SERVER SIDE TRACKING VIA GTM

BY HIMANSHU SHARMA
FOUNDER OF OPTIMIZESMART.COM

OptimizeSmart.com

Server Side Tracking via GTM

Written by Himanshu Sharma, Founder of Optimize Smart

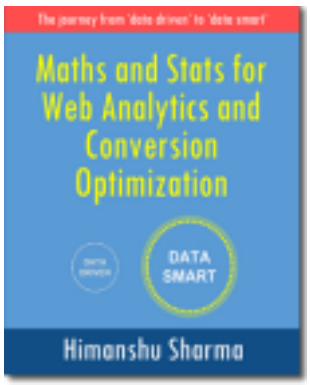
About the author

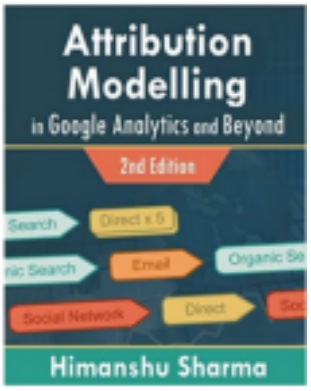
- Founder, OptimizeSmart.com
- Over 15 years of experience in digital analytics and marketing
- Author of four best-selling books on digital analytics and conversion optimization
- Nominated for Digital Analytics Association Awards for Excellence
- Runs one of the most popular blogs in the world on digital analytics
- Consultant to countless small and big businesses over the decade

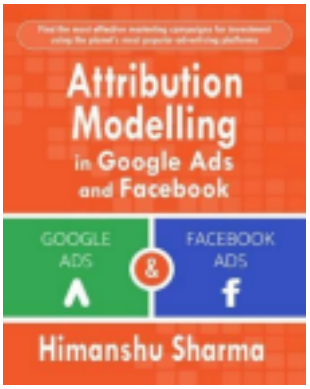
Website: https//www.optimizesmart.com

LinkedIn: https://www.linkedin.com/in/analyticsnerd

Facebook: https://www.facebook.com/optimizesmart







Following are our most downloaded ebooks for career advancement:

#1 Sales and ROI Accelerator (150+ pages)

WHAT'S INSIDE: My step-by-step blueprint for generating record sales and ROI by leveraging analytics data.

#2 Set Up Your Google Analytics 4 (GA4) Account Correctly And Fast (70 pages)

WHAT'S INSIDE: Learn to set up your GA4 account correctly and fast using this 62 points checklist.

FAQ: Do you show "How" to do each item on the checklist? If so, with screenshots?
Yes. There are links to the articles with detailed step by step instructions.

FAQ: Does this ebook cover GTM too?
Yes.

#3 Google Tag Manager Data Layers (100+ pages)

WHAT'S INSIDE: My step-by-step blueprint for getting started with data layers. Get the only ebook on GTM data layers ever published. Learn the JavaScript behind it.

#4 Learn to Read E-Commerce Reports in Google Analytics (100+ pages)

WHAT'S INSIDE: My step-by-step guide to reading both standard and enhanced e-commerce reports in Google Analytics. E-commerce reports are the most valuable reports in Google Analytics.

#5 Do you want better skills in digital analytics and marketing? If yes, then register for the free training:

Here's what we're going to cover…

1. Why digital analytics is the key to online business success.
2. The number 1 reason why most marketers are not able to scale their advertising and maximize sales.
3. Why Google and Facebook ads don't work for most businesses & how to make them work.
4. Why you won't get any competitive advantage in the marketplace just by knowing Google Analytics.
5. The number 1 reason why conversion optimization is not working for your business.
6. How to advertise on any marketing platform for FREE with an unlimited budget.
7. How to learn and master digital analytics and conversion optimization in record time.

Get helpful tips on a daily basis

If you are the type of person who finds it helpful to receive short tips on building your website traffic, improving conversions, fixing attribution issues and learning about analytics in general, then follow me on LinkedIn. I post a few short tips each day.

Click here and follow me on LinkedIn

Why use Server Side Tracking?

You probably know that an increasing number of users are now using ad blockers. And browsers continue to restrict access to more and more users' data.

All of these tracking restrictions are creating big data gaps on the conversion paths and making it very difficult to understand customers' purchase journeys and advertise profitably.

So how do you track users' data then?

Is this the end of website tracking as we know it?

No.

It is actually the dawn of new and more powerful tracking called '**server-side tracking**'. You need to switch from client-side to server-side tagging.

You move all your tracking pixels to the server-side. Whether it is Google Analytics, or GA4 or Google ads.

You run all tracking scripts from your web server instead of the client-side. You can then make third-party cookies, first-party. You can then prevent at least some of your data from being blocked by web browsers and ad blockers.

When it comes to tracking websites, or mobile apps, we generally use client-side tracking which is commonly achieved through tags in Google Tag Manager (aka GTM).

This method involves setting up a tag that collects data from the browser (client) and directly sends it to the individual data services like Google Analytics, Google Ads and Facebook.

What is GTM server-side tagging?

Server-side tagging is a new way to use Google Tag Manager (GTM) in Google's cloud environment.

It has its own benefits such as reduced page load time, better security, and better control over the data that you send to Google Analytics and third-party tools.

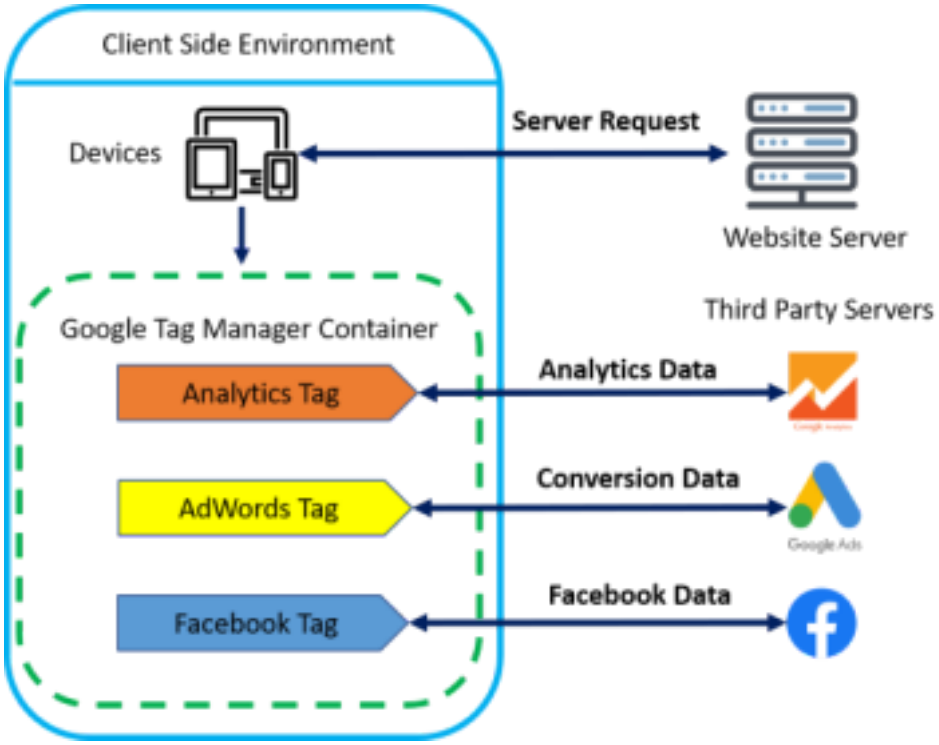
Like the normal Google Tag Manager container, which resides in a client-side environment, the server-side container resides in the Google cloud environment.

A server-side container also uses the same concepts, like tags, triggers and variables, which you have used earlier.

A server-side container acts like a proxy in-cloud environment that you own.

Instead of sending hits directly to the endpoint server (like in client-side GTM), you send hits to a server-side GTM container and then to the endpoint server which collects the data.

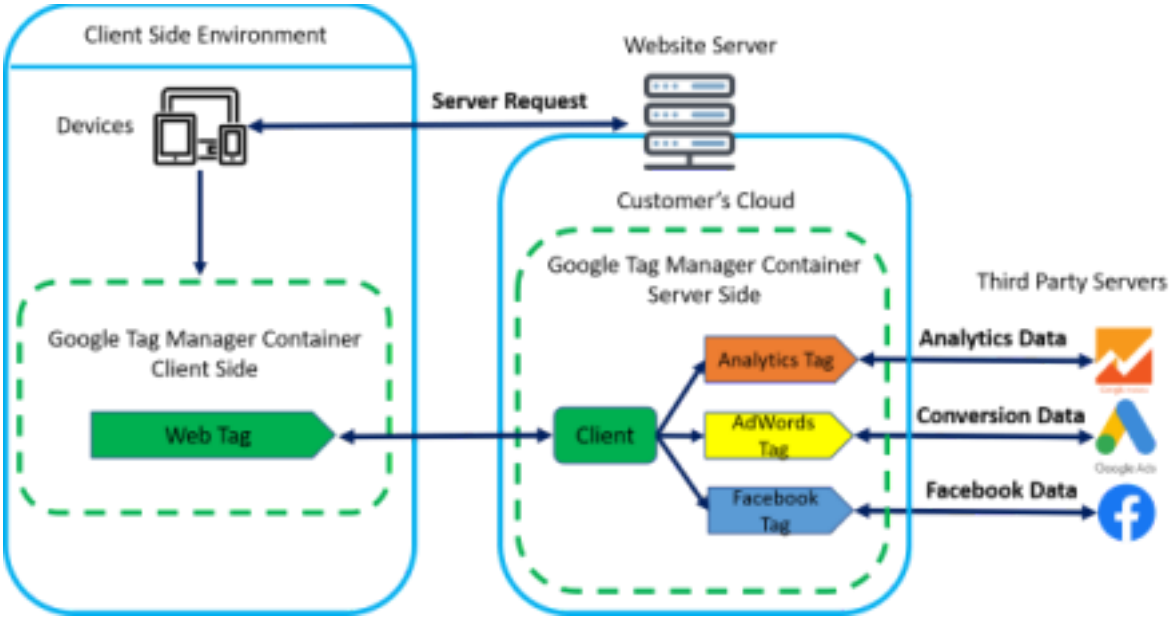
A typical client-side GTM container looks like below.



As you can see from the image above, whenever a server request is made by a user with different devices, the Google Tag Manager configuration without server-side tagging depends on a container in the website to send hits data to various third-party collection servers.

But in the case of server-side tagging, the Tag Manager configuration runs in a cloud environment and thus does not affect the website or application performance by having multiple scripts running on the devices.

Let's see the below image for the detailed working of the server-side tagging container.



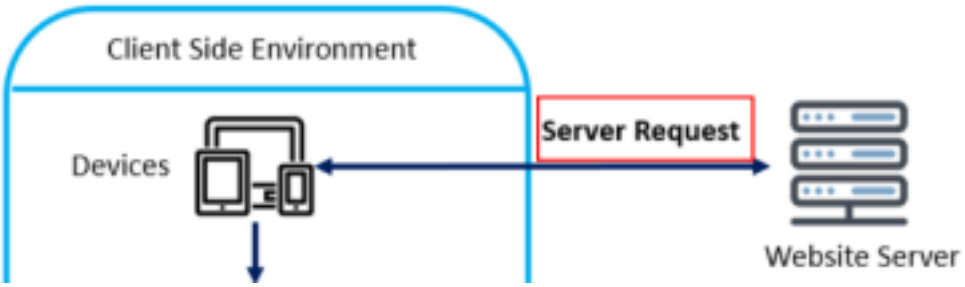
As you can see from the above image, the server-side container runs in the Google cloud platform owned by the customer and only the customer has access to the data which is being sent to third-party tools.

You have complete control over the data on how it's being routed to the third-party tool.

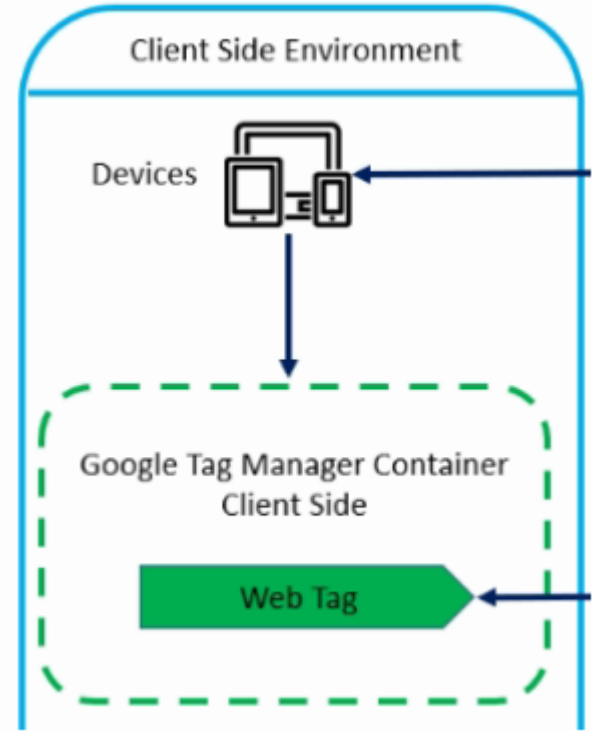
How server-side tagging works

Let me explain the process of how server-side tagging works in a step-by-step method.

Step 1: When a user visits your website, a pageview request is made to the web server through a server request.

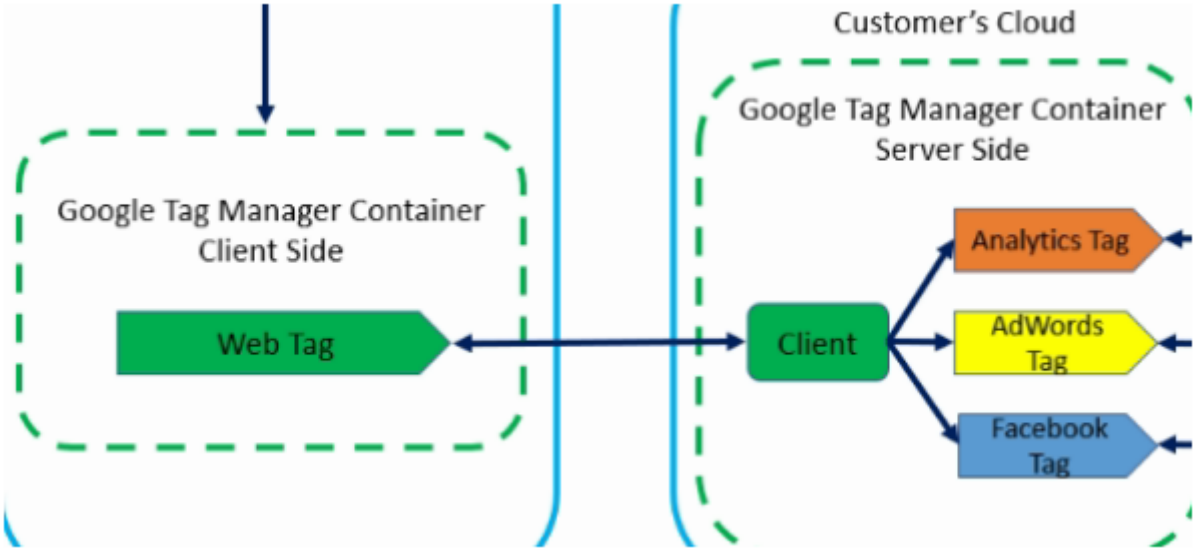


Step 2: At the same time, the client-side GTM container fires a ‘web tag’ that acts as an adapter between the scripts running on a user’s device and your server container. It takes the required measurement data from the user’s device and sends it to the client in the server-side container.

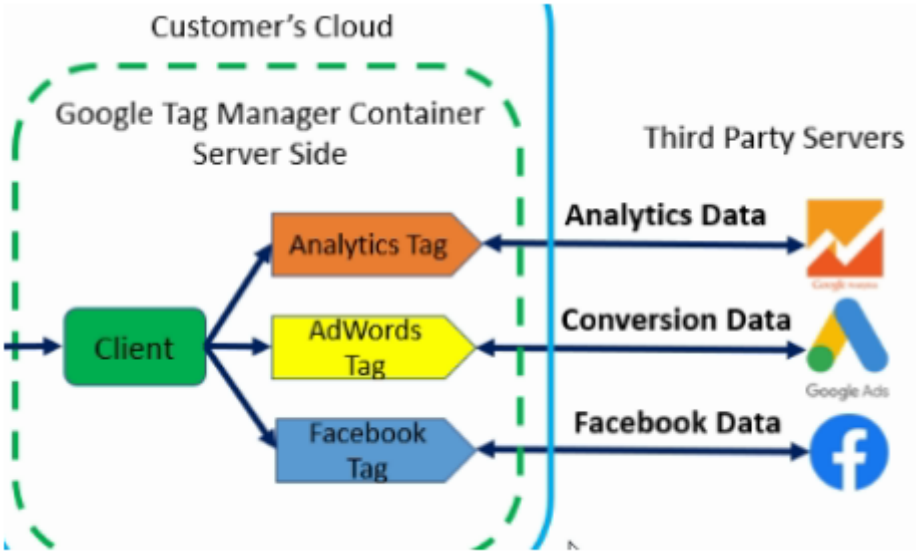


Note: We are still using the client-side GTM container but in this case instead of directly sending data to third-party tools like Google Analytics, Google Ads, and Facebook, it sends data to the client in the server-side container of GTM.

Step 3: The client in the server-side container is the type of adapter that receives measurement data from 'web tag', processes that data into one or more events and packages up the final data to send back to the requester.



Step 4: Now in the server-side container you can create multiple tags based on the requirements and then send them to the final third-party tools.



How much does server-side tagging cost?

Google recommends that you run at least 3 instances of App Engine on the Google Cloud Platform. The cost per server instance is around US $40 per month.

So for 3 server instances, it would cost you around $120 per month or $1440 per year. If you run a very high traffic website, you may need more than three server instances.

So if you need say 6 servers then it would cost you around $240 per month or $2880 per year. And here is the bummer. We are only getting started.

This is just the computation cost and like the bare minimum, you can expect to spend each month.

The actual cost would depend upon network egress (all outgoing HTTP traffic from your server endpoint) and logging.

I tested server-side tagging on my website for a month and it cost me $250.

What that means, if I start using it full time, I am looking at a bill of around $3k per year. And this is assuming that the website traffic doesn't increase over time.

Because the more traffic your websites get, the more third party tags you send data to, the more you will pay.

When you are using server-side tagging, you have to be very careful not to collect

unnecessary data.

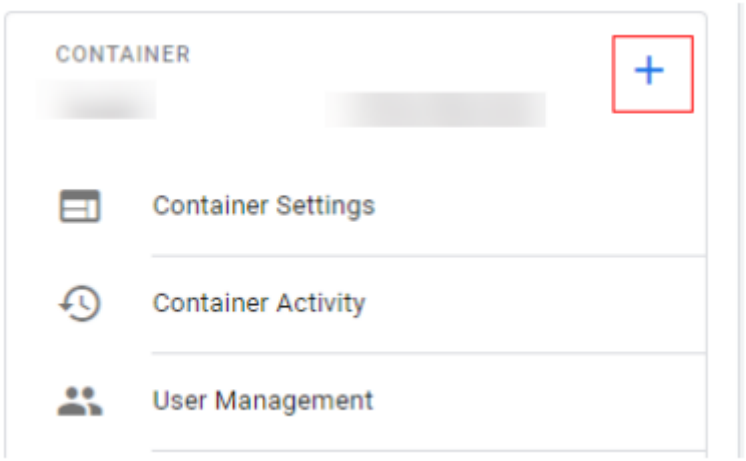
How to set up server-side tagging via GTM

To use server-side tagging, you need to create a new Tag Manager server container.

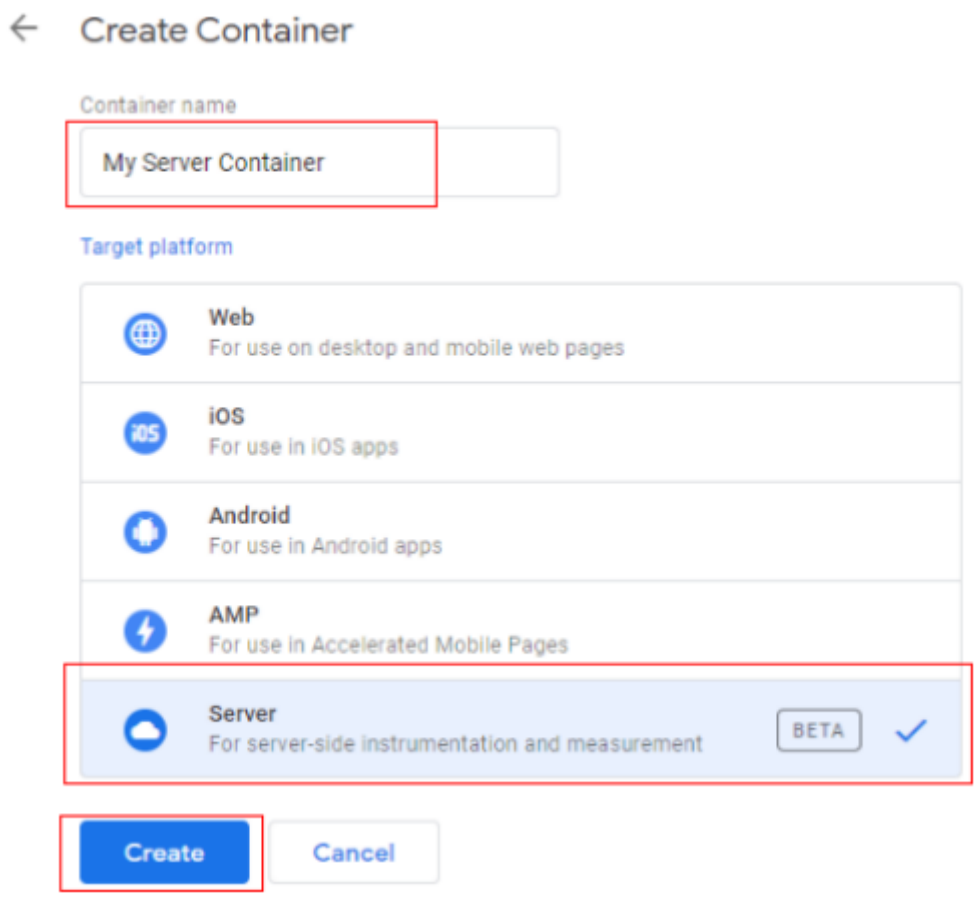
Follow the below steps to create it:

Creating a server-side GTM container

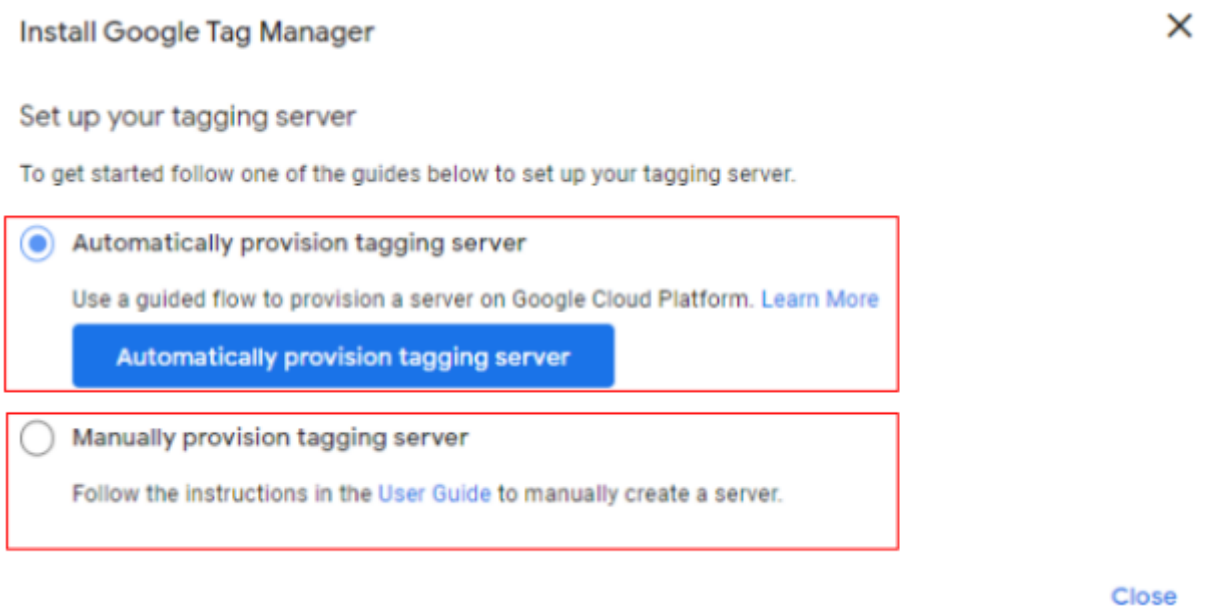
Step 1: Navigate to your Google Tag Manager account and click on the '+' button to create a container.



Step 2: Give your container a descriptive name, select 'Server' from the available target platforms and then click on 'Create'.

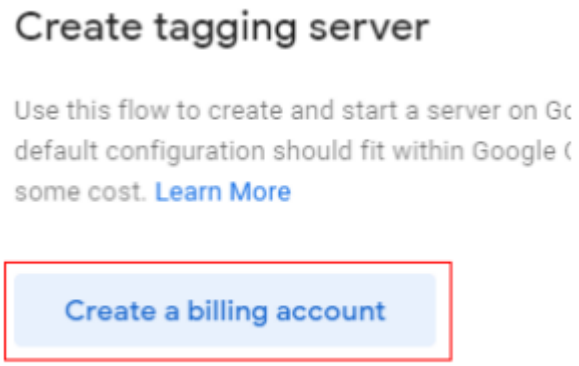


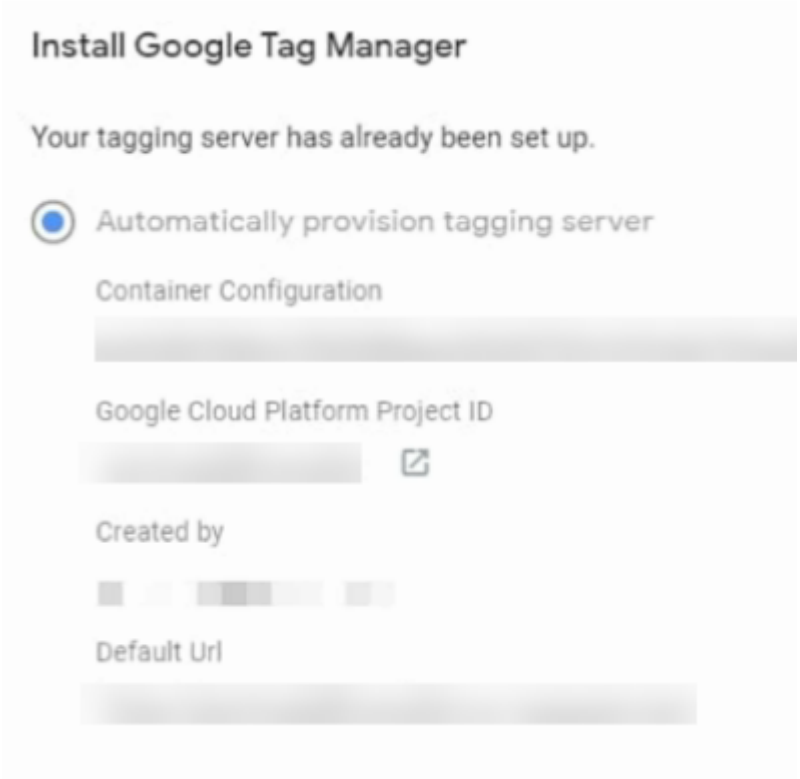
Step 3: Once you click on 'Create' a pop-up will come like the one below asking you to provide the container with a tagging server (Google Cloud App Engine Standard Instance).



You have two options to select from – '**Automatically provision tagging server**' and '**Manually provision tagging server**'.

If you are new to the Google Cloud Platform you can select the first option where you will need to create a billing account providing your financial details and Google will automatically create a tagging server for you.

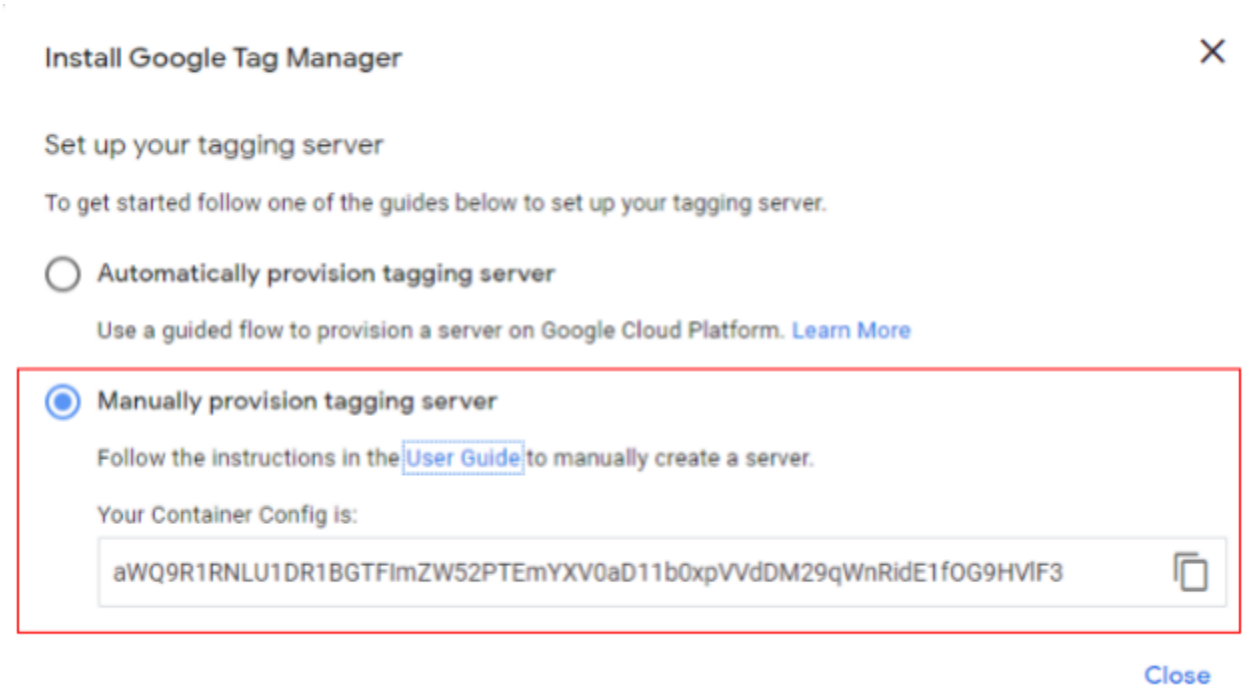




If you already have access to Google Cloud Platform and have your own server you can select the 'Manually provision tagging server' option.

In our case, I already have access to Google Cloud Platform so I will go with

manually configuring the tagging server.



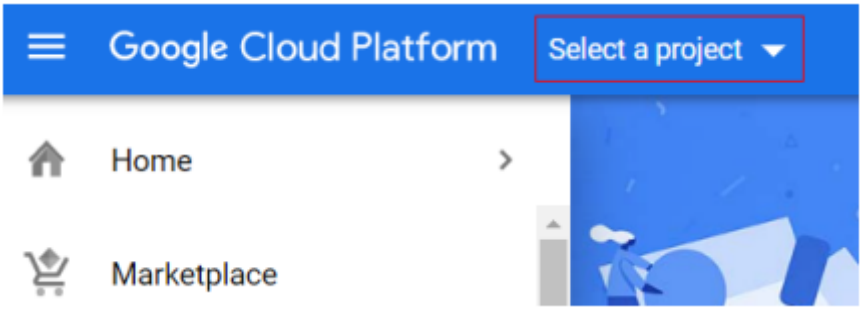
Note down the container configuration to configure the Google cloud server in the next steps.

Note: In case you have selected 'Automatically provision tagging server', the

container configuration will be loaded by default while setting up the Google server.

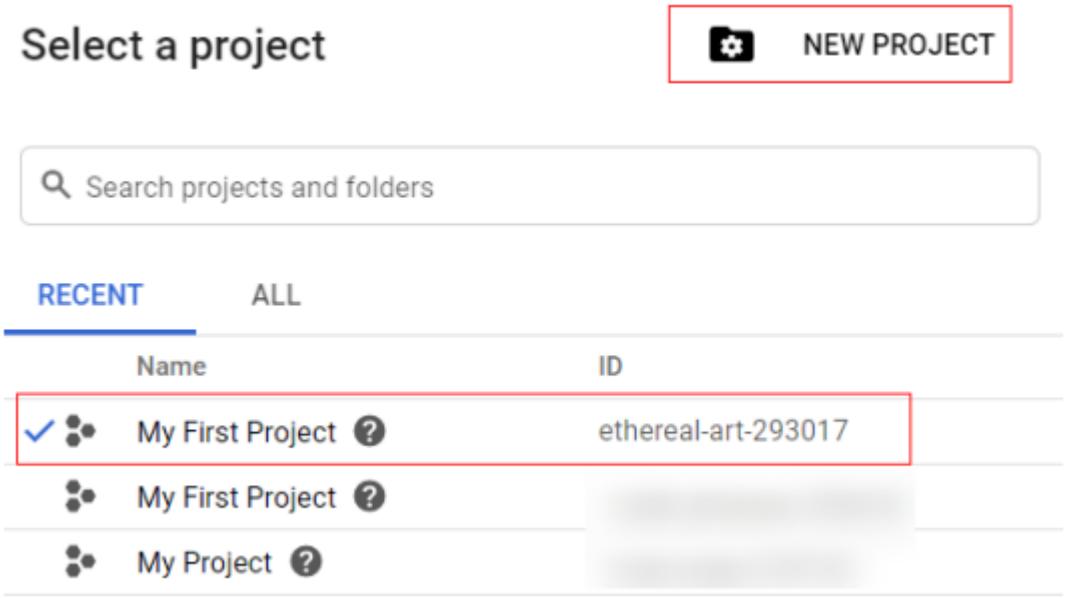
Configuring Google server for GTM container

Step 4: Navigate to https://console.cloud.google.com/ and click on ‘Select a project’.

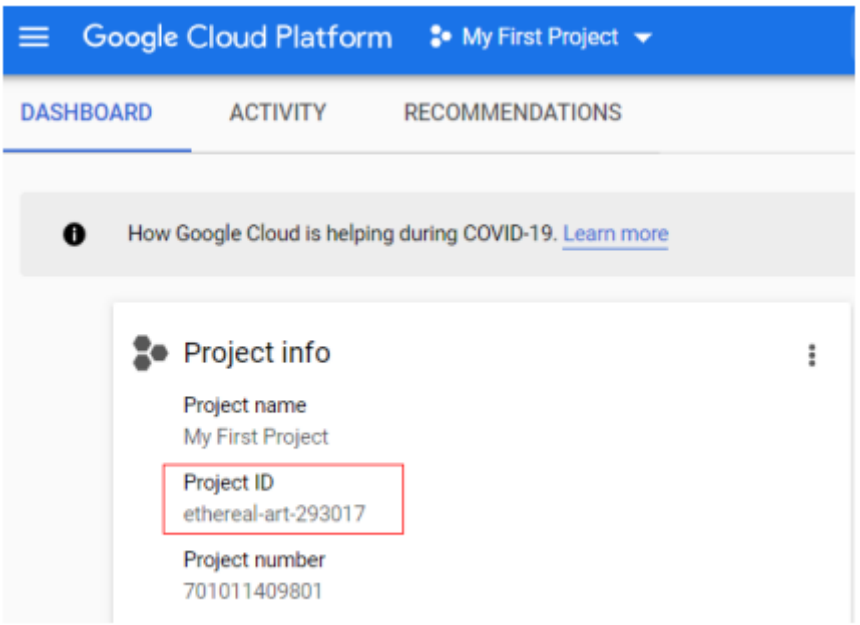


Step 5: An overlay will appear. You can select the project from the available list or you can also create a new project by clicking on ‘New project’. In our case, I will select from the list of available projects since I have already created one.

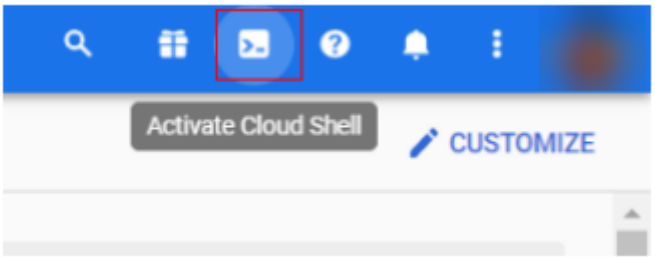
Click on ‘Open’ at the bottom of the overlay.



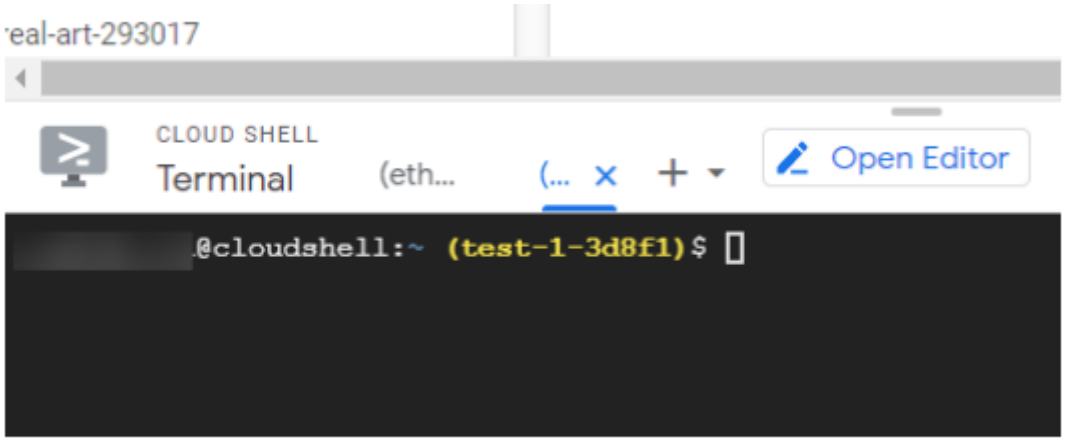
Step 6: A dashboard for the project will open. Note down the project ID.



Step 7: Now click on 'Activate Cloud Shell' from the menu bar in the upper-right corner.



Step 8: A command prompt will appear at the bottom of your window, like below.

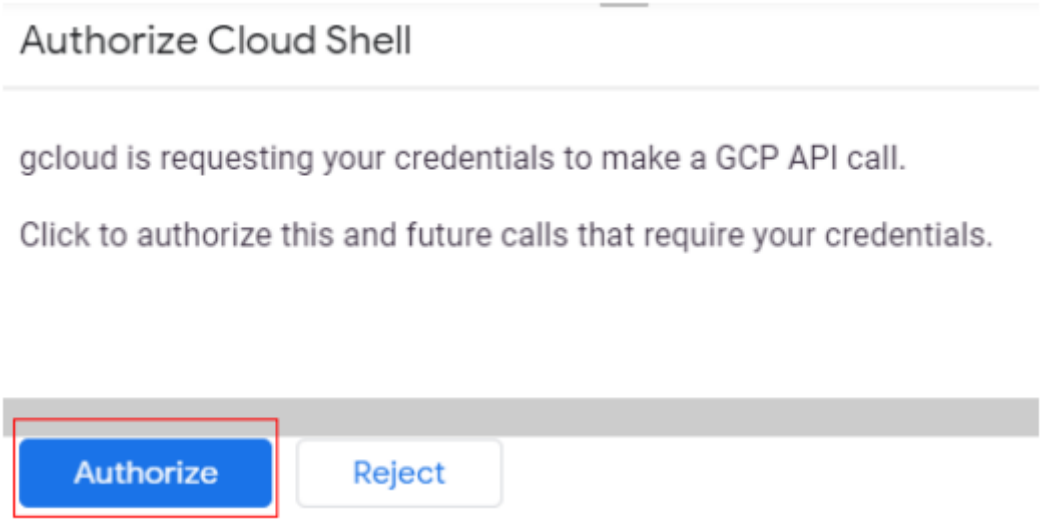


Step 9: Now we need to set the cloud platform project in the cloud shell.

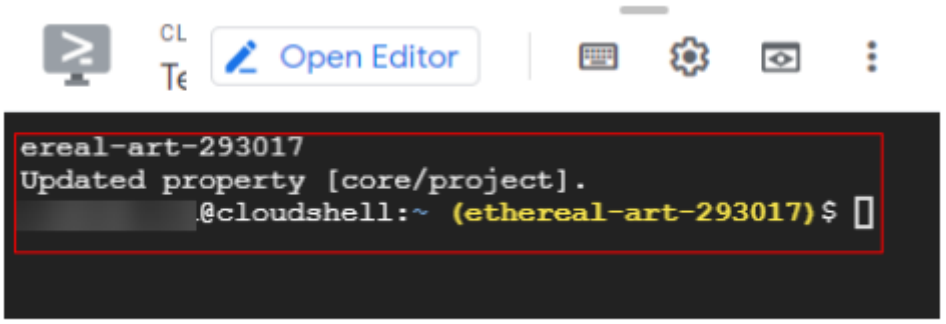
Copy the following command and paste it in the command prompt. Replace the <PROJECT ID> with the GCP project ID that you noted earlier in step 6.

gcloud config set project <PROJECT_ID>

Step 10: Press ‘Enter’, an authorization overlay will occur to make the GCP call. Click on ‘Authorize’.

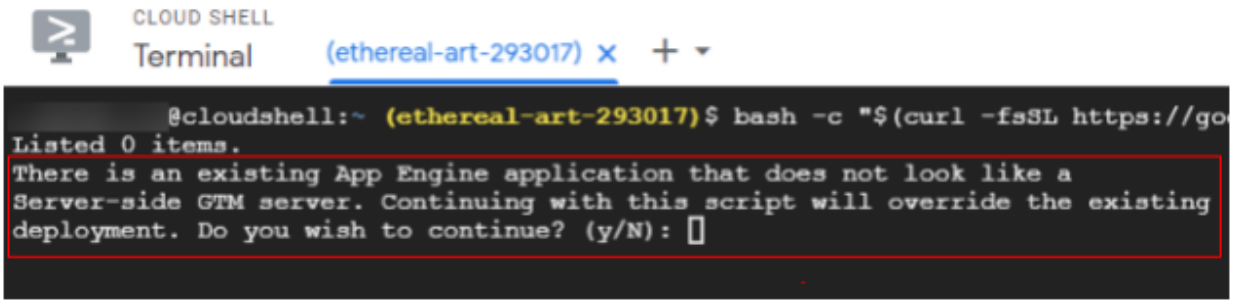


Step 11: You will now see the project ID set to the new ID for which we are going to set up the server.



Step 12: Copy and paste the following command in the command prompt and press ‘Enter’.

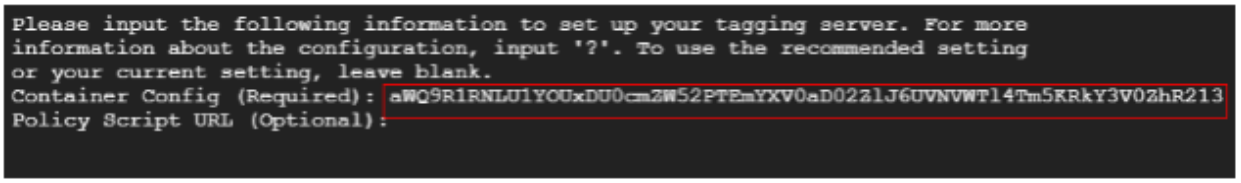
bash -c “$(curl -fsSL https://googletagmanager.com/static/serverjs/setup.sh)”



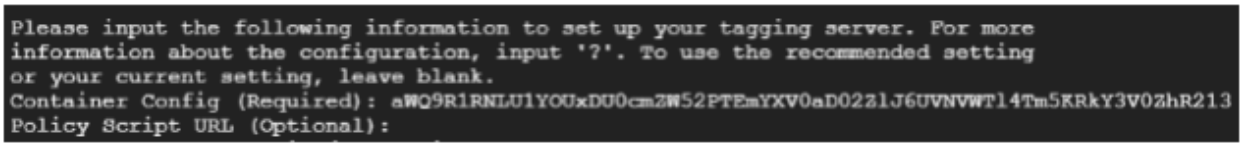
This script will configure the host for our server-side container. It will ask multiple questions for which we will need to provide input.

Step 13: Press “Y” and press enter. It will ask for “Container Config (Required)”.

Copy the container configuration we noted earlier in step 3 and paste in the command prompt. Press ‘Enter’.

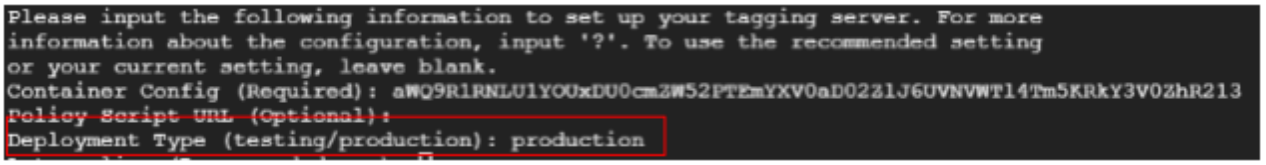

Please input the following information to set up your tagging server. For more
information about the configuration, input '?'. To use the recommended setting
or your current setting, leave blank.
Container Config (Required): aWQ9R1RNLU1YOUxDU0cmZW52PTEmYXV0aD02Zl1J6UVNVWTl4Tm5KRkY3V02hR213
Policy Script URL (Optional):

Step 14: Now It will ask for the policy script URL which is optional and we can skip it. Press ‘Enter’ to skip it.



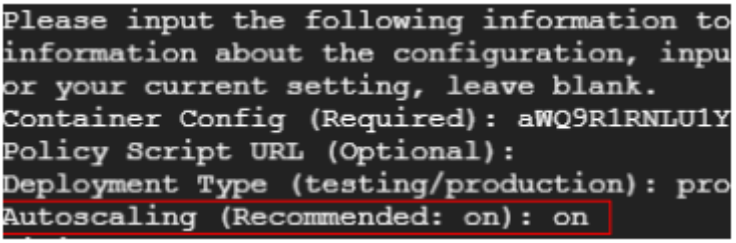
Step 15: The next input is to provide ‘Deployment type’. The options available are ‘testing’ and ‘production’. We are deploying this solution on a live website so we will choose ‘production’.

Type ‘production’ in the command prompt and press ‘Enter’.

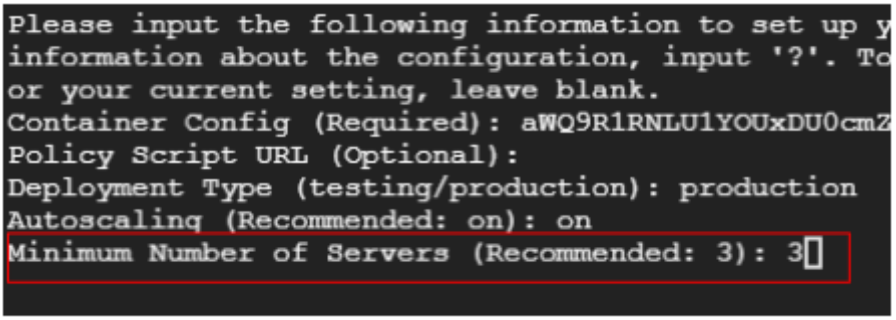


Step 16: The next step is to provide the autoscaling option. Autoscaling lets your apps gracefully handle increases in traffic, and it reduces costs when the need for resources is lower. It is recommended that this is ‘on’.

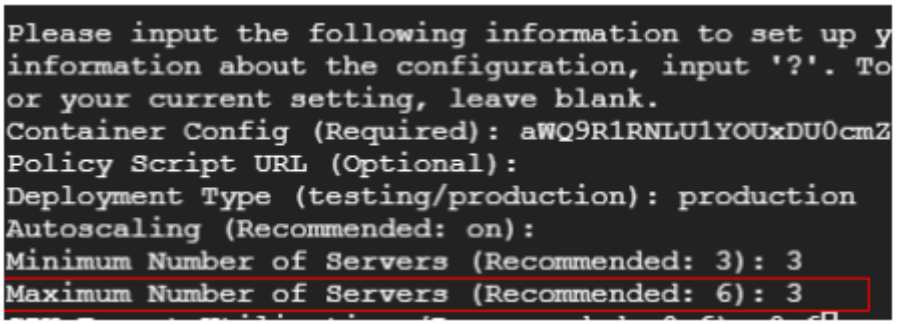
Type ‘on’ and press ‘Enter’.



Step 17: In the next step we have to provide the minimum number of servers to host the container (minimum 3 to maximum 6). Type ‘3’ and press ‘Enter’.



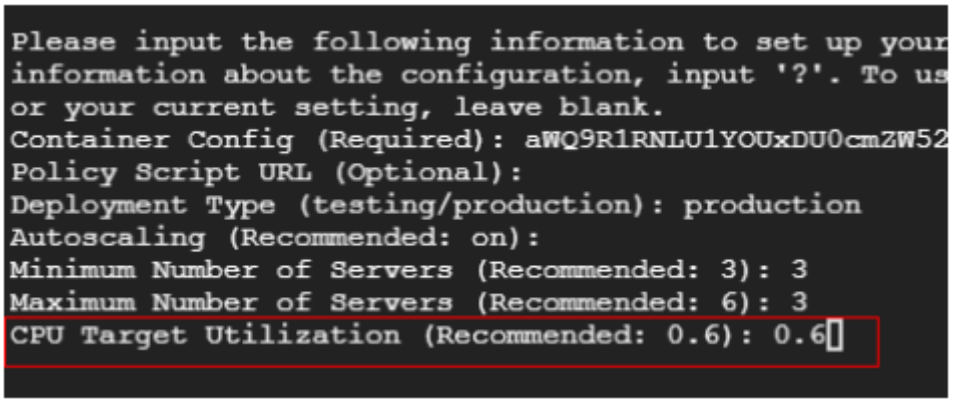
Step 18: You will be prompted to provide the maximum number of servers. You can add up to 6 maximum servers but in our case 3 are sufficient. Type ‘3’ and press ‘Enter’.



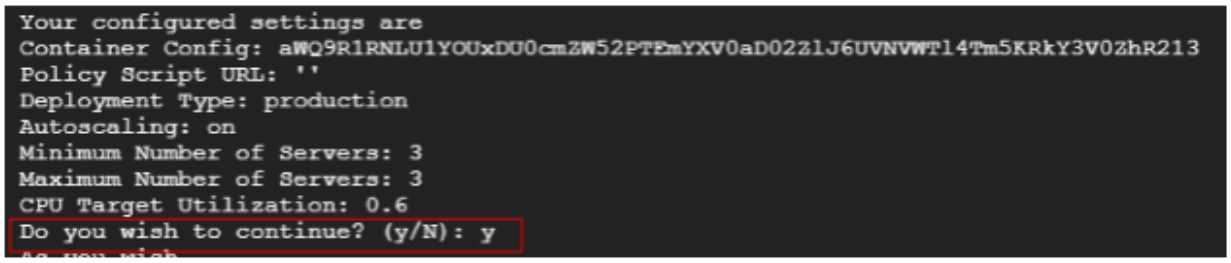
Step 19: Now you will be asked to provide CPU target utilization. The target utilization level is the level at which you want to maintain your virtual machine (VM) instances.

For example, if you scale based on CPU utilization, you can set your target utilization level at 75% and the autoscaler will maintain the CPU utilization of the specified group of instances at or close to 75%.

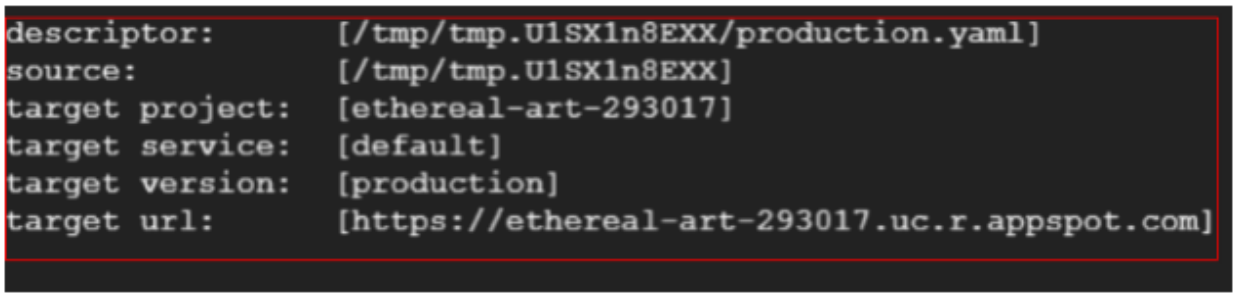
Recommended to put at 0.6 (60%). Type '0.6' and press 'Enter'.



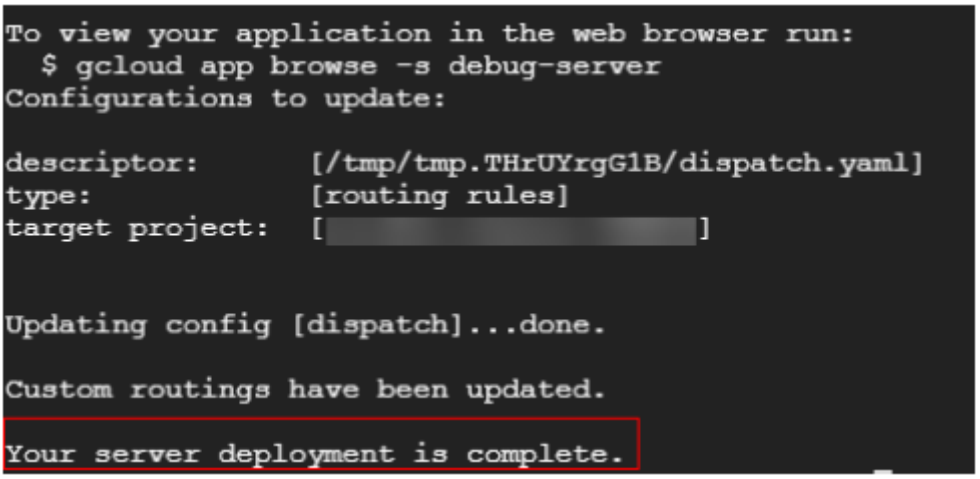
Step 20: You will be prompted with the inputs you have provided so far and asked if you wish to continue. Type ‘Y’ and press ‘Enter’.



Step 21: Once you press ‘Enter’, your server will be configured and you will be provided with confirmation like below.

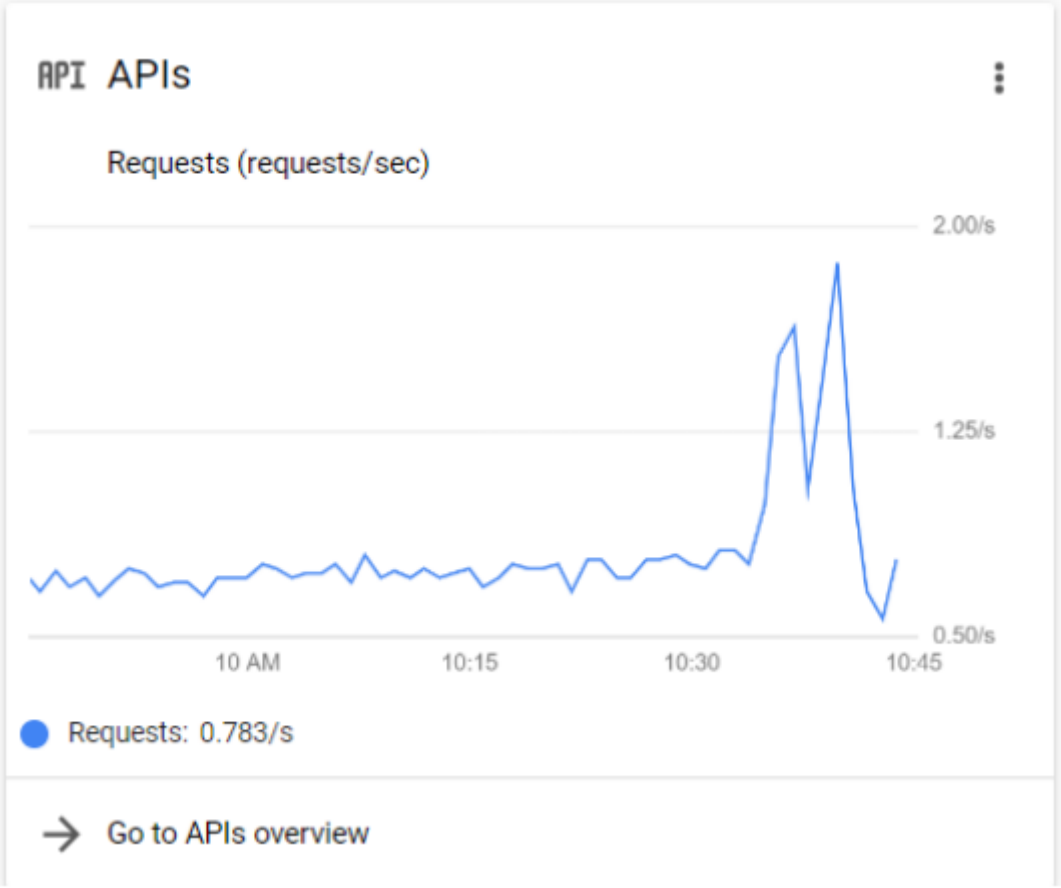


Note down the target URL from the above image since we need to add this in our Google Tag Manager server-side container.



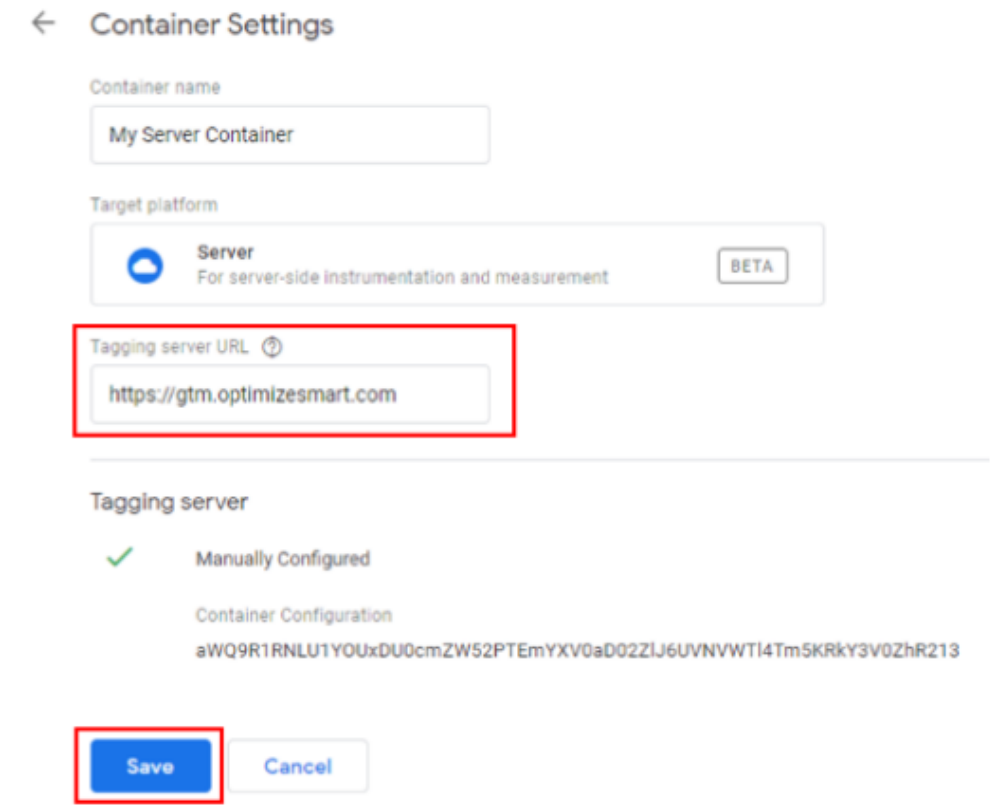
Congratulations! You have successfully configured your server now.

You can go to the dashboard and see the live requests coming in.



Now, let's set up the Google Tag Manager container with the target URL we noted earlier.

Step 22: Navigate to the admin section of your server-side Google Tag Manager container. Paste the target URL you have noted down in the input box named 'Tagging server URL' and click on 'Save'.



Your server-side container is now ready to use.

How to send data to server-side

Google Tag Manager

You can send data to a server-side container in the following ways:

1. Using GTM: You can use regular GTM on the client-side to send data.

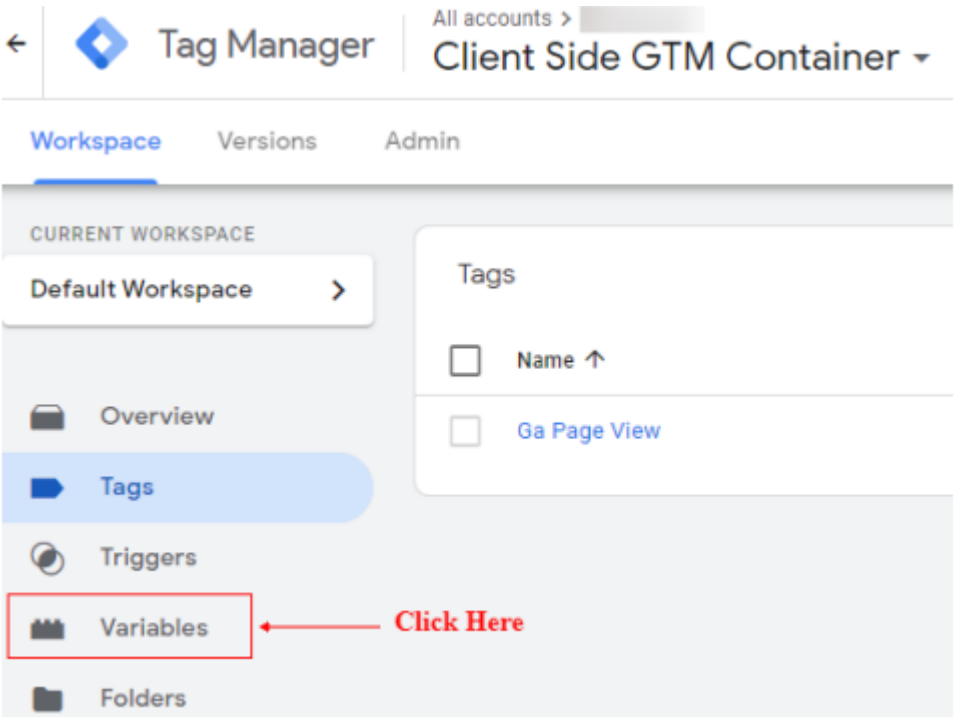
2. Using Gtag.js: You can also use Gtag.js to send data to a server-side container

3. Custom code: You can also write custom code.

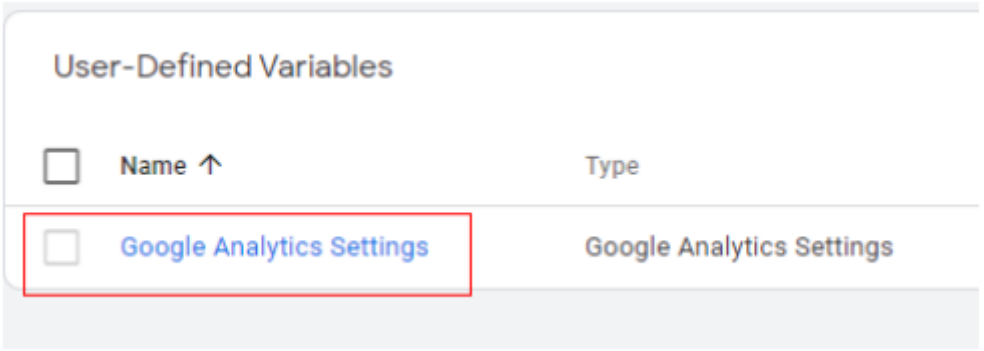
In our case, I am using the regular version of a client-side GTM container.

Follow the below steps to send data to a server-side GTM container:

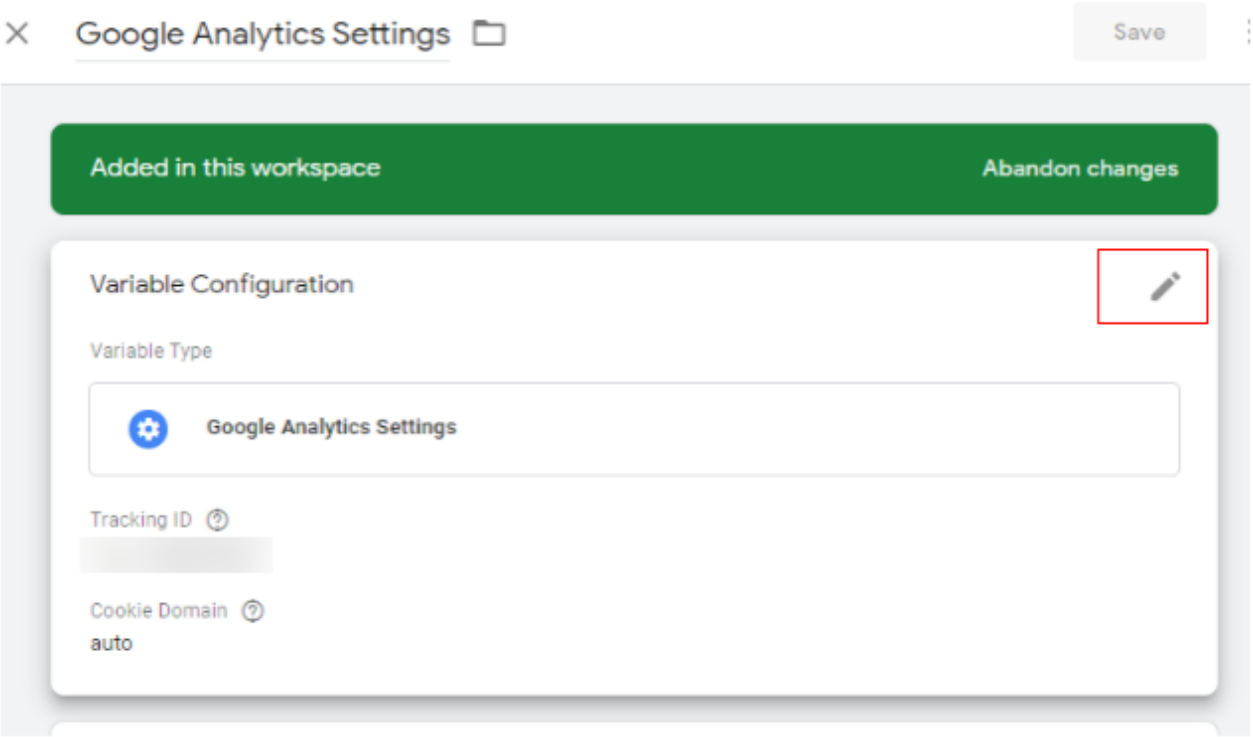
Step-1: Log in to your client-side GTM container and go to the 'Variables' tab.



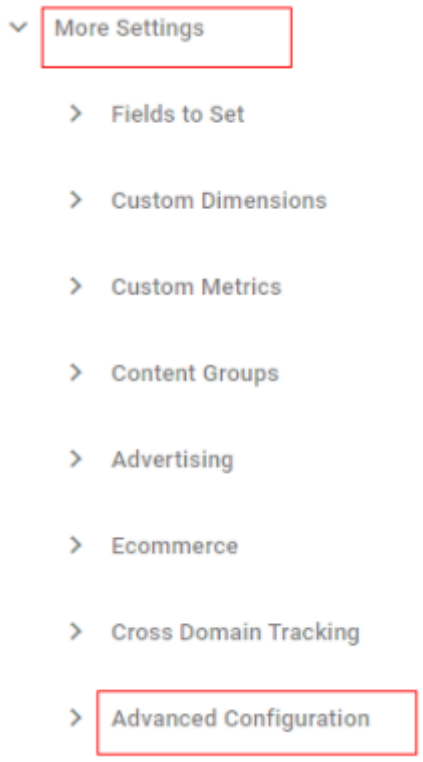
Step 2: Select the ‘Google Analytics Setting’ variable. This is the variable that sends data directly to the Google Analytics server. We are configuring it to send data to our server-side container.



Step 3: A new overlay will appear. Click on the pencil icon to edit the variable settings.

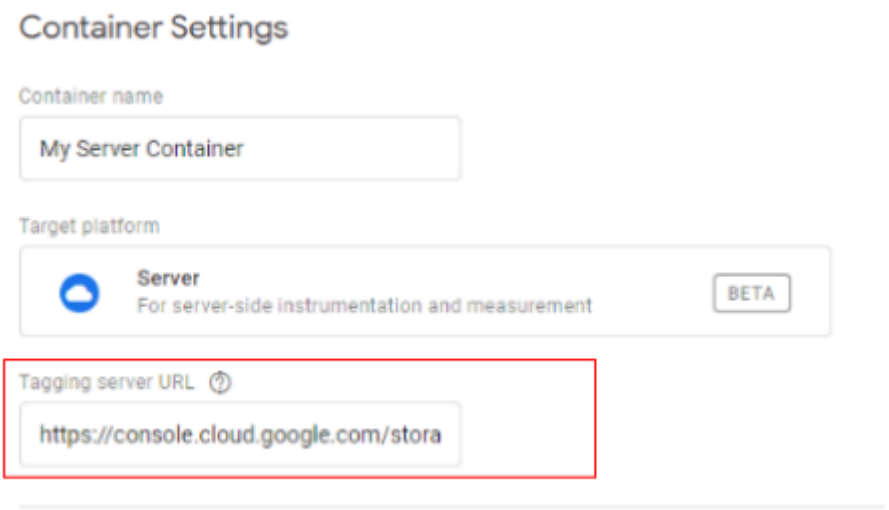


Step 4: Click on ‘More settings’ and then click on ‘Advanced configuration’.



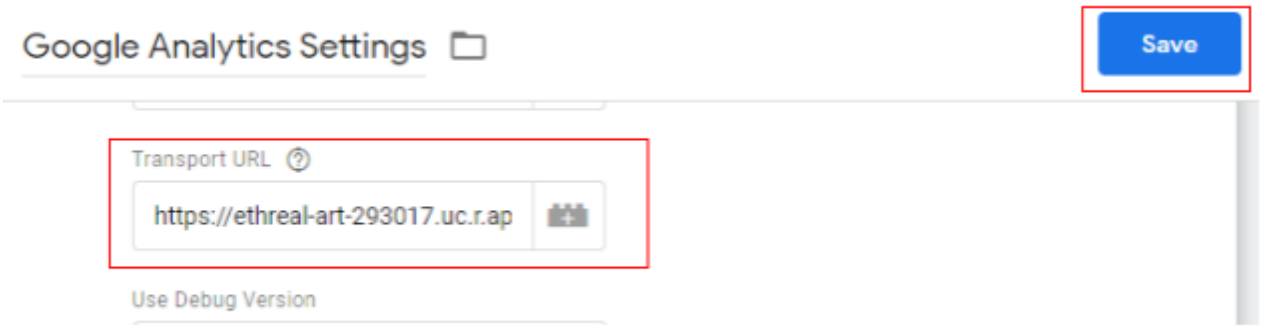
Step 5: In the ‘Transport URL’ input box, you need to provide the tagging server URL so that instead of directly sending the data to Google Analytics, it will send data to our server-side container.

You can find the tagging server URL in the admin section of the server-side GTM container.



Copy this URL

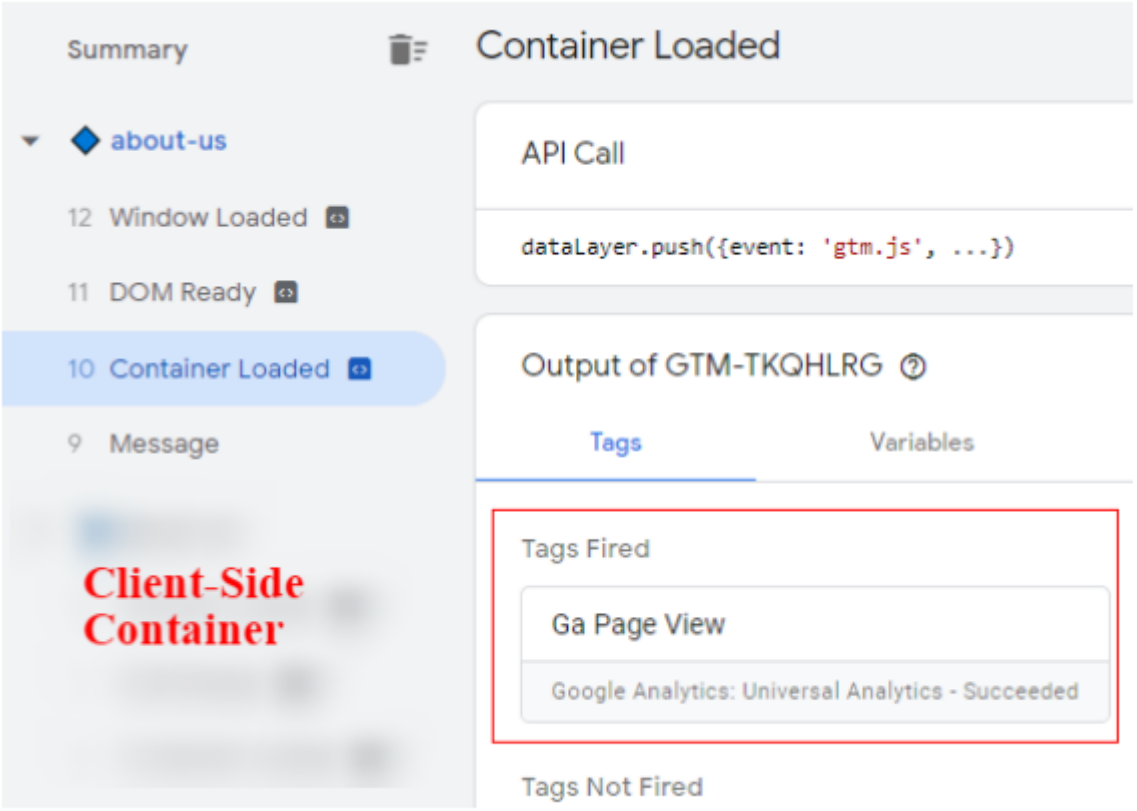
Step 6: Paste the tagging server URL in the 'Transport URL' input box and click on 'Save'.



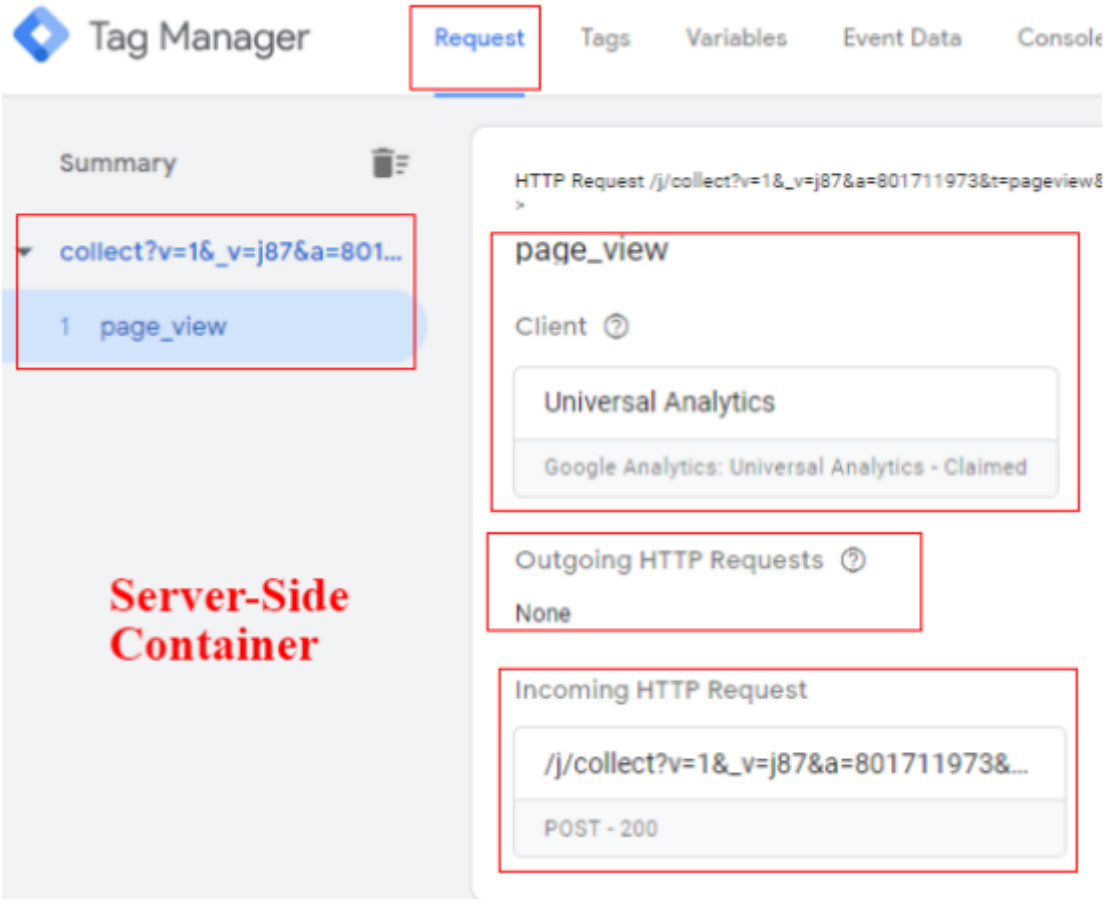
Congratulations!!! You have successfully configured your client-side container to send data to the server-side container.

You can check this by enabling the preview mode of both the containers (client-side GTM container and server-side GTM container).

As you can see from the below image, the Google Analytics pageview has fired on the client-side container.



The same is received in our server-side container preview window as a 'page_view' request.



Now our client-side GTM container is all set to send all hits (pageview, events, transactions) to the server-side container.

From the above image, you can also see that the client is Universal Analytics,

which has generated an incoming HTTP request. This means that that is collected from the website and is sent to our server-side container.

Outgoing HTTP requests are ‘None’ since we have not created any tag in the server-side container and hence no data is actually sent to Google Analytics.

Let’s proceed further and understand the preview and debug method for a server-side container.

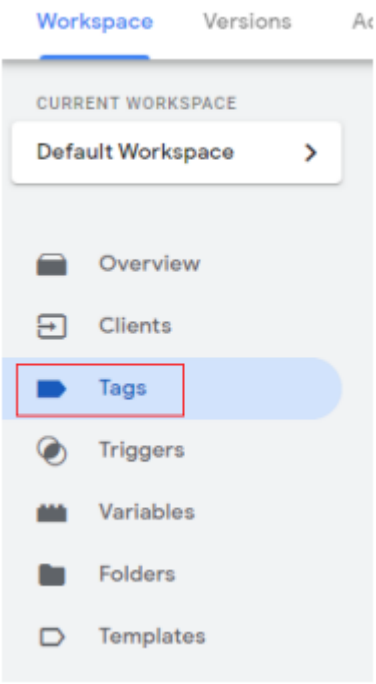
How to preview and debug the server-side container

You can preview the server-side container just like the typical client-side container by clicking on the ‘Preview’ button. It will open a new tab where you can debug your tags, variables, events, etc.

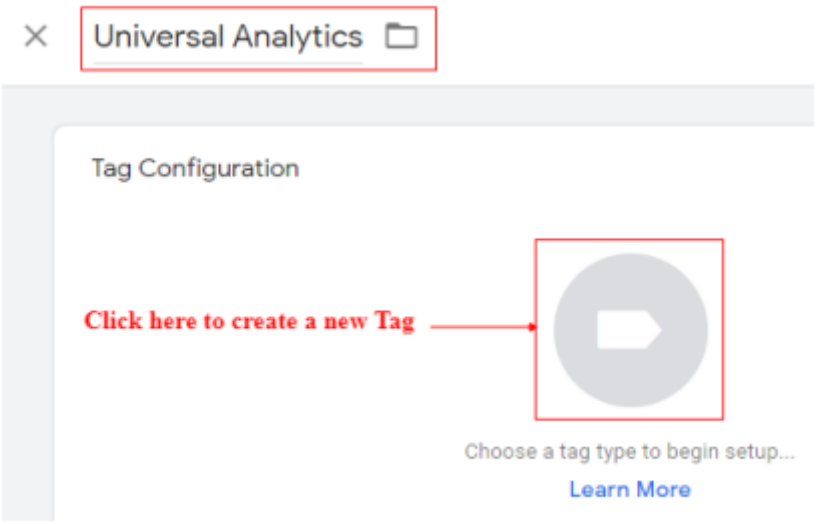
To understand it in more detail let’s create an analytics tag in the server-side container.

Step 1: Navigate to your server-side container and click on ‘Tags’ and click on

'New'.



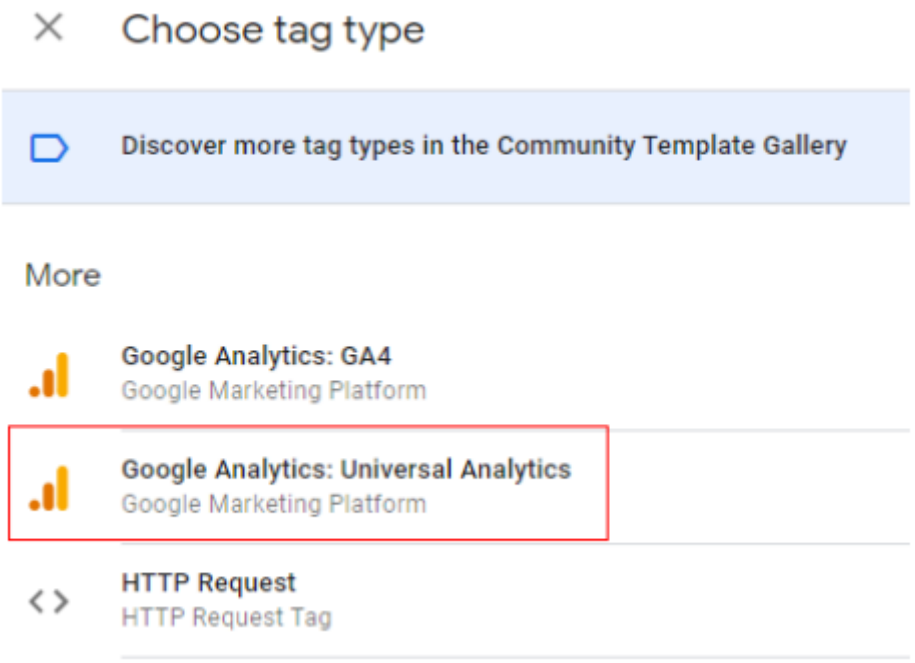
Step 2: Give your tag a descriptive name and click on 'Tag configuration'.



Step 3: An overlay will appear on the right-hand side with a list of tag templates.

Since server-side tagging is still in the beta version, there are currently only three types of tag templates. ‘Google Analytics: GA4’, ‘Google Analytics: Universal Analytics’, and ‘HTTP request’.

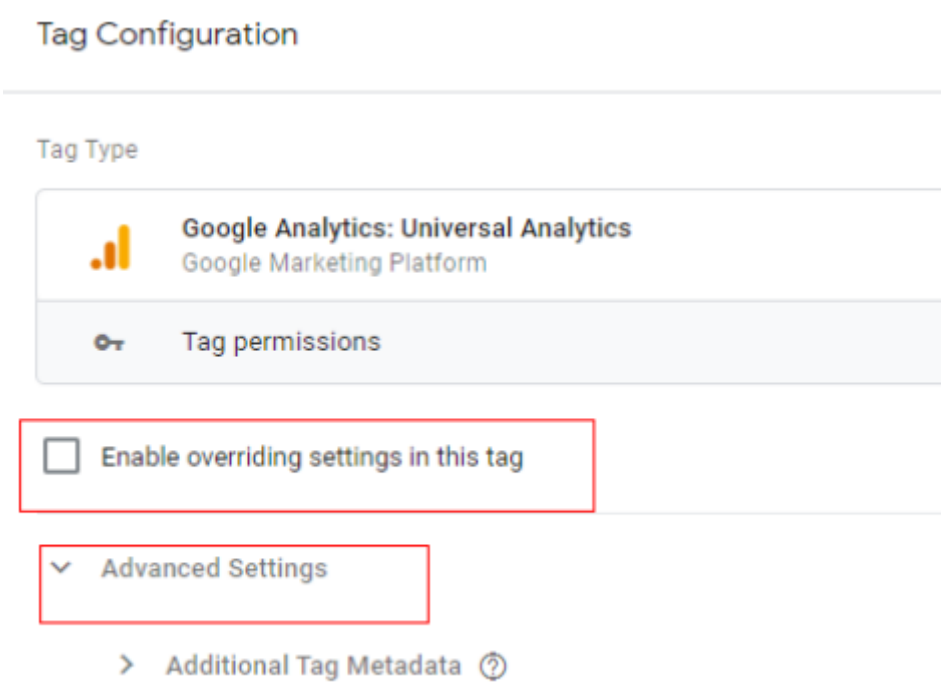
In our case, we will select ‘Google Analytics: Universal Analytics’



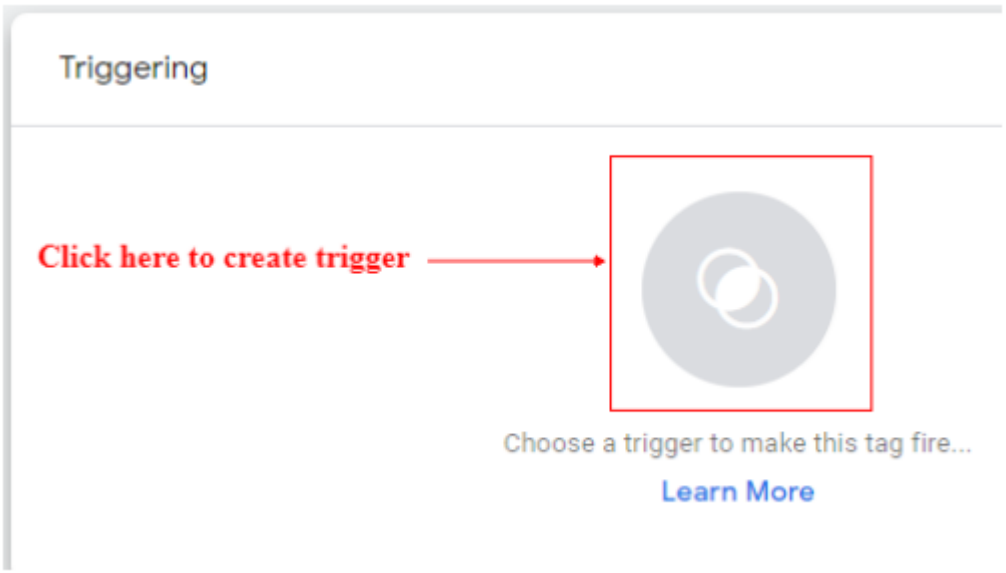
Step 4: In the 'Tag configuration' window you have the option to select the 'Enable overriding settings in this tag' checkbox.

We will keep it unchecked since, by default, the server-side container inherits the analytics ID from the client-side container.

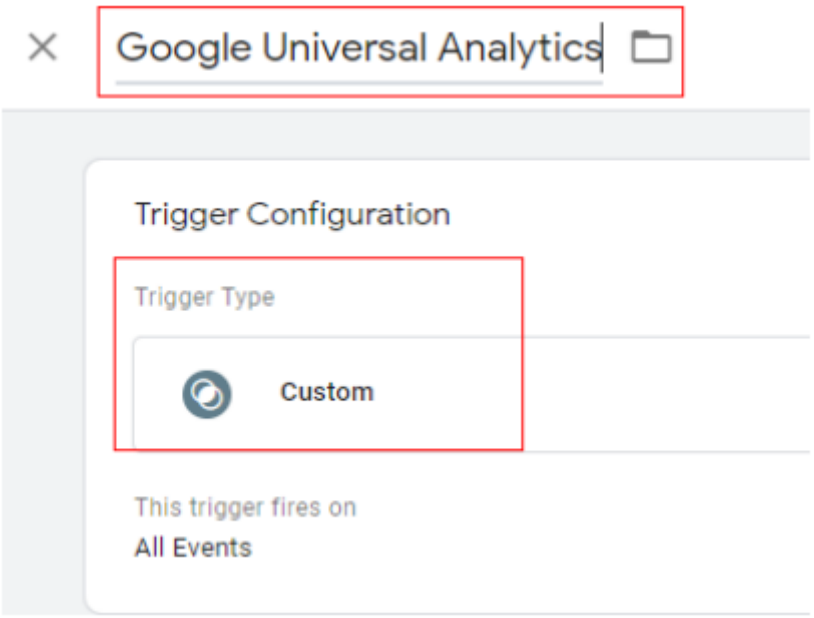
You also get the option of advanced settings which is similar to the typical GTM container settings.


Tag Configuration
Tag Type
Google Analytics: Universal Analytics
Google Marketing Platform
Tag permissions
Enable overriding settings in this tag
Advanced Settings
Additional Tag Metadata

Step 5: Click on ‘Triggering’ to create a trigger for our tag.

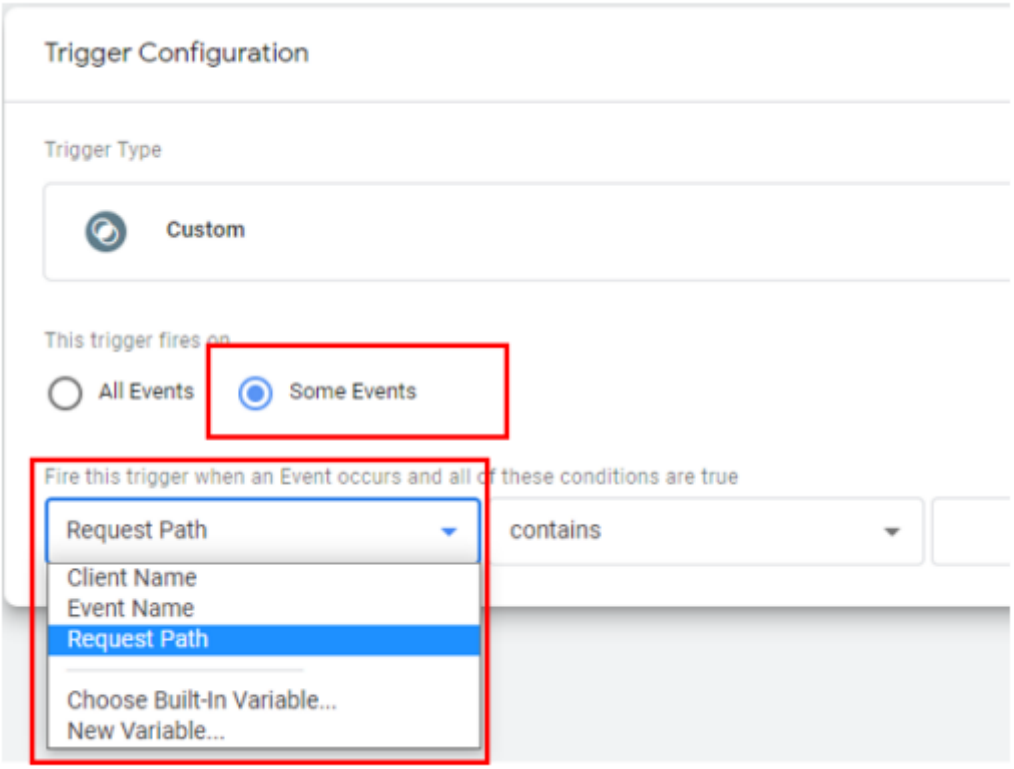


Step 6: Click on the '+' icon to create a new trigger. Give a descriptive name to your trigger.

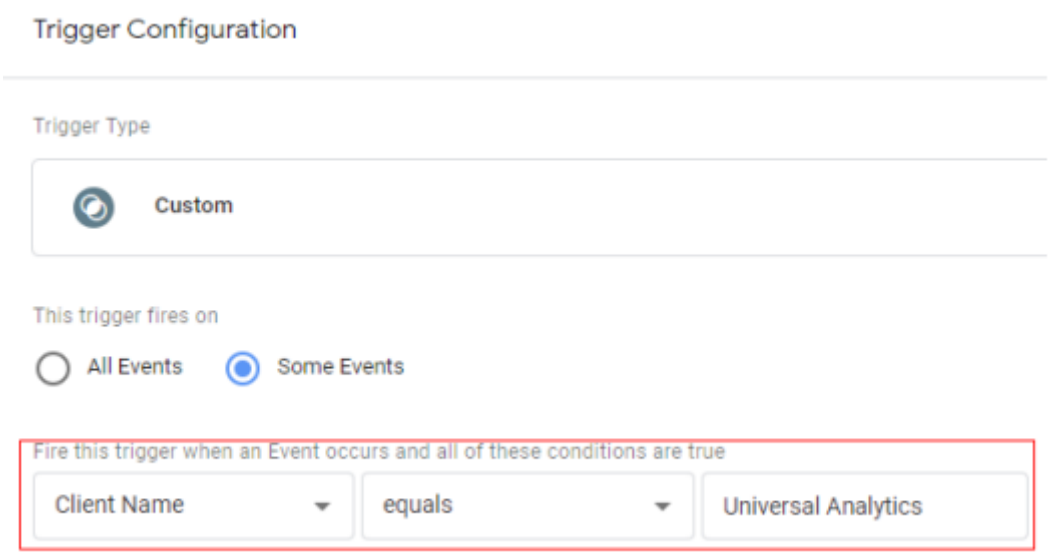


Note that like a typical GTM container you don't get the predefined trigger 'All Pages' here. In fact, there is only one trigger type available at the moment which is 'Custom Trigger'.

Step 7: Click on ‘Trigger type’ and you can specify the trigger condition here by selecting the ‘Some events’ option. You can also create a new variable to define your trigger condition or choose the built-in variables as well.



Step 8: In our case, we only want to pass data to Google analytics hence we will select the condition as “Client name” equals “Universal Analytics”.

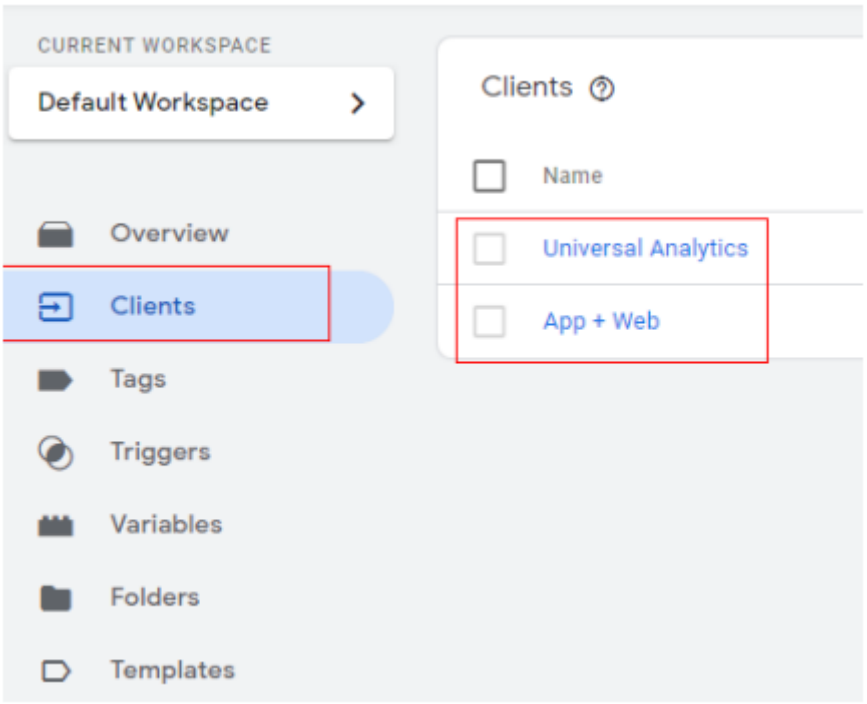


Click on ‘Save’.

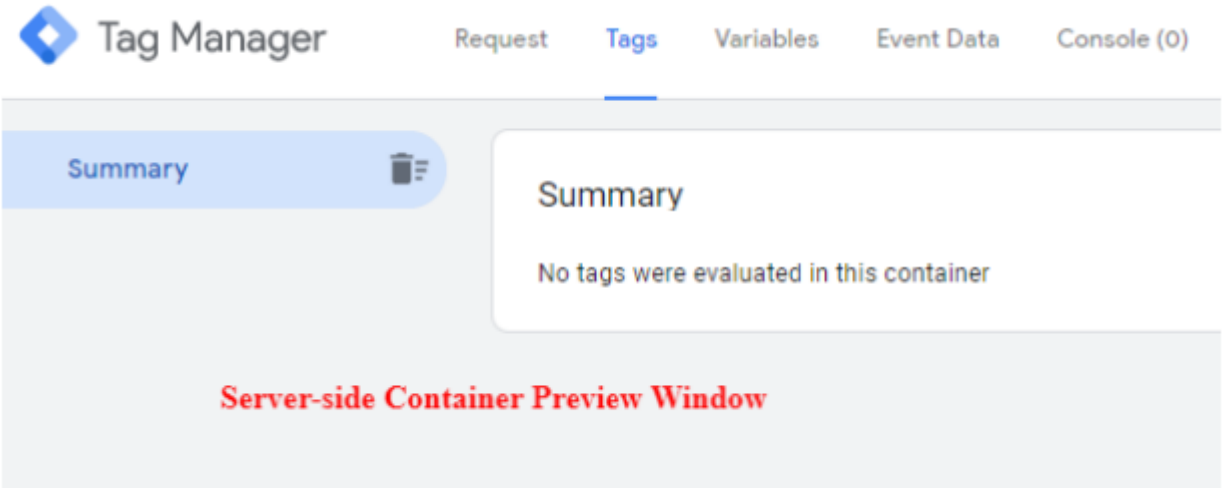
Just to give you a bit more of an idea about clients, you would have noticed that in the server-side container there is one extra menu called ‘Clients’.

Here you will find all the third-party tools to which you would like to send the data. Currently, there are only two clients available by default which are ‘Universal Analytics’ and ‘App + Web’.

Hopefully, Google will come up with various clients (Facebook, Google Ads, etc) here in near future.



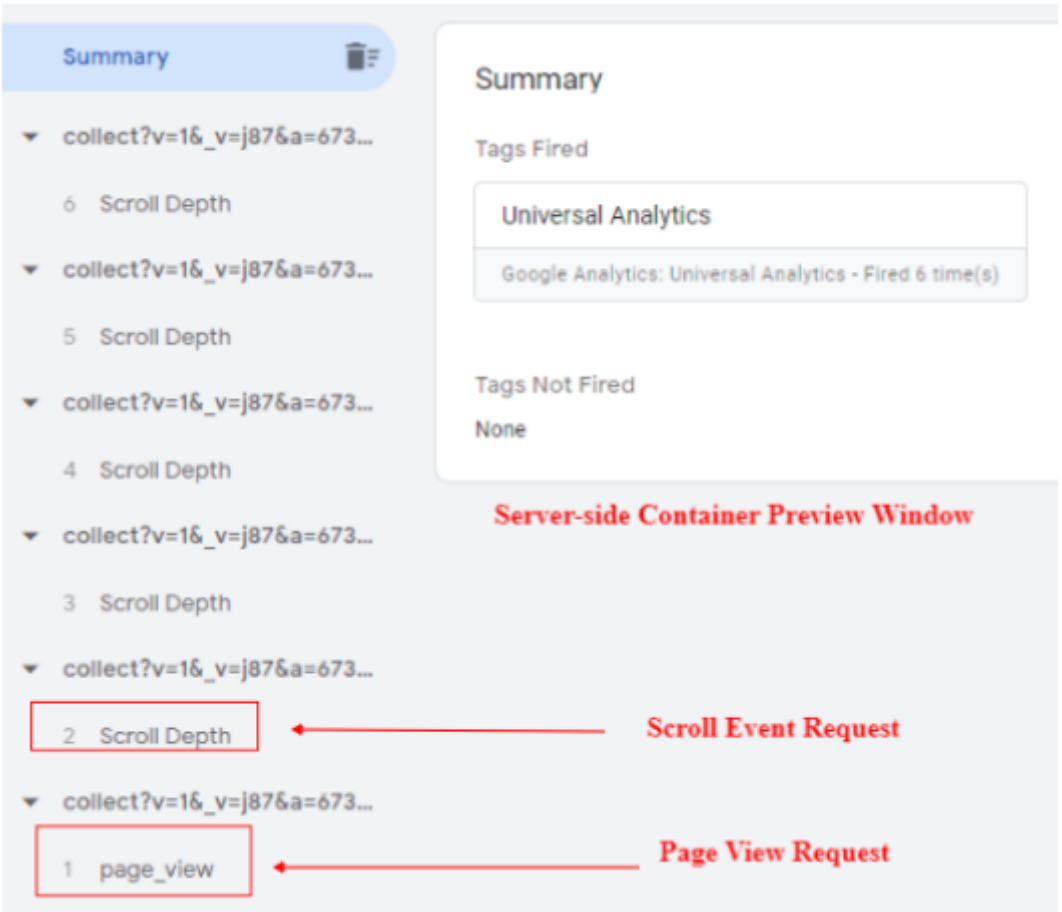
Step 9: Click on the 'Preview' button and a new window will open like the one below with a blank summary.



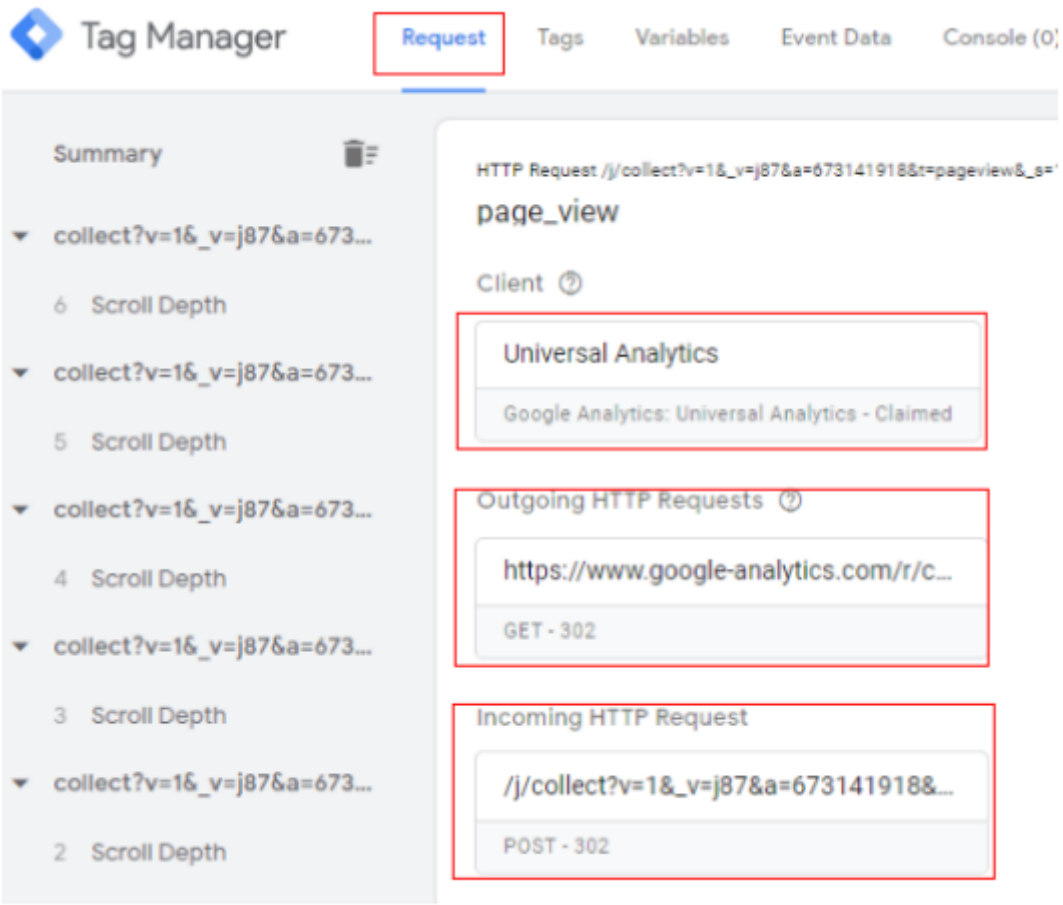
Step 10: Go to your website and visit any page which has the client-side GTM container code available on it.

Switch back to the preview window and you can see multiple hits received from the client-side container.

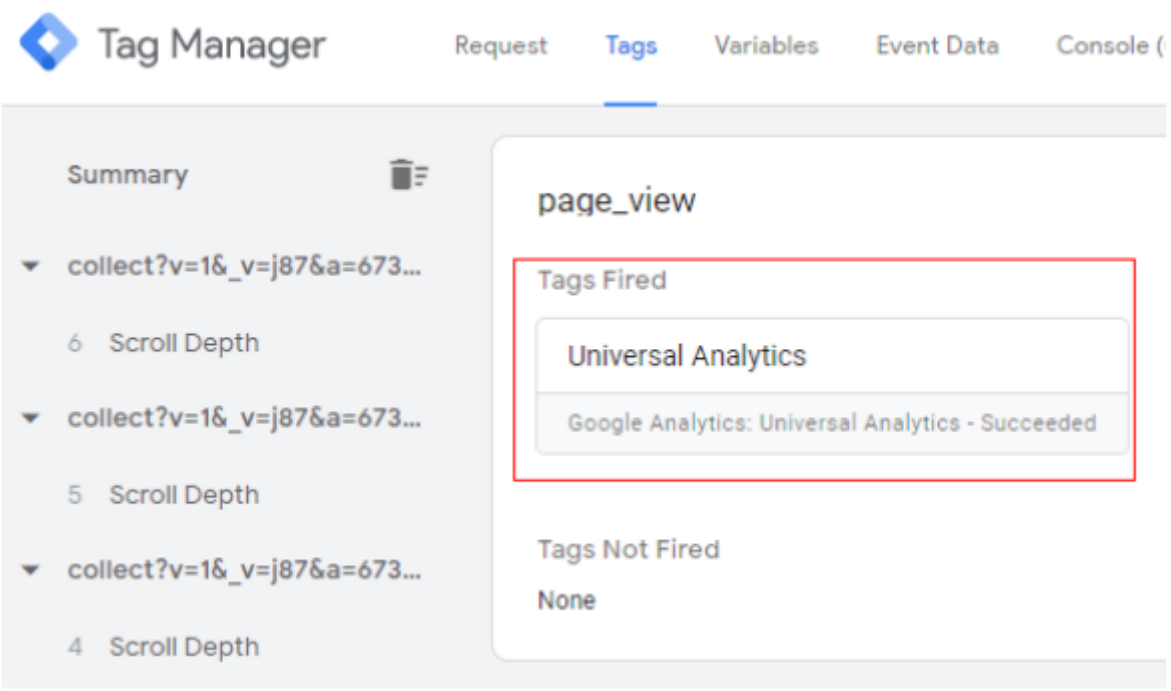
As you can see from the below image, a pageview request is generated and multiple scroll events are also generated (I have implemented scroll tracking in the client-side container).



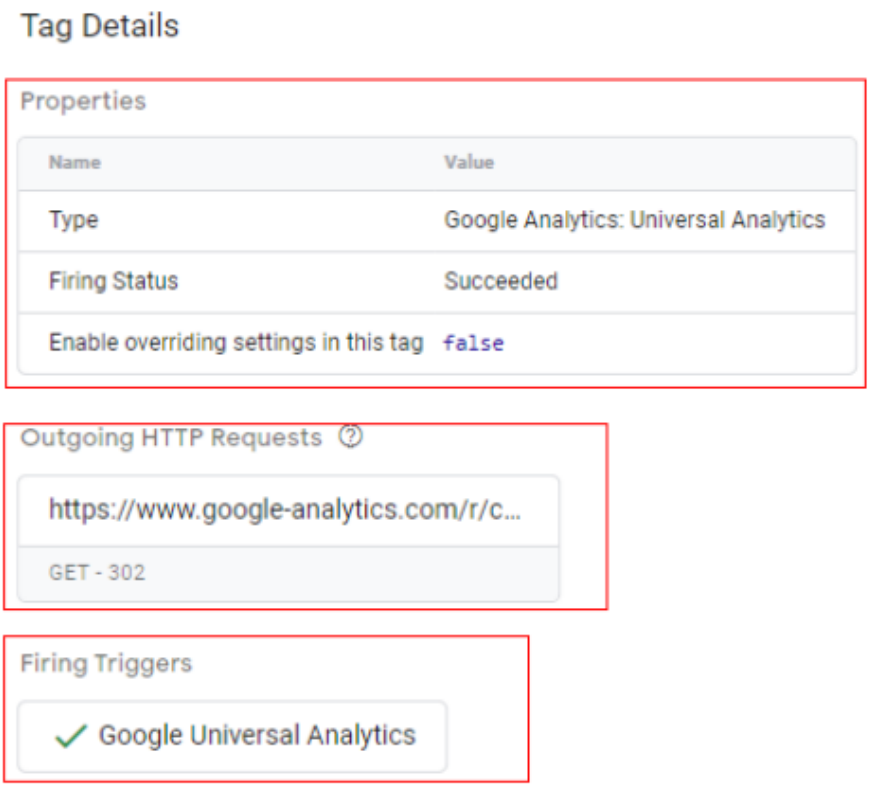
If you click on the ‘Request’ tab on the preview window you can check the Universal Analytics tag. You will also notice the incoming and outgoing HTTP requests to the Google Analytics server.



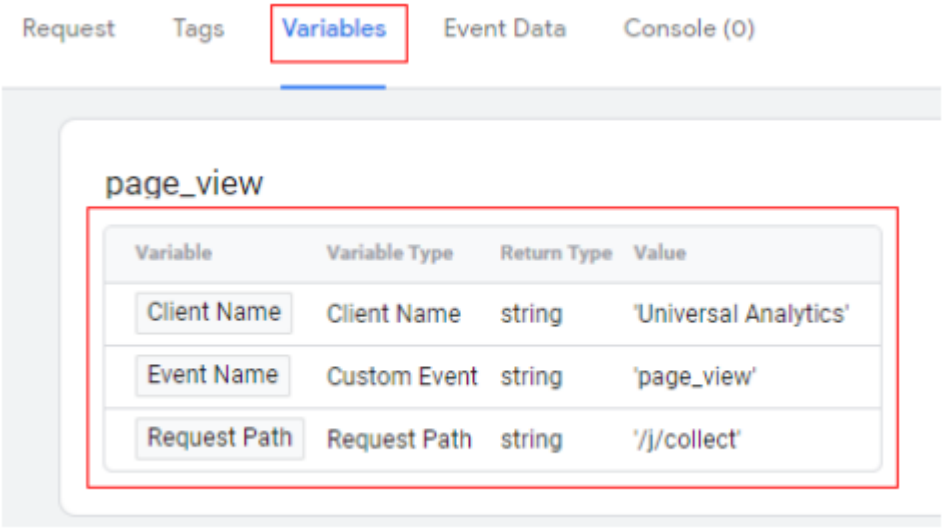
To see the details of tags fired you can click on the ‘Tags’ tab of the preview window and select ‘Universal Analytics’.



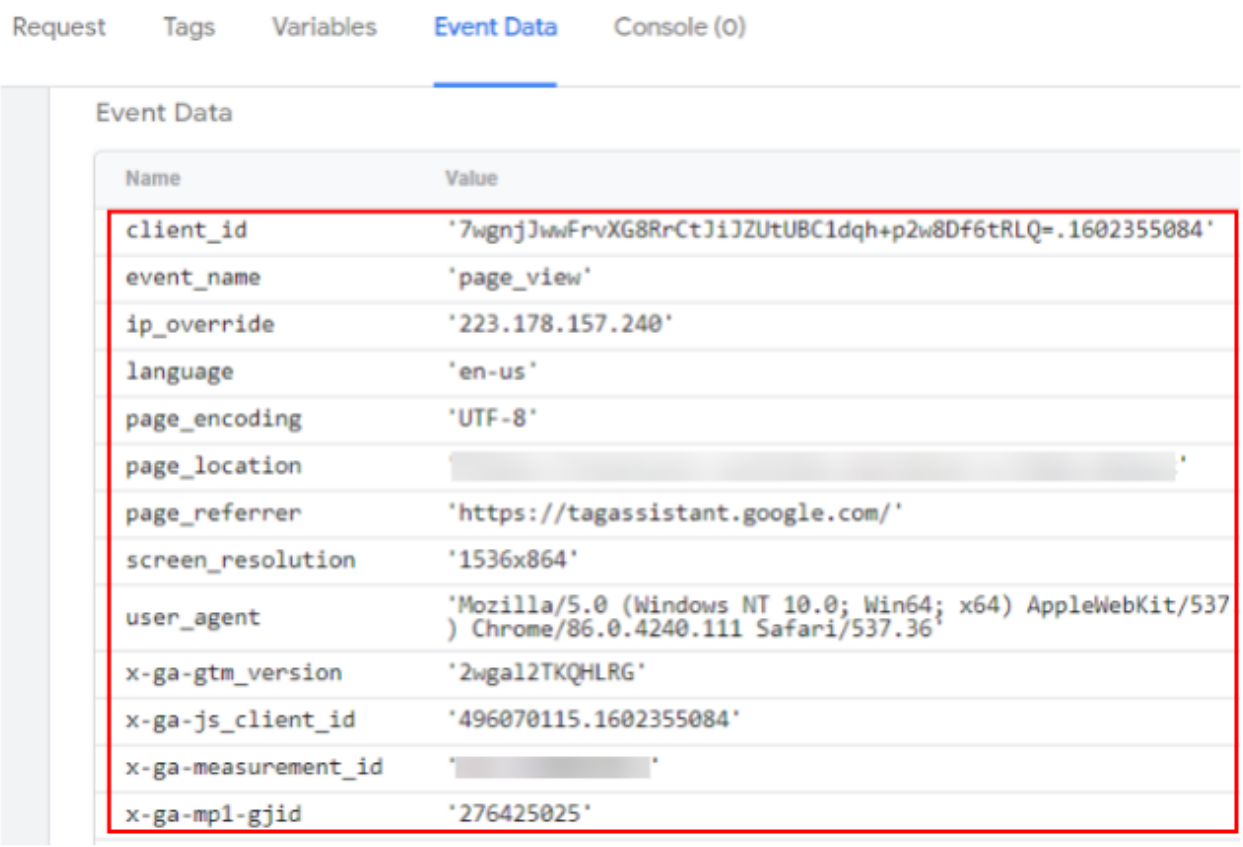
It will provide you with tag details, outgoing HTTP requests and the trigger condition of the server-side container.



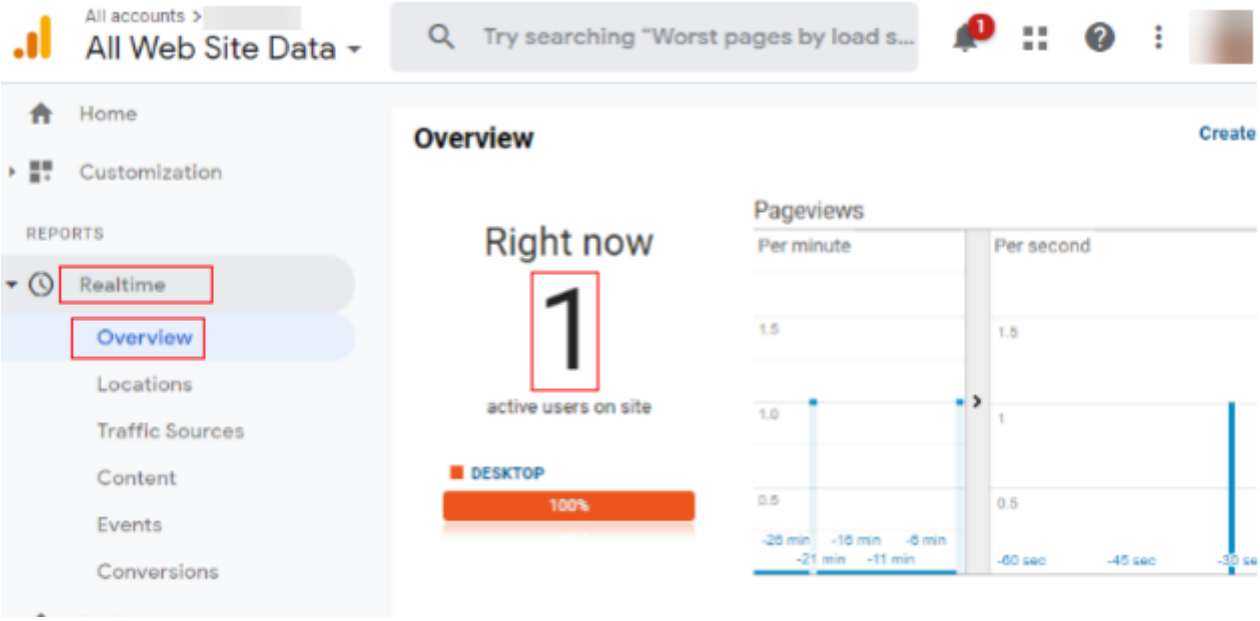
The ‘Variable’ tab in the preview window will provide the list of variables enabled and their corresponding values.



The ‘Event data’ tab will provide you with all the information which is collected and being passed to the endpoint tool, like Google Analytics in this case.



You can also check real-time reports in Google Analytics, just to be sure.



Congratulations !!! If you have reached this point you have now successfully…

1. Created a server-side container

2. Configured server

3. Create your first analytics tag in a server-side container

4. Debugged and previewed the analytics call

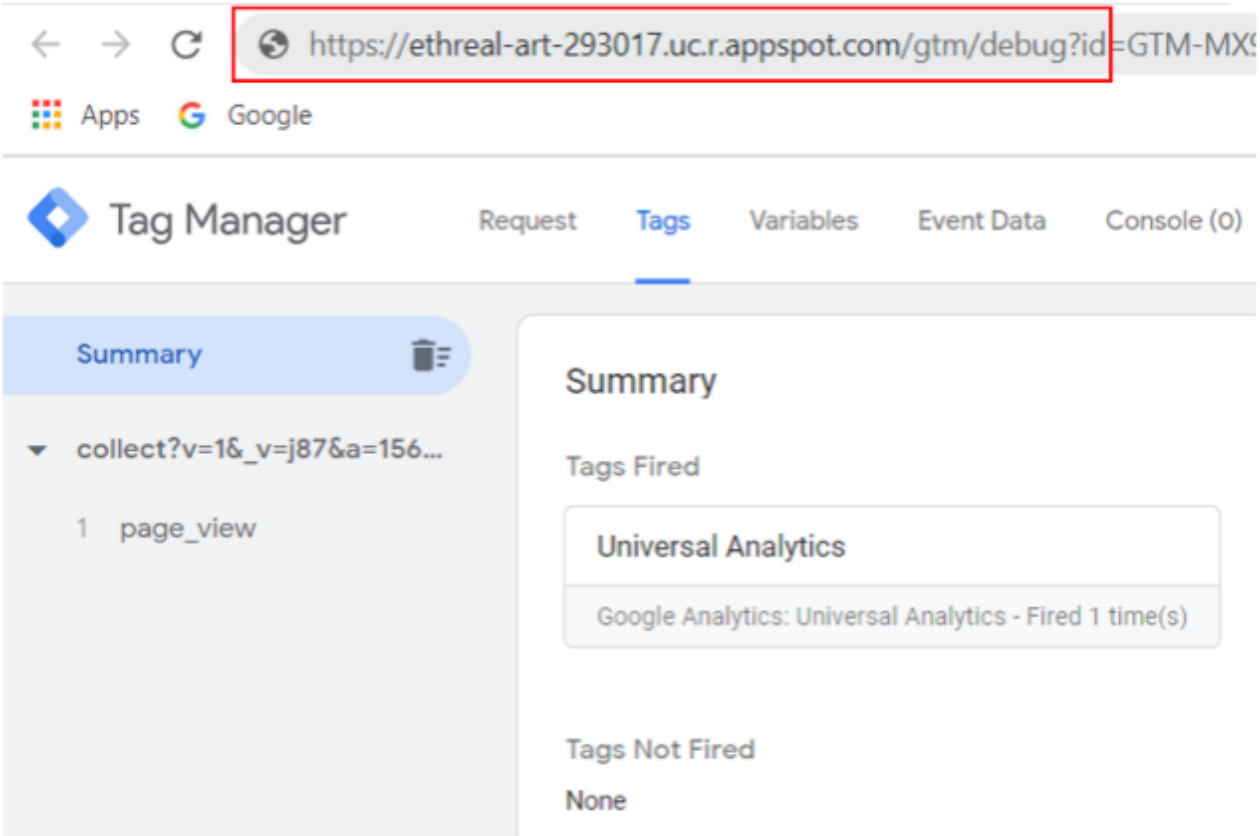
Set up a custom domain

It is strongly recommended to map your custom domain to your server container endpoint. This is because the default server-side tagging deployment is hosted on an App Engine domain.

You need to modify the deployment to use a subdomain of your website instead, in order to use first-party cookies on the user devices.

If you don’t map your custom domain you won’t be able to use the cookie information. It will consider the App Engine domain as a third-party context.

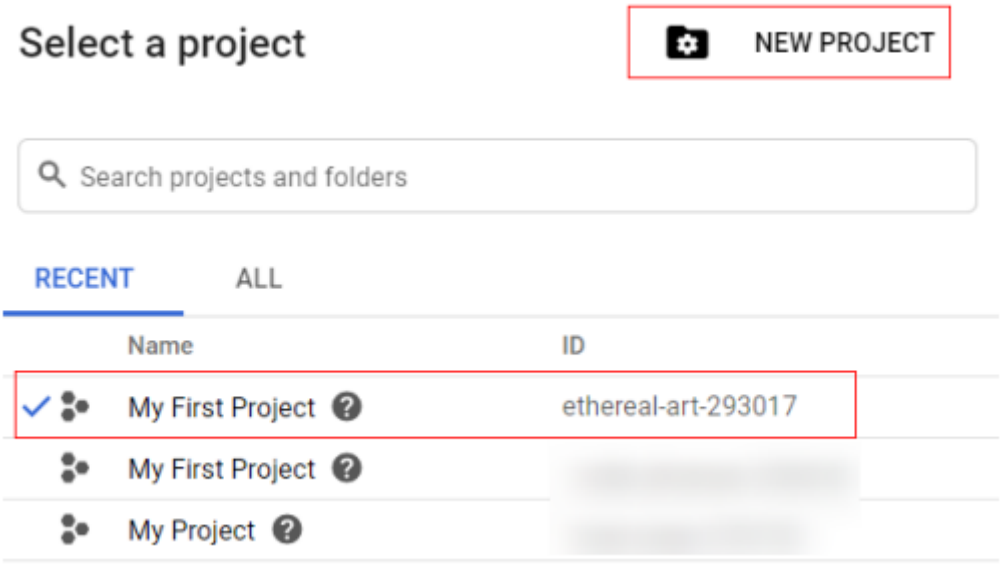
If you have noticed, the preview window opens a Google App Engine domain.



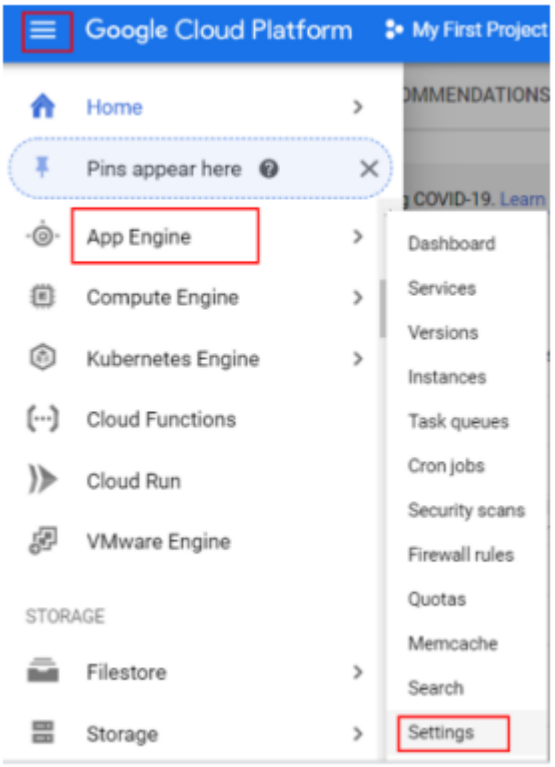
In order to consider all your server-side container in the first-party context we can

map your own domain to the server by following these steps:

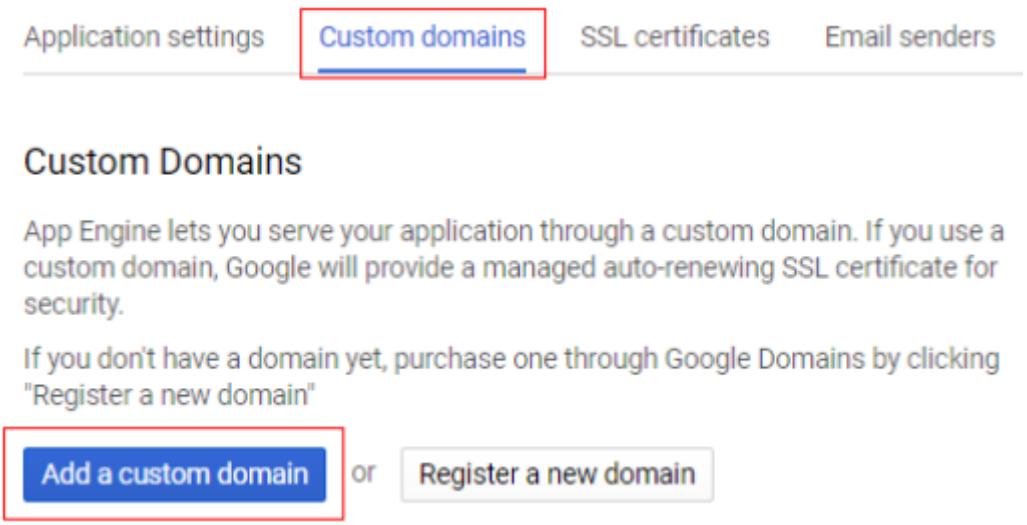
Step 1: Navigate to https://console.cloud.google.com/ and select the project you have created earlier.



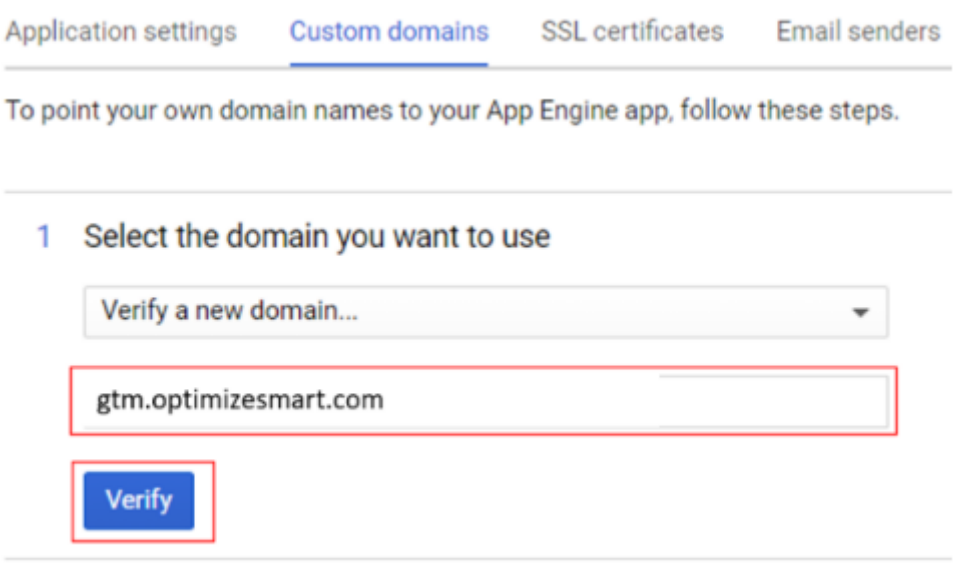
Step 2: Click on the three horizontal lines in the upper left corner and then click on ‘App Engine’. An overlay will appear, click on ‘Settings’.



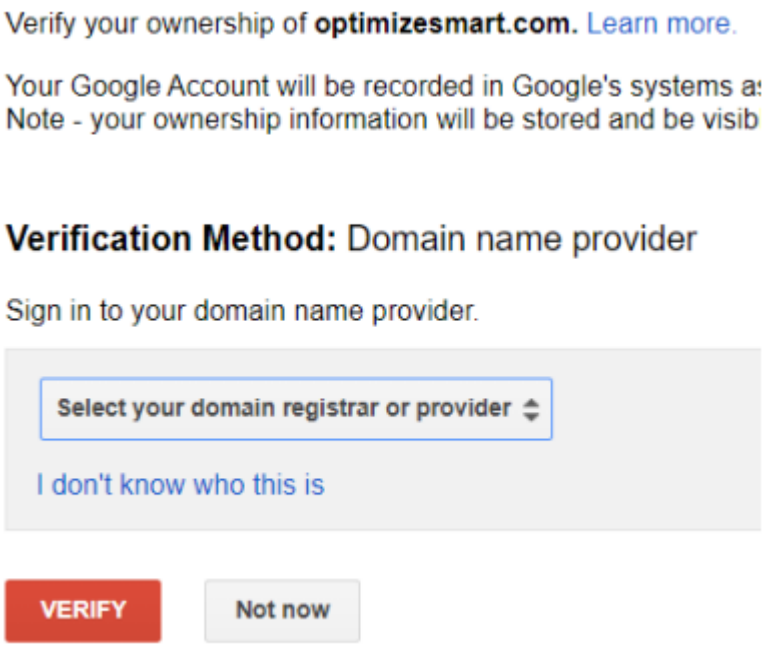
Step 3: A settings window will open. Click on 'Custom domains' and then click on 'Add a custom domain'.



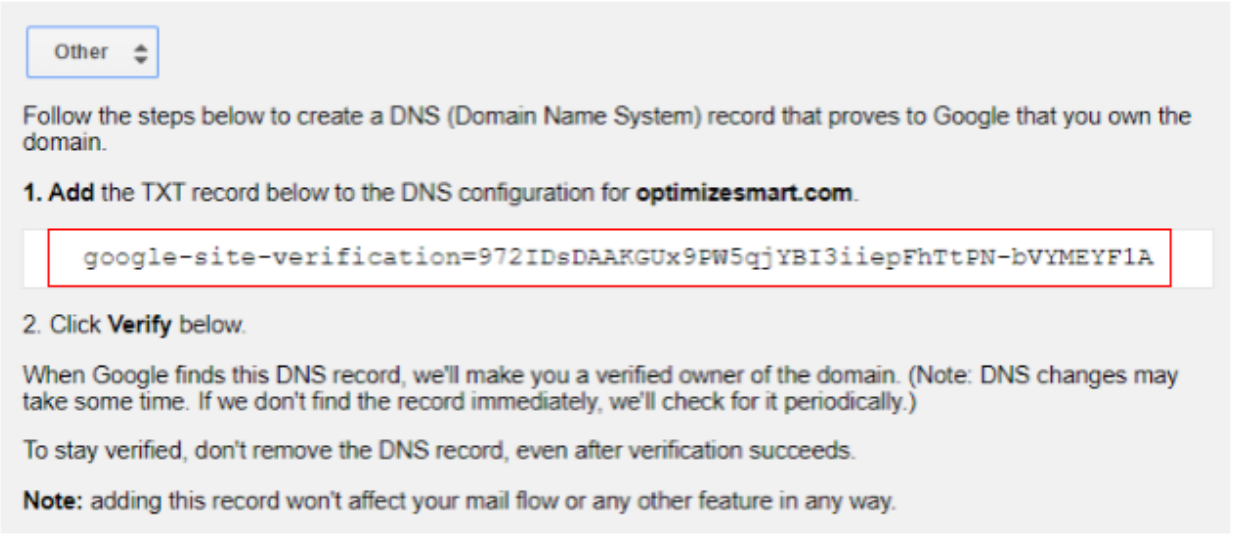
Step 4: Add your domain name without HTTPS:// or www and click on ‘Verify’. In our case, I will use gtm.optimizesmart.com



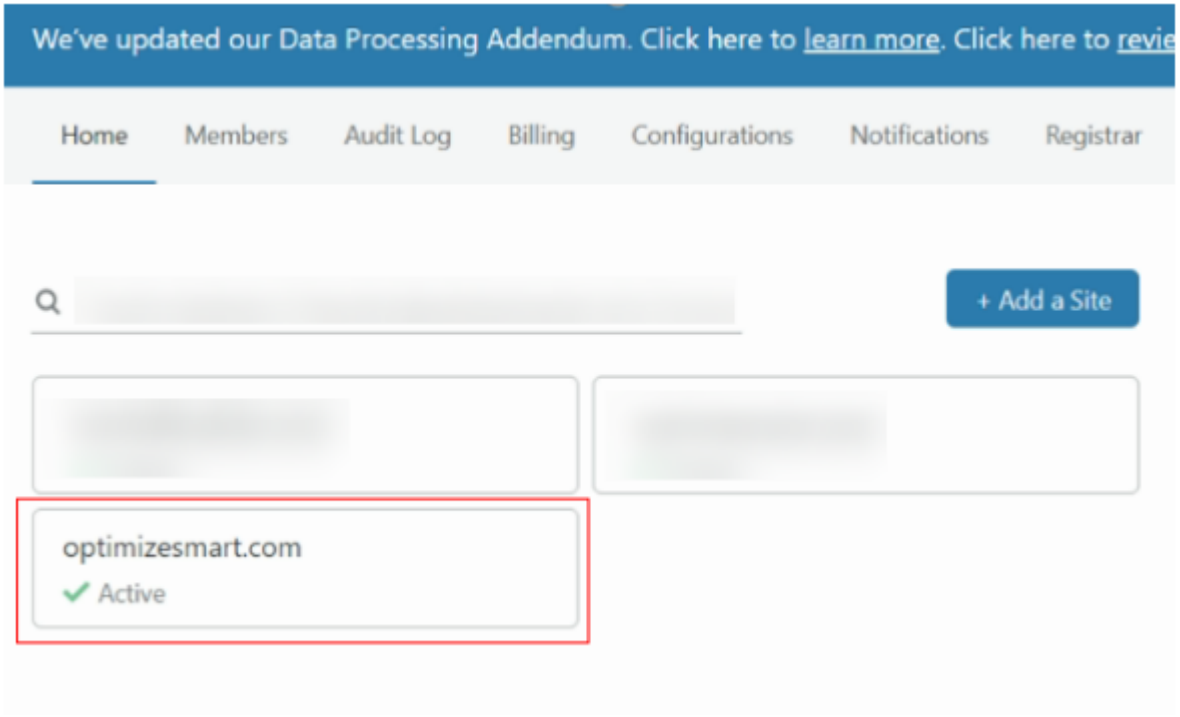
Step 5: A new window tab will open in which Google will ask you to verify your domain.



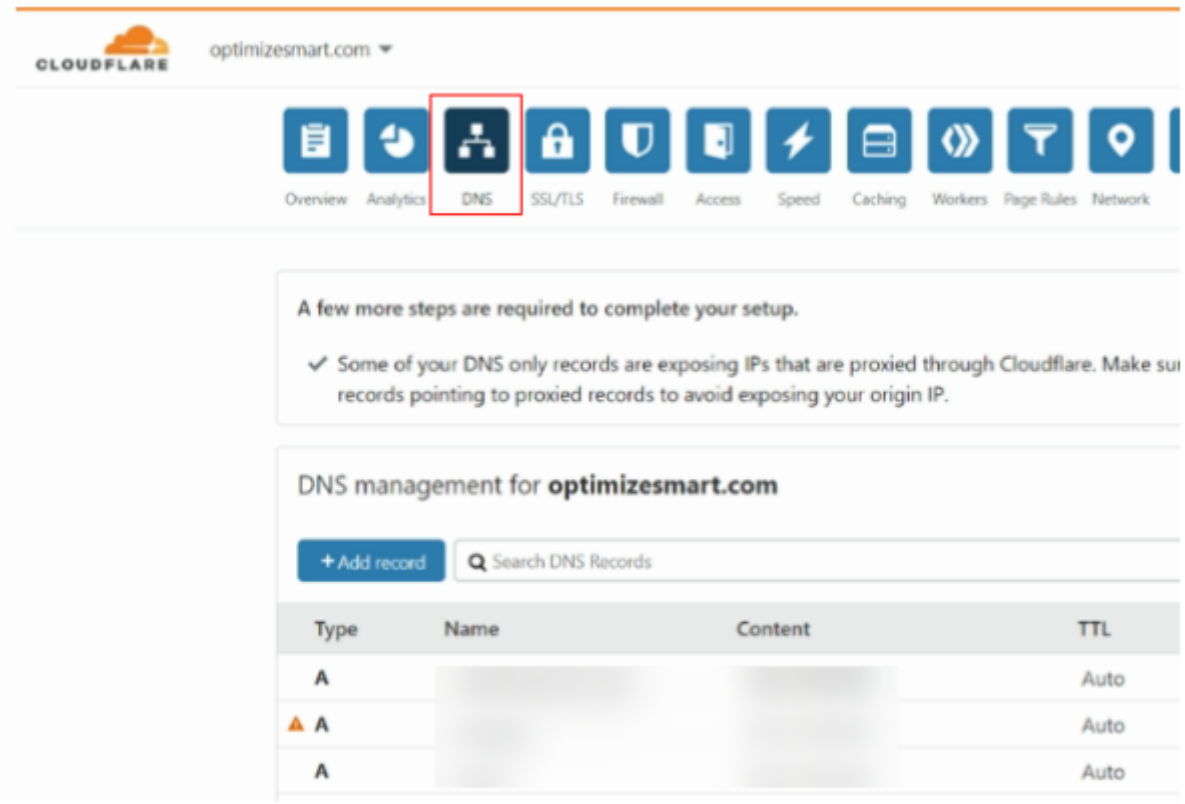
Step 6: Select 'Other' from the drop-down list and it will generate a TXT record which you will need to add to your DNS records to verify. Copy the TXT record.



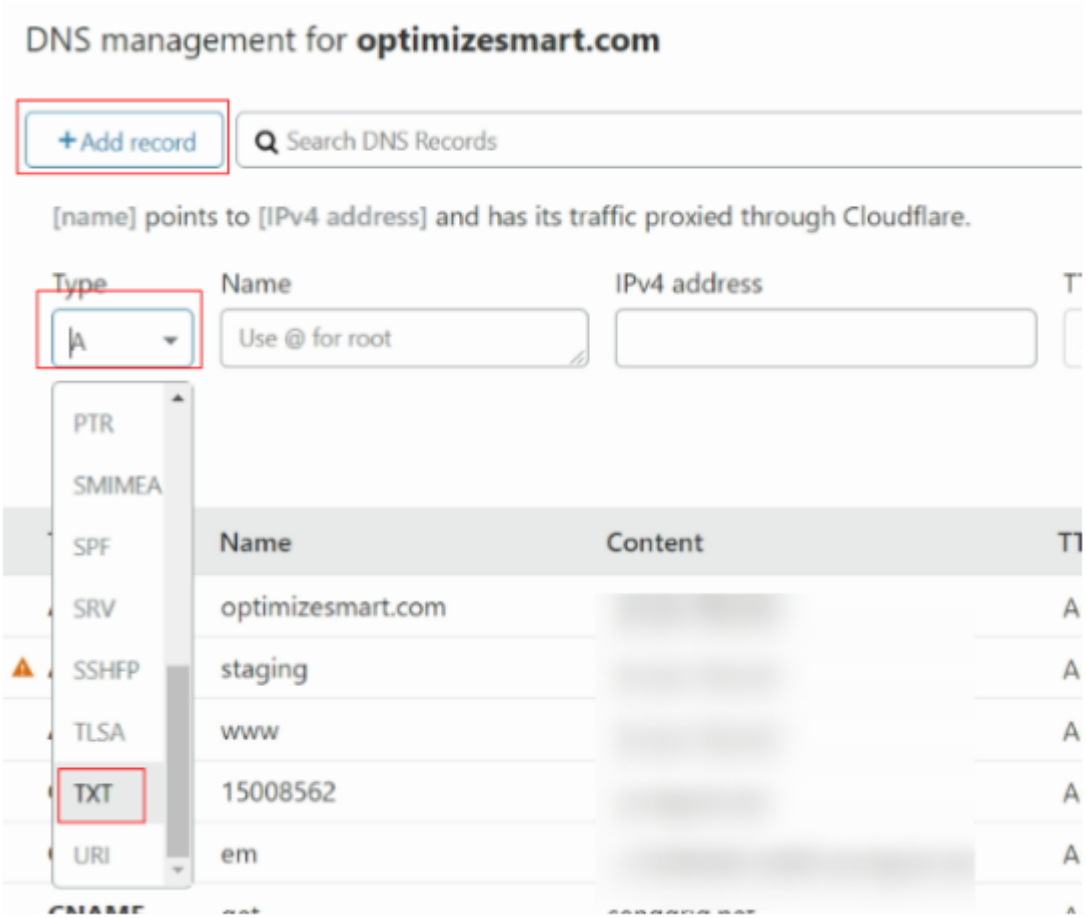
Step 7: Log in to your DNS record manager. In our case, I use Cloudflare to manage all my domains and their DNS records. Once you have logged in, select the domain name for which we will add the TXT record.



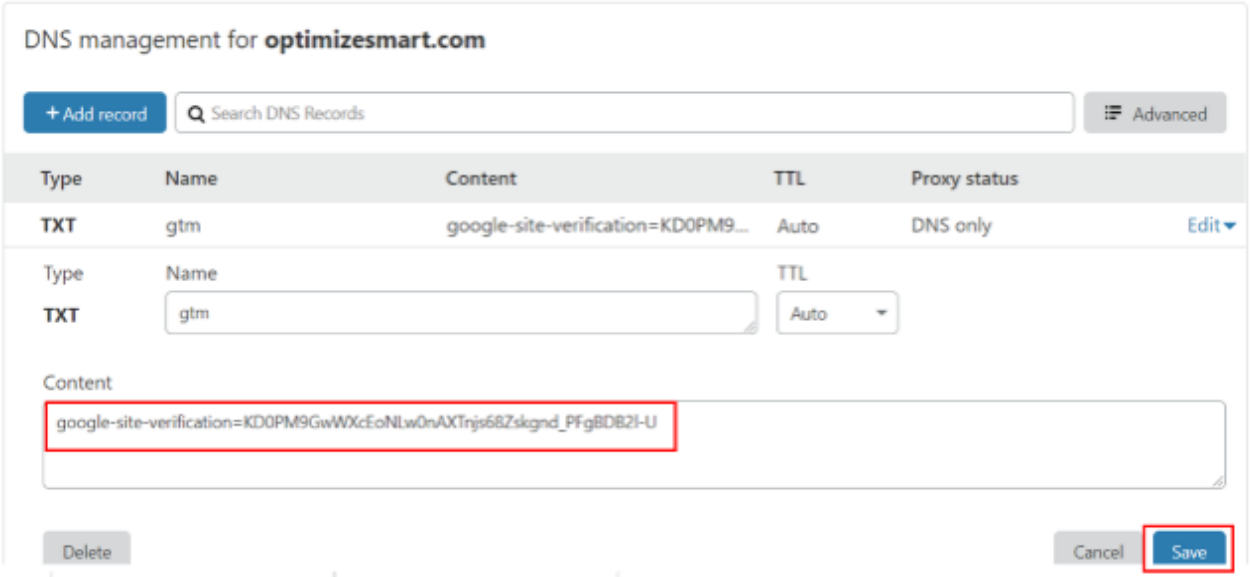
Step 8: Click on 'DNS' to enter the record settings and you will get a window like below:


CLOUDFLARE
optimizesmart.com
Overview
Analytics
DNS
SSL/TLS
Firewall
Access
Speed
Caching
Workers
Page Rules
Network
A few more steps are required to complete your setup.
Some of your DNS only records are exposing IPs that are proxied through Cloudflare. Make sur
records pointing to proxied records to avoid exposing your origin IP.
DNS management for optimizesmart.com
+ Add record
Search DNS Records
Type
Name
Content
TTL
A
Auto
A
Auto
A
Auto

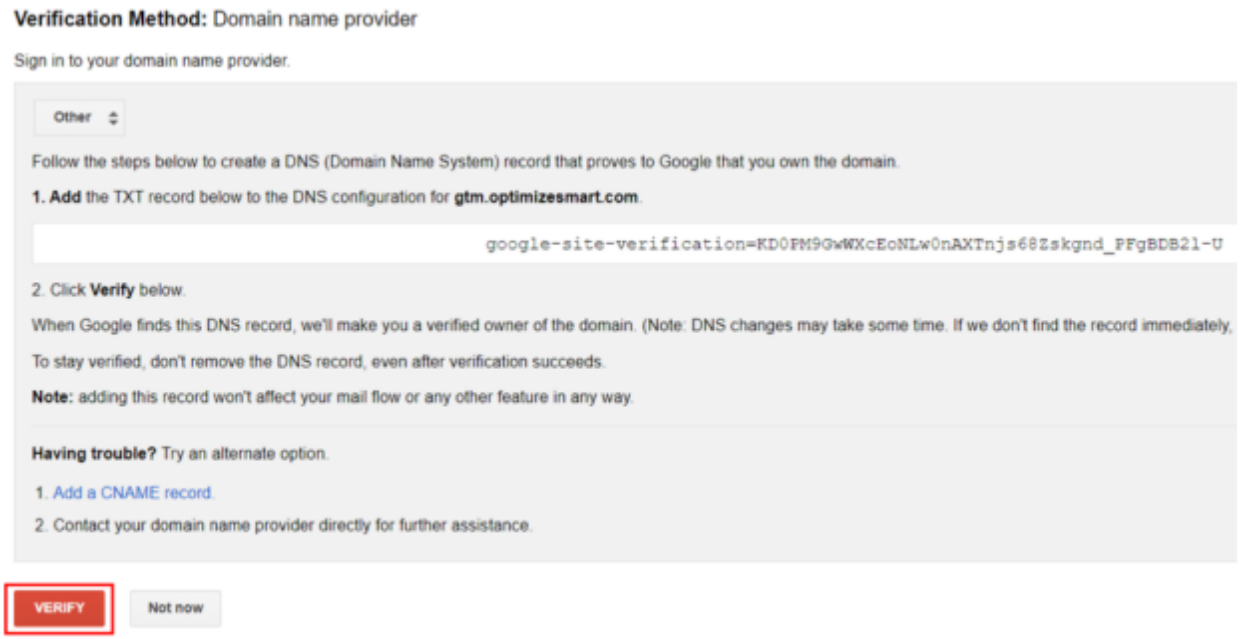
Step 9: Click on '+Add Record' and select the type as 'TXT' from the drop-down menu.



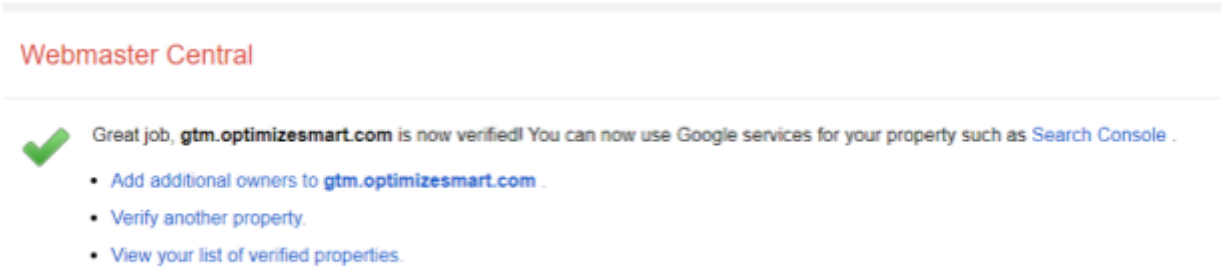
Step 10: Paste the TXT record that we have noted down earlier in step 6 and click on ‘Save’.



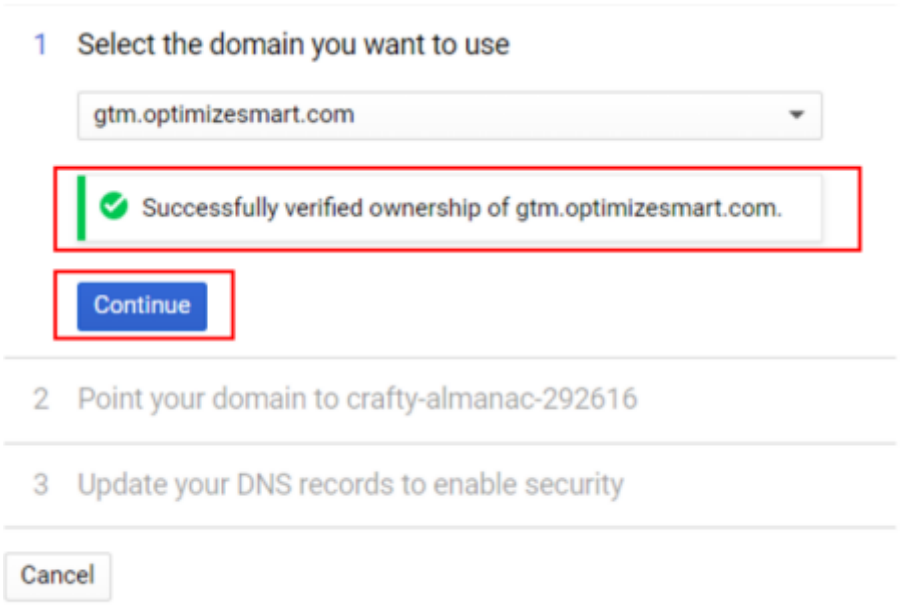
Step 11: Now go back to the Google webmaster window and click on ‘Verify’.



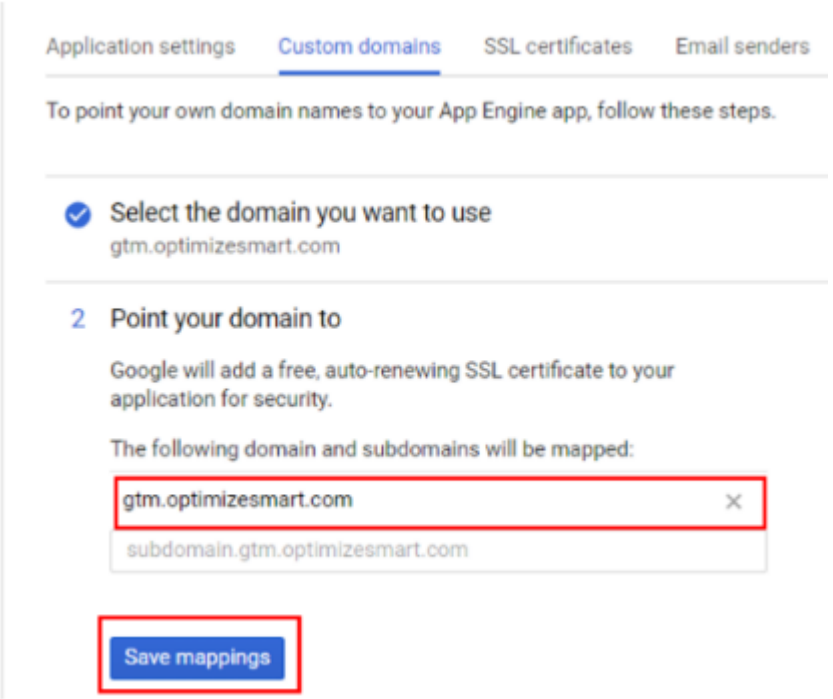
Upon successful verification you will get the below notification:



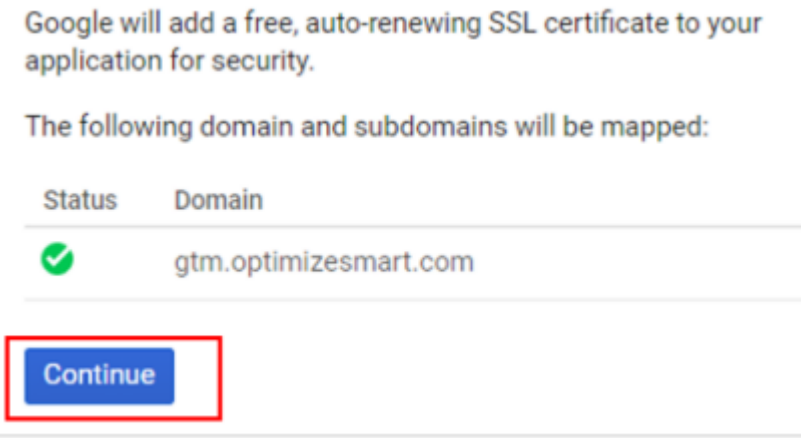
Step 12: Now go to Google cloud console. You will see the message as 'Successfully verified ownership of <your domain name>'. Now click on 'Continue'.



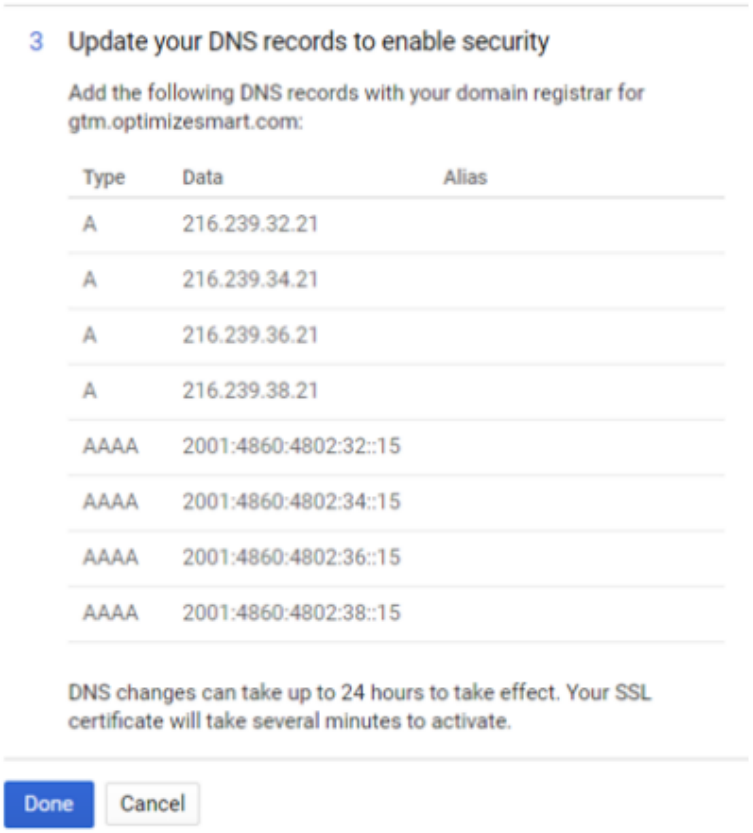
Step 13: Select your newly added domain and map it with the Google cloud project by clicking on ‘Save Mapping’.



Step 14: You will get status marked as a green tick mark. Click on ‘Continue’.



Step 15: It will now provide you with a list of records that you need to add to your DNS to enable security. Navigate to your DNS console and add the provided list of records one by one accordingly.

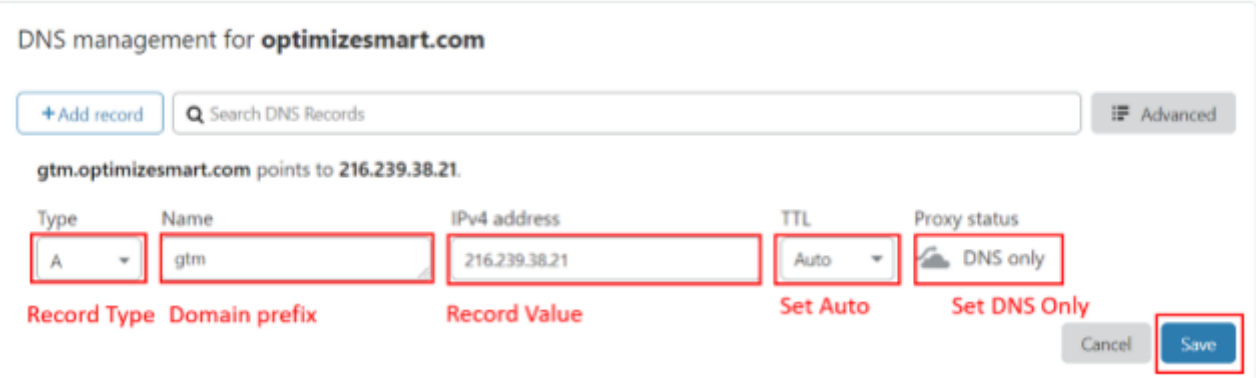


Note that there are four “A” type records and four “AAAA” type records, you need to add these records to your DNS using the domain names like below.

- Record type: “A” or “AAAA”
- Name: gtm (prefix to our main domain)
- Ipv4 address: record values which are provided in earlier step for ex: 216.239.38.21
- TTL: Set to auto

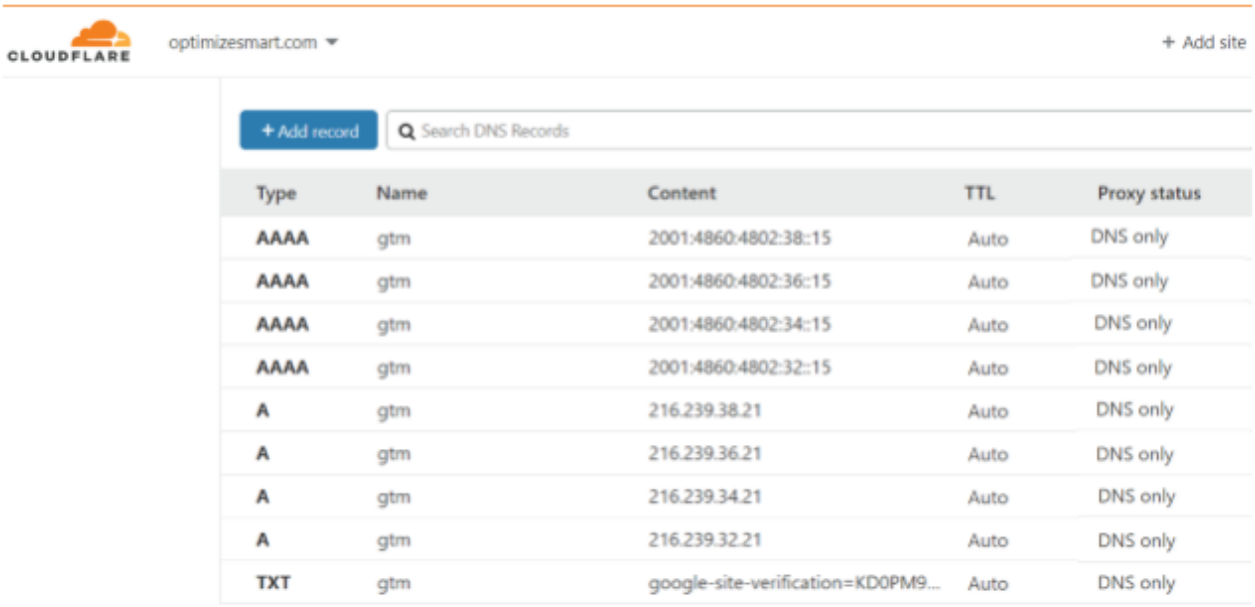
- Proxy status: set to DNS only

Click on ‘Save’.

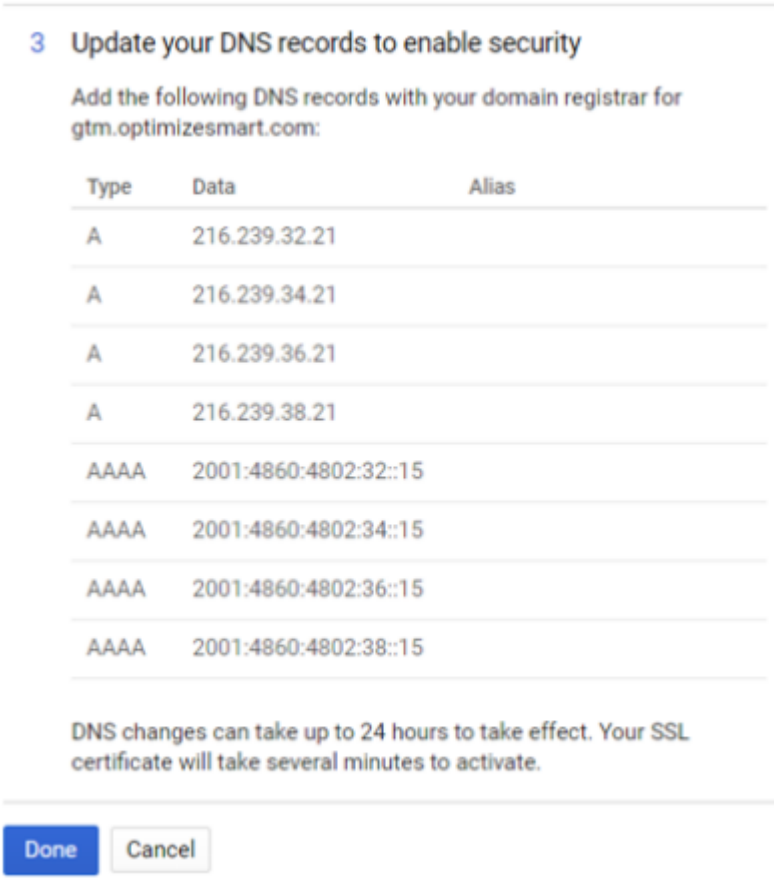


Repeat this step for the remaining records.

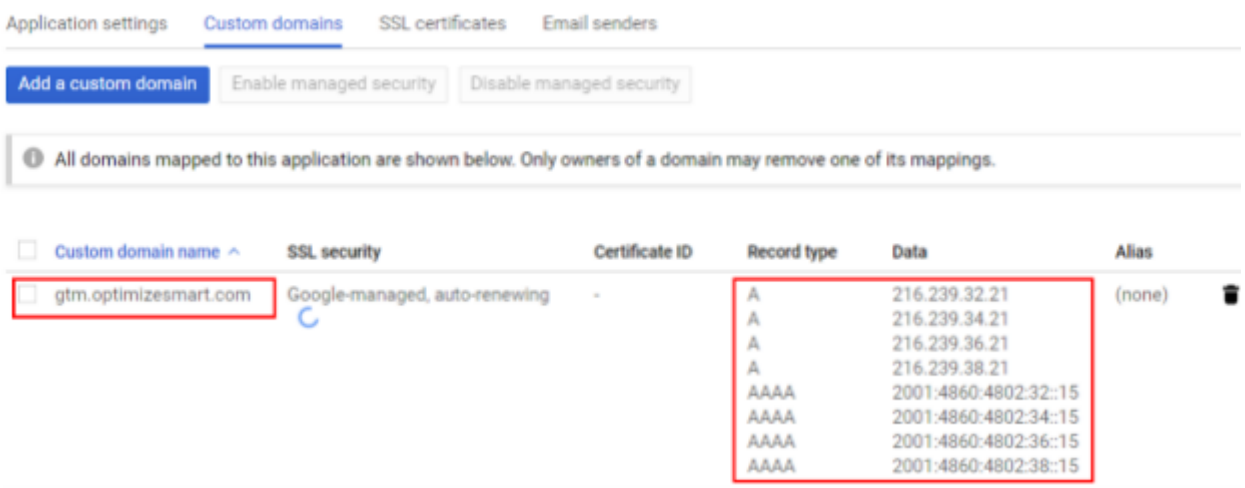
Step 16: Once you have added all the provided records from the list you will see the below window:



Step 17: Navigate to Google cloud console and click on ‘Done’.



You will see the below screen with all records mapped to your custom domain:



It may take an hour or two for mapping to update in Google cloud. Once updated you will see the preview mode window in your own domain mapped.

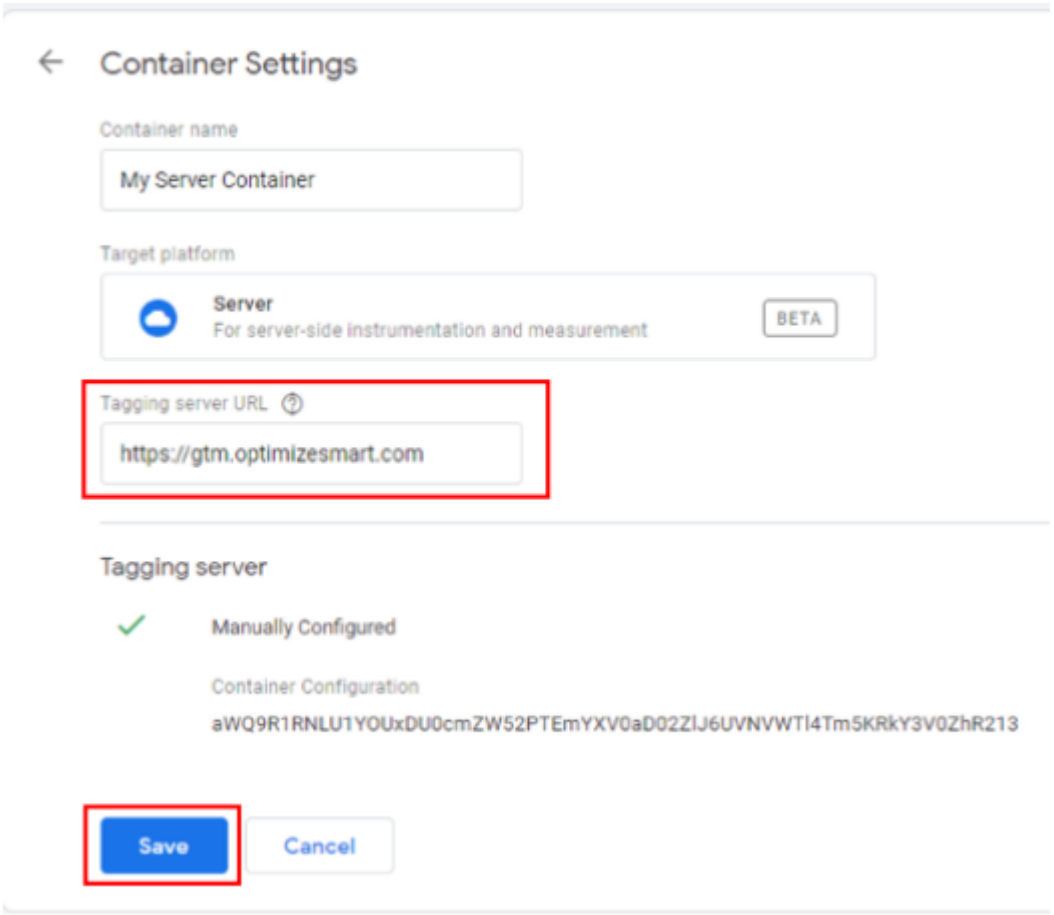
But hold on, there is one more setting that we need to change before you can validate your custom domain mappings.

Step 18: Navigate to your server-side container admin settings and change the tagging server URL to your newly mapped domain.

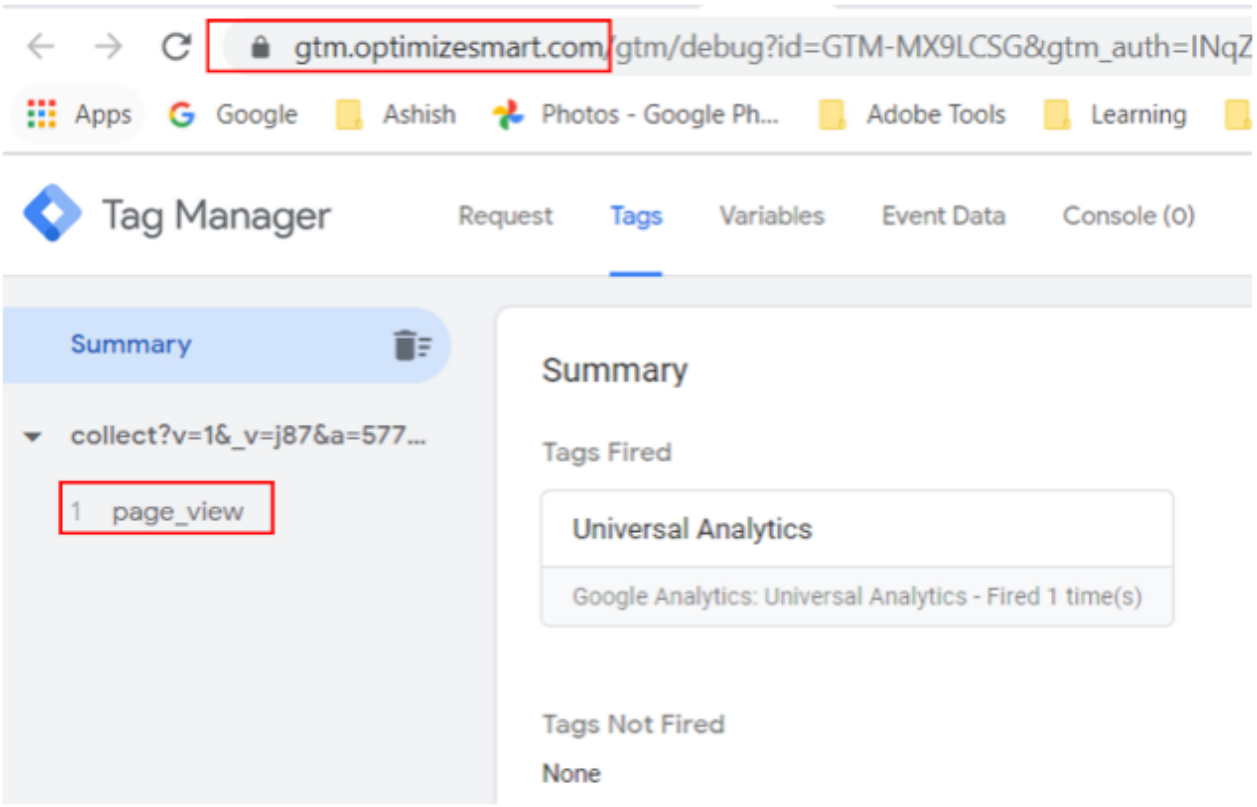
For example, earlier our tagging server URL was “https://ethreal-art-293017.uc.r.appspot.com” (App Engine domain). Change it with your new domain URL like ‘https://abc.yourdomainname.com’.

In our case, I will use ‘https://gtm.optimizesmart.com’.

Click on ‘Save’.



Step 19: Enable the server-side container preview mode and visit your website. You will now see your custom domain URL along with the page_view tag firing.



Great, you have configured your custom domain to the server-side GTM container.

You are doing Google Analytics all wrong. Here is why...

I have dealt with hundreds of Google Analytics accounts in my career.

I have seen a lot of issues from incorrect tracking code, selecting the wrong KPIs to analyzing data without using custom reports or advanced segments.

But do you know the biggest issue of all in Google analytics?....

It is the "misinterpretation of analytics data".

Many marketers make the mistake of crediting conversions to the wrong marketing channel.

And they seem to be making this mistake over and over again.

They give the credit for conversions to the last touchpoint (campaign, ad, search term...).

They can't help themselves because they believe that the Google Analytics reports are 'what you see is what you get'.

But they are actually 'what you interpret is what you get'.

This has resulted in marketers making wrong business decisions and losing money.

All the data you see in Google Analytics reports **today lies to you unless you know exactly how to interpret it correctly.**

For example, let's talk about direct traffic.

All untagged or incorrectly tagged marketing campaigns from display ads to emails could be reported as direct traffic by Google.

Whenever a referrer is not passed, the traffic is reported as direct traffic by Google.

Mobile applications don't send a referrer. Word/PDF documents don't send a referrer.

'302 redirects' sometimes cause the referrer to be dropped. Sometimes browsers don't pass the referrer.

During an HTTP to HTTPS redirect (or vice versa) the referrer is not passed because of security reasons.

All such traffic is reported as direct traffic by Google.

So on the surface, it may look like that most people are visiting your website directly but this is not usually the case.

But this analysis does not end here, because you are still not looking at the complete picture.

People do not always access your website directly and then make a purchase straight away.

They are generally exposed to multiple marketing channels (display ads, social media, paid search, organic search, referral websites, email etc) before they access your website directly.

Before they make a purchase.

So if you are unaware of the role played by prior marketing channels, you will credit conversions to the wrong marketing channels.

Like in the present case to direct traffic.

To get this type of understanding you need to understand and implement web analytics.

But **you learn data analysis and data interpretation from web analytics and not from Google Analytics.**

The direction in which your analysis will move will determine the direction in which your marketing campaigns will move.

You get that direction from ‘web analytics’ and not from ‘Google Analytics’.

Web/Digital analytics is not about Google Analytics (GA) or Google Tag Manager (GTM). It is about analyzing and interpreting data, setting up goals, strategies and KPIs.

It’s about creating a strategic roadmap for your business.

That's why the knowledge of web/digital analytics is so important.

So, what I have done is put together some completely free training for you.

This training will teach you what digital analytics really is and how I have been able to leverage it to generate floods of new sales and customers.

I will also show you how you can copy what I have done to get similar results.

You can sign up for the free training here:

Reserve My Seat Now

I hope you find it helpful.

All the best,

Himanshu