## **IPN- Instant Payment Notification (IPN)**

## OTRA COSA DIFERENTE PARA LOS QUE USAN “botones” para pagar:

## Payment Data Transfer (PDT).

https://developer.paypal.com/docs/classic/paypal-payments-standard/integration-guide/get-notifications/?mark=IPN

Instant Payment Notification (IPN) notifies merchants almost instantly about transaction events, such as:

* Payments received, including Express Checkout, and Adaptive Payments.
* Credit card authorizations.
* eCheck payments and related pending, completed, or denied status events.
* Recurring payments and subscription actions.
* Chargebacks, disputes, reversals, and refunds.

Merchants can use their backend processes to act based on the information they receive. For example, they can:

* Update their inventory.
* Email a purchase confirmation.
* Trigger order fulfillment.
* Enable download of digital media.
* Update their customer list.
* Email a shipping notification.
* Issue a refund.
* Update accounting-related information.

## Recurring Payments Notifications

PayPal notifies you of recurring payments events through IPN and email. Typically, however, you can call GetTransactionDetails to obtain the information you need.

PayPal notifies you of certain events through IPN. For recurring payments profiles created using Express Checkout, PayPal also notifies buyers of specific events by email. The following table indicates when PayPal generates IPN and emails:

Table 7. Recurring payments IPN messages and email

| **Event** | **IPN** | **Buyer Email** |
| --- | --- | --- |
| Profile successfully created | Yes | Yes |
| Profile creation failed | Yes | Yes |
| Profile canceled from paypal.com interface | Yes | Yes |
| Profile status changed using API | No | Yes |
| Profile updated using API | No | Yes |
| Initial payment either succeeded or failed | Yes | Yes |
| Payment either succeeded or failed (during either trial period or regular payment period) | Yes | Yes |
| Outstanding payment either succeeded or failed | Yes | Yes |
| Maximum number of failed payments reached | Yes | No |

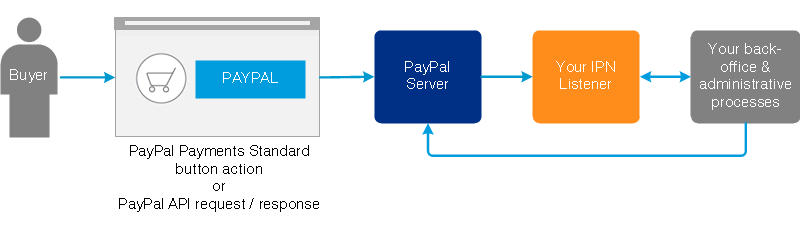
**Note:** API transactions such as ManangeRecurringPaymentsProfileStatus do not trigger IPN notification. The API response immediately provides the success or failure of the call.

## How it works

Merchants create an IPN listener page on their website and then specify the URL of the listener page in their PayPal account profile. PayPal then sends notifications of all transaction-related events to that URL. When customers pay for goods or services, PayPal sends a secure FORM POST containing payment information (IPN messages) to the URL. The IPN listener detects and processes IPN messages using the merchant backend processes. The IPN listener page contains a custom script or program that waits for the messages, validates them with PayPal, and then passes them to various backend applications for processing.

Listed below are some events that can trigger a notification.

* Instant payments — includes [Express Checkout](https://developer.paypal.com/docs/classic/products/express-checkout/) payments and [direct credit card](https://developer.paypal.com/docs/classic/paypal-payments-pro/integration-guide/) payments.
* eCheck payments — includes payments with a status of pending, completed, and denied
* Pending payments — includes payments being reviewed for potential fraud
* Authorizations — a payment that is [authorized](https://developer.paypal.com/docs/classic/admin/intro/) but has not yet been collected
* [Recurring payments](https://developer.paypal.com/docs/classic/admin/intro/#recurring-payments) and subscription payment actions
* Chargebacks, disputes, reversals, and refunds associated with different transactions



If you enable the IPN service, PayPal sends messages to the IPN listener at the URL you specify in your account profile. If you want, you can override this URL in order to associate a different IPN listener with a specific transaction. To do this, you can either:

* Specify the URL of a different listener in your definition of a PayPal Payment Standard button *or*
* Pass the URL of a different listener to a call of a PayPal API operation

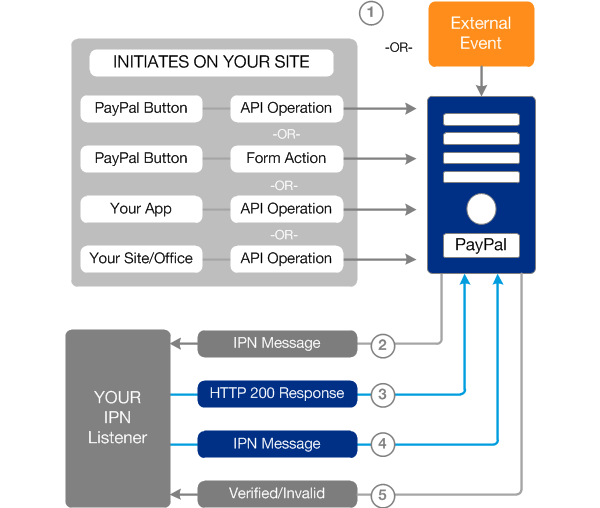
The IPN message authentication protocol consists of four steps:

1. PayPal HTTP POSTs an IPN message to your listener that notifies it of an event.
2. Your listener returns an empty HTTP 200 response to PayPal.
3. Your listener HTTP POSTs the complete, unaltered message back to PayPal; the message must contain the same fields (in the same order) as the original message and be encoded in the same way as the original message.
4. PayPal sends a single word back - either VERIFIED (if the message matches the original) or INVALID (if the message does not match the original).

Your listener must respond to *every* IPN message it gets, whether you take action on it or not. If you do not respond, PayPal assumes the IPN was not received and re-sends it. Further, PayPal continues to re-send the message periodically until your listener responds, although the interval between retries increases with each attempt. An IPN will be resent for up to four days, with a maximum of 15 retries.

**Important:**

* Every IPN message you receive from PayPal includes a User-Agent HTTP request header whose value is PayPal IPN ( https://www.paypal.com/ipn ). Do ***not*** use this header to verify that an IPN really came from PayPal and has not been tampered with. Rather, to verify these things, you ***must*** use the IPN authentication protocol outlined above.
* PayPal expects to receive a response to an IPN message within 30 seconds. Consequently, your listener must not perform time-consuming operations (such as updating a database) before responding to an IPN.



The numbers in the diagram correspond to the following actions:

1. A user clicks a PayPal button to kick off a checkout flow; your web application makes an API call; your back-office system makes an API call; or PayPal observes an event.
2. PayPal HTTP POSTs your listener an IPN message that notifies you of this event.
3. Your listener returns an empty HTTP 200 response.
4. Your listener HTTP POSTs the complete, unaltered message back to PayPal.

**Note:** This message must contain the same fields, in the same order, as the original IPN from PayPal, all preceded by cmd=\_notify-validate. Further, this message must use the same encoding as the original.

1. PayPal sends a single word back - either VERIFIED (if the message matches the original) or INVALID (if the message does not match the original).

**Important:** To prevent fraud, your IPN listener must implement the IPN authentication protocol (steps 2, 3, 4, and 5 in the diagram above). Upon receipt of a VERIFIED response, your back-office process can parse the contents of the IPN message and respond appropriately - print a packing list, enable a digital download, etc.

**Important:** After you have authenticated an IPN message (received a VERIFIED response from PayPal), ***you must perform these important checks*** before you can assume that the IPN is both legitimate and has not already been processed:

1. Check that the payment\_status is Completed.
2. If the payment\_status is Completed, check the txn\_id against the previous PayPal transaction that you processed to ensure the IPN message is not a duplicate.
3. Check that the receiver\_email is an email address registered in your PayPal account.
4. Check that the price (carried in mc\_gross) and the currency (carried in mc\_currency) are correct for the item (carried in item\_name or item\_number).

### IPN Simulator

Because the most complex part of implementing and IPN solution is creating your listener, PayPal provides templates and examples of listener code to help you. And to help you in testing your listener code, PayPal provides an IPN Simulator tool that you can use to send test IPN messages to the URL at which your listener is running. The IPN Simulator tool lets you verify that your listener is receiving IPN messages and handling them correctly. IPN Simulator notifications include a test\_ipn variable, which is not found in live IPN messages, so the IPN test messages can be distinguished from the real IPN messages.

See [IPN Simulator](https://developer.paypal.com/docs/classic/ipn/integration-guide/IPNSimulator/) for details on using the IPN Simulator.