab from the main menu. The Triggers screen appears.
3. Press New. The Trigger Configuration screen appears.
4. Name it accordingly, e.g. **First Party Cookie - Block Performance - OT**.
5. Press the Trigger Configuration and set the Trigger type to the relevant use case.
6. Set the Event name to the event the on which affected tag will fire. This differs from the Data Layer variable `OneTrustActiveGroups` that must trigger on the event `OneTrustGroupsUpdated`.
7. Select the relevant trigger and set it to fire when the following is true:

`[OptanonConsent] [does not contain] [C0002:1]`

8. Save the Trigger.
9. Repeat this process for the remaining Cookie Categories.
10. Apply the Triggers to Tags as an exception trigger.