

# CountBOX Shopify Connector

- WebHook documentation link: <https://help.shopify.com/api/tutorials/webhooks>
- Service location: [terminal.countbox.us/shopify-connector/](https://terminal.countbox.us/shopify-connector/)

## Регламент подключения клиента к системе

1. Создаём запись БД в таблице аккаунтов облака. Здесь:

1.1. api\_email и api\_password - реквизиты для подключения к WebAPI облака CountBOX для выгрузки данных клиента

1.2. sync\_with\_stage - значение b'0' или b'1' в зависимости от того, не требуется или требуется выполнять синхронизацию с тестовым сервером соответственно.

```
INSERT INTO shopify.countbox_accounts (api_email, api_password, sync_with_stage)
VALUES ("api_email", "api_password", "sync_with_stage");
```

2. Запоминаем internal\_id созданной записи

3. Создаём запись БД в таблице торговых точек. Здесь значение cb\_id - это значение internal\_id, полученное на шаге 1-2. Значения countbox\_shop\_id, countbox\_shop\_id\_tt, countbox\_shop\_name - значения реквизитов ТП, доступной под указанными в шаге 1 реквизитами WebAPI, в которую требуется выполнять выгрузку данных.

```
INSERT INTO shopify.shops (cb_id, countbox_shop_id, countbox_shop_id_tt, countbox_shop_name)
VALUES (cb_id, countbox_shop_id, "countbox_shop_id_tt", "countbox_shop_name");
```

4. Запоминаем internal\_id созданной записи

5. Создаём запись в таблице алиасов. Здесь countbox\_accounts\_internal\_id - значение полученное на шаге 2, shops\_internal\_id - значение полученное на шаге 4.

```
INSERT INTO shopify.account_relations (countbox_accounts_internal_id, shops_internal_id)
VALUES (countbox_accounts_internal_id, shops_internal_id);
```

6. Формируем вебхук-ссылку, которую необходимо передать клиенту для подключения его ТП к системе с его стороны. Здесь \${shopId} - это значение, полученное на шаге 4.

```
http://terminal.countbox.us/shopify-connector/${shopId}/orderFulfillment.do
```

7. Передаём клиенту сформированную ссылку клиенту, запрашиваем у клиента:

7.1.Название ТП у него в системе;

7.2.Токен для проверки подлинности получаемых от Shopify данных;

7.3.Информацию о том, требуется ли не учитывать транзакции с нулевой суммой;

7.4 Информацию о том, в каком часовом поясе находится подключаемая ТП.

8.Выполняем следующий запрос к нашей БД, где:

8.1.shopify\_shop\_name - значение, полученное в ответ на 7.1

8.2.shopify\_shop\_name - значение, полученное в ответ на 7.2

8.3.not\_include\_zero\_payments - либо b'0', либо b'1' в зависимости от того, положительный или отрицательный ответ на вопрос пункта 7.3

8.4.time\_shift\_minutes - значение смещения локального времени ТП относительно GMT-0, подсчитанное на основе ответа на 7.4

```
UPDATE shopify.shops SET
shopify_shop_name = "shopify_shop_name", shopify_webhooks_token = "shopify_webhooks_token",
is_enabled = "b'1'", not_include_zero_payments = "not_include_zero_payments",
time_shift_minutes = time_shift_minutes
WHERE `internal_id` = internal_id;
```

9.Просим проверить клиента не заблокирован ли и не удалён ли настроенный им веб-хук. Если это произошло, то переходим к шагу 7. Отключение веб-хука может произойти так как система Shopify автоматически отключает неработоспособные хуки. А хук не может быть готов к работе ранее, чем все данные в базе коннектора будут корректно заполнены. Для уменьшения вероятности этого события рекомендуется выполнять шаги 7-8 в максимально-сжатое время, при этом желательно в момент времени, когда подключаемая ТП не функционирует.

## Описание действий, выполняемых клиентом в процессе подключения

1.Заходим в Settings

2.Переходим в Notifications

~ General ~ Shopify - Chromium

Denis Ryabikov

Settings / General Save

**Store details**

Shopify and your customers will use this information to contact you.

Store name

Account email  Customer email

We'll use this address if we need to contact you about your account. Your customers will see this address if you email them

**Store address**

This address will appear on your invoices. You can edit the address used to calculate shipping rates in your [shipping settings](#).

Legal name of business

Phone

Street

Apt, suite, etc. (optional)

City  Postal / ZIP code

SETTINGS

- General
- Payments
- Checkout
- Shipping
- Taxes
- Notifications**
- Gift cards
- Files
- Sales channels
- Account

DR

3. Листаем до раздела Webhooks. Нажимаем кнопку Create webhook

~ Notifications ~ Shopify - Chromium

Denís Ryabíkov

Settings / Notifications

Notification Location

[Redacted]	All	<a href="#">send test notification</a>	<a href="#">Disable</a>
------------	-----	--	-------------------------

**Desktop notifications**

Get desktop notifications when you receive a new order. This setting only applies to your current browser — you must have the Shopify admin open in this browser to receive desktop notifications.

[Enable desktop notifications](#)

**Webhooks**

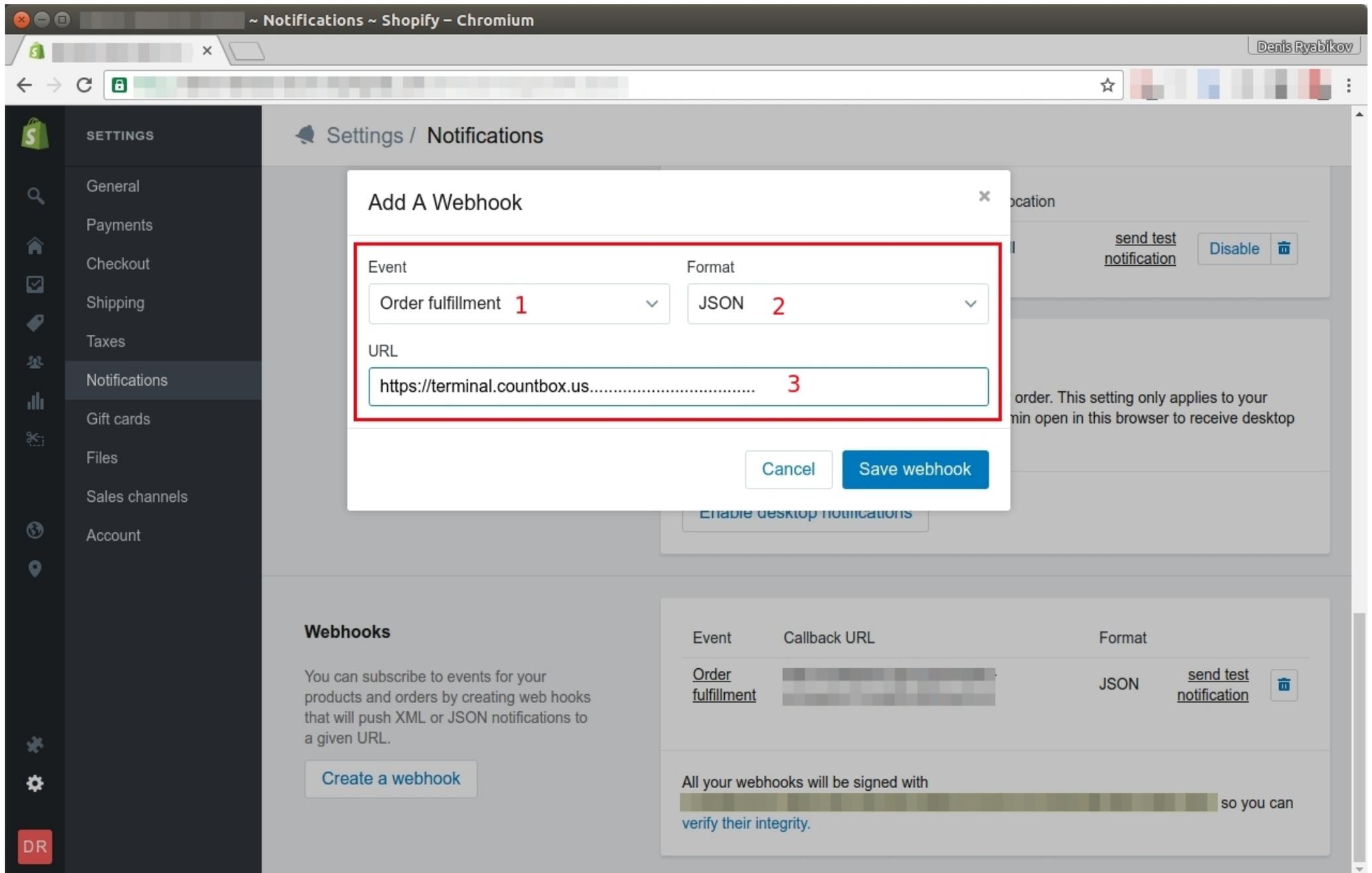
You can subscribe to events for your products and orders by creating web hooks that will push XML or JSON notifications to a given URL.

[Create a webhook](#)

Event	Callback URL	Format	
<a href="#">Order fulfillment</a>	[Redacted]	JSON	<a href="#">send test notification</a>

All your webhooks will be signed with [Redacted] so you can [verify their integrity](#).

4. Заполняем поля 1 и 2 как показано на скриншоте, в поле 3 указываем ссылку на вебхук, сформированную в пункте 6 "регламента подключения клиента к системе"



5. Клиент передаёт значение подписи ("webhooks will be signed with"). После нашей готовности проверяет наличие созданного вебхука в табличке над "подписью", при необходимости пересоздаёт вебхук

~ Notifications ~ Shopify - Chromium

Denís Ryabíkov

Settings / Notifications

SETTINGS

- General
- Payments
- Checkout
- Shipping
- Taxes
- Notifications
- Gift cards
- Files
- Sales channels
- Account

Notification	Location	
[Redacted]	All	<a href="#">send test notification</a> <a href="#">Disable</a>

### Desktop notifications

Get desktop notifications when you receive a new order. This setting only applies to your current browser — you must have the Shopify admin open in this browser to receive desktop notifications.

[Enable desktop notifications](#)

### Webhooks

You can subscribe to events for your products and orders by creating web hooks that will push XML or JSON notifications to a given URL.

[Create a webhook](#)

Event	Callback URL	Format	
<a href="#">Order fulfillment</a>	[Redacted] connector/1/orderFullfillment.do	JSON	<a href="#">send test notification</a>

All your webhooks will be signed with `a3d[Redacted]ed52a34ba82795[Redacted]68350b1526e7fabcd` so you can verify their integrity.