Pixel & Events API Configuration

TikTok Marketing Suite for Magento 2

Complete reference for TikTok Pixel and Events API settings. Use both for optimal tracking accuracy.

Location: Stores → Configuration → Magmodules → TikTok - Pixel & CAPI

Overview

The module supports two tracking methods:

Method	Type	Description
TikTok Pixel	Client-side (browser)	JavaScript-based tracking on your website
Events API	Server-side	Direct server-to-TikTok communication

Recommendation: Enable both for maximum accuracy. Events API captures events that Pixel might miss due to ad blockers or browser restrictions.

Pixel Configuration

Enable

Enable TikTok Pixel tracking on your store.

Pixel ID

Your unique TikTok Pixel identifier.

Where to find it:

1. Go to TikTok Ads Manager
2. Go to Assets → Events
3. Select your Pixel
4. Copy the Pixel ID from Settings

Events

Select which events to track via Pixel:

Event	Triggers when
PageView	Any page is viewed
ViewContent	Product page is viewed

Event	Triggers when
ViewCategory	Category page is viewed (custom event)
CustomizeProduct	An option is chosen on the product page (configurable attribute, custom option, bundle option); once per page view
AddToCart	Product added to cart
ViewCart	Cart page is viewed (custom event)
InitiateCheckout	Checkout is started
AddToWishlist	Product added to wishlist
Search	Search is performed
CompleteRegistration	Customer account is created
Purchase	Order is completed

Recommendation: Enable all events for complete funnel tracking.

ViewCategory and ViewCart are not TikTok standard events: TikTok reports them as custom events, which you can use for custom audiences and custom conversions.

Advanced Matching

Identify customers to the Pixel with their hashed email, phone number and customer ID (default: Yes). See Data Captured.

Events API Configuration

Enable

Enable server-side event tracking via Events API.

Access Token

Your TikTok Events API access token (stored encrypted).

Where to find it:

1. Go to TikTok Events Manager
2. Select your Pixel

3. Go to Settings → Events API
4. Click "Generate Access Token"

Events

Select which events to track via Events API:

Event	Triggers when
ViewContent	Product page is viewed
ViewCategory	Category page is viewed (custom event)
CustomizeProduct	An option is chosen on the product page; once per page view
AddToCart	Product added to cart
AddToWishlist	Product added to wishlist
Search	Search is performed
InitiateCheckout	Checkout is started
AddPaymentInfo	Payment method is selected
CompleteRegistration	Customer account is created
Subscribe	Newsletter subscription
Lead	Contact form submission
Purchase	Order is completed

Event names

In May 2025 TikTok renamed **CompletePayment** to **Purchase** and **SubmitForm** to **Lead**. The module sends the new names. Old names still reach the right event: a configuration value or a `window.mmTikTokPixel.track()` call from a theme or integration with `CompletePayment` or `SubmitForm` is mapped to `Purchase` or `Lead`. TikTok also converts the old names itself, so existing ad optimisation keeps working.

Advanced Matching

Send hashed customer data (email, phone, customer ID, name, zip code) and city, state and country with Events API events (default: Yes). See Data Captured.

Test Event Code

Optional field for testing your Events API integration.

How to use:

1. Go to TikTok Events Manager → Data Sources → Your Pixel → Test Events
2. Copy the Test Event Code
3. Enter it in this field
4. Perform actions in your store
5. View real-time events in TikTok's Test Events tab

Important: Remove the Test Event Code after testing is complete!

How Events Are Tracked

Pixel Events (Client-Side)

Events fire via JavaScript when the action occurs in the browser:

```
Customer action → Browser JavaScript → TikTok Pixel → TikTok
```

Pros: Real-time, captures client data **Cons:** Blocked by ad blockers, affected by browser privacy settings

Events API (Server-Side)

Events are queued and sent from your server:

```
Customer action → Magento Observer → Processing Queue → Cron → TikTok API
```

Pros: Not affected by ad blockers, more reliable **Cons:** Slight delay (queue processed every 5 minutes)

Event Deduplication

When both Pixel and Events API are enabled, TikTok uses event_id to deduplicate events, ensuring each action is counted only once.

The Pixel event and the Events API event for the same action share one `event_id`:

Event	Where the event_id comes from
ViewContent, **ViewCategory**, **Search**	Generated by the Pixel in the browser and posted to the store, which queues the Events API event with it. Works on full-page-cached pages.

Event	Where the event_id comes from
CustomizeProduct	Generated in the browser when the first option is chosen, used for the Pixel event and posted to the store for the Events API event.
AddToCart (product page)	Generated by the Pixel on the add-to-cart click and submitted with the add-to-cart form.
InitiateCheckout	Generated once per cart and kept in the checkout session.
Purchase	Derived from the order number, so a re-send with `magmodules-tiktok:sync-order` carries the same id.
AddToWishlist, **CompleteRegistration**	Generated when the action happens; the Pixel event fires on the next page with that id.

To check, enable a Test Event Code and look in TikTok Events Manager: an action should show up once, marked as deduplicated.

Consent

Track only visitors who consented. Set **Consent Source** under Stores → Configuration → Magmodules → TikTok Marketing → Consent:

Consent Source	Consent is given when
Off (always track)	Always (default, previous behaviour)
Magento Cookie Restriction Mode	The visitor accepted the "Allow Cookies" notice (cookie `user_allowed_save_cookie`). When Cookie Restriction Mode is disabled in Magento, there is no notice and tracking is always on.
Cookie name and value	The configured cookie exists and its value contains the configured text (case-insensitive). For Cookiebot: cookie `CookieConsent`, value `marketing:true`. An empty value accepts any value of the cookie.

Pixel: starts with `ttq.holdConsent()`. Events are held until the visitor decides, then sent with `ttq.grantConsent()` or discarded with `ttq.revokeConsent()`. The decision is picked up as soon as the consent banner sets its cookie, without a page reload. The TikTok script itself is loaded in all cases.

Events API: events are not queued without consent. ViewContent and Search are sent once consent is given on that page. For Purchase, consent is checked when the order is placed and recorded on the quote, so a re-send with `magmodules-tiktok:sync-order` follows the customer's choice. Orders placed without a visitor (e.g. in the

admin) are not sent while consent is required.

Customer Groups

Exclude Customer Groups (Stores → Configuration → Magmodules → TikTok Marketing → Customer Groups) turns tracking off for customers in the selected groups, e.g. staff, test or wholesale accounts: the Pixel is not loaded for them and no Events API events are queued. Select **NOT LOGGED IN** to exclude guests.

For events tied to a cart or order (AddToCart, InitiateCheckout, AddPaymentInfo, Purchase) the group of the cart or order counts; for CompleteRegistration the group of the new account; otherwise the group of the visitor. The full-page cache keeps a separate page per customer group, so cached pages follow the setting too.

Limited Data Use

Limited Data Use (Events API, default: No) adds `limited_data_use: true` to every Events API event, so TikTok processes the data under its Limited Data Use rules, e.g. for US state privacy laws.

Data Captured

Advanced Matching (Customer Data)

With **Advanced Matching** enabled (default), customer data is normalised and sent to help TikTok match events to people. Email, phone, customer ID, names and zip code are SHA-256 hashed before they leave your store; city, state and country are sent as plain lowercase values, as TikTok expects them.

Data	Pixel (`ttq.identify`)	Events API	Source	Normalisation
Email	✓	✓	Customer account, checkout email field, order	lowercase, trimmed, SHA-256
Phone	✓	✓	Default billing address, quote or order billing address	E.164 (e.g. `+31612345678`), SHA-256
External ID	✓	✓	Customer ID (logged-in customers)	SHA-256
First name		✓	Address or customer account	lowercase, trimmed, SHA-256

Data	Pixel (`ttq.identify`)	Events API	Source	Normalisation
Last name		✓	Address or customer account	lowercase, trimmed, SHA-256
Zip code		✓	Address	lowercase, no spaces (US: 5 digits), SHA-256
City		✓	Address	lowercase, not hashed
State		✓	Address (region code)	lowercase, not hashed
Country		✓	Address	ISO code, lowercase, not hashed

The address is the order billing address for Purchase, the checkout address for InitiateCheckout and AddPaymentInfo, and the default billing address of a logged-in customer for other events.

The Pixel identifies logged-in customers on every page (loaded separately, so full-page caching keeps working), and guests on the checkout once they enter their email address and on the order success page.

Advanced Matching has a separate setting under **Pixel** and under **Events API**. When disabled, the Pixel sends no customer data and the Events API sends only the technical data below. Make sure your privacy policy covers sharing hashed customer data with TikTok.

Technical Data

Data	Purpose
IP Address	Geo-targeting
User Agent	Device targeting
_ttp cookie	Cross-device tracking
ttclid	TikTok Click ID from URL

Processing Queue

Events API events are processed via a queue for reliability:

1. Event occurs → Added to queue (status: pending)
2. Cron runs every 5 minutes → Processes pending events

3. Success → Status updated to "success"
4. Failure → Retried up to 3 times, then marked "failed"

Monitor the queue at: **Marketing → Magmodules TikTok → Processing Queue**

Best Practices

Enable Both Pixel and Events API

For maximum event capture rate:

- Pixel catches immediate interactions
- Events API provides backup for blocked/missed events
- TikTok deduplicates automatically

Select Same Events for Both

Enable the same events in both Pixel and Events API settings to ensure complete coverage.

Monitor the Processing Queue

Regularly check the queue for failed events:

1. Go to **Marketing → Magmodules TikTok → Processing Queue**
2. Filter by Status = Failed
3. Check error messages
4. Fix configuration issues if needed

Verify in Events Manager

After setup, verify events in TikTok:

1. Go to Events Manager
2. Select your Pixel
3. Check "Overview" for recent events
4. Use "Test Events" for real-time debugging

Need More Help?

Documentation:

- All Help Articles - Complete documentation overview

Support:

- Contact Support - Get help from our team

For a complete overview of features and configuration options, see the TikTok Marketing Suite extension on magmodules.eu

All articles for TikTok Marketing Suite

Installation

1	Install through FTP and SSH
2	Installation using Composer (recommended)
3	Installation using the Adobe Marketplace

Configuration

1	Quick Start Guide
2	Feed Configuration
3	**Pixel & Events API Configuration (current)**

Troubleshooting

1	Troubleshooting

Grids

1	Data Feed Logs
2	Processing Queue

Background

1	CLI Commands

