BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2021
http://excel.fit.vutbr.cz

Darkmarket Forensics

Daniel Dolejška*

ToRReZ

CANNAZON

Abstract
Overlay networks (like Tor or I2P) create a suitable environment for criminality to thrive on the Internet. Dark marketplaces (a.k.a. cryptomarkets) are one such example of criminal activities. They act as an intermediary in the trade of illegal goods and services. This project focuses on forensic analysis of such web services and subsequent extraction of non-trivial information about the realised orders and payments from selected marketplaces. The main goal is to pinpoint the time interval when an order has been completed on selected marketplaces and its following correlation with cryptocurrency blockchains. The implemented program provides fully automated non-stop monitoring of selected cryptomarkets. That, under certain conditions, allows detection of realised purchases, detailed product and vendor monitoring and collection of various meta-data entries. Law enforcement agencies can use acquired data as support evidence regarding the operation of selected cryptomarkets and their vendors. The acquired information can also indicate current trends in products supply and demand.

Keywords: dark web — darknet — dark market — cryptomarket — forensic analysis — crawling — scraping — cryptocurrency — bitcoin — blockchain — transaction — correlation — detection

Supplementary Material: Demonstration Video (Monitoring and Results) — Dark Web Podcast (in Czech) — Additional Public Resources (Live Metrics, Dashboards and More)

*dolejska@fit.vut.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

With the unstoppable evolution of the Internet, cryptography and privacy related software, a new dangerous place on the Internet has emerged and is gaining popularity — the dark web. Criminality has always been present in human history, but now it reaches a whole new global level. Instead of the necessity to *know people* to buy or trade with illegal goods, information or services, people can now go online and purchase such things with a few simple keystrokes and clicks. Thanks to modern privacy tools, it is pretty challenging to fight against illegal activity sources on the Internet effectively. The presented project aims to create a highly modular and extensible software toolset capable of uninterrupted monitoring of websites (e.g., cryptomarket) running behind the Tor proxy — a specialised and customisable crawler and scraper framework. By implementing modules tailored for individual sites, the program can gather precise and remarkably detailed information over time. Long-term monitoring can produce vital information about the website, offered products, active vendors and even the purchases themselves. Such data can later provide a deep insight into the operations of the selected sites, show current trends and reveal otherwise hidden knowledge.

Due to the nature of the required data (accurate with the necessity of high scraping precision), this program is not meant to be a generic scraper capable

of accessing any darknet website. It should instead be deployed as a targeted solution for concrete, hand-picked websites. This project currently focuses on cryptomarkets operating in the Tor network only but is not dependent on it. The implementation is expected to allow:

- product purchase detection (this further depends on the data available on the websites);
- marketplace, product and vendor meta-data collection (names, images, dates, PGP[1] keys and others);
- website archivation (creation of a chain of custody, offline marketplace browsing);
- and at a later point cryptocurrency blockchain correlation (flagging transactions and addresses potentially relevant to the previously detected purchases).

Mapping and indexing the surface web is typically not a too difficult task. That is because almost every website on the surface web wants to be accessible and seen and voluntarily takes steps to improve its presence on the Internet. One such example can be the usage of the Open Graph protocol[2]. Adding meta tags as defined by the OGP helps web scraping tools better understand a particular page's contents, which improves the presence of said page on the Internet. That is not the case for the services on the dark web. Typically, every Tor hidden service operator is trying to prevent any bots from accessing their web pages. That incorporates custom CAPTCHA challenges, rate-limiting server requests, user account login requirements, and various other anti-bot features as described, for example, by Andres Baravalle [1].

Julia Buxton [2] offers an overview of the issues and challenges cryptomarkets pose on the Internet. It discusses the history of the subject, presents used technologies and underlines issues faced by law enforcement agencies. A more in-depth look at the structure and contents of various marketplaces is offered by Julian Broséus [3]. It provides a detailed view of the marketplace listings, product categories and vendors. It also shows several aggregated statistics which identify popular product categories and vendor practices across marketplaces. Finally, Matthew Ball [4] presents some previously implemented web scraping tools allowing monitoring of dark marketplaces and some acquired results.

[1] Pretty Good Privacy (RFC4880)
[2] https://ogp.me/

2. Nature of Dark Marketplaces

Dark marketplaces are websites operating within the dark web (behind Tor[3] proxy). They provide an environment to allow easy contact between customers (potential buyers) and vendors worldwide. These websites differ mainly in what can be bought there in contrast to conventional marketplaces on the Internet. An overwhelming majority — if not all — of offered substances, goods or services are not legal. These sites can be seen in two primary forms on the dark web:

- structured forum;
- e-shop.

There are not many public marketplaces on the surface web operating as a structured forum. That is not the case on the dark web. Many cryptomarkets still work as a forum. That impairs consumers' comfort and simplicity of the purchase process, though, allows for much more diversity in the offered goods and services.

E-shop based websites are not too different from the well-known and used surface web e-shops like eBay[4] or Amazon[5]. Nevertheless, the dark marketplace sites are generally lightweight and *much* more straightforward than their public counterparts.

2.1 The Building Blocks

Dark marketplaces and their users must employ some specialised software/technology to stay "safe" since most (if not all) of the offered products there are usually not legal. There are three significant areas that need to be secure and anonymous to enable the existence and continuous operation of cryptomarkets on the Internet (visualised in Figure 1):

1. *network access* — highly anonymous network access can be provided by Tor proxy (for both the client and the server), allowing access to and the existence of the marketplace on the Internet;
2. *purchase order payment* — cryptocurrencies offer highly anonymous financial transactions between any number of parties, allowing trades to take place;
3. *safe information exchange* — PGP assures the secure transmission of highly sensitive information between the trade participants (PGP signing capabilities are also used for identity verification of web servers, vendors and users alike).

[3] https://www.torproject.org/
[4] https://www.ebay.com/
[5] https://www.amazon.com/

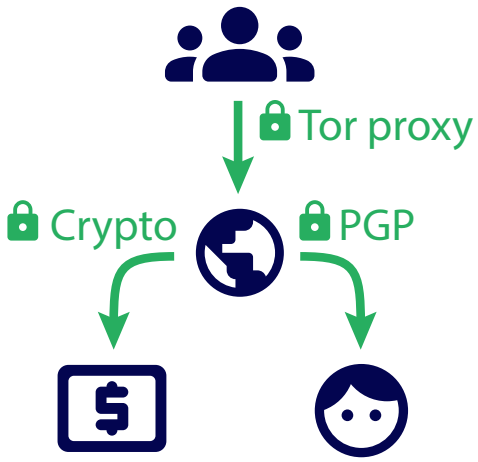


Figure 1. Cryptomarket Building Blocks
This figure shows three main building blocks allowing the existence of cryptomarkets on the Internet. Tor allows anonymous website access, cryptocurrencies allow anonymous payments, and PGP allows anonymous communication.

Secure and highly anonymous network access can be provided by Tor proxy (or other similar overlay networks such as JonDonym[6], I2P[7] and others). Tor primarily offers anonymity to the clients but also allows the creation of so-called hidden services (abbreviated further as HS). Servers deployed as HS in Tor are provided with the same security benefits as the clients. The connection established between a client and an HS is end-to-end encrypted by Tor (HTTPS is therefore typically not used by the server in this case).

Cryptocurrencies are a secure and highly anonymous payment option on the dark web, though some cryptocurrencies provide more anonymity than others by design. Based on DarkDotFail [5], ordinarily supported cryptocurrencies consist mainly of well-known Bitcoin (BTC) and Monero (XMR); some markets may also support Litecoin (LTC) and Zcash (ZEC). The most secure marketplaces only support payments via Monero as it has been designed to maximise its users' privacy.

Asymmetric encryption via PGP is used when sending highly sensitive information (such as delivery address) to the vendor as part of the purchase order. This step is mandatory on virtually any existing dark marketplace. Nevertheless, the signing capabilities of PGP are heavily used too. All the involved parties, cryptomarkets, vendors and buyers alike use them to verify each other's identities based on signed messages and previously presented public keys. Marketplaces rely on PGP signatures to prove the validity of `.onion` domains, as chances of targeted phishing attacks are typically high.

The combination of previously mentioned technologies allows highly anonymous, practical and pretty straightforward trading to take place "anywhere" on the Internet. It is important to note that the technologies allowing such trading are free and relatively simple to use by virtually anyone. The user looks up what they wish to buy and contact the corresponding vendor through the marketplace by placing an order (this is highly anonymous, thanks to Tor). Then, they provide necessary delivery information (this can only be seen by the vendor thanks to PGP asymmetric encryption). Finally, they pay for the order using cryptocurrencies (again anonymous, under certain conditions).

2.2 Website Standards

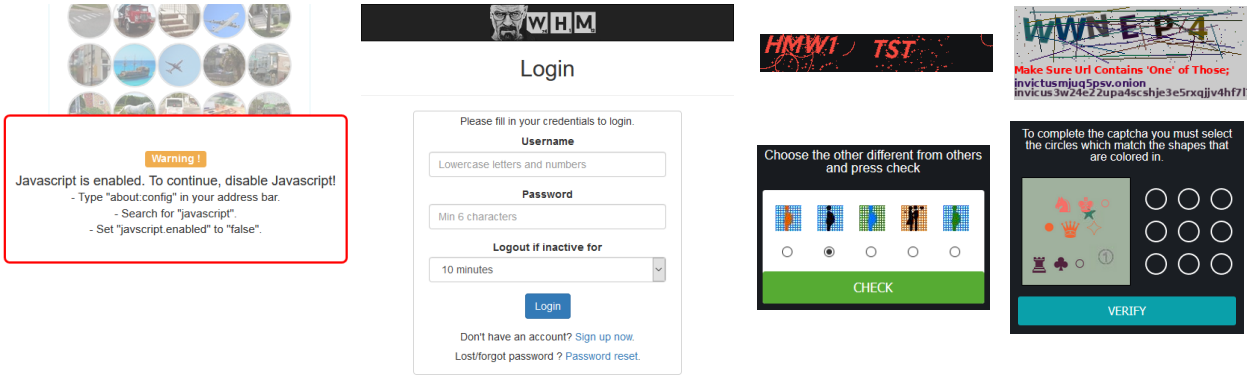
This section will now describe some features typically present on dark marketplaces. The mentioned features mainly cover potential security issues and do not generally provide any purchase-wise advantage. Cryptomarket operators deploy and enforce such measures and procedures to better protect the marketplace and its users.

An overwhelming majority of web services running as part of the dark web follow and enforce a strict no JavaScript policy. One such enforcement web page can be seen in Figure 2a. JavaScript can be used to acquire some additional information about the user's session, as demonstrated by Keaton Mowery [6] or Martin Mulazzani [7]. That is due to the various browser features like Cookies, LocalStorage or IndexDB being available from JavaScript. Furthermore, browser fingerprinting principles can be employed to track the user's session even further.

CAPTCHA challenges are a necessary part of any dark marketplace. Or almost any website on the dark web, for that matter. They help prevent unwanted bots from accessing the actual website and help to mitigate frequent (D)DoS attacks. The challenge types vary but ordinarily do not surprise and are typically custom implemented by each website. Examples from various dark marketplaces can be seen in Figure 2c.

Many popular darknet websites follow and implement the OMG[8] — Onion Mirror Guidelines — a set of instructions defined by dark.fail[9] to "...reduce the impact of phishing and to ease automatic PGP verification of mirrors..." The actual verification of PGP signed messages from websites following the OMG is very quick and straightforward for any user. That is

[6] https://anonymous-proxy-servers.net/
[7] https://geti2p.net/
[8] https://dark.fail/spec/omg.txt
[9] https://dark.fail



2a No-JavaScript Enforcement **2b** Account Login Requirement **2c** CAPTCHA Challenge Examples

Figure 2. Various Website Access Policies

Some of the most notorious features of the vast majority of dark marketplaces are shown in these figures. Figure 2a shows a message requiring JavaScript to be disabled in the browser before continuing. The landing page requiring a user to sign in to an existing user account before proceeding to the market can be seen in figure 2b. Both screenshots in figures 2a and 2b were taken from White House Market's website. Various CAPTCHA challenges taken from different darknet websites are then shown in Figure 2c.

thanks to a publicly available PGP tool[10] also implemented by dark.fail.

The websites will often require logging into a previously created user account before allowing browsing the market freely. Some very exclusive markets may require invitation or reference to allow a user to sign up, but an overwhelming majority will not require this step. Account registrations are email-less and typically do not require any further verification. Because there is no way to recover lost passwords via email, the sites will ordinarily generate a mnemonic string (some 20 random words long) which can be used to reset the account password.

3. Proposed Solution and its Challenges

The proposed solution is to create an automated monitoring tool for selected cryptomarkets. This application would *constantly* monitor the web pages — extracting available metadata and appropriately archiving its contents to create a valid chain of custody. The data collected by the monitoring application will later allow the program to detect changes over time. Detection of such changes can indicate some significant event, e.g. a purchase being finalised, new stock being added, new products being published, or new vendors registering.

The visualisation of the solution can be seen in Figure 3. The program will map a given marketplace (1) looking for pages with essential data (2) and extracting them (3). Over time, the detected changes can set up a purchase time frame (4) and designate relevant blockchain blocks (5). Finally, blockchain analysis will try to correlate relevant transactions based on various available parameters (current cryptocurrency value to fiat currency, advertised product price in time, ordered product amount, shipping, etc.)

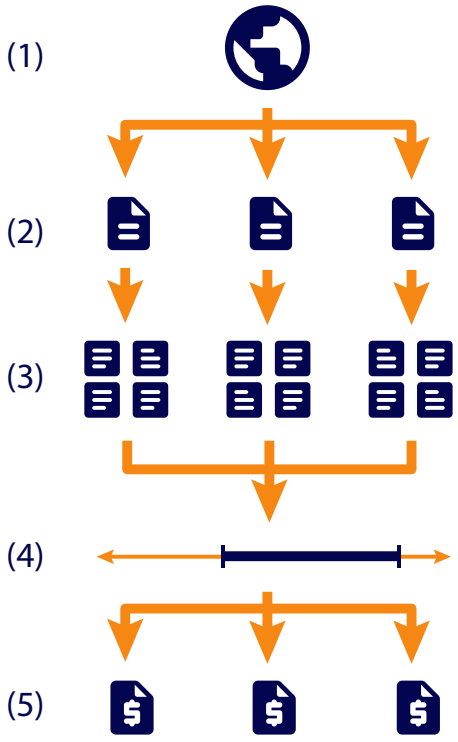


Figure 3. Proposed Solution Overview

This figure shows a simplified overview of the proposal depicting individual solution steps — from a cryptomarket website (1) to extracted data entries (3) to purchase-relevant cryptocurrency blocks (5).

It is expected that data (and its aggregates), collected by the monitoring tool over time, should match

[10] https://dark.fail/pgp

any statistics provided by the marketplace itself — such as rankings of the most successful vendors, the most sold product and virtually any other.

3.1 Website Access

Given that most service operators are actively trying to make automated access to their web servers difficult, the program has a few issues to deal with.

As mentioned before, CAPTCHA prompts are customarily a major part of this bother. Thankfully, services offering automated CAPTCHA challenge solving — like 2Captcha[11], AntiCaptcha[12] and others — do exist on the Internet. The whole CAPTCHA system can be automatically solved and effectively evaded for a reasonable price per challenge entry thanks to these services. From experience with 2Captcha, the price for a 1,000 "normal" (meaning non-reCAPTCHAs) challenge solutions typically lies around 1 USD (solving Google's reCAPTCHAs is significantly more expensive at around 3 USD per 1,000 solutions). Operational expenses for the program running 24/7 for a whole month at a single marketplace with user session of 6 hours and solution success rate roughly at 40% are less than 0.5 USD.

Before being able to access any valuable data, sign-in pages are typically the program's next issue. Solving this is pretty straightforward since the program has to keep track of the HTTP session to prevent CAPTCHAs from popping up again (until session timeout). Manually creating an account beforehand and then making a POST request to log in during the initial program setup is the simplest solution to this problem.

Last major complication: addresses of the web servers are prone to sudden changes. The implementation must keep track of server mirrors and switch to different base URLs in case of need. An up-to-date list of mirrors is usually published somewhere on the website. That is much easier to deal with when the service follows the beforementioned OMG.

3.2 Website Mapping

Another part of the implemented system is website explorer — a web crawler. The crawler module is tasked with page discovery and website map creation. It is necessary mainly in order to detect newly created pages.

The crawler leverages the fact that the website can be reduced to a graph. Each node in such a graph represents a single page on the site; edges represent links in the HTML source code of the corresponding pages. A simplified graph (tree) representation can be seen in Figure 4. Each arrow represents a link (either internal or external) within a given page. The graph shown in this figure has already been converted to a corresponding tree representation with a root at the site's landing page. One such tree representation could be generated by a BFS (breadth-first search) based crawling algorithm.

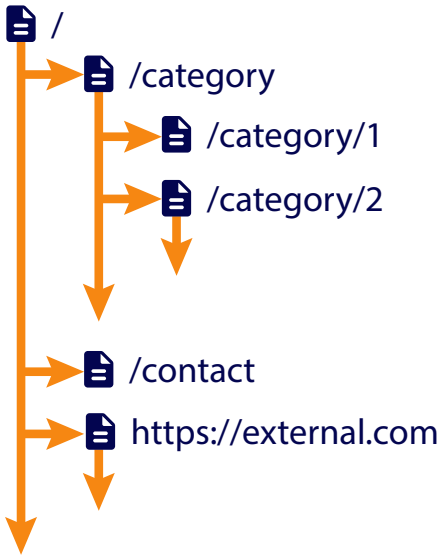


Figure 4. Website Graph Representation
A simplified tree graph representation of a website is shown in this figure.

Newly detected pages are evaluated, categorised and scheduled for further processing. Persisting page metadata such as HTTP cookies and headers is also a responsibility of the crawler. Outputs of this module are then used further down the processing pipeline.

3.3 Data Acquisition

Important pages discovered by the crawler are next selected by the scraper for data extraction. Marketplace-specific strategies are selected based on the web page type, content, location or other attributes. These strategies are implemented by hand for each required page to be processed. That allows fine-tuning and extraction of precisely the information that is required by the implementation.

Extracted data are then stored in a structured format in persistent storage. That allows simple aggregation, filtering and lookup of concrete events. Entries are also linked with other relevant metadata (such as selected HTTP request and response contents, Tor relay context and more). Structured data and record relationships allow complex queries. Such queries may be able to reveal some vital information or relations.

[11] https://2captcha.com/
[12] https://anti-captcha.com/

4. Evaluation

This chapter introduces some of the achieved results and then discusses the validity of the implemented program.

4.1 Currently Available Results

The aggregated statistics to follow are from a single dark marketplace within the Tor network (its name will not be disclosed for obvious reasons). The selected marketplace has around 100 active vendors and around 800 active listings. All marketplace listings are drug-related.

The scraper has processed 53.6 pages on average each minute, with a maximum going up to 66 and an average request timeout rate below 1%. More than 3,000 purchases of 727 different products from 75 distinct vendors have been detected. These statistics were based on data from 30th of March to 12th of April 2021.

Table 1 shows the most popular subcategories with their corresponding top-level category. The subcategories should refer to a single drug (LSD, cocaine, etc.) — not a drug class (psychedelics, stimulants, etc.)

Table 1. Top 10 Subcategories by Number of Sales

Category	Subcategory	Sales
Cannabis	Buds and Flower	2,491
Stimulants	Cocaine	2,194
Benzos	Pills	1,394
Psychedelics	LSD	1,203
Dissociatives	Ketamine	765
Stimulants	Speed	723
Ecstasy	MDMA	455
Cannabis	Carts	452
Psychedelics	Shrooms	421
Prescription	Genuine	330

This table shows drugs with the highest count of product purchases. Calculated from over 12,300 detected purchases between 1 March – 25 April 2021.

Data shown in Figure 5 are based on the same dataset as Table 1. It displays purchase ratios between the available drug categories (classes) on the marketplace. It is clear that cannabis listing sales dominate the marketplace and account for almost a third of all the detected sales. The cannabis category includes buds and flower, edibles, hashish, distillates and more. The second sales rank goes to stimulants which include cocaine, methamphetamine, amphetamine and others. These two drug categories alone account for more than half of all registered purchases.

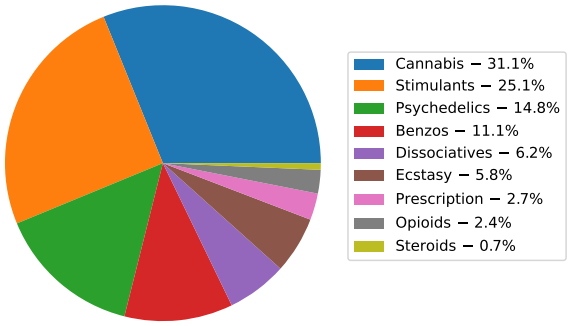


Figure 5. Categories by Number of Sales

This figure displays purchase ratios between main drug categories (classes) available for purchase on the monitored marketplace. Calculated from over 12,300 detected purchases between 1 March – 25 April 2021.

Finally, the histogram shown in Figure 6 shows detected purchases in relation to the time of day. The data follow a normal distribution and contain no abnormal spikes, which is to be expected. This information can hint at where most customers are ordering from in correlation to their time zone and day/night activity.

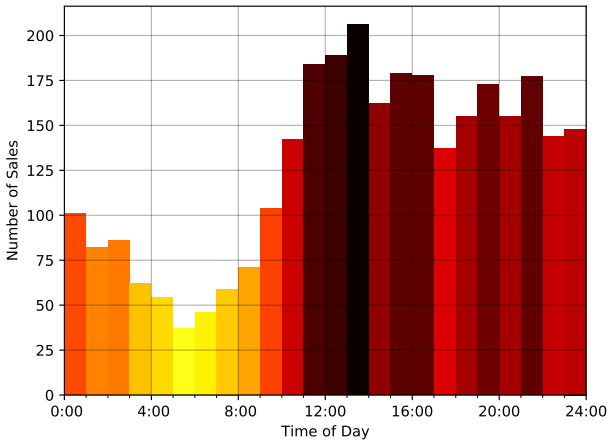


Figure 6. Sales by Time of Day

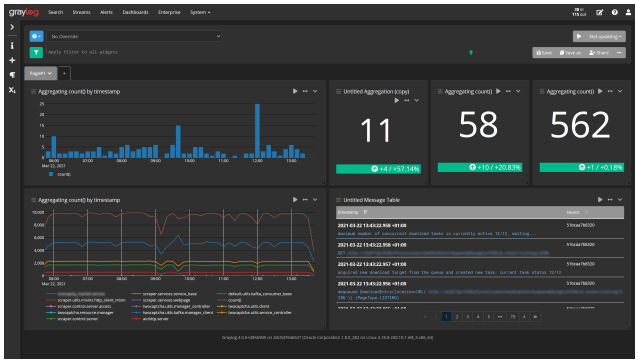
The histogram in this figure shows a summary of detected sales during the time of day. The purchase count disregards categories of the sold products. Calculated from over 3,000 detected purchases during uninterrupted scraping between 30 March – 12 April 2021.

4.2 Monitoring, Reporting and Verification

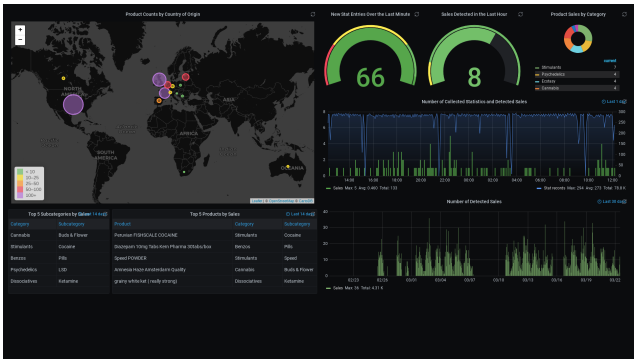
The deployment currently uses two mutually independent sources of system metrics — Graylog[13] and Grafana[14].

Graylog uses logging messages sent directly from the application to create custom metrics and statistics. That allows precise monitoring of what the application is doing at any given time. It also allows live alerts

[13] https://www.graylog.org/
[14] https://www.grafana.com/



7a Graylog Dashboard



7b Grafana Dashboard

Figure 7. System Monitoring and Reporting

This figure shows dashboards with various real-time metrics of the implemented system. A Graylog dashboard, providing information obtained directly from the verbose application logging, is shown in figure 7a. Another dashboard from the Grafana system is shown in figure 7b. Grafana, in contrast with Graylog, uses the application's database as its only source of data.

to be set up for any required events and significantly eases debugging of the system.

In contrast to Graylog, Grafana relies solely on the application's database and is entirely independent of the application itself. It allows easy aggregation, visualisation, presentation and sharing of the data. Grafana, same as Graylog, also provides a module for real-time alerts, which can be set up to track any system metrics based on provided SQL queries. Information loaded from the database can also validate the information from the application logging and vice versa.

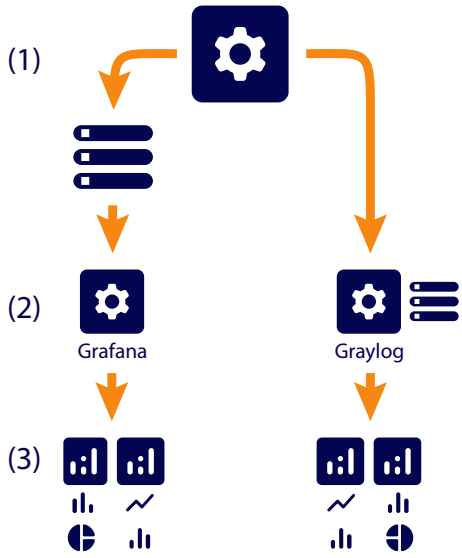


Figure 8. Application Monitoring Deployment

This figure shows the current monitoring deployment. (1) The application produces data into the database (left branch) and logging messages (right branch). (2) The monitoring systems process the data independently. (3) Data are aggregated and presented understandably.

Graphical visualisation of the deployment can be seen in Figure 8. An example with two real-time monitoring dashboards is shown in Figure 7 (Graylog in Figure 7a, Grafana in Figure 7b).

Furthermore, the monitored marketplace provides a publicly available list of the most successful vendors there. Aggregated data from the application do match with the rankings published by the marketplace. Based on the analysis of the collected data, it is safe to say that the application is acquiring correct information and is working as intended.

5. Conclusions

The main goal of this project was to design and implement a highly modular and extensible web scraping toolset capable of working on the dark web. That has been successfully done.

The program has been implemented using the Python[15] language. It uses asynchronous programming and master-worker principles along with the publisher-subscriber pattern to achieve the necessary speed and versatility. Some logic behind the program implementation has been described in Section 3. The implementation uses PostgreSQL[16] database for data persistence and Redis[17] for effective inter-process communication and content sharing. The whole stack of cooperating services (the application itself, its database, Tor proxy, Graylog and Grafana systems with their dependencies and potentially other parts) is containerised and can be deployed in a single command using Docker[18]. Deployment in Docker also offers some level of platform

[15] https://www.python.org/
[16] https://www.postgresql.org/
[17] https://redis.io/
[18] https://www.docker.com/

independence.

Section 4 has shown some of the actual results of the project and demonstrated their validity. The data acquired by the program can be used, and they do provide a real insight into the selected marketplace's operations. Data clearly indicate when a product purchase has been registered by the website allowing further blockchain analysis. Aggregated results shown before are only a small part of the results — raw data can subsequently provide even more knowledge. You can visit the web page with additional public resources for more information and even some real-time metrics of the actual implementation (if the system will be running at the time of your visit).

The core part of the project's implementation (the framework itself) is planned to be open-source. That should allow anyone who would wish to use the framework or continue working and extending it, to do so freely. The author will continue improving and extending the implementation as necessary while being actively used for research purposes here at FIT BUT. Furthermore, the author plans on implementing a separate blockchain correlation module. The module will use the acquired information about purchases to try to correlate, detect and flag potentially related transactions in various cryptocurrency blockchains. That could provide even more information relevant to the marketplaces' operations, vendors, and even customers.

Acknowledgements

I want to thank my supervisor Ing. Vladimír Veselý, Ph.D., for his invested time, expertise, support, passion and ideas.

Icons used in figures 1, 3, 4 and 8 are publicly available at Material.io[19].

References

[1] Andres Baravalle, Mauro Sanchez Lopez, and Sin Wee Lee. Mining the dark web: Drugs and fake ids. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 350–356, 2016.

[2] Julia Buxton and Tim Bingham. The rise and challenge of dark net drug markets. *Policy brief*, 7:1–24, 2015.

[3] Julian Broséus, Damien Rhumorbarbe, Caroline Mireault, Vincent Ouellette, Frank Crispino, and David Décary-Hétu. Studying illicit drug trafficking on darknet markets: structure and organisation from a canadian perspective. *Forensic science international*, 264:7–14, 2016.

[4] Matthew Ball and Roderic Broadhurst. Data capture and analysis of darknet markets. *Available at SSRN 3344936*, 2021.

[5] dark.fail (@DarkDotFail). And now: a whirlwind tour of today's notable darknet markets. [thread]. Retrieved from https://twitter.com/DarkDotFail/status/1299443832228995073, Aug 2020. (Online; Accessed on 2021-03-24).

[6] Keaton Mowery, Dillon Bogenreif, Scott Yilek, and Hovav Shacham. Fingerprinting information in javascript implementations. *Proceedings of W2SP*, 2(11), 2011.

[7] Martin Mulazzani, Philipp Reschl, Markus Huber, Manuel Leithner, Sebastian Schrittwieser, Edgar Weippl, and FC Wien. Fast and reliable browser identification with javascript engine fingerprinting. In *Web 2.0 Workshop on Security and Privacy (W2SP)*, volume 5. Citeseer, 2013.

[19] https://material.io/