- **All requests should be done only on the hero's turn (otherwise will get an error).**

- **For production address please contact support, must be subscribed to the API service!**

- **If a postflop spot was used, make sure to send the “finish” field when that hand is over! Otherwise your account may automatically get temporarily restricted!**

- **Make sure to replace variables $username and $password with YOUR username and password. Pay attention, it appears in the JSON body <u>and</u> URL.**

- **Default 3-day trial license that can be acquired after downloading Checkmath application has API access but with limited amount of requests!**

- **In the Dev/Test environment only a limited number of flops are available: AsTd7s, all turns and rivers.**
  In Production environment all flops/turns/rivers are available

- **Credentials for dev/test environment:**
  - **Username: apitest**
  - **Password: apitest-password**

- **After subscribing to the API service, production credentials will be sent to your email address.**

- **No SSL certificate verification on the test server**

- Not supported lines:
  - For 6max: Limp on any position except SB, multiple calls vs. raise
  - For Spins 3way: Button limps

PREFLOP

POST
https://dev.cmpkr.com:443/api/pokeron/**$username**/api/multiway

--Headers--
Accept: application/json
Content-Type: application/json

--Body--
JSON

"username" - String

"password" - String

"**gameType"** - String, enum {"cash", "spins", "sng"} - Must have proper license. “spins” is for 3way Spin&Go stage, sng for husng and hu in Spins.

"position" - String, Position at the table, enum {"EP", "MP", "CO", "BTN", "SB", "BB"}

"myCards" - String, hero's cards (e.g. AsKd, 2cAc etc...)

"stackSize" - String, effective stack size for the start of the hand (before any bet made), should stay the same as bets being conducted, will also stay the same for transition to Postflop! For HU-cash (and only heads-up cash!), add "-HU" to a stacksize (e.g. 105bb-HU)

"state" - String. Should start with "r:0", representing Preflop action up until the hero's turn! If the hero is on the button in a 6max game and everyone has folded until btn, the state would be "r:0:f:f:f". Another example, hero on the BB, action before hero can be "r:0:b22:f:b75:b200:f". f = fold, c = call, b75 = bet 75. For cash games, all bets must be for 5/10 blinds structure (**BB=10**). For Spins/SNG all bets must be for 50/100 blinds structure(**BB=100**)! Limp is a call! (c) For 6max cash state should always include 6 players, so if the table is not full (e.g. 3handed), add appropriate amount of folds at the start of the state (e.g. r:0:f:f:f for 3handed game)

"protocol" - Integer. Must equal 2 in current implementation!

Preflop Examples:

====6max Cash====

{"username": $username, "password": $password, "position": "SB", "myCards": "AsAd", "state": "r:0:b30:f:f:f", "stackSize": "105bb", "gameType": "cash", "protocol": 2}

{"username": $username, "password": $password, "position": "EP", "myCards": "AsAd", "state": "r:0", "stackSize": "142bb", "gameType": "cash", "protocol": 2}

{"username": $username, "password": $password, "position": "SB", "myCards": "AsAd", "state": "r:0:f:f:f:f", "stackSize": "142bb", "gameType": "cash", "protocol": 2}

{"username": $username, "password": $password, "position": "EP", "myCards": "AsAd", "state": "r:0:b22:f:f:c:b111:f", "stackSize": "142bb", "gameType": "cash", "protocol": 2}

{"username": $username, "password": $password, "position": "EP", "myCards": "AsAd", "state": "r:0:b24:f:f:c:b111:f", "stackSize": "142bb", "gameType": "cash", "protocol": 2}

====HU Cash====

{"username": $username, "password": $password, "position": "SB", "myCards": "AsAd", "state": "r:0", "stackSize": "105bb-HU", "gameType": "cash", "protocol": 2}

{"username": $username, "password": $password, "position": "SB", "myCards": "AsAd", "state": "r:0:c:b25", "stackSize": "24bb-HU", "gameType": "cash", "protocol": 2}

{"username": $username, "password": $password, "position": "BB", "myCards": "AsAd", "state": "r:0:b23:b100:b235", "stackSize": "100bb-HU", "gameType": "cash", "protocol": 2}

====Spins 3-way====

{"username": $username, "password": $password, "position": "BTN", "myCards": "AsAd", "state": "r:0", "stackSize": "11.5bb", "gameType": "spins", "protocol": 2}

{"username": $username, "password": $password, "position": "BB", "myCards": "AsAd", "state": "r:0:b200:f", "stackSize": "11.5bb", "gameType": "spins", "protocol": 2}

{"username": $username, "password": $password, "position": "BTN", "myCards": "AsAd", "state": "r:0:b200:b428:b2444", "stackSize": "24.3bb", "gameType": "spins", "protocol": 2}

====HU SNG(Also HU for Spins)====

{"username": $username, "password": $password, "position": "SB", "myCards": "AsAd", "state": "r:0", "stackSize": "11.5bb", "gameType": "sng", "protocol": 2}

{"username": $username, "password": $password, "position": "SB", "myCards": "AsAd", "state": "r:0:c:b322", "stackSize": "11.5bb", "gameType": "sng", "protocol": 2}

Response:

JSON

“success” - Integer. 1 = ok

“target_state” - String. Real state in the sim that your request was mapped to (should be closest in terms of bet sizings)

“effectiveStack” - Integer. Effective stack in the sim (starting from preflop, before posting blinds) - should be the closest to requested stack size.

“options” - Array of Nodes. Each one is a strategy node in the requested state. Each node has following fields:

-**”full”** - String. Real state for this strategy option.

-**”last”** - String. Last action in full state.

-**”type”** - String. Can be {END_NODE, SPLIT_NODE or empty}. END_NODE = this option leads to the end of a hand, SPLIT_NODE - this option leads to the next street. Empty is a regular node that has children in the tree for the current street.

-**”normalizedBet”** - Integer. If this node is a betting (or raising) action, this integer represents bet size on the current street. Will be 0 if this node is not betting option (call, check or fold)

“strategyQuota” - Integer. Remaining amount of requests available for your license.

POSTFLOP

POST
https://dev.cmpkr.com:443/api/pokeron/**$username**/api

--Headers--
Accept: application/json
Content-Type: application/json

--Body--
JSON

Same as Preflop with addition of 2 new fields:

"defender" - String. Split node that leads to a postflop! This is a "state" that you've been sending preflop, just its split node, where no more actions can be made preflop. Don't send "r:0" here, only preflop actions. Must NOT be a multiway node (>2 active players). Stays the same for the course of the hand, can be treated as "spot"

"flopCards" - String. Only flop cards (no turns or rivers!). E.g. **AsTd7s**

"state" - String. Like preflop, all postflop actions. Turns and Rivers are written here as well in a format of postflop action. E.g. "r:0:b20:c:Qh:c:b50:c:7d:c"

"finish" - Integer, **Optional**. When the postflop hand is over, set finish to 1 ("finish" : 1). This is a technical request to make your next response faster and avoid your account being restricted for too many “unfinished” postflop spots! This request doesn't count as "request" and the quota should stay the same.

PostflopExamples:

Flops available: AsTd7s

====6max Cash====

```
{"gameType":"cash","myCards":"JsTc","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"152bb","position":"BTN","state":"r:0:c","username":$username,"defender":"f:f:f:b22:f:c"}
```

```
{"gameType":"cash","myCards":"JsTs","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"45bb","position":"SB","state":"r:0","username":$username,"defender":"f:b22:f:f:b80:f:c"}
```

```
{"gameType":"cash","myCards":"JsTs","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"105bb","position":"BB","state":"r:0","username":$username,"defender":"f:b22:f:c:f:b122:c:f"}
```

====HU Cash====

```
{"gameType":"cash","myCards":"JsTs","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"105bb-HU","position":"SB","state":"r:0:c","username":$username,"defender":"b25:c"}
```

```
{"gameType":"cash","myCards":"JsTs","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"105bb-HU","position":"BB","state":"r:0","username":$username,"defender":"b25:b100:c"}
```

====Spins 3-way====

```
{"gameType":"spins","myCards":"AdAc","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"15.3bb","position":"BTN","state":"r:0:c","username":$username,"defender":"b200:f:c"}
```

```
{"gameType":"spins","myCards":"JsTs","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"15.3bb","position":"SB","state":"r:0","username":$username,"defender":"f:c:c"}
```

====HU SNG====

```
{"gameType":"sng","myCards":"JsTs","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"13bb","position":"SB","state":"r:0:c","username":$username,"defender":"c:c"}
```

```
{"gameType":"sng","myCards":"JsTs","flopCards":"AsTd7s","password":$password,"protocol":2,"stackSize":"13bb","position":"BB","state":"r:0","username":$username,"defender":"c:b300:c"}
```

Response:

JSON

Most of the fields are identical to Preflop, but there are a few additional fields.

“freqs” - Array of Floats. Represents average strategy for the entire hero's range for node i in options! This is NOT the hero's hand strategy!!

“strategy” - Array of Floats. Represents hero strategy for node i in options!

“options” - same as Preflop, but these have additional fields:

-**”ev”** - Float. EV of this option

-**”pot”** - Integer array [3] representing pot for this option where array[0] = OOP invested money in the pot, array[1] = IP invested money in the pot, array[2] = Dead money (invested on previous streets)

“hero_combo_eq” - Float. Hero's hand Equity.

“current_pot” - Integer array [3] representing pot for this state where array[0] = OOP invested money in the pot, array[1] = IP invested money in the pot, array[2] = Dead money (invested by all players Preflop)

ERROR codes:

1 - wrong username/password
7 - no active license for the requested spot
9 - license expired
15 - out of requests