

Solidcoin24 API

19th November 2011

Contents

I	How to use the API	2
1	Information about the API	2
1.1	Introduction	2
1.2	User authentication	3
1.3	Errors	3
II	The API	3
2	General	3
2.1	Balance	3
2.1.1	Get balance	3
2.1.2	Balance history	3
2.2	Internal transfer	4
2.2.1	Transfer to user	4
2.2.2	Check address for Solidcoin24 affiliation	4
3	Exchange	5
3.1	Trade data	5
3.1.1	Ticker	5
3.1.2	Orderbook	5
3.1.3	Trades	5
3.2	Trading	6
3.2.1	New order	6
3.2.2	Cancel order	6
3.2.3	Order info	7
3.2.4	List orders	7

4	Transfer	7
4.1	Deposits	7
4.1.1	Create deposit address	7
4.1.2	Get deposit address info by ID	8
4.1.3	Get deposit address info by address	8
4.1.4	Get deposit address info by group	8
4.1.5	Update deposit address by ID	9
4.1.6	Update deposit address by address	9
4.1.7	Get deposits by deposit address ID	9
4.1.8	Get deposits by deposit address	10
4.1.9	Get deposit addresses	10
4.2	Withdrawals	10
4.2.1	Create withdrawal address	10
4.2.2	Create withdrawal address and withdraw	11
4.2.3	Create withdrawal addresses and withdraw	11
4.2.4	Withdraw to withdrawal address by ID	12
4.2.5	Get withdrawal address info by ID	12
4.2.6	Get withdrawal address info by address	12
4.2.7	Get withdrawal address info by group	12
4.2.8	Update withdrawal address by ID	13
4.2.9	Update withdrawal address by address	13
4.2.10	Get withdrawals by withdrawal address ID	13
4.2.11	Get withdrawals by withdrawal address	14
4.2.12	Get withdrawal addresses	14
III	Services	14
5	Transfer service	14
5.1	Introduction	14
5.2	Deposit addresses and deposits	15
5.3	Withdrawal addresses and withdrawals	15

Part I

How to use the API

1 Information about the API

1.1 Introduction

The API can be used to automatically trade, check your balance or transfer Solidcoins externally and internally.

1.2 User authentication

There are currently two ways to authenticate for the API. You can pass `base64({id_string}:{password})` after `api/` like that: `https://slc24.com/api/base64({id_string}:{password})/balance/{currency}`

The second option is to use WWW-authentication. The username is your id string which can be found in the API section of the Solidcoin24 site when you've logged in.

1.3 Errors

If an error occurs during an API call you will get a json array with one field. The field's name is `error` and its content is a message describing why the call failed.

Part II

The API

2 General

2.1 Balance

2.1.1 Get balance

`https://slc24.com/api/balance/{currency}`

Parameters

`{currency}` specifies the currency and currently can be `slc` (Solidcoins), `btc` (Bitcoins) or `nmc` (Namecoins); this parameter is optional

Returns The amount of Solidcoins, Bitcoins or Namecoins if a currency is provided. If there's no currency passed it returns a json array with all balances as values and the currencies as keys.

Examples

Without currency (`api/balance`) `{"slc":14.39,"btc":0.691235,"nmc":0}`

With currency (`api/balance/slc`) `{"balance":14.39}`

2.1.2 Balance history

`https://slc24.com/api/balance/history/{currency}`

This call requires user authentication.

Parameters

`{currency}` specifies the currency and currently can be `slc` (Solidcoins), `btc` (Bitcoins) or `nmc` (Namecoins); this parameter is optional

Returns An array containing all the transactions. Each transactions is an array with the fields `type` (internal or external), `dir` (in or out), `amount`, `currency`, `balance` (balance after this transaction), `time` (UNIX timestamp), `info` (withdrawal, deposit, trade_placement, ...) and `info_id` (e.g. id of a trade order).

2.2 Internal transfer

2.2.1 Transfer to user

`https://slc24.com/api/internal_transfer/{currency}?r={recipient_email}&am={amount}`

Parameters

`{currency}` specifies the currency and currently can be `slc` (Solidcoins), `btc` (Bitcoins) or `nmc` (Namecoins)

`{amount}` amount to transfer

Returns Nothing (“”).

2.2.2 Check address for Solidcoin24 affiliation

`https://slc24.com/api/internal_transfer/check_address/{currency}?a={address}`

Parameters

`{currency}` specifies the address type and currently can be `slc` (Solidcoins), `btc` (Bitcoins) or `nmc` (Namecoins); this parameter is optional (Solidcoin is default)

`{address}` Solidcoin, Bitcoin or Namecoin address

Returns A unique hash that anonymously identifies a user or null if it's not a Solidcoin24 address.

Examples

Successful `{"hash": "28b8de7cb9f7da1656fda6bf472d6fab"}`

Unsuccessful { "hash":null }

3 Exchange

3.1 Trade data

3.1.1 Ticker

<https://slc24.com/api/exchange/ticker/{currency}>

Parameters

`{currency}` specifies the currency and currently can be `btc` (Bitcoins) or `nmc` (Namecoins); this parameter is optional

Returns A json array, containing the fields `ask` (highest ask price), `bid` (lowest bid price), `last` (price of the last trade), `last_when` (unix timestamp of the last trade), `volume` (amount of Solidcoins traded during the last 24h), `high` (highest price of the trades during the last 24h), `low` (lowest price of the trades during the last 24h) and `average` (average price of the last 24h weighted by volume) if a currency is provided. If there's no currency passed it returns a json array containing all tickers as values and the currencies as keys.

Example (`api/exchange/ticker/btc`) { "ask":0.0125,"bid":0.01,"last":0.01,"last_when":1320846297,"

3.1.2 Orderbook

<https://slc24.com/api/exchange/orderbook/{currency}>

Parameters

`{currency}` specifies the currency and currently can be `btc` (Bitcoins) or `nmc` (Namecoins); this parameter is optional

Returns If a currency is provided the call returns a json array containing the fields `ask` and `bid`. Each of those two fields is an array that has an unlimited number of trade orders as entries. Each trade order is represented by an array with the fields `price` (price of the trade order) and `amount` (amount of Solidcoins at this price). If there's no currency passed it returns a json array with all orderbooks as values and the currencies as keys. The entries are sorted in the same way as on the Solidcoin24 site.

3.1.3 Trades

<https://slc24.com/api/exchange/trades/{currency}?t={timestamp}>

Parameters

`{currency}` specifies the currency and currently can be `btc` (Bitcoins) or `nmc` (Namecoins); this parameter is optional (i.e. may be left out; remove the slash in this case)

`{timestamp}` only trades that are newer than or equal to this given UNIX timestamp will be shown; this parameter is optional

Returns If a currency is provided this call returns a json array containing trades ascendingly sorted by time. Each trade is an array with the fields `id` (id of the trade), `time` (UNIX timestamp), `price` and `amount` (Solidcoin amount). If there's no currency passed it returns a json array with all trades as values and the currencies as keys. The maximum number of trades shown is 2000.

3.2 Trading

3.2.1 New order

`https://slc24.com/api/exchange/order/new/{currency}/{type}?p={price}&am={amount}`

Parameters

`{currency}` specifies the currency and currently can be `btc` (Bitcoins) or `nmc` (Namecoins)

`{type}` can be either `buy` or `sell`

`{price}` price of the order in Solidcoins per Bitcoins or Namecoins

`{amount}` amount of Solidcoins to buy or sell

Returns The id of the newly created trade order.

Example `{"id":9}`

3.2.2 Cancel order

`https://slc24.com/api/exchange/order/cancel?id={id}`

Parameters

`{id}` ID of the order to cancel

Returns Nothing (`""`).

Example `[]`

3.2.3 Order info

<https://slc24.com/api/exchange/order/info?id={id}>

Parameters

`{id}` ID of the order to cancel

Returns A json array with the fields `creation_time` (UNIX timestamp of the creation), `finishing_time` (UNIX timestamp of the completion or cancellation), `currency`, `type` (buy or sell), `price`, `amount` (amount of Solidcoins), `completed` (amount of Solidcoins already completed) and `active` (yes or no).

3.2.4 List orders

<https://slc24.com/api/exchange/orders/{currency}>

Parameters

`{currency}` specifies the currency and currently can be `btc` (Bitcoins) or `nmc` (Namecoins); this parameter is optional

Returns If a currency is provided this call returns a json array containing all of your active and inactive (finished) orders. Each entry of the array has the same fields as the order info call plus the field `id`. If there's no currency passed it returns a json array with all orders as values and the currencies as keys.

4 Transfer

4.1 Deposits

4.1.1 Create deposit address

https://slc24.com/api/transfer/deposit/new?g={group}&m={send_mail}&c={callback}&d={data}

Parameters

`{group}` group to identify the deposit address; this parameter is optional

`{send_mail}` can be yes or no depending on whether you wish to receive an email upon deposit or not

`{callback}` a callback URL that is called upon deposit (see above for more information); this parameter is optional

`{data}` up to 1024 characters where you can store whatever you want (e.g. serialized data); this parameter is optional

Returns A json array with the fields `id` and `address` (Solidcoin deposit address).

4.1.2 Get deposit address info by ID

`https://slc24.com/api/transfer/deposit/info?id={id}`

Parameters

`{id}` deposit address ID

Returns A json array containing the fields `address` (Solidcoin address), `group`, `send_mail` (yes or no), `callback`, `data`, `total_amount` (total received amount not including `pending_amount`), `pending_amount` (pending amount, i.e. deposits with less than 6 confirmations) and `new_amount` (additional amount since last check, becomes 0 after this call).

4.1.3 Get deposit address info by address

`https://slc24.com/api/transfer/deposit/info?a={address}`

Parameters

`{address}` deposit address

Returns A json array containing the fields `id`, `group`, `send_mail` (yes or no), `callback`, `data`, `total_amount` (total received amount not including `pending_amount`), `pending_amount` (pending amount, i.e. deposits with less than 6 confirmations) and `new_amount` (additional amount since last check, becomes 0 after this call).

4.1.4 Get deposit address info by group

`https://slc24.com/api/transfer/deposit/info?g={group}`

Parameters

`{group}` deposit address group

Returns A json array containing an array for every deposit address of that group. Every deposit address has the fields `id`, `address`, `send_mail` (yes or no), `callback`, `data`, `total_amount` (total received amount not including `pending_amount`), `pending_amount` (pending amount, i.e. deposits with less than 6 confirmations) and `new_amount` (additional amount since last check, becomes 0 after this call).

4.1.5 Update deposit address by ID

`https://slc24.com/api/transfer/deposit/update?id={id}&g={group}&m={send_mail}&c={callback}&d={data}`

Parameters

`{id}` deposit address ID

`{group}` group to identify the deposit address; this parameter is optional

`{send_mail}` can be yes or no depending on whether you wish to receive an email upon deposit or not

`{callback}` a callback URL that is called upon deposit (see above for more information); this parameter is optional

`{data}` up to 1024 characters where you can store whatever you want (e.g. serialized data); this parameter is optional

Returns Nothing (``[]``).

4.1.6 Update deposit address by address

`https://slc24.com/api/transfer/deposit/update?a={address}&g={group}&m={send_mail}&c={callback}&d={data}`

Parameters

`{address}` deposit address

`{group}` group to identify the deposit address; this parameter is optional

`{send_mail}` can be yes or no depending on whether you wish to receive an email upon deposit or not

`{callback}` a callback URL that is called upon deposit (see above for more information); this parameter is optional

`{data}` up to 1024 characters where you can store whatever you want (e.g. serialized data); this parameter is optional

Returns Nothing (``[]``).

4.1.7 Get deposits by deposit address ID

`https://slc24.com/api/transfer/deposit/deposits?id={id}`

Parameters

{id} deposit address ID

Returns A json array containing an array for every deposit. Every deposit has the fields **txid** (Solidcoin transaction id), **time** (UNIX timestamp), **amount** and **type** (intern or extern).

4.1.8 Get deposits by deposit address

<https://slc24.com/api/transfer/deposit/deposits?a={address}>

Parameters

{address} deposit address

Returns A json array containing an array for every deposit. Every deposit has the fields **txid** (Solidcoin transaction id), **time** (UNIX timestamp), **amount** and **type** (intern or extern).

4.1.9 Get deposit addresses

<https://slc24.com/api/transfer/deposit/addresses>

Returns A json array containing an array for every deposit address. Every deposit address has the fields **id**, **address** and **group**, **callback** and **send_mail** (yes or no).

4.2 Withdrawals

4.2.1 Create withdrawal address

<https://slc24.com/api/transfer/withdrawal/new?a={address}&g={group}&d={data}>

Behaviour Creates a new withdrawal address with the provided address, group and data.

Parameters

{address} Solidcoin address to which the Solidcoins are to be withdrawn

{group} group to identify the withdrawal address; this parameter is optional

{data} up to 1024 characters where you can store whatever you want (e.g. serialized data); this parameter is optional

Returns An array containing `id` (the ID of the withdrawal address) and `type` (intern in case the withdrawal address belongs to a Solidcoin24 account or extern).

4.2.2 Create withdrawal address and withdraw

`https://slc24.com/api/transfer/withdrawal/withdraw?a={address}&am={amount}&g={group}&d={data}`

Behaviour Creates a new withdrawal address with the provided address, group and data and withdraws the provided amount of Solidcoins. If a withdrawal address with this address already exists the provided fields are not updated and the amount withdrawn to the address. This is the standard withdrawal call.

Parameters

`{address}` Solidcoin address to which the Solidcoins are to be withdrawn

`{amount}` Amount of Solidcoins to withdraw

`{group}` group to identify the withdrawal address; this parameter is optional

`{data}` up to 1024 characters where you can store whatever you want (e.g. serialized data); this parameter is optional

Returns An array containing `id` (the ID of the withdrawal address), `txid` (the Solidcoin transaction id in case the address does not belong to a Solidcoin24 account) and `type` (intern in case the withdrawal address belongs to a Solidcoin24 account or extern).

4.2.3 Create withdrawal addresses and withdraw

`https://slc24.com/api/transfer/withdrawal/withdraw?a[0]={address1}&a[1]={address2}&...&am[0]={amount1}&am[1]={amount2}&...&g={group}`

Behaviour Creates new withdrawal addresses with the provided addresses and group and withdraws the provided amounts of Solidcoins. If a withdrawal address with this address already exists the group field is not updated and the amount withdrawn to the address.

Parameters

`{address1},{address2},...` Solidcoin addresses to which the Solidcoins are to be withdrawn

`{amount1},{amount2},...` Amounts of Solidcoins to withdraw

`{group}` group to identify the withdrawal address; this parameter is optional

Returns An array containing an entry for each address with the fields `id` (the ID of the withdrawal address), `txid` (the Solidcoin transaction id in case the address does not belong to a Solidcoin24 account), `address`, `amount` and `type` (intern in case the withdrawal address belongs to a Solidcoin24 account or extern). The `txid` field is the same for all external transactions.

4.2.4 Withdraw to withdrawal address by ID

<https://slc24.com/api/transfer/withdrawal/withdraw?id={id}&am={amount}>

Parameters

`{id}` withdrawal address ID

`{amount}` Amount of Solidcoins to withdraw

Returns An array containing `txid` (the Solidcoin transaction id or an intern transaction ID in case the withdrawal address belongs to a Solidcoin24 user) and `type` (intern or extern).

4.2.5 Get withdrawal address info by ID

<https://slc24.com/api/transfer/withdrawal/info?id={id}>

Parameters

`{id}` withdrawal address ID

Returns A json array containing the fields `address`, `group`, `data`, `total_amount` (total sent amount) and `type` (intern in case the withdrawal address belongs to a Solidcoin24 account or extern).

4.2.6 Get withdrawal address info by address

<https://slc24.com/api/transfer/withdrawal/info?a={address}>

Parameters

`{address}` withdrawal address

Returns A json array containing the fields `id`, `group`, `data`, `total_amount` (total sent amount) and `type` (intern in case the withdrawal address belongs to a Solidcoin24 account or extern).

4.2.7 Get withdrawal address info by group

<https://slc24.com/api/transfer/withdrawal/info?g={group}>

Parameters

`{group}` withdrawal address group

Returns A json array containing an array for every withdrawal address of that group. Every withdrawal address has the fields `id`, `address`, `data`, `total_amount` (total sent amount) and `type` (intern in case the withdrawal address belongs to a Solidcoin24 account or extern).

4.2.8 Update withdrawal address by ID

<https://slc24.com/api/transfer/withdrawal/update?id={id}&g={group}&d={data}>

Parameters

`{id}` withdrawal address ID

`{group}` group to identify the withdrawal address; this parameter is optional

`{data}` up to 1024 characters where you can store whatever you want (e.g. serialized data); this parameter is optional

Returns Nothing (“”).

4.2.9 Update withdrawal address by address

<https://slc24.com/api/transfer/withdrawal/update?a={address}&g={group}&d={data}>

Parameters

`{address}` withdrawal address

`{group}` group to identify the withdrawal address; this parameter is optional

`{data}` up to 1024 characters where you can store whatever you want (e.g. serialized data); this parameter is optional

Returns Nothing (“”).

4.2.10 Get withdrawals by withdrawal address ID

<https://slc24.com/api/transfer/withdrawal/withdrawals?id={id}>

Parameters

`{id}` withdrawal address ID

Returns A json array containing an array for every withdrawal. Every withdrawal has the fields `txid` (Solidcoin transaction id in case the withdrawal address is external or an intern transaction id in case the withdrawal address is intern), `time` (UNIX timestamp) and `amount`.

4.2.11 Get withdrawals by withdrawal address

`https://slc24.com/api/transfer/withdrawal/withdrawals?a={address}`

Parameters

`{address}` withdrawal address

Returns A json array containing an array for every withdrawal. Every withdrawal has the fields `txid` (Solidcoin transaction id in case the withdrawal address is extern or an intern transaction id in case the withdrawal address is intern), `time` (UNIX timestamp) and `amount`.

4.2.12 Get withdrawal addresses

`https://slc24.com/api/transfer/withdrawal/addresses`

Returns A json array containing an array for every withdrawal address. Every withdrawal address has the fields `id`, `address`, `group` and `type` (intern or extern).

Part III

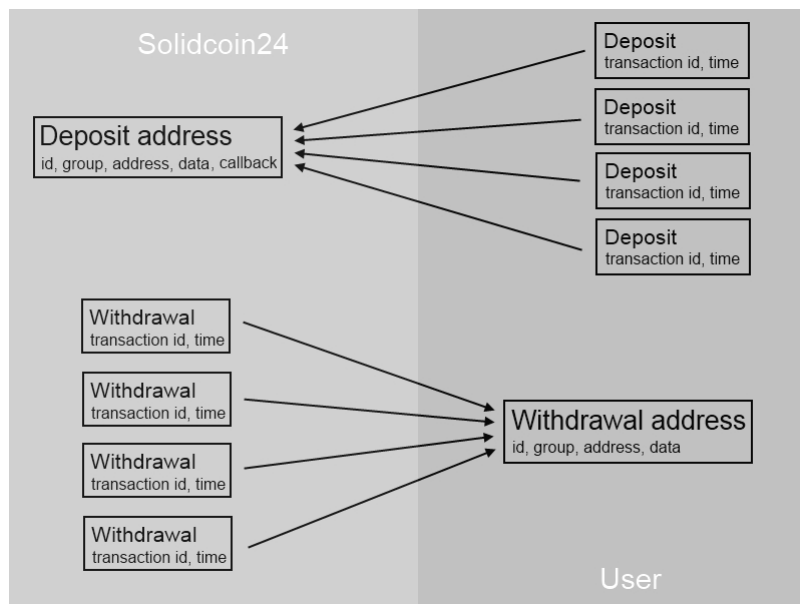
Services

5 Transfer service

5.1 Introduction

To understand the transfer calls in section 4 you need to know how the transfer service works.

Fig. 1



As you can see, there are no longer checkouts or payouts but rather deposit addresses and withdrawal addresses. The reason for this decision is that by keeping the functionality of the transfer service as close to the original Solidcoin client as possible you have maximum flexibility. The old functions can still easily be implemented in your favourite language. Even though deposits and withdrawal addresses are on the user-side here, they're both stored on the Solidcoin24 servers.

5.2 Deposit addresses and deposits

A deposit address can be created in order to receive deposits or payments. You can pass a group and data field when creating a deposit address. The group field can be used to identify a group of deposit addresses. You can use a group for each customer for instance. The data field can be used to store serialized data and can hold up to 1024 characters. This way you don't need to create a database for every small data that needs to be stored. Deposits are created automatically when someone sends Solidcoins to a deposit address. Every time this happens the callback URL is called (provided you set one) with the following GET parameters: `id`, `group`, `address`, `data` and `total_amount`, `new_amount` (the amount that was added on this call) and `txid` (the transaction id).

5.3 Withdrawal addresses and withdrawals

Before you can send Solidcoins to someone you need to create a withdrawal address. After that you can send Solidcoins to this address and create a withdrawal

this way. These two steps can be done in one call, however (see 4.2.2). The group and data fields have the same purpose as above. If an address provided belongs to a Solidcoin24 account (a regular deposit address or a transfer deposit address) this will be noticed and the transfer happens instantly.