



# Web Services Support

Spotware Systems Ltd

Version 78, 2022-04-27



# Table Of Contents

1. General Description .....	1
2. Operations with accounts .....	2
2.1. Generate Manager's token .....	2
2.2. Create Trader .....	3
2.3. Read Trader's Details .....	4
2.4. Read Details of all Traders .....	5
2.5. Read SubAccount Details .....	6
2.6. Update Trader .....	7
2.7. Delete Trader .....	8
2.8. Change Trader's Balance .....	9
2.9. Change Trader's Password .....	10
2.10. Check Trader's Password .....	11
2.11. Change Trader's Bonus .....	12
2.12. List Trader Groups .....	13
2.13. List of open positions .....	14
2.14. List of closed positions .....	15
2.15. Create Withdrawal Request .....	17
3. Operations with cTID .....	18
3.1. Create cTID .....	19
3.2. Link account to cTID .....	20
3.3. Change Whitelabel of the Trading account .....	21
4. Error codes .....	22

# 1. General Description

We expose the following RESTful web services for operations with Accounts (see the table below).

The content type for requests and responses is text/xml or text/json.

Type definitions can be found at <https://HOST:PORT/v2/webserv/schema>

**All URLs are relative to <https://HOST:PORT/v2>**

The web services API is authenticated under the same Manager credentials which are used to login to the cBroker application. Usage of the API requires inclusion of an authentication token with each request by appending `?token={token}` to the end of the request URL. Token doesn't have expiration time and could be generated according to section [Generate Manager's token](#).

## Notes:

1. The Volume parameter contain the amount in  $10^2$  (e.g. `volume=234512` is 2345.12 base asset units).
2. All fields related to money (balance, commission, etc.) contain the amount in  $10^{\text{moneyDigits}}$  (e.g. `balance=234512` with `moneyDigits=2` is 2345.12 currency units; `balance=234512` with `moneyDigits=8` is 0.00234512 currency units).
3. Number of requests per hour may be limited due to abnormal and harmful usage. In case limit is activated all new PUT/POST/DELETE requests will be rejected. GET requests won't be limited.

## 2. Operations with accounts

### 2.1. Generate Manager's token

**HTTP Method:**

POST

**URL:**

/webserv/managers/token

**Inputs:**

GetWebservTokenReqTO

- **login** is your cBroker Manager's login
- **hashedPassword** is MD5 of your cBroker Manager's password

**Output:**

GetWebservTokenResTO

**Expected HTTP status code:**

200

**Example:**

```
curl -X POST "https://HOST:PORT/v2/webserv/managers/token" -H "accept: application/json" -H "Content-Type: application/json" -d "{ \"hashedPassword\": \"0f94e246908667af85916300c57f74b6\", \"login\": 9017800}"
```

```
{  
  "webservToken": "1dd4ef40-c5b3-61c0-0689-b1b40c97fadc"  
}
```

## 2.2. Create Trader

### HTTP Method:

POST

### URL:

/webserv/traders

### Inputs:

CreateTraderReqTO

### Output:

TraderTO

### Expected HTTP status code:

201

### Example:

```
curl -X POST "https://HOST:PORT/v2/webserv/traders?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fad" -H "accept: application/json" -H "Content-Type: application/json" -d "{
  \"accessRights\": \"FULL_ACCESS\", \"accountType\": \"HEDGED\", \"balance\": 1000,
  \"brokerName\": \"BESTBROKER\", \"contactDetails\": { \"address\": \"Moon\", \"city\":
  \"Lake\", \"countryId\": 8, \"documentId\": \"0123\", \"email\":
  \"president@moon.moon\", \"phone\": \"+50987654321\", \"state\": \"RE\",
  \"zipCode\": \"5500\" }, \"defaultSplitRevenue\": true, \"depositCurrency\": \"EUR\",
  \"description\": \"string\", \"enabled\": true, \"frenchRisk\": true, \"groupName\":
  \"Default\", \"hashedPassword\": \"d8578edf8458ce06fbc5bb76a58c5ca4\", \"lastName\":
  \"President\", \"leverageInCents\": 10000, \"maxLeverage\": 100000, \"name\":
  \"Best\", \"sendOwnStatement\": true, \"swapFree\": false,
  \"totalMarginCalculationType\": \"MAX\"}"
```

```
{"login":9017800,"groupName":"Default","depositCurrency":"EUR","name":"Best","lastName
":"President","description":"string","accessRights":"FULL_ACCESS","balance":1000,"bonu
s":0,"nonWithdrawableBonus":0,"leverageInCents":10000,"contactDetails":{"email":"presi
dent@moon.moon","address":"Moon","city":"Lake","state":"RE","zipCode":"5500","countr
yId":8,"documentId":"0123","phone":"+50987654321"},"registrationTimestamp":15988620138
24,"lastUpdateTimestamp":1598862013825,"equity":1000,"usedMargin":0,"freeMargin":1000,
"accountType":"HEDGED","introducingBroker":true,"introducingBrokerCommissionRate":0,
"pocketCommissionRate":0,"pocketMarkupRate":0,"defaultIntroducingBrokerCommissionRate
":0,"defaultPocketCommissionRate":0,"defaultPocketMarkupRate":0,"defaultRebateRate":0,
"defaultSplitRevenue":true,"limitedRisk":false,"sendOwnStatement":true,"swapFree":fal
se,"splitRevenue":false,"rankId":1,"ranks":{"rank":[{"id":1,"name":"Standard","volume
":0,"parentIbPercentage":35,"brokerPercentage":30}]}, "totalMarginCalculationType":"MAX"
,"maxLeverage":100000,"frenchRisk":true,"isLimitedRisk":true,"limitedRiskMarginCalcu
lationStrategy":"ACCORDING_TO_LEVERAGE","moneyDigits":2,"sendStatementToBroker":false,
"defaultIbCommissionsType":"USD_PER_LOT","ibCommissionsType":"USD_PER_MILLION_USD"}
```

## 2.3. Read Trader's Details

**HTTP Method:**

GET

**URL:**

/webserv/traders/{login}

**Inputs:**

login

**Output:**

TraderTO

**Expected HTTP status code:**

200

**Example:**

```
curl -X GET "https://HOST:PORT/v2/webserv/traders/9017800?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fadc" -H "accept: application/json"
```

```
{
  "login": 9017800,
  "groupName": "Default",
  "depositCurrency": "EUR",
  "name": "Best",
  "lastName": "President",
  "description": "string",
  "accessRights": "FULL_ACCESS",
  "balance": 1000,
  "bonuses": 0,
  "nonWithdrawableBonus": 0,
  "leverageInCents": 10000,
  "contactDetails": {
    "email": "president@moon.moon",
    "address": "Moon",
    "city": "Lake",
    "state": "RE",
    "zipCode": "5500",
    "countryId": 8,
    "documentId": "0123",
    "phone": "+50987654321"
  },
  "registrationTimestamp": 1598862013824,
  "lastUpdateTimestamp": 1598862013825,
  "equity": 1000,
  "usedMargin": 0,
  "freeMargin": 1000,
  "accountType": "HEDGED",
  "introducingBroker": true,
  "introducingBrokerCommissionRate": 0,
  "pocketCommissionRate": 0,
  "pocketMarkupRate": 0,
  "defaultIntroducingBrokerCommissionRate": 0,
  "defaultPocketCommissionRate": 0,
  "defaultPocketMarkupRate": 0,
  "defaultRebateRate": 0,
  "defaultSplitRevenue": true,
  "limitedRisk": false,
  "sendOwnStatement": true,
  "swapFree": false,
  "splitRevenue": false,
  "rankId": 1,
  "ranks": {
    "rank": [
      {
        "id": 1,
        "name": "Standard",
        "volume": 0,
        "parentIbPercentage": 35,
        "brokerPercentage": 30
      }
    ]
  },
  "totalMarginCalculationType": "MAX",
  "maxLeverage": 100000,
  "frenchRisk": true,
  "isLimitedRisk": true,
  "limitedRiskMarginCalculationStrategy": "ACCORDING_TO_LEVERAGE",
  "moneyDigits": 2,
  "sendStatementToBroker": false,
  "defaultIbCommissionsType": "USD_PER_LOT",
  "ibCommissionsType": "USD_PER_MILLION_USD"
}
```

## 2.4. Read Details of all Traders

### HTTP Method:

GET

### URL:

/webserv/traders/

### Inputs:

- from timestamp (inclusive) formatted as `yyyy-MM-ddTHH:mm:ss.SSS` in UTC (example: 2018-01-01T12:12:12.000)
- to timestamp (exclusive) formatted as `yyyy-MM-ddTHH:mm:ss.SSS` in UTC (example: 2018-01-01T12:12:12.000)
- groupId int if specified only accounts of this group will be returned
- token Manager token

### Output:

TraderListTO and field required bool hasMore

### Expected HTTP status code:

200

### Example:

```
curl -X GET "https://HOST:PORT/v2/webserv/traders/?from=2020-09-03T11:55:32.773&to=2022-09-03T11:58:08.355&token=1dd4ef40-c5b3-61c0-0689-b1b40c97fadc" -H "accept: application/json"
```

```
{ "trader": [ { "login": 47, "groupName": "testUSD", "depositCurrency": "USD", "name": "brahmaputra", "accessRights": "FULL_ACCESS", "balance": 10000000, "bonus": 0, "nonWithdrawableBonus": 0, "leverageInCents": 10000, "contactDetails": { "email": "default@ctrader.com", "city": "delhi", "countryId": 356 }, "registrationTimestamp": 1632960000000, "lastUpdateTimestamp": 1632960000000, "lastConnectionTimestamp": 1632960000000, "equity": 10000000, "usedMargin": 0, "freeMargin": 10000000, "accountType": "HEDGED", "introducingBroker": false, "introducingBrokerCommissionRate": 0, "pocketCommissionRate": 0, "pocketMarkupRate": 0, "rebateRate": 0, "defaultIntroducingBrokerCommissionRate": 0, "defaultPocketCommissionRate": 0, "defaultPocketMarkupRate": 0, "defaultRebateRate": 0, "defaultSplitRevenue": false, "limitedRisk": false, "sendOwnStatement": false, "swapFree": false, "splitRevenue": false, "ranks": { "rank": [] }, "totalMarginCalculationType": "MAX", "frenchRisk": false, "isLimitedRisk": false, "moneyDigits": 2, "defaultIbCommissionsType": "USD_PER_MILLION_USD", "ibCommissionsType": "USD_PER_MILLION_USD" }, { "login": 60, ...
```

## 2.5. Read SubAccount Details

**HTTP Method:**

GET

**URL:**

/webserv/traders/{login}/subaccounts

**Inputs:**

login

**Output:**

SubAccountsTotalTO

**Expected HTTP status code:**

200

**Example:**

```
curl -X GET "https://HOST:PORT/v2/webserv/traders/9017800/subaccounts?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fadc" -H "accept: application/json"
```

```
{"subAccountLogins":{"login":[9006731,9006762,9006765]}, "totalSubAccountsBalance":46016, "totalSubAccountsEquity":47324}
```



## 2.6. Update Trader

**HTTP Method:**

PUT

**URL:**

/webserv/traders/{login}

**Inputs:**

UpdateTraderReqTO

**Output:**

none

**Expected HTTP status code:**

204

**Important note:**

On update all fields of the the trader must be sent otherwise they will be set to null or to a default value

**Example:**

```
curl -X PUT "https://HOST:PORT/v2/webserv/traders/9017800?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fad" -H "accept: application/json" -H "Content-Type: application/json" -d "{ \"accessRights\": \"FULL_ACCESS\", \"brokerName\": \"string\", \"contactDetails\": { \"address\": \"string\", \"city\": \"string\", \"countryId\": 0, \"documentId\": \"string\", \"email\": \"string\", \"introducingBroker1\": \"string\", \"introducingBroker2\": \"string\", \"phone\": \"string\", \"phonePassword\": \"string\", \"state\": \"string\", \"status\": \"string\", \"zipCode\": \"string\" }, \"defaultIntroducingBrokerCommissionRate\": 0, \"defaultPocketCommissionRate\": 0, \"defaultPocketMarkupRate\": 0, \"defaultRebateRate\": 0, \"defaultSplitRevenue\": true, \"description\": \"string\", \"frenchRisk\": true, \"groupName\": \"string\", \"introducedByBroker\": 0, \"introducingBroker\": true, \"introducingBrokerCommissionRate\": 0, \"isLimitedRisk\": true, \"lastName\": \"string\", \"leverageInCents\": 0, \"limitedRisk\": true, \"limitedRiskMarginCalculationStrategy\": \"ACCORDING_TO_LEVERAGE\", \"login\": 0, \"maxLeverage\": 0, \"name\": \"string\", \"pocketCommissionRate\": 0, \"pocketMarkupRate\": 0, \"rankId\": 0, \"ranks\": { \"rank\": [ { \"brokerPercentage\": 0, \"id\": 0, \"name\": \"string\", \"parentIbPercentage\": 0, \"volume\": 0 } ] }, \"sendOwnStatement\": true, \"splitRevenue\": true, \"swapFree\": true}"
```

## 2.7. Delete Trader

**HTTP Method:**

DELETE

**URL:**

/webserv/traders/{login}

**Inputs:**

login

**Output:**

none

**Expected HTTP status code:**

204

**Example:**

```
curl -X DELETE "https://HOST:PORT/v2/webserv/traders/9017800?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fad0"
```

## 2.8. Change Trader's Balance

### HTTP Method:

POST

### URL:

/webserv/traders/{login}/changebalance

### Inputs:

- login
- ChangeTraderBalanceReqTO

### Output:

ChangeTraderBalanceResTO

### Expected HTTP status code:

200

### Example:

```
curl -X POST
"https://HOST:PORT/v2/webserv/traders/9017800/changebalance?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fad0" -H "accept: application/json" -H "Content-Type: application/json"
-d '{"amount": 12345, "comment": "string", "externalId": "string",
"externalNote": "string", "login": 9017800, "preciseAmount": 123.45,
"source": "string", "type": "DEPOSIT"}'
```

```
{"balanceHistoryId":6340680}
```

## 2.9. Change Trader's Password

**HTTP Method:**

POST

**URL:**

/webserv/traders/{login}/changepassword

**Inputs:**

- login
- ChangeTraderPasswordReqTO

**Output:**

none

**Expected HTTP status code:**

204

**Example:**

```
curl -X POST
"https://HOST:PORT/v2/webserv/traders/9017800/changepassword?token=1dd4ef40-c5b3-61c0-
0689-b1b40c97fad" -H "accept: application/json" -H "Content-Type: application/json"
-d '{"hashedPassword": "\0f94e246908667af85916300c57f74b6", "login": 9017800}'
```

## 2.10. Check Trader's Password

**HTTP Method:**

POST

**URL:**

/webserv/traders/{login}/checkpassword

**Inputs:**

- login
- CheckTraderPasswordReqTO

**Output:**

none

**Expected HTTP status code:**

204

**Example:**

```
curl -X POST
"https://HOST:PORT/v2/webserv/traders/9017800/checkpassword?token=1dd4ef40-c5b3-61c0-
0689-b1b40c97fad" -H "accept: application/json" -H "Content-Type: application/json"
-d '{"hashedPassword": "\0f94e246908667af85916300c57f74b6", "login": 9017800}'
```

## 2.11. Change Trader's Bonus

### HTTP Method:

POST

### URL:

/webserv/traders/{login}/changebonus

### Inputs:

- login
- ChangeTraderBonusReqTO

### Output:

ChangeTraderBonusResTO

### Expected HTTP status code:

200

### Example:

```
curl -X POST "https://HOST:PORT/v2/webserv/traders/9017800/changebonus?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fadc" -H "accept: application/json" -H "Content-Type: application/json" -d "{ \"amount\": 123, \"comment\": \"string\", \"externalNote\": \"string\", \"login\": 9017800, \"preciseAmount\": 1.23, \"type\": \"DEPOSIT\"}"
```

```
{"bonusHistoryId":2825}
```

## 2.12. List Trader Groups

### HTTP Method:

GET

### URL:

/webserv/tradergroups

### Inputs:

none

### Output:

TraderGroupListTO

### Expected HTTP status code:

200

### Example:

```
curl -X GET "https://HOST:PORT/v2/webserv/tradergroups?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fad" -H "accept: application/json"
```

```
{
  "traderGroup": [
    {
      "id": 51350,
      "name": "Disabled Group",
      "description": ""
    },
    {
      "id": 51351,
      "name": "Disabled 2",
      "description": "demo group"
    },
    {
      "id": 51352,
      "name": "EURUSD unknown",
      "description": ""
    },
    {
      "id": 51354,
      "name": "Dasha_del",
      "description": ""
    },
    {
      "id": 51355,
      "name": "Fair Stop Out",
      "description": ""
    },
    {
      "id": 51100,
      "name": "DN717iZXZePvIaCsmsi12jZb",
      "description": ""
    },
    {
      "id": 51356,
      "name": "30 stopout",
      "description": ""
    },
    {
      "id": 51101,
      "name": "Cj0D0QgqwUpfzXqt02IpGkS2",
      "description": ""
    },
    {
      "id": 51357,
      "name": "234234",
      "description": ""
    },
    {
      "id": 51102,
      "name": "KgWzWUiWPausD7DILrNxCuCP",
      "description": ""
    },
    {
      "id": 51358,
      "name": "Test13",
      "description": ""
    },
    {
      "id": 51366,
      "name": "DS_stopout",
      "description": ""
    },
    {
      "id": 51367,
      "name": "arta_test",
      "description": ""
    },
    {
      "id": 1000,
      "name": "Default",
      "description": "demo group"
    },
    {
      "id": 51368,
      "name": "arta_test2",
      "description": ""
    },
    {
      "id": 51369,
      "name": "OnlyAUDCAD",
      "description": ""
    },
    {
      "id": 51050,
      "name": "TEST",
      "description": "Testing Group Symbols"
    },
    {
      "id": 51370,
      "name": "load-cat",
      "description": ""
    },
    {
      "id": 51371,
      "name": "swap test",
      "description": ""
    },
    {
      "id": 51250,
      "name": "for_stopout_test",
      "description": ""
    },
    {
      "id": 51000,
      "name": "Test1",
      "description": ""
    },
    {
      "id": 51001,
      "name": "Test12",
      "description": ""
    }
  ]
}
```

## 2.13. List of open positions

### HTTP Method:

GET

### URL:

/webserv/openPositions

### Inputs:

none or login

### Output:

Comma-separated values

### Expected HTTP status code:

200

### Example:

```
curl -X GET "https://HOST:PORT/v2/webserv/openPositions?token=1dd4ef40-c5b3-61c0-0689-b1b40c97fadc"
```

```
Login,positionId,openTimestamp,entryPrice,direction,volume,symbol,commission,swap,book  
Type,stake,spreadBetting,usedMargin  
1010,711,2020-09-  
03T11:51:07.902,1.1,BUY,100000.00,EURUSD,0.00,0.00,BOOK_B,0.00,false,1000.00  
1011,734,2020-09-  
02T09:41:22.361,1.2,SELL,20000.00,EURGBP,0.00,0.00,BOOK_A,0.00,false,1000.00
```

or with filtering by the login:

```
curl -X GET "https://HOST:PORT/v2/webserv/openPositions?login=1000&token=1dd4ef40-c5b3-61c0-0689-b1b40c97fadc"
```

```
Login,positionId,openTimestamp,entryPrice,direction,volume,symbol,commission,swap,book  
Type,stake,spreadBetting,usedMargin  
1010,711,2020-09-  
03T11:51:07.902,1.1,BUY,100000.00,EURUSD,0.00,0.00,BOOK_B,0.00,false,1000.00
```



## 2.14. List of closed positions

### HTTP Method:

GET

### URL:

/webserv/closedPositions

### Inputs:

- from timestamp (inclusive) formatted as `yyyy-MM-ddTHH:mm:ss.SSS` in UTC (example: 2018-01-01T12:12:12.000)
- to timestamp (exclusive) formatted as `yyyy-MM-ddTHH:mm:ss.SSS` in UTC (example: 2018-01-01T12:12:12.000)
- login (optional)

### Notes:

Period cannot be more than 2 (two) days

### Output:

Comma-separated values

### Expected HTTP status code:

200

### Example:

```
curl -X GET "https://HOST:PORT/v2/webserv/closedPositions?from=2020-09-03T11:55:32.773&to=2020-09-03T11:58:08.355&token=04d95575-c9af-42fb-a72e-2f0ce93f01d4"
```

```
login,positionId,openTimestamp,closeTimestamp,closePrice,direction,volume,symbol,commission,swap,pnl,depositConversionRate,usdConversionRate,bookType,spread,spreadBetting
1012,716,2020-09-03T11:58:07.580,2020-09-03T11:58:07.626,1.20000,SELL,200000.00,EURJPY,0.00,0.00,0.00,0.8333333333333334,1.1,BOOK_B,0.00,false
1033,1244,2020-09-03T13:08:11.607,2020-09-03T14:22:04.889,1.25400,BUY,87000.00,EURJPY,0.00,0.00,0.00,0.82,1.2,BOOK_A,0.00,false
```

or with filtering by the login:

```
curl -X GET "https://HOST:PORT/v2/webserv/closedPositions?from=2020-09-03T11:55:32.773&to=2020-09-03T11:58:08.355&login=2033&token=04d95575-c9af-42fb-a72e-2f0ce93f01d4"
```

```
login,positionId,openTimestamp,closeTimestamp,closePrice,direction,volume,symbol,commi  
ssion,swap,pnl,depositConversionRate,usdConversionRate,bookType,spreadBetting  
2033,1231,2020-09-03T11:58:07.580,2020-09-  
03T11:58:07.626,1.20000,SELL,200000.00,EURJPY,0.00,0.00,0.00,0.8333333333333334,1.1,BO  
OK_B,0.00,false  
2033,1244,2020-09-03T13:08:11.607,2020-09-  
03T14:22:04.889,1.25400,BUY,87000.00,EURJPY,0.00,0.00,0.00,0.82,1.2,BOOK_A,0.00,false
```

## 2.15. Create Withdrawal Request

**HTTP Method:**

POST

**URL:**

/webserv/traders/{login}/withdrawRequest

**Inputs:**

- login of the trader
- preciseAmount as decimal amount
- comment (optional)

**Output:**

none

**Expected HTTP status code:**

200

**Example:**

```
curl -X POST
"https://HOST:PORT/v2/webserv/traders/1007/withdrawRequest?token=04d95575-c9af-42fb-
a72e-2f0ce93f01d4" -H "accept: application/json" -H "Content-Type: application/json"
-d "{ \"comment\": \"test\", \"login\": 1007, \"preciseAmount\": 0.31234567}"
```

### 3. Operations with cTID

We expose the following RESTful web services for operations with cTID (users) (see the table below).

The content type for requests and responses is text/json.

All URLs are relative to <https://HOST:PORT/cid>

The web services API is authenticated under the same Manager credentials which are used to login to the cBroker application. Usage of the API requires inclusion of an authentication token with each request by appending `?token={token}` to the end of the request URL. Token doesn't have expiration time and could be generated according to section Generate Manager's token.

## 3.1. Create cTID

### HTTP Method:

POST

### URL:

/cid/ctid/create

### Inputs:

- "brokerName": "string"
- "email": "string"
- "preferredLanguage": "string"

### Output:

- "userId": "integer"
- "nickname": "string"
- "email": "string"
- "utcCreateTimestamp": "integer"
- "status": "string"

### Expected HTTP status code:

200

### Example:

```
curl -X POST --header 'Content-Type: application/json' --header 'Accept: application/json' -d '{ \
  "brokerName": "BESTBROKER", \
  "email": "president%40bestbroker.com", \
  "preferredLanguage": "EN" \
}' 'https://HOST:PORT/cid/ctid/create?token=04d95575-c9af-42fba72e-2f0ce93f01d4'
```

```
{
  "userId": 10333,
  "nickname": "ctid10333",
  "email": "president@bestbroker.com",
  "utcCreateTimestamp": 1598938735839,
  "preferredLanguage": "en",
  "status": "CTID_NEW"
}
```

## 3.2. Link account to cTID

### HTTP Method:

POST

### URL:

/cid/ctid/link

### Inputs:

- "traderLogin" : "integer"
- "traderPasswordHash" : "string"
- "userId" : "integer"
- "brokerName" : "string"
- "environmentName" : "string"

### Output:

none

### Expected HTTP status code:

200

### Example:

```
curl -X POST --header 'Content-Type: application/json' --header 'Accept: application/json' -d '{ \
  "environmentName": "demo", \
  "brokerName": "BESTBROKER", \
  "kycNotPassed": true, \
  "traderLogin": 9017800, \
  "traderPasswordHash": "0f94e246908667af85916300c57f74b6", \
  "userId": 10333 \
}' 'https://HOST:PORT/cid/ctid/link?token=04d95575-c9af-42fba72e-2f0ce93f01d4'
```

## 3.3. Change Whitelabel of the Trading account

### HTTP Method:

POST

### URL:

/cid/ctid/changeCtidTraderAccount

### Inputs:

- "brokerName": "string"
- "traderLogin": "integer"
- "environmentName": "string"

### Output:

none

### Expected HTTP status code:

200

### Example:

```
curl -X POST --header 'Content-Type: application/json' --header 'Accept: application/json' -d '{ \
  "environmentName": "demo", \
  "brokerName": "BESTBROKER", \
  "traderLogin": 9017800, \
}' 'https://HOST:PORT/cid/ctid/changeCtidTraderAccount?token=04d95575-c9af-42fba72e-2f0ce93f01d4'
```

## 4. Error codes

Please find below the list of possible error codes you can encounter in a response body:

Error Code	Error Description
UNKNOWN_ERROR	In case of server error.
INVALID_REQUEST	In case of client invalid request (request validation failure).
NOT_ENOUGH_MONEY	In case the trader doesn't have enough money or has some open positions with unrealized profit/loss.
CHANGE_BALANCE_BAD_AMOUNT	In case the change balance amount is not valid. (e.g. amount < 0)
TRADER_NOT_FOUND	In case trader is not found.
TRADER_GROUP_NOT_FOUND	In case trader group not found on trader create/update operation.
WRONG_PASSWORD	In case trader's password is wrong on "Check Trader's Password" operation.
NOT_ENOUGH_RIGHTS	In case the client doesn't have enough rights to perform the operation.
REQUEST_FREQUENCY_EXCEEDED	In case client sends more requests per hour than allowed. GET requests are not limited.

If you have any questions please email us at [support@spotware.com](mailto:support@spotware.com)