

# Potato Sniper Bot

## Contact Info

Telegram Chat: <https://t.me/PotatoSniperBot>

Dev-Telegram: @poormanmentality

Dev-Discord: Anonymous101#5251

GitHub: [https://github.com/PotatoSniper/Potato\\_Bot](https://github.com/PotatoSniper/Potato_Bot)

YouTube Demo: <https://www.youtube.com/watch?v=3t74O6JmtNI>

## Disclaimer

Mr. Potato holds no responsible or reliable for any error or losses, if you persist on using this tool then you agree to hold full responsibility and liability for all consequences. If you are uncomfortable or disagree with any of the terms of this policy, please discontinue use of Potato Sniper Bot.

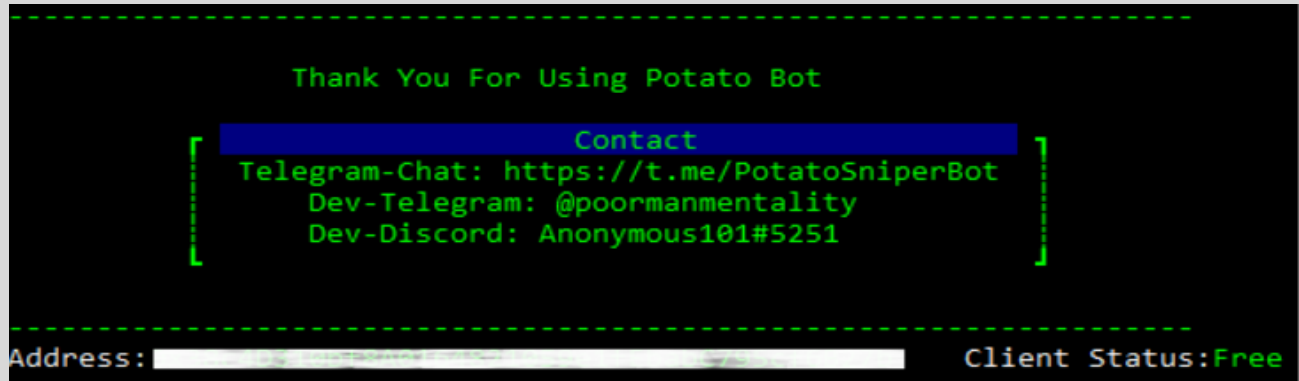
## Warning

⚠ Only use test wallet and never use main wallet ⚠

## Table of Contents

|  |    |
|--|----|
| Potato Sniper Bot Introduction .....   | 3  |
| <i>Supported Chains</i> .....  | 3  |
| <i>Functionality</i> .....   | 3  |
| Setup .....  | 4  |
| <i>Linux</i> .....   | 4  |
| <i>Window</i> .....  | 4  |
| <i>Mac</i> .....   | 4  |
| ContractDeployer .....   | 5  |
| <i>Deploy Contract</i> .....   | 5  |
| <i>Withdraw</i> .....  | 5  |
| PotatoBot .....  | 6  |
| <i>Mempool Mode</i> .....  | 6  |
| <i>Telegram Mode</i> .....   | 7  |
| <i>Chat Link Mode [1]</i> .....  | 8  |
| <i>Chat ID Mode [2]</i> .....  | 8  |
| <i>Tradable Mode</i> .....   | 9  |
| <i>Set Fee</i> .....   | 9  |
| Config.json Setup.....   | 9  |
| <i>api ID &amp; Hash</i> .....   | 9  |
| <i>url</i> .....   | 9  |
| <i>block_chain, router, factory, governance_token_addr</i> .....                                   | 9  |
| <i>Intermediate_token</i> .....  | 10 |
| <i>my_contract, hp_contract</i> .....  | 10 |
| <i>contract_owner_addr, contract_owner_key</i> .....   | 10 |
| <i>sell_account_addr, sell_account_key</i> .....   | 10 |
| <i>auto_sell config</i> .....  | 11 |
| <i>Telegram_Honeypot, Telegram_Download_File</i> .....   | 11 |
| <i>target_buytax, target_selltax</i> .....   | 11 |
| <i>buy_amount, buy_count, test_buy_amount, buy_delay_timer, gas, gaslimit, sell_slippage</i> ..... | 11 |
| <i>target_owner</i> .....  | 11 |

## Potato Sniper Bot Introduction



### Supported Chains

#### **Binance Smart-Chain**

- PancakeSwap
- ApeSwap
- BabySwap
- KnightSwap
- MDEXSwap

#### **Ethereum**

- UniSwapV2
- SushiSwap
- ShibaSwap

#### **Chronos**

- MeerkatSwap

#### **Polygon**

- QuickSwap

#### **Fantom**

- SpookySwap

### Functionality

#### **Mempool Mode**

- ✓ Buy Transaction conducted on smart contract.
- ✓ Detects blacklist and high tax (safe from getting rekt)
- ✓ Multiple buys within one transaction
- ✓ Auto sell / Manual sell

#### **Telegram Mode**

- ✓ Buy Transaction conducted on smart contract.
- ✓ Scrape Telegram and reconstruct to contract address
- ✓ Convert LP pair address to contract address
- ✓ Download txt file > read content > construct contract address
- ✓ Filter greeting messages
- ✓ Honeypot checker for buy/sell tax

#### **Tradable Mode**

- ✓ Buy Transaction conducted on smart contract.
- ✓ Buy when target contract is tradable/sellable
- ✓ Honeypot checker for buy/sell tax

## Setup

### Linux

△ Replace *filename* with actual name of the file △

△ Replace *NameOfDirectory* with actual name of the folder △

#### Download & Edit permission

1. cd home/
2. gitclone [https://github.com/PotatoSniper/Potato\\_Bot](https://github.com/PotatoSniper/Potato_Bot)
3. cd Potato\_Bot/
4. unzip Linux.zip
5. cd Linux/
6. chmod u+x ContractDeployer
7. chmod u+x PotatoBot

#### Modify config.json file

1. nano config.json                      To save: [CTRL+X > Y > ENTER]

#### Run

1. *./filename*

#### Clean-Up (Optional)

1. mv *filename* /home/*NameOfDirectory*/
2. cd /home/
3. rm -r Potato\_Bot/

### Window

1. Download files at [https://github.com/PotatoSniper/Potato\\_Bot](https://github.com/PotatoSniper/Potato_Bot)
2. Modify config.json file
3. Run

### Mac

1. Not sure, try Linux setup and change **Linux** to **Mac**

## ContractDeployer

```
-----  
                          Thank You For Using Potato Bot  
                          [ Contact ]  
                          Telegram-Chat: https://t.me/PotatoSniperBot  
                          Dev-Telegram: @poormanmentality  
                          Dev-Discord: Anonymous101#5251  
                          -----  
Address:  Client Status:Free  
!!!!!!!!!!!!!! Please Select Mode Of Operation !!!!!!!!!!!!!!!  
Deploy Buy-Contract      [1]:  
Deploy HoneyPot-Contract [2]:  
Withdraw                 [3]:  
Exit                     [x]:
```

### Deploy Contract

- **url** parameter in config.json file will determine which chain the contract is deployed.
- All contracts will utilize **contract\_owner\_addr** and **contract\_owner\_key** for deployment, make sure you have enough to cover gas fee.  $gas\ fee = \frac{gas}{10^9} \times gaslimit$

### Example.

Config.json file, "gas": "5" and "gaslimit": "2000000". Maximum Gas cost would be 0.01 BNB

$$gas\ cost\ in\ BNB = \frac{5}{10^9} \times 2000000 = 0.01\ BNB$$

### Withdraw

⚠ Only contract owner (**contract\_owner\_addr**) can execute this. ⚠

- Withdraw balance from buy contract.
- Make sure contract owner wallet have enough to cover gas fee.
- Restart Bot to load updated config.json file data.

## PotatoBot

⚠ Ensure **Contract Owner Wallet** has enough to cover gas fee ⚠

⚠ Ensure **Buy Contract** has enough to cover  $(buy\_amount * buy\_count + test\_buy\_amount)$  ⚠

### Mempool Mode

⚠ **PinkSale/Dxsale/All** Options Only for BSC ⚠

⚠ **Will Not Work If LP Is Already Added** ⚠

```
-----
Thank You For Using Potato Bot

[ Contact ]
Telegram-Chat: https://t.me/PotatoSniperBot
Dev-Telegram: @poormanmentality
Dev-Discord: Anonymous101#5251
]

-----
Address: [ ] Client Status:Free

!!!!!!!!!!!!!! Please Select Sniper Mode !!!!!!!!!!!!!!!
Mempool Snipe [1]:
Telegram Snipe [2]:
Tradable Snipe [3]:
1

!!!!!!!!!!!!!! Please Select A Launching Platform !!!!!!!!!!!!!!!
Mempool (NON-BNB) [0]:
Mempool (BNB) [1]:
PinkSale [2]:
Dxsale [3]:
Don't Fucking Know (All) [x]:
```

- ✓ Scan Mempool for Add\_LP actions
- ✓ Auto determine buy gas if "**buy\_delay\_timer**": null, to ensure buying in block 0.
- ✓ If Add\_LP is detected and added successfully but buy transaction failed, the bot will assume trading is disabled and automatically switch to monitor owner-contract interaction for enable trade action. If **contract\_owner** is empty in confirmation step, you must manually add it in config.json file for this to work.
- ✓ Buy from smart contract to prevent anti bot mechanism (prevent losing money/get rekt)

△ Fill in *"intermediate\_token"* parameter in config.json file, if target token is pairing with non-governance token. Otherwise leave it as *null*. △

### Example.

Cake token paired with BUSD. Fill in *"intermediate\_token"* parameter in config.json file with BUSD address.

Pcv2 **Cake/BUSD** LP Holdings:  
8,125,967.86 BUSD (\$8,125,968) | [Chart](#) | [Holders](#)

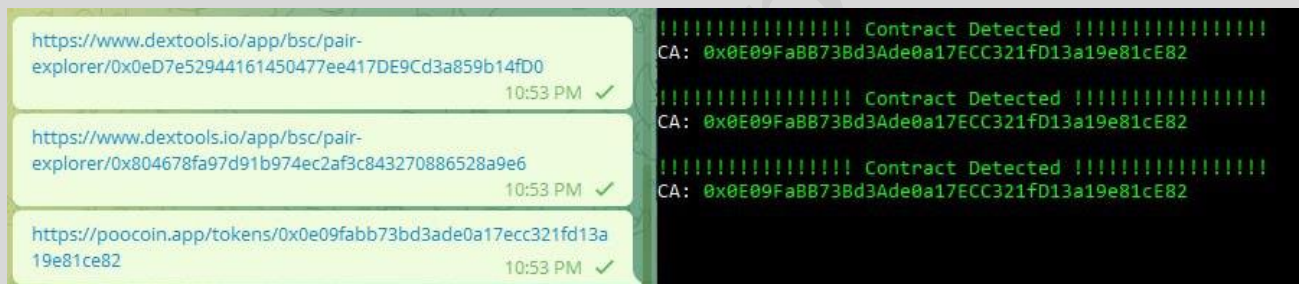
| Block Chain         | Governance Token |
|---------------------|------------------|
| Binance Smart-Chain | BNB              |
| Ethereum            | ETH              |
| Chronos             | CRO              |
| Polygon             | MATIC            |
| Fantom              | FTM              |

## Telegram Mode

△ Refer to *Config.json Setup* to obtain apiID and apiHash △

△ Require telegram login information for first time use △

△ Telegram scrapper is slower compare to other telegram bot due to more complex algorithm. Increase gas to compensate for this △



- ✓ Scan Telegram Public/Private Groups and Channels
- ✓ Convert letter number to decimal number such as *oNe* > *1*, *Two* > *2*, *ThReE* > *3*
- ✓ Convert number emoji to number
- ✓ txt file > download > read
- ✓ Support basic arithmetic calculation
- ✓ Support blurred text in telegram chat
- ✓ Convert pair address to token address
- ✓ Remove garbage content and attempt to reconstruct to valid address
- ✓ Ignore greeting messages.
- ✓ Check Honeypot and Buy/Sell tax

### Chat Link Mode [1]

```

!!!!!!!!!!!!!!!!!!!! Please Select Telegram Chat Mode [Link/ID] !!!!!!!!!!!!!!!!!!!!!
Telegram Chat Link [1]:
Telegram Chat ID [2]:
1
Please Enter Telegram Chat Link [https://t.me/Example_Chat]:
https://t.me/PotatoSniperBot

!!!!!!!!!!!!!!!!!!!! Running !!!!!!!!!!!!!!!!!!!!!

```

This mode is only used in public group/channels where the link is provided. Simply copy and paste link into bot terminal as shown in diagram above. Right click on the terminal to paste.

### Chat ID Mode [2]

[illegible]

⚠ DO NOT PRESS CTRL+C TO COPY ⚠

This mode is only used in private group/channels where chat link is not provided. Simply copy and paste desired Chat ID into bot terminal as shown in diagram above. To copy, simply highlight the text and right click on the terminal to paste.



## Tradable Mode

### ⚠ INCREASE GAS TO BUY FASTER ⚠

This mode will buy target token as soon as target contract is tradable and sellable and buy/sell tax is less than or equal to “*target\_buytax*” and “*target\_selltax*” parameter in config.json file.

## Set Fee

Enter desire buy amount and gas price, expected format *buy\_amount/gas\_price* or enter *0* to use default setting in config.json file.

### Example.

Buy 0.01 BNB with gas price of 5 gwei.  
Simply enter 0.01/5 into the terminal

```
Enter Fee [Buy/Gas] or [0=Default]:  
0.01/5
```

## Config.json Setup

### api ID & Hash

Method 1 (PC).

1. Obtain from <https://my.telegram.org/auth>

Method 2 (Android).

1. Download Opera Mini from Play Store
2. In Opera Mini Browser enter <https://my.telegram.org/auth>

### url

Endpoint/Node url, rate limited Endpoint/Node not supported.

### block\_chain, router, factory, governance\_token\_addr

**Refer to Address.txt, copy and paste from there**

### Example.

```
"block_chain": "BSC",  
"router": "0x10ED43C718714eb63d5aA57B78B54704E256024E",  
"factory": "0xcA143Ce32Fe78f1f7019d7d551a6402fC5350c73",  
"governance_token_addr": "0xbb4CdB9CBd36B01bD1cBaEBF2De08d9173bc095c",  
"intermediate_token": null,
```

## Intermediate\_token

⚠ Fill in **"intermediate\_token"** parameter in config.json file, if target token is pairing with non-governance token. Otherwise leave it as **null**. ⚠

### Example.

Cake token paired with BUSD. Fill in **"intermediate\_token"** parameter in config.json file with BUSD address.

```
Pc v2 Cake/BUSD LP Holdings:  
8,125,967.86 BUSD ($8,125,968) | Chart | Holders
```

| Block Chain         | Governance Token |
|---------------------|------------------|
| Binance Smart-Chain | BNB              |
| Ethereum            | ETH              |
| Chronos             | CRO              |
| Polygon             | MATIC            |
| Fantom              | FTM              |

## my\_contract, hp\_contract

⚠ Ensure **Buy Contract** has enough to cover ( $buy\_amount * buy\_count + test\_buy\_amount$ ) ⚠

**my\_contract** = Buy Contract

**hp\_contract** = Honeypot Contract

Obtain from ContractDeployer, Honeypot contract can be shared, if you can find someone who had already deployed the Honeypot Contract, you can use their contract instead of deploying your own.

## contract\_owner\_addr, contract\_owner\_key

⚠ Ensure **Contract Owner Wallet** has enough to cover gas fee ⚠

### Warning

⚠ Only use test wallet and never use main wallet ⚠

**contract\_owner\_addr** enter desire wallet address to deploy contracts.

**contract\_owner\_key** 64-character key

## sell\_account\_addr, sell\_account\_key

⚠ Ensure **Sell Wallet** has enough to cover gas fee ⚠

### Warning

⚠ Only use test wallet and never use main wallet ⚠

**sell\_account\_addr** enter desire wallet address for selling.

**sell\_account\_key** 64-character key

### auto\_sell config

|                             |  |
|-----------------------------|--|
| <b>auto_sell</b>            | Only accept <i>true</i> or <i>false</i> .  |
| <b>auto_sell_interval</b>   | Sell interval at every x pure profit. Pure Profit = Current Balance - Investment Amount. |
| <b>auto_sell_amount</b>     | Sell x % at every sell interval.   |
| <b>auto_sell_all_target</b> | Sell all balance at x pure profit.   |

### Telegram\_Honeypot, Telegram\_Download\_File

|                               |   |
|-------------------------------|---|
| <b>Telegram_Honeypot</b>      | Perform Honeypot check in <i>Telegram</i> mode. Only accept <i>true</i> or <i>false</i> .   |
| <b>Telegram_Download_File</b> | Enable download txt file in <i>Telegram</i> mode. Only accept <i>true</i> or <i>false</i> . |

### target\_buytax, target\_selltax

|                       |   |
|-----------------------|---|
| <b>target_buytax</b>  | Only buy if buy tax is $\leq x$ , only applies for <i>Telegram/Tradable</i> mode  |
| <b>target_selltax</b> | Only buy if sell Tax is $\leq x$ , only applies for <i>Telegram/Tradable</i> mode |

### buy\_amount, buy\_count, test\_buy\_amount, buy\_delay\_timer, gas, gaslimit, sell\_slippage

|                        |  |
|------------------------|--|
| <b>buy_amount</b>      | Amount To Buy, Actual Buy                                  |
| <b>buy_count</b>       | How many times to buy within one transaction               |
| <b>test_buy_amount</b> | Amount To Buy, Sample Buy                                  |
| <b>buy_delay_timer</b> | Buy delay in seconds. Leave it as <i>null</i> if not used. |
| <b>gas</b>             | Gas price in gwei  |
| <b>gaslimit</b>        | Maximum allowed gas limit per transaction                  |
| <b>sell_slippage</b>   | Maximum acceptable difference > expected vs actual.        |

### target\_owner

|                     |  |
|---------------------|--|
| <b>target_owner</b> | Leave It as is, bot will automatically fetch this. If bot failed to fetch owner address, then you need to manually enter it here. Only applies to <i>Mempool</i> Mode. |
|---------------------|--|