BlockChain Life

Mohamed LEGHERABA Dinesh Bisesser
Philip Blankendal Michail Vrachasotakis Ka-Wing Man

February 1, 2018

1 Introduction

This work is an extension of the Cloudomate[1] project for automatically buying Virtual Private Servers. Its aim is to create a network of self-autonomous replicating entities that buy servers using cryptocurrencies and use their bandwidth to earn trust in order to trade it in the Tribler decentralized market.The current version of the system updates the Cloudomate utility with Virtual Private Network support for a variety of providers, captcha solving functionality and a wallet for the Ethereum cryptocurrency.

Our goal was to not only re-enforce existing functionalities, but also add (extra) functionalities in order to realize the Block-chain Life's main purpose. We have added several new functionalities to this project based on certain perceptions on what was lacking in previous Block-chain Life project in order to strive and make this concept truly successful. These new implemented features ensure the agent's p2p traffic and vps buying functionalities are equipt to not only handle but also surpass the modern blockades as self-sustaining bots in todays web-based traffic, making a large and exiting step into the direction of becoming truly autonomous.

2 Feature selection and Design Overview

Features implemented

- Captcha-Solver
- Re-Captcha Solver
- Ethereum Wallet
- Torguard VPN
- Vpn.ac VPN
- Mullvad VPN

[1] https://github.com/Tribler/cloudomate

3 Captcha Solver

For various VPNs and VPSs an autonomous way of captcha solving is needed for instance to create an account. In order to overcome this obstacle it was decided to implement a captcha solver.

3.1 Solver Choice

Initially, various command line OCR tools were tested for simple captchas that would enable us to avoid using a system like Amazon Mechanical Turk or Anti-Captcha. Some of the tools used were Tesseract, Gocr, Ocrad and Tesseract in Python with captcha image preprocessing by OpenCV. The results were disappointing and we had to resort to a system that employs other humans to solve it for our bot. This approach has a margin for human error and takes at least five seconds but seems to be enough for most systems. Amazon Mechanical Turk was rejected because it involves a more lengthy account generation process and it is not captcha-centered unlike Anti-Captcha. The Anti-Captcha API was not only chosen for their specialization in all main CAPTCHA types but also because the fact that they have been operational for over 10 years and have thereby proven their services to be resilient against the evolvement of anti-bot google captchas during the past years. Next to the afore mentioned advantages they are also cheap (1/1000 per CAPTCHA); and not to mention the fact that they provide means of buying their services trough BITCOIN, ETHEREUM as well as LITECOIN.

3.2 Captcha Reload

Solving CAPTCHAS typically costs 1/1000 US DOLLARS per CAPTCHA.
To ensure that bots always have sufficient balance on their Anti-Captcha account, we have provided a script to automate the adding of funds to the account. An amount in dollars is given as parameter. The payment can be paid in either Bitcoin, Litecoin or DASH. The script will return a dictionary with the amount of the desired cryptocurrency and the adddress that needs to be paid.

4 Ethereum Wallet

Bitcoin transaction fees have rised to a point where they cost a significant amount of money and other cryptocurrencies have to be used in order to avoid spending too much money. As a popular alternative to Bitcoin, Ethereum is chosen for the payments.

4.0.1 Infura scraping

To access the Ethereum network, we need a node. For Bitcoin we use a light node included in the Electrum wallet. We tested the light node of Ethereum (both Geth and Parity) but they are "experimental" version and they don't

work at all. We can install a full node but we need a tens of gigabytes so it is not suitable for a small automate script.

The other solution is to use a remote node. We tried different API (like MyEtherAPI or Etherscan API) but they ask for a lot of personal data to use them and they don't allow all ethereum network requests. The most simple way to get a node is to use infura (https://infura.io/).

Infura is a service that provide a full access to an ethereum node, free of charge and without limits. We have scraped the infura website and our automate can now have access to an ethereum node automatically.

4.0.2 Wallet creation

To create an ethereum wallet, we use the web3 and ethereum libraries. We need an access to an ethereum node and a private key. The user can give it's own private key or/and ethereum node access or we can generate an access to the infura service and generate a new private key.

4.1 Send of transactions

Now that we have a node and a private key, we can send a transactions (of course we need some ether on the account). You need to provide the address where you want to send and the amount. If you want you can precise the fees amount (in Gwei) and the number of gas but by default the script calculate it (using http://gasprice.dopedapp.com/) and 21000 as default gas number.

5 VPN

The basic component of the system is a process that uses the Cloudomate Python2 and Python3 package to buy a VPS among a list of certain providers. These providers do not allow the use of BitTorrent-like services in their servers, which in turn would not be a use-full hosting-provider for a bot that uses Tribler to upload p2p traffic and gain value-coins.
For this reason we decided to implement multiple VPN providers for the use of our agents.
The solution lies in that the VPN service provides a means for masking the torrent traffic. Buying a VPN subscription is similar to the original functionality of purchasing a VPS subscription. Therefore, the rationale is that a process now buys a not only server and logs in to it, but also buys a VPN using the same cloudomate package. In order to be versatile enough, we implemented numerous VPN providers all of them allow several cryptocurrencies as a way to pay for their services.

5.1 Mullvad VPN

Mullvad[2] is a provider that offers real anonymity as it generates an account number for a customer without asking them any personal information (Which in turn eases the scraping process greatly). In order to get this account number, the contents of a simple CAPTCHA image must be filled by the client. (For solving this CAPTCHA we use our implemented solver) This number is then used for any interaction with the provider like buying or updating a subscription and downloading the necessary files for the service. Installation is done trough OpenVPN (as it is supported on almost every type of machine). An end-to-end implementation is added to cloudomate, meaning that you can create an account, buy a subscription and install this VPN completely automatically.

5.2 VPN.ac & TorGuard

Coinpayments.net is one of the fastest growing payment providers platform (supports over 125 Coins) for paying with different cryptocurrency, including Bitcoin, Ethereum, Litecoin, DASH, DODGE, MINT and many others. We decided to implement TorGuard (due to its popularity) as well as VPN.ac (due to its stability) in order have VPN support that can bypass BITCON's transaction fees using the Ethereum wallet, and in order to provide payment support for any future currency.
For scraping these VPN's, Selenium with headless Chrome is used. This is proven to be much better than libraries such as MECHANICALSOUP and ROBOBROWSER, as those libraries cannot run JavaScript and thus websites where JavaScript was necessary could not run and therefore could not be scraped properly. With Selenium, a real browser is simulated and thus navigating through websites containing javascript with much more ease. These VPNs have a normal account (email/user-name and password) and make use of the Coinpayments gateway for their transaction.
After payment a different username (given by VPN) and password (given by VPN, but can be changed) are used for setting up and installing the VPN. Both these VPNs have an end-to-end implementation as well.

5.3 Code structure Design and Design choice

Multiple VPNs were added to CLOUDOMATE, so that the bot can host an exit node anonymously. We divided the code of each VPN provider into two parts: 1. for buying the service (contained within the cloudomate folder) and 2. one for installing and running the VPN service obtained trough purchase (each time randomly choosing a country trough which to route traffic).
We structured it this way because it is typical for VPN providers to provide services that can be used simultaneously on different machines. Therefore by choosing this design we note the fact that an once an agent has bought a VPN: it can create at least four children that can use the VPN SERVICE bought by

[2]https://www.mullvad.net/en

it parent and therefore only needs to utilize the the script to run a certain VPN ass opposed to first needing to buy one.(perhaps even share this with agents that are not necessarily related)

6 System overview

As mentioned before Agents are supplied with a either a Bitcoin or an Ethereum wallet which can be used for acquiring a VPN service for both themselves as well as their children. This part of the system needs to become more flexible and that is why other well established cryptocurrencies have to be included, which means that a different wallet needs to be implemented for each of them. To increase the range of providers, additional features like captcha solving and phone authentication need to be utilized. Figure 1 shows this whole process.

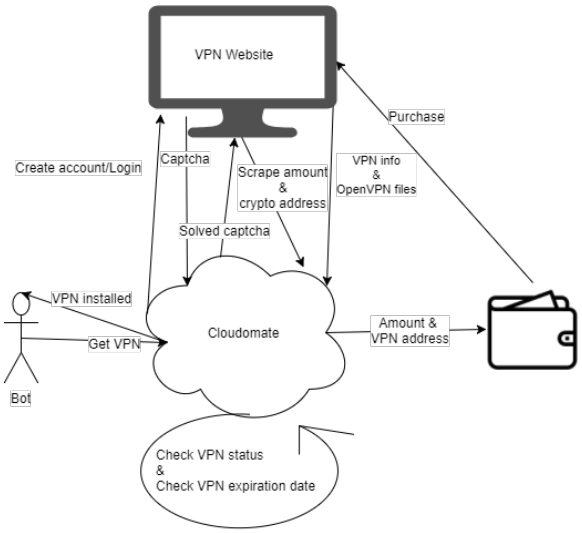

Figure 1: BlockChain Life process

7 Testing

Most of this work was tested on a server bought by Cloudomate. Additionally, a test of Plebnet's installation and functionality was performed in this VPS. After a successful setup, it was run after enabling the VPN and a check of its Tribler plugin was also successful. An issue occurs though: using the VPN changes the routing table and some services like ssh. As a result the VPS is not reachable by other machines even in its new IP although it is connected to the Internet. Despite trying various approaches discussed online like firewall and

routing rules, the issue was not resolved and may potentially lead to a problem with the communication of the Plebnet community.

8 Conclusion

Cloudomate has been enriched with quite some stunning new features. It can now buy and use VPNs with multiple types of cryptocurrencies and solve captchas. And now has a means of bypassing BITCOIN transactional fees, due to ETHEREUM support. By using scraping, crowdsourcing and blockchain techniques Tribler is step closer to dominating the world of privacy and peer-to-peer.

9 Future work

There are several features that should be included by future blockchain engineers. Due to time shortage, the litecoin wallet had to be dropped. This wallet could have been used for reloading the anti-captcha, for they also except litecoin payments. Another aspect that should be done, is looking at the distributing part of the bots and how they would share all this information (wallet, VPN, VPS). More unit tests could have also been included to better test the performance.