

Taboola S2S Conversion Tracking With AppsFlyer

Web to App

Last updated: April 22, 2024

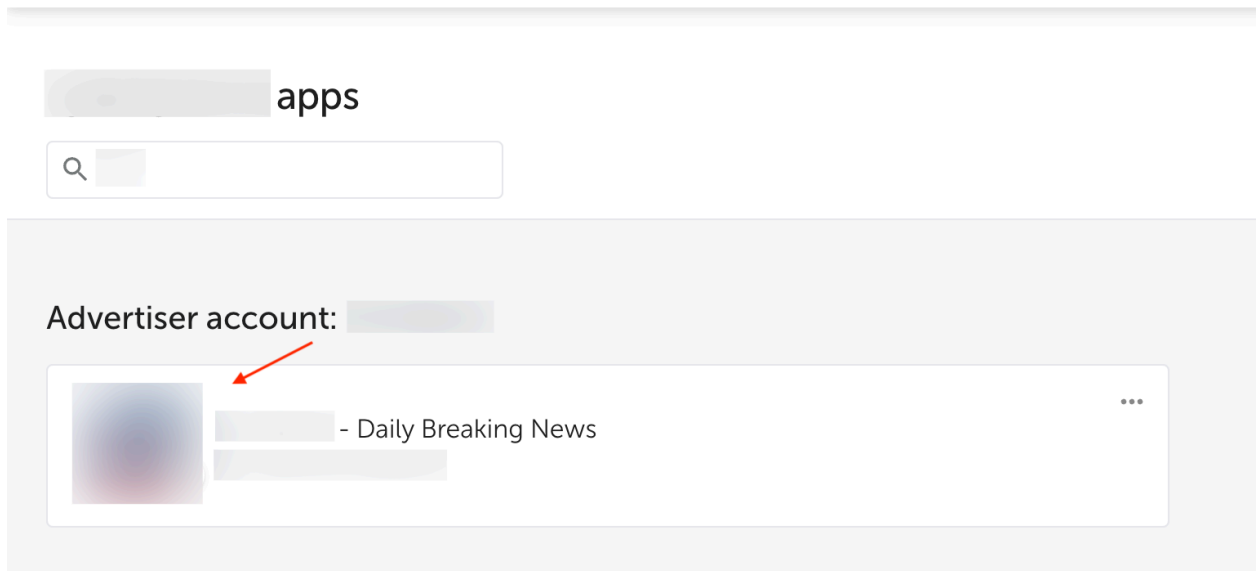
This **does** require Appsflyer Smart Script installed on Landing Page
 For an *overview*, see [S2S Conversion Tracking With AppsFlyer - Overview](#).

Your version of **AppsFlyer** might differ from the one described here.
 For **AppsFlyer** support, visit: <https://support.appsflyer.com/hc/en-us>

Integrate AppsFlyer with Taboola

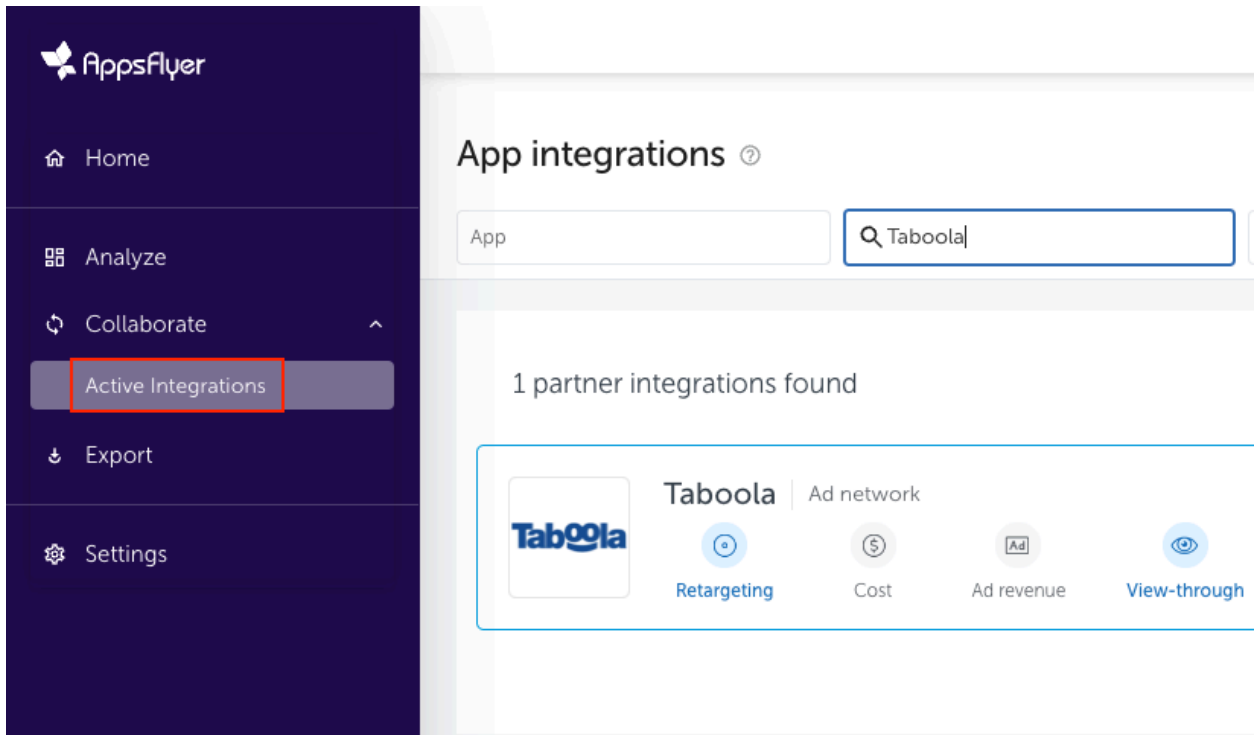
Enable the Taboola Integrated Partner

1. Log into the **AppsFlyer** dashboard: <https://hq1.appsflyer.com/auth/login>
2. Click on **My Apps** (*top, right*).
3. Start typing the relevant **App ID** and click on the *thumbnail* to open the **App**:

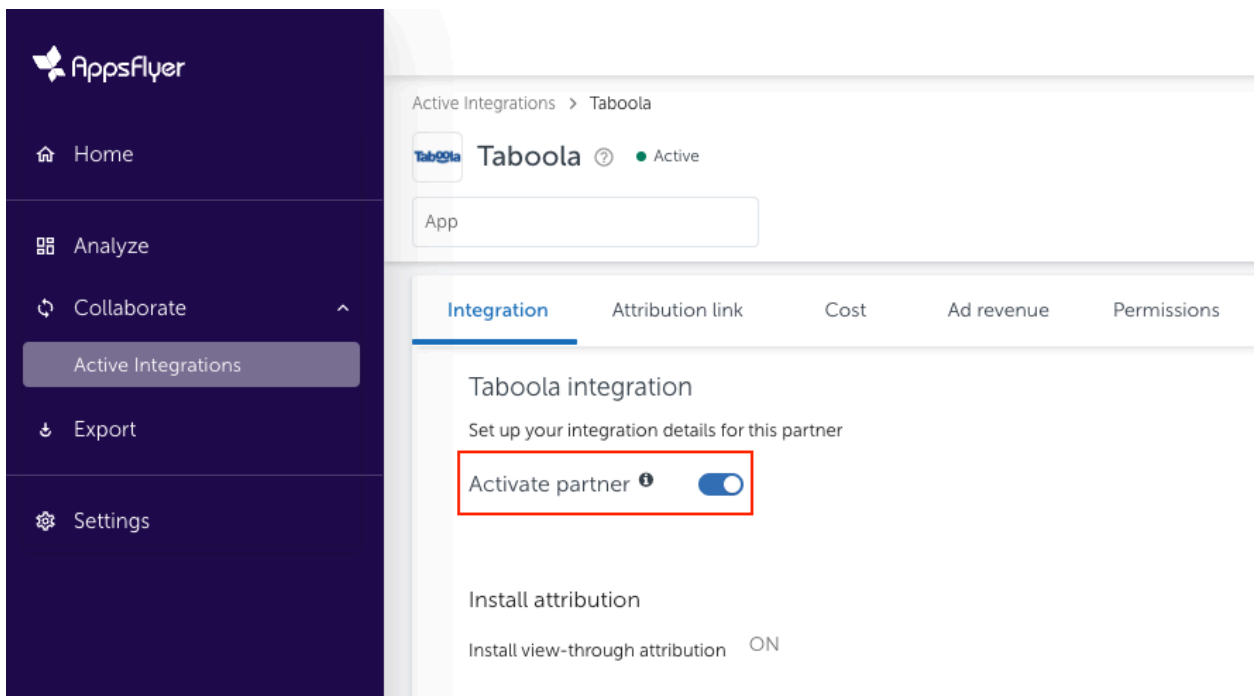


4. Under **Collaborate** (*bottom, left*), select **Active integrations**.

5. Start typing “**Taboola**” and click on the *thumbnail* to open the **Taboola Partner**:

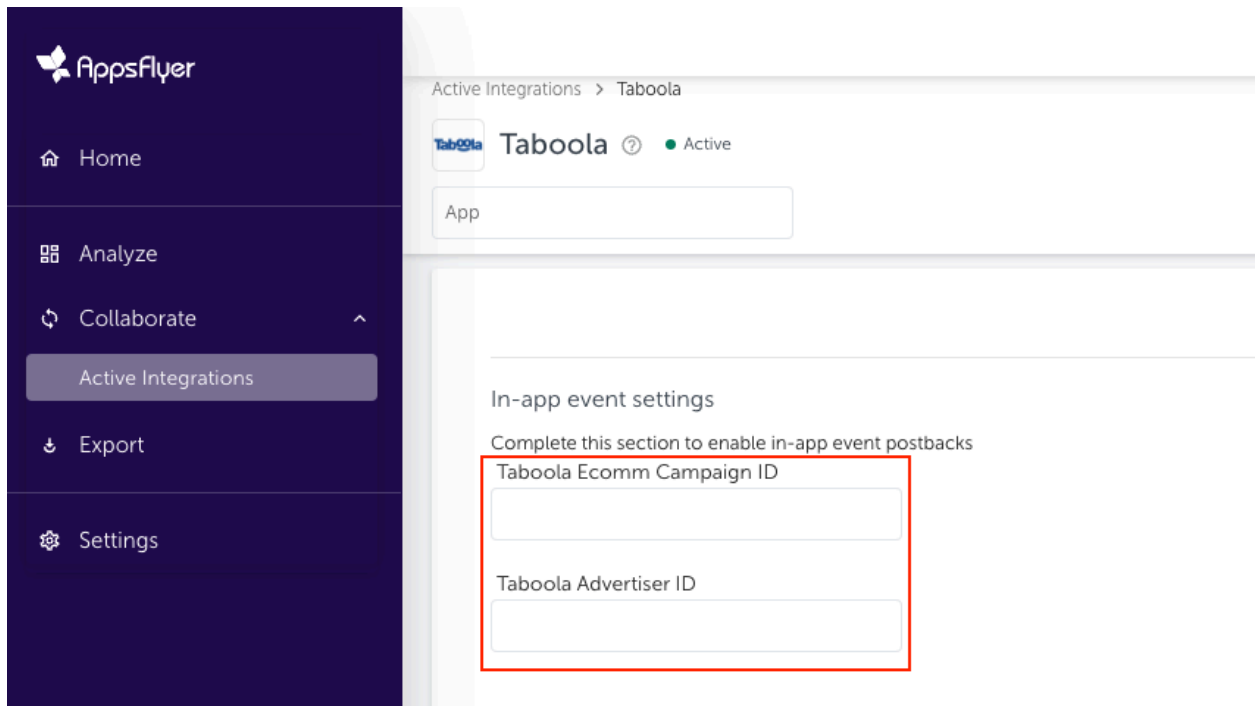


6. Under **Taboola Integration**, toggle **Activate Partner** to **ON**:



Enable Partner “Taboola” General Settings

7. Under **In-app event settings**
 - a. Leave the **Taboola Ecomm Campaign ID** field *blank*.
 - b. Fill in **Taboola Advertiser ID**.



8. Click **Save Integration** (bottom, right).

Which Taboola Advertiser ID?

- To track conversions for a **given account**, fill in that **Taboola Account ID**.
- To track conversions for a **network**, fill in the **Taboola Network Account ID**.

Configure Partner “Taboola” Attribution Link

9. Next, switch to the “**Attribution Link**” tab.
10. Depending on your intended use case, choose either ‘**Use single platform link**’ or ‘**Use Onelink**’ as the “**Attribution Link Type**”. For more detail, see the [AppsFlyer documentation](#).



11. Map the selected AppsFlyer “**Parameter Name**” from the provided list to its corresponding Taboola parameter (referred to as the “**Value**”) in the “**Attribution link parameters**” section.
12. Skip the “**Retargeting settings**” section.
13. Under the **Click-through attribution** tab, locate the **Click-through lookback window** section and set it to the desired duration - e.g. 30 days.
14. If you also want to track views, navigate to the **View-through attribution** tab and locate the **View-through lookback window** section. Set the desired duration, for example, 24 hours.

For a detailed understanding of which parameters to use, review the [Parameters](#) guide.

For guidance on selecting appropriate values, refer to [Creating and Adding URL Parameters for Tracking](#).

Best Practice

Configure **AppsFlyer** and **Taboola** with the *same* lookback window (e.g. 30 days).

If **AppsFlyer** has a *shorter* lookback window, then any conversions outside that window will *not* be passed to **Taboola** for attribution.

The **Taboola** lookback window will be configured below (under [Set up corresponding events in Taboola](#)). By *default*, the **Taboola** lookback window is *30 days*.

Other Useful links

- [Adding an App to AppsFlyer](#)
- [Integrated partners dashboard](#)
- [Set up an integrated partner](#)
- [Partner activation/deactivation](#)

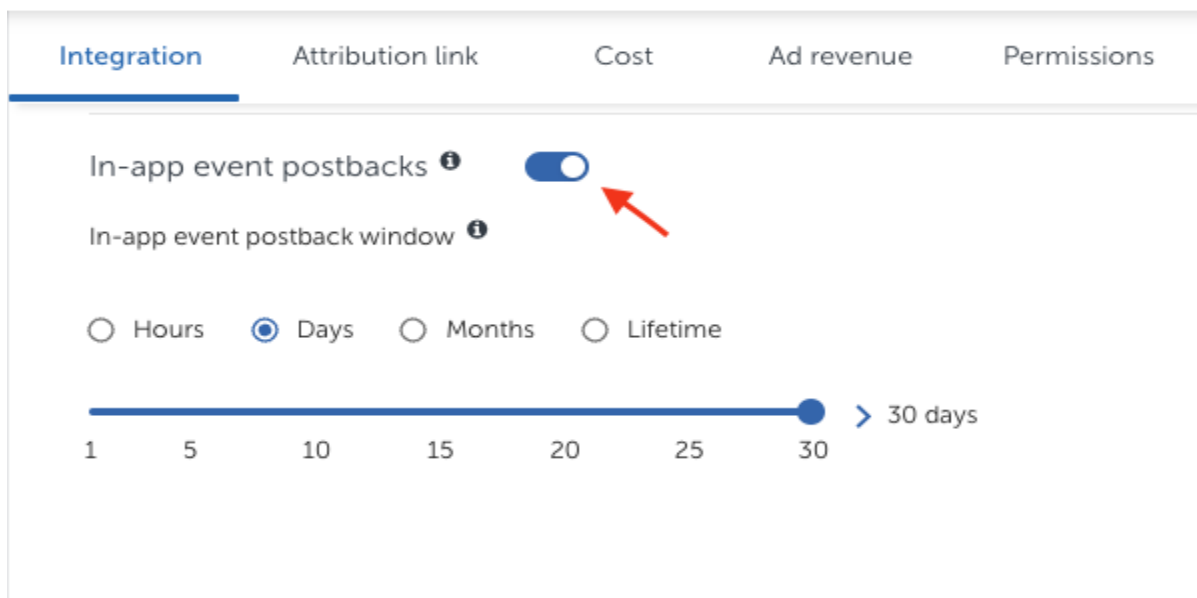
(Optional) Enable Partner “Taboola” In-app event postbacks & Create In-app events

The steps taken in the *previous* section allow you to track **Install Events**. This section describes how to track **In-App Events**.

In-App Event Tracking allows for *granular campaign optimization*.

If you followed the previous section, you already have the necessary pane open - switch to the “**Integration**” tab & *skip* to **Step 3**.

1. Open the **AppsFlyer** dashboard.
2. Select the relevant **App** and navigate to **Collaborate > Active Integrations > Taboola** (see *previous* section: [Enable the Taboola Integrated Partner](#)).
3. Toggle In-app event postbacks to **ON** if you wish to track any in-app conversions.



4. Define the desired duration for the postback window (e.g.30 days).
5. Initiate event configuration by clicking “**Add Event**”
 - a. Choose an **AppsFlyer event** from the list or add a new one by clicking “**Add item**”



- b. Link the chosen **AppsFlyer event** with the Taboola event name it corresponds to (aka **mapped to partner event**).

<input type="checkbox"/>	AppsFlyer event ⓘ		mapped to partner event ⓘ
<input type="checkbox"/>	af_registration	>	complete_registration
<input type="checkbox"/>	FirstDeposit	>	FTD

this should match with event name on "tracking" page of taboola ads.

- c. For the **user from** field, pick **"This Partner Only"** for exclusive Taboola events or **"All Media Sources Including Organic"** to include all sources.
- d. In the **including** field, select **"No Values & No Revenue"** for non-revenue events, or **"Values & Revenue"** for transactions like purchases. Choose **"Values & No Revenue"** for specific non-revenue related values.

Best Practice

Keep the **AppsFlyer** postback window consistent with the **Taboola** lookback window to ensure all in-app conversions are **attributed** correctly.

For instance, if your Taboola lookback window is set to 7 days, your AppsFlyer postback window should also be set to 7 days. Choosing a shorter postback window in AppsFlyer means any conversions that happen after that period won't be credited to Taboola, potentially missing out on valuable attribution data

Useful links

- [In-app event postback configuration](#)

Set up 2 links

Edit the campaign creative/ad



Link the Campaign Creative/Ad *directly* to your **Landing Page**:

1. Log into **Taboola Ads**.
2. **Add or edit** your **Campaign Creative/Ad**. Fill in the **Landing Page** as the **URL** and save.
Sample URL: https://www.example.com/taboola_landing_page

Use the **Landing Page** as the URL - *not* the link obtained from **AppsFlyer**.
Add the **Landing Page** at the *creative/ad* level (*not* the campaign level).

Add URL parameters at the campaign level

1. **Add or edit** the relevant *campaign*.
2. Extract the url parameter from Appsflyer Click Tracker, *Sample link*

https://app.appsflyer.com/com.appsflyer.adNetworkTest?pid=taboola_int&af_siteid={site_id}&c={campaign_name}&af_ad_id={campaign_item_id}&af_click_lookback=7d&clickid={click_id}

3. Add the shown above in **red** under **Campaign Settings > Tracking > Tracking Code** :

The screenshot shows a user interface for adding tracking code. At the top, there is a header with a code icon and the text '</> Tracking'. Below this is a section titled 'Tracking Code' with a help icon. A descriptive text states: 'This code will be added to your landing page URLs so you will be able to track the performance on other analytics platforms.' A text input field contains the following URL parameters: 'pid=taboola_int&af_siteid={site_id}&c={campaign_name}&af_ad_id={campaign_item_id}&af_click_lookback=7d&clickid={click_id}'.

For additional information about adding **passback macros** at the *campaign* level, see: [Creating and Adding URL Parameters for Tracking](#)

Create links for your Landing Page CTAs



- **Configure Onelink Smart Script on your Landing Page**

1. Create a Onelink smart script by following the guide on the AppsFlyer help center.
2. Use the example in the information box below to learn how to integrate Taboola URL parameters with a Onelink script.

Example:

Sample Appsflyer link:

https://app.appsflyer.com/com.appsflyer.adNetworkTest?pid=taboola_int&af_siteid={site_id}&c={campaign_name}&af_ad_id={campaign_item_id}&af_click_lookback=7d&clickid={click_id}

JavaScript

```
//Initializing Smart Script arguments
var oneLinkURL = "https://yourappname.onelink.me/XyZK/";
var mediaSource = {keys: ["utm_source", "pid"]};
var campaign = {keys: ["c"]};
var adSet = {keys: ["af_ad_id"]};
var af_siteid = {paramKey:"af_siteid", keys: ["af_siteid"]};
var af_click_lookback = {paramKey:"af_click_lookback", keys: ["af_click_lookback"]};
var clickid = {paramKey: "clickid",keys: ["clickid"]};
//Onelink URL is generated.
var result = window.AF_SMART_SCRIPT.generateOneLinkURL({
  oneLinkURL: oneLinkURL,
  afParameters: {
    mediaSource: mediaSource,
    campaign: campaign,
    adSet: adSet,
    afCustom: [af_click_lookback,af_siteid,clickid]
  }
});
var result_url = "No output from script"
if (result) {
  result_url = result.clickURL;

  document.getElementById('andrd_link').setAttribute('href',
  result_url);

  document.getElementById('ios_link').setAttribute('href',
  result_url);
}
document.getElementById('output_url1').innerHTML =
result_url;
```

Onelink url:

https://yourappname.onelink.me/XyZK/?af_js_web=true&af_ss_ver=2_2_0&pid=taboola_int&c={campaign_name}&af_adset={campaign_item_id}&af_click_lookback=7d&af_siteid={site_id}&clickid={click_id}



3. The generated **Onelink URL** will be used for the **Landing Page CTA**.

The **PID** portion (in orange) tells **AppsFlyer** to attribute the conversion to **Taboola**.

The **Click ID** param (in red) should look *exactly* as shown.

Useful links:

- [OneLink Smart Script V2 Setup](#)
- [Test the Smart Script](#)

What is the Taboola Click ID?

When a user clicks on your campaign creative, Taboola's **{click_id} macro** passes a unique identifier to the destination URL.

Creative link:

```
https://www.example.com/taboola_landing_page?pid=taboola_int&af_siteid={site_id}&c={campaign_name}&af_ad_id={campaign_item_id}&af_click_lookback=7d&clickid={click_id}
```

Upon clicking, the user is directed to:

```
https://www.example.com/taboola_landing_page?pid=taboola_int&af_siteid={site_id}&c={campaign_name}&af_ad_id={campaign_item_id}&af_click_lookback=7d&clickid=abc123
```

The *abc123* identifier is a **Click ID**. It stores information about the *click event* - e.g. which campaign was clicked and which user, etc. When **AppsFlyer** passes back the **Click ID**, **Taboola** uses it for attribution and tracking.

A real **Click ID** is an alphanumeric, case-sensitive string of up to 51 characters. E.g.

[GiAsAGJvFovGp9sM6kcwLmGqMeNUF6PYvd7Xwp1pA_mkcSC292Qo-5-A3lجت9yQAQ](#)

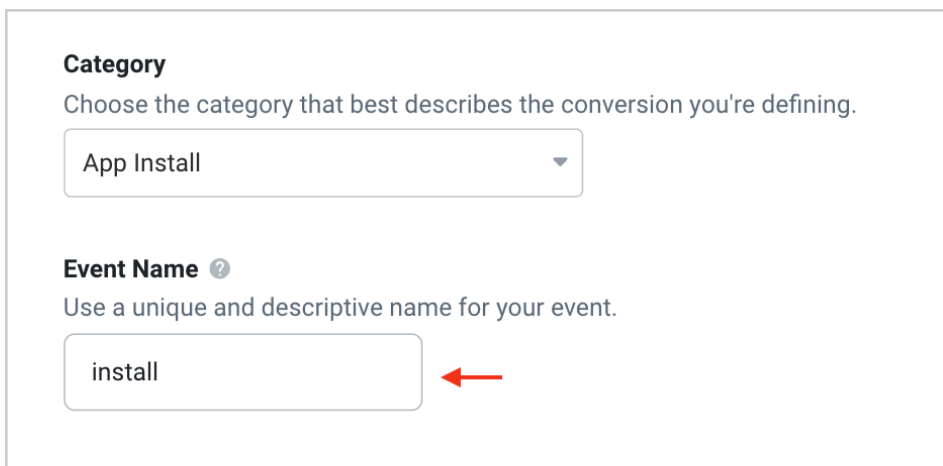
Set up corresponding events in Taboola

You will need to set up **Taboola Ads** with the *same* events as **AppsFlyer**.

That way, when **AppsFlyer** notifies **Taboola** of a conversion, **Taboola** can record the event and reflect it in reporting.

Create the app install event in *Taboola*

1. Log into **Taboola Ads**.
2. In the navigation bar, select '**TRACKING**' to open the **Conversions** screen.
3. Click on **+ New Conversion**.
4. For **Type**, click on **Event**.
5. Under **Conversion Properties**:
 - a. Select Category as "**App Install**"
 - b. After selecting the category in the previous step as App Install, the auto-filled **Event Name** will be 'app_install'; change it to 'install'.



Category
Choose the category that best describes the conversion you're defining.

App Install

Event Name ?
Use a unique and descriptive name for your event.

install

IMPORTANT: Type in the Event Name *exactly* as shown.

- c. Fill in the remaining fields as desired and click on **Create Conversion**.

(Optional) Create in-app events in Taboola

First create in-app events in **AppsFlyer** - see above: [\(Optional\) Enable Partner "Taboola" In-app event postbacks & Create In-app events.](#)

1. Log into **Taboola Ads**.
2. In the navigation bar, select '**TRACKING**' to open the **Conversions** screen.
3. Click on **+ New Conversion**.
4. For **Type**, click on **Event**.
5. Fill in the remaining **conversion rule** details, according to your specific needs. The same guidelines apply as per [Pixel Based Conversions](#).



6. Under **Conversion Properties**, fill in:
 - a. **Conversion Name** - This is *not* used for the server-to-server communication, and can be changed at any time. Choose any name that helps you to identify the conversion rule.
 - b. **Event Name** - Use the *same* name that you provided in **AppsFlyer**, *exactly* as it appears there. See: [\(Optional\) Create In-App Events in AppsFlyer](#) (above).

- Use the *identical* event name, as it appears in **AppsFlyer**.
- Event names are *case-sensitive*.

Start Tracking Conversions

Well done! Your setup is complete, and you are now ready to track conversions.

Test your configuration

We recommend *testing* your configuration.

Note that you can test your configuration *before* your campaign goes live.

For more information, see: [Testing Your Server-to-Server \(S2S\) Tracking Accuracy](#)

Understanding the end-to-end flow

The following steps illustrate the end-to-end flow:

1. **John** clicks on your *creative/ad*. **Taboola** appends the actual **Click ID** value (*abc123*) and directs **John** to your **Landing Page**:

https://www.example.com/taboola_landing_page?pid=taboola_int&af_siteid={site_id}&c={campaign_name}&af_ad_id={campaign_item_id}&af_click_lookback=7d&clickid=abc123

2. The **Landing Page** records the **Click ID** value for **John**'s session (using Smart Script)
3. A bit later, **John** clicks on a **CTA** in the **Landing Page**:

<https://yourappname.onelink.me/XyZK/>



4. The **Smart Script** appends the relevant **query string params** and **Click ID**, and directs **John** to the target URL:

```
https://yourappname.onelink.me/XyZK/?af_js_web=true&af_ss_ver=2_2_0&pid=taboola_int&c={campaign_name}&af_adset={campaign_item_id}&clickid=abc123
```

5. **AppsFlyer** stores John's **Click ID** (*abc123*) and directs **John** to the **App Store/Play Store**.
6. A bit later, **John** *installs the App* - which you defined as a *conversion* in the **AppsFlyer** dashboard and **Taboola Ads**.*
7. **AppsFlyer** invokes the **Taboola postback URL**, inserting John's **Click ID**, and the **event name**:

```
https://trc.taboola.com/actions-handler/log/3/s2s-action?click-id=abc123&name=install
```

8. **Taboola Ads** reflects the conversion in the relevant reports.

* A similar flow will apply for any *in-app* events that were defined.

Additional Resources

- [AppsFlyer Help Center](#)
- [Server-to-Server Events](#)