Serversided Tracking - Einbindung im GTM

- Server-to-Server Tracking

Server-to-Server Tracking

Es ist möglich, neben einem clientseitigen Tracking auch ein serverseitiges Tracking zu implementieren und dieses über den Google Tag Manager steuern zu lassen.

Um ein First-Party-Tracking Server-to-Server zu ermöglichen, sollte eine Subdomain eingerichtet werden. Diese dient als **Server Endpoint** und ist notwendig, damit das `HttpOnly`-Cookie auf der Domain des Werbetreibenden gespeichert werden kann.

Um den Traffic nicht über die Google Cloud zu leiten, wird ein eigener Tracking-Server empfohlen.

Weitere Informationen zur manuellen Einrichtung findest du hier:

- **Leitfaden für die manuelle Einrichtung**
- **Google Tag Manager – Server-side**
- **Google for Developers**

Funktionsweise

Für das serverseitige Tracking werden Informationen von der Website an den serverseitigen Container übergeben. Dieser kommuniziert anschließend mit einem sogenannten **Client**.

Übergabe der Daten an den serverseitigen Container durch `gtag.js`

```
1  <!-- Global site tag (gtag.js) - Google Analytics -->
2
3  <script async src="https://www.googletagmanager.com/gtag/js?id=MEASUREMENT_ID"></script>
```

```
<script>
window.dataLayer = window.dataLayer || [];

function gtag() {
    dataLayer.push(arguments);
}

gtag('js', new Date());

gtag('config', 'MEASUREMENT_ID', {
    transport_url: 'https://Ihresubdomain.de'
});

gtag('event', 'checkout', {
    token: 'ordertoken',
    campaign_id: 'campaign_id',
    trigger_id: 'trigger_id',
    basket: JSON.stringify([
        {
            article_number: 'article_number_1',
            amount: 'article_amount_1',
            price: '100.00',
            productname: 'article_name_1',
            category: 'article_category',
            additional: {
                "VARIABLE": "VALUE",
                "VARIABLE2": "VALUE2"
            }
        },
        {
            article_number: 'article_number_2',
            amount: 'article_amount_2',
            price: '15.00',
            productname: 'article_name_2',
            category: 'article_category',
            additional: {
                "VARIABLE": "VALUE",
                "VARIABLE2": "VALUE2"
```

```
            }
        },
        {
            article_number: 'article_number_N',
            amount: 'article_amount_N',
            price: '110.00',
            productname: 'article_name_N',
            category: 'article_category',
            additional: {
                "VARIABLE": "VALUE",
                "VARIABLE2": "VALUE2"
            }
        }
    ])
});
</script>
```

Parameter

Parameter	Beschreibung
`MEASUREMENT_ID`	Muss durch die Google Analytics 4 oder Universal Analytics ID ersetzt werden.
`transport_url`	URL des Tagging-Servers (First-Party-Subdomain).

> **Hinweis:** Alle Parameter im Beispiel müssen dynamisch befüllt werden.

Client im serverseitigen Container einrichten

Im nächsten Schritt wird ein Client im serverseitigen Container eingerichtet. Dieser verarbeitet die übergebenen Daten und erstellt das Event `createPurchase`, welches schließlich den serverseitigen Trackingcode aufruft.

Dies erfolgt über einen **Basket-Tracking-Aufruf**, der in einem **Custom Tag** hinterlegt wird.

Client einrichten

Im **Server Container Account** wird im Reiter **Clients** zusätzlich zum bereits vorhandenen **GA4 Client** ein **Google Tag Manager: Web Container Client** eingerichtet.

Innerhalb dieses Clients werden mehrere **Tags** und **Trigger** definiert.

Diese übernehmen folgende Aufgaben:

- Setzen des notwendigen First-Party-Cookies
- Verarbeitung der Trackinginformationen (`emid`)
- Auslösen des eigentlichen Trackingaufrufs

Setzen des First-Party-Cookies

Sobald Daten an den Client des Server-Containers gesendet werden, kann anhand bestimmter Kriterien eine Aktion ausgelöst werden.

Relevant ist in diesem Fall die **`emid`** als Tracking-Parameter, der in der Regel via "&emid" übergeben wird.

Der übermittelte Wert muss im Rahmen der HTTP-Response als Cookie gespeichert werden.

Variable anlegen

Es wird zunächst eine Variable vom Typ **URL** angelegt.

Beispielsweise:

- Name: `emid Parameter`
- Komponententyp: **Suchanfrage**
- Suchanfrageschlüssel: `emid`

Cookie setzen

Der Wert wird anschließend als `HttpOnly`-Cookie im First-Party-Kontext gespeichert.

Dies erfolgt über die `setCookie`-API.

```
const setCookie = require('setCookie');

setCookie('emid', '{{emid Parameter}}', {
    'max-age': 2592000,
    httpOnly: true,
    secure: true,
    samesite: 'none'
});
```

Bedeutung der Einstellungen

Einstellung	Beschreibung
`max-age`	30 Tage in Sekunden
`httpOnly`	Verhindert das Auslesen des Cookies durch JavaScript
`secure`	Cookie wird ausschließlich über HTTPS übertragen
`samesite: none`	Erlaubt die Verwendung im Cross-Site-Kontext

Damit das Tag ausgeführt wird, wird ein entsprechender **Page View Trigger** verwendet.

Der eigentliche Trackingaufruf

Nachdem der Benutzer die Bestellung erfolgreich abgeschlossen hat, muss im Client ein weiteres Tag ausgeführt werden.

Dieses liest

- die per `gtag()` in den `dataLayer` geschriebenen Daten,
- sowie die `emid` aus dem zuvor gesetzten First-Party-Cookie

aus und übergibt diese an den serverseitigen Conversion-Aufruf.

Dokumentationshilfen zum serverseitigen Tagging

Weitere Informationen stellt Google in der offiziellen Dokumentation bereit: Google Tag Manager – Server-side