



Media Metrix Reporting

Publisher Tag Specification

document version: 2.4.0; released on December 14, 2020

for further information, please contact:

Comscore
Tag Support
+1 866 276 6972

Contents

| | |
|--|---|
| Introduction | 3 |
| Unified Methodology | 3 |
| What are Tags? | 3 |
| Implementing the Publisher Tag | 4 |
| Appendix A: Additional Instructions for Specific Use Cases | 5 |
| Dynamically Generated Webpages | 5 |
| Custom Implementation Without JavaScript | 5 |
| Appendix B: User Consent Tagging | 6 |
| Using a Consent Management Platform | 6 |
| Manually Communicating Consent | 6 |
| Appendix C: Update an Existing Comscore Publisher Tag Implementation | 8 |
| Appendix D: Implementation FAQs | 9 |

Introduction

In our continuous efforts to provide the market with highest quality audience measurement data, Comscore Media Metrix uses the *Unified Digital Measurement* methodology.

Unified Methodology

Unified Digital Measurement is a best-of-breed approach that puts the consumer — the human, not the machine — at the center of Comscore measurement and relies on Panel data as well as Census or Server data. Data from the Comscore panel provides a 360° view of the consumer including demographics; cross visitation, etc. while Census data from tags provides overall, site-specific usage activity.

This document explains the benefits of Comscore Unified Digital Measurement and covers the specifications to implement tags on a Publisher site.

What are Tags?

A tag, sometimes referred to as a web bug or tracking pixel, is a transparent one by one pixel graphic image that is placed on a web site or other web based content asset and is used to track an “event”. Each time the content asset — in the case of publishers a page — is served upon the end-consumer, the tag call is made and that consumption event is logged on the data collection server, thus enabling all tagged events to be tracked. These events include visiting a web site or page within a website, watching a video, delivery of an advertisement or delivery of distributed content. Information such as the IP address of the computer that consumed the content, a timestamp of when the tag was retrieved, the type of browser, previously set cookie values, etc. can be passed along with the tag call.

Additional values are used to capture more granular information about the tag call including where it was viewed, who delivered it, which advertising package, campaign or content channel, etc. it belonged to. These values are typically included as part of most tag calls through query string parameters.

When client-side tags are combined with insights provided by the Comscore panel, they deliver significant additional value to internet audience measurement, bringing breadth, flexibility and enhanced granularity to the rich consumer insights and behavioral data of the Comscore panel.

Benefits of tagging:

- Census-adjusted metrics (Home + Work) for your site reported in all Media Metrix reports.
- Greater consistency with internal and server-side metrics
- Profiling of your channels against wide area of planning criteria
 - Demos, Lifestyle, Online Behavior, Product Ownership, 3rd party segmentation schemes.
- Coverage of smaller sites/channels and niche audiences.

Implementing the Publisher Tag

The Comscore Publisher Tag uses asynchronous loading of a JavaScript file so that the processing of the webpage is unaffected by loading of the tag. A `` HTML element is also present for cases where JavaScript cannot be executed. The tag uses the parameters listed below, which are required to ensure appropriate data categorization.

Comscore publisher tag parameters

| Parameter | Required or Optional | Description | Explanation | Example value |
|-----------|----------------------|-----------------------|---|---|
| c1 | Required | Tag Type | Pre-populated with fixed value 2 | 2 |
| c2 | Required | Comscore Publisher ID | A Comscore-provided number with at least 7 digits that is unique for each Publisher ⁽¹⁾ . This value is also used in the URL of the <code>beacon.js</code> file. | 1234567 |
| c4 | Optional | Full Page URL | Manually populated full web page URL including query string variables | http://www.website.com/mypage.aspx |

To implement the tag, copy its code into your web page content and provide the appropriate parameter values:

```

1. <!-- Begin Comscore Tag -->
2. <script>
3.     var _comscore = _comscore || [];
4.     _comscore.push({ c1: "2", c2: "1234567" });
5.
6.     (function() {
7.         var s = document.createElement("script"), e1 = document.getElementsByTagName("script")[0]; s.async = true;
8.         s.src = "https://sb.scorecardresearch.com/cs/1234567/beacon.js";
9.         e1.parentNode.insertBefore(s, e1);
10.    })();
11. </script>
12. <noscript>
13.     
14. </noscript>
15. <!-- End Comscore Tag -->

```

In addition to the mentioned parameters, the following parameters are automatically collected by the JavaScript code of the tag:

Automatically collected parameters

| Parameter | Description |
|-----------|--------------------------|
| c7 | Full Page URL (required) |
| c8 | Page Title |
| c9 | Referring URL (required) |



About the locations where tags can be implemented...

Publishers should refrain from implementing the tag inside ad calls and/or iframes as this may result in the c7 parameter — required for correct attribution of traffic — to not populate correctly.

Please also do not add the tags to pop-up or pop-under locations that may contain page content.

(1) The Publisher ID is also known as the *Client ID* or *c2 value*.

Appendix A: Additional Instructions for Specific Use Cases

Content can be delivered in many ways. Some use cases require additional implementation steps. If these additional instructions apply to your content then please consult with your Comscore account team.

Dynamically Generated Webpages

In the case of dynamic content such as when using AJAX, infinite scroll, HTML5 History `pushState` and `popState` API or similar technologies, Publishers should follow the following steps:

1. Consult with your Comscore account team to ensure that the dynamic content meets Comscore's rules for measurement and will be measured on the Comscore panel.
2. Implement the tag as instructed in this document for the initial page load.
3. When the page content changes for more than 50% due to user interaction, call the JavaScript `COMSCORE.beacon()` function, where the appropriate parameters are passed as argument:

```
81. | <script>
82. | (self.COMSCORE && COMSCORE.beacon({ c1: "2", c2: "1234567" }));
83. | </script>
```

This is the desired method for implementing the tag with dynamic content. The above code example checks for the existence of the `COMSCORE` reference before executing the function call, which will be available after the tag was executed for the initial page load.

Often times for panel measurement it is required to perform additional implementation steps for *Pageview Candidate* measurement. Please consult with your Comscore account team to ensure these additional steps are executed where applicable.

Custom Implementation Without JavaScript

Sometimes the automatically collected URL in the `c7` parameter does not represent the URL associated with the content the user is viewing. In other cases the content might not be allowed to contain or execute JavaScript code, as is the case with mobile-optimized WAP content.

For these use cases Publishers should use **only** the `` HTML code of the tag and populate the parameters with appropriate values, **including the `c4` parameter**. The `c4` parameter must be populated with the full URL associated with the content being consumed by the user (which is not necessarily the URL in the address bar of the web browser).

```

```



About URL encoding of query string parameter values in the URL of the `src` attribute of `` HTML element...

Values of query string parameters in the URL of the `src` must be properly URL encoded as illustrated in the example shown above. This is particularly important when the `c4` parameter is provided with a URL as its value.

Appendix B: User Consent Tagging

Applicable privacy and data protection laws and regulations may require companies to capture and/or document a user's consent for measurement. For example, the European Union's General Data Protection Regulation ("GDPR") and the Privacy and Electronic Communication Directive 2002/58/EC require capturing user consent. The California Consumer Privacy Act ("CCPA") requires companies to provide users with the ability to "opt-out of the sale of personal information." Please note that the implications of applicable privacy and data protection laws and regulations may vary and are best evaluated by each individual business.

Using a Consent Management Platform

If you are using a Consent Management Platform (CMP) which implements iAB Transparency and Consent Framework (TCF) version 2.0 then Comscore Publisher Tag integrates with the CMP to automatically collect user consent. No additional steps are necessary to enable this integration, other than to make sure the Comscore Publisher Tag is in the web page where it can access the CMP as per the TCF 2.0 technical specification.

Sometimes JavaScript code execution might not be possible. Publishers are expected to provide two extra parameters in the `` HTML code of the tag, but only if they can retrieve and incorporate Transparency and Consent data. The two parameters are `gdpr` and `gdpr_consent`, both used according to the Consent Management Platform (CMP) specification.

For populating `gdpr_consent` it is important to note Comscore's TCF vendor id is 77.

If JavaScript code execution is not possible and Transparency and Consent data cannot be retrieved, then please implement manual consent communication in the `` HTML code of the tag as explained in the next section.

Manually Communicating Consent

If you are not using a Consent Management Platform then you need to manually communicate user consent by adding parameter `cs_ucfr` to the collected data. The required values for this user consent parameter are:

Label `cs_ucfr` values for communicating user consent

| Value | Interpretation | Usage |
|-------|---|---|
| 0 | User has not given consent or has opted out | Use this value to indicate the user <ol style="list-style-type: none"> has been asked for consent where the user did not give consent, or enabled the option to opt out (e.g., opt out of the sale of personal information) |
| 1 | User has given consent | Use this value to indicate the user has been asked for consent where the user has given consent to collect data for measurement |
| | User has not taken an action | Use an empty string value (i.e., blank) to indicate the user has not taken an action |



About including parameter `cs_ucfr` when not communicating user consent...

If you do not communicate consent for a user, then **do not populate** label `cs_ucfr`.

A publisher must add parameter `cs_ucfr` with an appropriate value to the web page impression tagging by adding the parameter and its value to the HTML and JavaScript code. With this change of the HTML and JavaScript code the publisher **should not change any other collected data values**.

For example, assuming the user has given consent, the aforementioned tag code would be changed into:

```

1.  <!-- Begin Comscore Tag -->
2.  <script>
3.      var _comscore = _comscore || [];
4.      _comscore.push({ c1: "2", c2: "1234567", cs_ucfr: "1" });
5.
6.      (function() {
7.          var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
8.          s.src = "https://sb.scorecardresearch.com/cs/1234567/ beacon.js";
9.          el.parentNode.insertBefore(s, el);
10.         })();
11. </script>
12. <noscript>
13.     
14. </noscript>
15. <!-- End Comscore Tag -->

```

Likewise, assuming the user has given consent, the aforementioned code statement for tagging dynamically generated webpages would be changed into:

```

81. <script>
82.     (self.COMSCORE && COMSCORE.beacon({ c1: "2", c2: "1234567", cs_ucfr: "1" }));
83. </script>

```

Appendix C: Update an Existing Comscore Publisher Tag Implementation

To update an existing Comscore Publisher Tag Implementation typically only the URL of the *beacon.js* file needs to be updated. This URL is typically found in one of the JavaScript statements. For example:

```

1. | <!-- Begin comScore Tag -->
2. | <script>
3. |     var _comscore = _comscore || [];
4. |     _comscore.push({ c1: "2", c2: "1234567" });
5. |
6. |     (function() {
7. |         var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
8. |         s.src = (document.location.protocol == "https:" ? "https://sb" : "http://b") + ".scorecardresearch.com/
9. |         beacon.js";
10. |         el.parentNode.insertBefore(s, el);
11. |     })();
12. | </script>
13. | <noscript>
14. |     
15. | </noscript>
16. | <!-- End comScore Tag -->

```

Please replace the old URL with one using the Comscore Publisher ID and let the URL use secure transmission:

```

1. | <!-- Begin Comscore Tag -->
2. | <script>
3. |     var _comscore = _comscore || [];
4. |     _comscore.push({ c1: "2", c2: "1234567" });
5. |
6. |     (function() {
7. |         var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
8. |         s.src = "https://sb.scorecardresearch.com/cs/1234567/beacon.js";
9. |         el.parentNode.insertBefore(s, el);
10. |     })();
11. | </script>
12. | <noscript>
13. |     
14. | </noscript>
15. | <!-- End Comscore Tag -->

```

In the above example, the Comscore Publisher ID is [1234567](#) and is used in the URL of the *beacon.js* file.

If your existing implementation has additional parameters being collected aside from the *c1* and *c2* shown in the examples above then **please do not change those parameters as part of your update.**

Appendix D: Implementation FAQs

1. Where should I place the tag?

We recommend that the clients place the Comscore tag code as close to the top of the page body as possible.

2. How can I validate my tag is working as intended?

Log into Comscore Direct and click on the “Verify URL” button. In the pop-up, enter in the URL that you desire to test. The Verify URL process will make a request to the page and verify that the tag is physically present on the page. For those sites that embed the Comscore tag in a separate file, the Verify URL feature will not detect the presence of the tag.

We suggest that you load your web page and verify the tag HTTP request calls using a tool like Fiddler or the developer console of your web browser. The instructions for looking for the appropriate tag calls are as follows:

1. On initial deployment of the Comscore tag, the user's machine will request the Comscore beacon.js file⁽²⁾. The resulting HTTP request URL will have the following structure: <https://sb.scorecardresearch.com/cs/1234567/beacon.js> where instead of 1234567 your Comscore Publisher ID will appear.
2. The beacon.js code is executed and then provides the instruction the web browser to fire the actual tag call, which will have a HTTP request URL with this structure:
https://sb.scorecardresearch.com/b?c1=2&c2=1234567&ns__t=1524476354599&ns_c=UTF-8&cv=3.1&c8=My%Page%20Title&c7=http%3A%2F%2Fwww.mywebsite.com%2Fmypage.aspx&c9=
3. The HTTP request will have a response code of **204: No Content**.
4. The important parameters to confirm in the tag call are:
 - **c1=2**
 - **c2=** with your specific Publisher ID as assigned by Comscore
 - **c7=** with the full page URL including query string variables

3. Will placing tags slow the site increasing page load time?

No. Comscore has partnered with Akamai and delivers industry-leading response times. It is expected that the Comscore tag will have a response time of 60-90ms.

4. Can I place the Comscore tag on secure pages?

Yes, the JavaScript code of the default Comscore tag auto-detects that it is placed on a secure page and in those cases fires a HTTPS version of the tag.

The URL in the `src` attribute of `` HTML element of the tag already uses an HTTPS URL.

(2) Unless the beacon.js file has been requested before and has been cached.