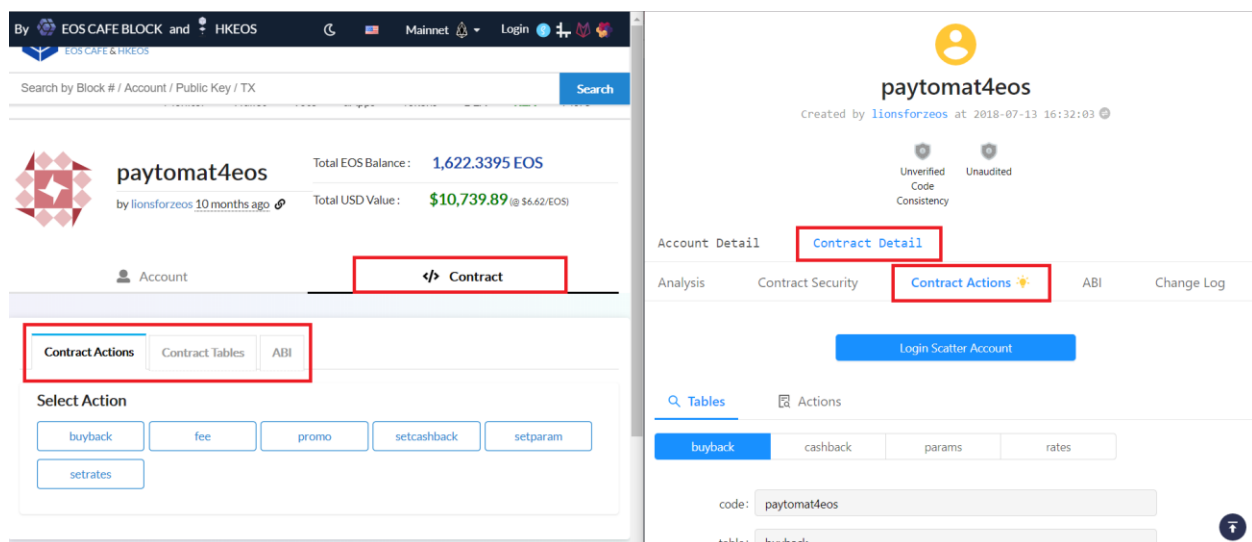


# How to Monitor Revenue Smart Contract

We know there are lots of tech people in our community. We appreciate your support and want to give you something curious to work with. In the [last article](#), we described how to read Tables and Actions in RSC. As a follow-up, we want to show you how do those transactions look like on the blockchain.

Turns out, it's not easy to create an interface for working with smart contracts but thanks to Bloks.io and EOSPark teams we can do it. First and foremost, make sure to open up a *Contract* ([Bloks.io](#)) or *Contract Detail* ([EOSPark](#)) tab and switch to Contract Actions. Typically this tab has subtabs with *Tables* and *Actions*. You may choose the one that is more interesting for you.



The image displays two screenshots of the EOSPark interface for the `paytomat4eos` contract. The left screenshot shows the 'Contract' tab with subtabs 'Contract Actions', 'Contract Tables', and 'ABI'. The right screenshot shows the 'Contract Detail' tab with subtabs 'Analysis', 'Contract Security', 'Contract Actions', 'ABI', and 'Change Log'. Both screenshots show the 'Contract Actions' subtab selected, displaying a table of actions with columns 'code' and 'table'.

code	table
buyback	buyback
cashback	cashback
params	params
rates	rates

In order to receive the data of any contract table, you just choose the name of the table and hit Submit. It will show you the properties this particular table contains and their values. For instance, the table `buyback` displays 3 properties - *symbol*, *balance* and *amount*. You can do the same thing with all of the tables, it works pretty much the same.

Tables

Actions

buyback

cashback

params

rates

code:

paytomat4eos

table:

buyback

scope:

paytomat4eos

string

lower\_bound:

string

upper\_bound:

string

limit:

10

int

index\_position:

primary

▼

key\_type:

string

Submit

Show Code

symbol	balance	amount	⬆
4,PTI	30944.3362 PTI	774575.0239 PTI	<div>⬆</div> <div>⬇</div> <div>💬</div>

You can't use this method with the Actions though. You can try obviously but unless you have an Active key of paytomat4eos account, it's not gonna work.

We assume this type of information can be valuable to people who want to know the changes in the cashback system, auto purchase, and rates settings. Other than that, it's just a service data and may not be that useful for daily usage.

# Tracking transactions

For those who love digging deep, you can watch and monitor the transactions happening in RSC.

Let us walk you through the main ones.

## Revenue generating event

As you already know, at the moment the revenue through Paytomat Wallet is generated during cryptocurrency exchange and credit card withdrawals. As soon as this happens, a fee action executes and RSC initiates 660.1373 PTI transfer from [paytomat4eos](#) to EOS account that triggered this event. In our example that would be *cornsyllable* who successfully exchanged 0.174 LTC. You can also see that 1540.3206 PTIs were sent to *ptistockpile* which means that those tokens were taken out of circulation.

c83d7...	2019-05-16 17:29:03	ptitokenhome::Token Transfer	paytomat4eos → ptistockpile 1540.3206 PTI:ptitokenhome MEMO
c83d7...	2019-05-16 17:29:03	ptitokenhome::Token Transfer	paytomat4eos → cornsyllable 660.1373 PTI:ptitokenhome MEMO Cashback for using Paytomat services
c83d7...	2019-05-16 17:29:03	paytomat4eos::fee	{ "user": "cornsyllable", "amount": "0.17476269 LTC" }

## Setting rates event

Set rates is an action that is executed every 5 minutes to set EOS exchange rates for all of the supported cryptocurrency pairs in the smart contract (currently BTC, ETH, LTC, DASH, EOS).

89b3c...	2019-05-16 17:33:03	paytomat4eos::setrates	{ "symbols": [ "8,BTC", "8,ETH", "8,LTC", "8,DASH", "4,EOS" ], "rates": [ "1219.90369181380424379", "39.85317252228631446", "15.10934393638170903", "23.94077807528744728", "1.0000000000000000" ] }
----------	---------------------	------------------------	---

## Revenue balance indicator check event

This indicator is really important to us since it shows whether we need to purchase extra tokens from an exchange after the cashback funds were sent to the wallet user.

This event is generated every 48 seconds (the value can be adjusted) and triggers the purchase of the tokens from an exchange when the *balance* property in the *buyback* table becomes negative.

As you can see in the sample below, the *buyback* action was executed every 48 seconds until the *growthfather* exchanged 0.2154 ETH. After that happened, a portion of revenue (211.6032 PTI) was transferred to the user in a form of cashback, the other part was sent to a burning account (*ptistockpile*). At this point the value of *balance* in the *buyback* table become negative and it triggered an action to create a market order of 0.2636 EOS worth of PTI on *Newdex* exchange (you can see that at the latest block of the image below).

c2eb5...	2019-05-16 14:01:45	ptitokenhome::Token Transfer	newdexpocket → paytomat4eos 286.5467 PTI:ptitokenhome MEMO {"type":"delivery-transfer","symbol":"ptitokenhome-pti-eos","price":"0.000919","count":"286.5467","random":"190516140144116146"}
a731a...	2019-05-16 14:01:42	Token Transfer	paytomat4eos → newdexpocket 0.2636 EOS:eosio.token MEMO {"type":"buy-market","symbol":"ptitokenhome-pti-eos","price":"0.000000","count":0,"amount":"0.2636","channel":"web","ref":"PAYOMAT","receiver":""}
a731a...	2019-05-16 14:01:42	paytomat4eos::buyback	00
bbe4a...	2019-05-16 14:01:04	ptitokenhome::Token Transfer	paytomat4eos → ptistockpile 493.7408 PTI:ptitokenhome MEMO
bbe4a...	2019-05-16 14:01:04	ptitokenhome::Token Transfer	paytomat4eos → growthharder 211.6032 PTI:ptitokenhome MEMO Cashback for using Paytomat services
bbe4a...	2019-05-16 14:01:04	paytomat4eos::fee	{ "user": "growthharder", "amount": "0.02154861 ETH" }
4cee1...	2019-05-16 14:00:54	paytomat4eos::buyback	00
c29e5...	2019-05-16 14:00:06	paytomat4eos::buyback	00
f893e...	2019-05-16 13:59:18	paytomat4eos::buyback	00
21860...	2019-05-16 13:58:30	paytomat4eos::buyback	00

## Promo event

There's a separate method that is used to arrange different promotional campaigns. In this case, *xiaazhuyiym* receives 10 PTI for creating an EOS account, a promo campaign that was recently started in [Paytomat Wallet](#).

c38d4... 2019-05-16 16:48:21

ptitokenhome::Token Transfer

paytomat4eos → xiaozhuyiym 10.0000 PTI:ptitokenhome  
MEMO Promo from paytomat.com

c38d4... 2019-05-16 16:48:21

paytomat4eos::promo

```
{
  "user": "xiaozhuyiym",
  "amount": "10.0000 PTI"
}
```