Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich

Technical University of Munich

Department of Informatics

Research Internship in Informatics

Evaluating Client Discrimination in Anonymization Networks Using Active Network Scans

Jan Lauinger

Technical University of Munich

Department of Informatics

Research Internship in Informatics

Evaluating Client Discrimination in Anonymization Networks Using Active Network Scans

Auswertung von Client Diskriminierung in Anonymisierungs Netzwerken Anhand von Aktiven Netzwerk Scanns

Author:	Jan Lauinger
Supervisor:	Prof. Dr.-Ing. Georg Carle
Advisor:	M. Sc. Oliver Gasser, M. Sc. Sree Harsha Totakura
Date:	Nov 10, 2017

I confirm that this Research Internship is my own work and I have documented all sources and material used.

Garching, Nov 10, 2017

Location, Date

Signature

Abstract

This work investigates client discrimination in anonymization networks. It thereby inspects discrimination techniques and motivations, how website responses differ with regard to client discrimination, and whether the anonymous or regular client becomes discriminated. Focusing on application layer client discrimination, the work handles dynamic website content as well as geo-fencing problems with the help of the Tor anonymization network and the Planet Lab network. Targeting the Alexa top 1 million list through the infrastructure setup enables to perceive and evaluate discrimination within comparable website responses. Results show discrimination in form of IP layer blocking, CAPTCHA, and reduced content detection towards anonymous internet users. The large variety of differing website responses together with the scanning domain information allow assumptions about discrimination motivations.

Zusammenfassung

Dieser Forschungsbericht untersucht Internetdiskriminierung in Anonymisierungsnetzwerken. Dabei werden sowohl Diskriminierungstechniken, Diskriminierungsmotive, Unterschiede in HTTP Antworten, in Bezug auf Internetdiskriminierung, sowie der Bezug auf den jeweiligen Internetnutzer näher betrachtet. Diese Arbeit legt Fokus auf die Auswertung der Diskriminierung in der Anwendungsschicht. Es wird Einfluss dynamischer Websiteinhalte und durch Geofencing hervorgerufene Probleme, durch die Verwendung von Tor und Planet Lab Rechnernetzen ausgeschlossen. Die Verwendung der populären Alexa Domainliste und der Rechnerinfrastruktur ermöglicht das Empfangen und Auswerten von vergleichbaren HTTP Websiteantworten. Resultate zeigen Diskriminierung bezüglich anonymer Internetnutzer in Form von IP-Schicht Blockierungen, Mensch und Maschinen Unterscheidungstests und reduzierten Nachrichteninhalten. Die große Vielzahl unterschiedlicher Websiteantworten im Zusammenhang mit Domain Informationen erlaubt das Erschließen von Annahmen über Diskriminierungsmotive gegen Internetnutzer.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Since the leaks about government in-depth surveillance on live communication and stored information [10], the awareness of personal information and privacy with regard to the use of the Internet has increased. Not only due to privacy protection reasons, more and more people start to use anonymization networks for different reasons [18]. The increasing popularity of anonymization networks also influences average Internet users to think and make decisions about different practices of Internet usage. The anonymous approach of Internet usage uses client requests which propagate through an anonymization network such as the Tor network [4]. Requests of regular or non-anonymous users take a more direct trace to request web services. These different practices of Internet usage induce occurrences of client discrimination.

Distinct motivations of use cases of client discrimination with regard to anonymous network usage appear in different scenarios. The business model of a website host might consist of selling user data. When a client hides more information through anonymization techniques, it is less valuable for a website provider. Therefore, counter measures of the website provider might block or discriminate the customers which hides information. Less request processing means less costs and saves resources of the website provider [17]. Another occurrence of client discrimination is CAPTCHA solving by internet users. Website service providers might defend their services of automated exploitations at scale [21]. Automated exploitations are easily perceivable due to the fact that these many requests originate from the same location. Hence obfuscated website request through anonymous networks can profit of changing connection circuits of anonymization networks. Anticipation of exploiting actions through anonymous networks leads to another view and intention about setting up CAPTCHA solving. The

uncertainty and arbitrariness of reasons and intentions of client discrimination is an problem which this work captures as far as possible.

Another motivation and occurrence of client discrimination is government censorship [26]. China for instance restricts the use of anonymization networks to further keep control over transparent Internet traffic originating from China. Blocking of anonymization network nodes represent used restrictions. Hence, development of services and provisions to bypass restrictions increases [1]. New investigations about client discrimination enables governments to better censor Internet usage. On the other side, new findings about discrimination help users to extend anonymity with regard to Internet usage.

To clarify the intention of this research report it is necessary to mention that this work does not investigate solutions which solve the discrimination drawbacks of anonymous users. Therefore evaluation of circumvention techniques such as domain fronting [9], content caching [11], and network traffic obfuscation [5] do not represent the purpose of this report. On the contrary the following problems occur while clarifying the reasons, the extend, and factors which point out client discrimination due to the usage of anonymization networks.

- It is necessary to provoke time stable and locational conform non anonymous web responses as a reference to corresponding anonymous website requests.
- Evaluation of many discrimination methods only allows assumptions about discrimination motivations.
- Different web services can have a large variety of discrimination implementations besides varying intentions of whom to discriminate.

The overall intention of this project is to provide insights into client discrimination in anonymization networks to further clear the picture and facilitate future questions about the controversial use cases of anonymization networks. It is therefore essential to investigate client requests which originate from an anonymization network exit nodes and client requests which request data in a non-anonymous and more direct approach.

1.1 Research Questions

This research project investigates and provides an evaluation of client discrimination in anonymization networks. Since discrimination in general opens a wide field of different options, the first research question tries to answer what discrimination possibilities and motivations for discrimination exist. Understanding discrimination possibilities

allows assumptions and might even prove the original motivation. Since application of client discrimination happens on different network communication layers [9], this report focuses on application layer discrimination possibilities. The investigation of other network layer effects concering our research topic goes beyond the scope of the project.

As it is unclear where and how web service responses of different request practices differ, the second research question investigates to what extent web service responses of different request practices differ. Knowing different discrimination possibilities and motivations does not explain concrete implementation details of responses. Differences in HTTP headers and bodies might already indicate uneaqual treatment of internet users. The goal aims to find the dominating components of a response message for client discrimination. These findings would further reinforce the implications of anonymous network usage.

An important aspect set the target list because it affects the coverage of discrimination intentions of web service providers. A broad list of addresses ensures the experience of almost all cases of discrimination techniques and motivations. Additionally a wide variety of geographical locations of the targets increases the probability to find time stable and locational conform responses. This is an important approach to provide enough consistent responses of non anonymous requests. More responses of non anonymous requests means more opportunities of comparing responses of Tor client requests.

Lastly it is necessary to remember that discrimination incidents can affect regular clients as well as anonymous clients. Therefore the third research questions asks which of both requesting nodes becomes discriminated. It is important to evaluate the scanning results with regard to both discrimination directions to uncover all different discrimination implementations and motivations. Qualified for these two different discrimination directions are clients which use either TOR exit nodes or regular Planet Lab nodes for requesting web services.

1.2 Contribution

It was possible to achieve the exploration of client discrimination in anonymization networks through active networks scans. The Alexa top 1 million list set the target list. Time and location conform and stable web responses of non anonymous requests had to be found before the comparison of web resonses of respective anonymous requests. Not following this measure would distort the impact of client discrimination on anonymization network usage. Discrimination differences on already non anonymous

requests could lead to wrong insights about anonymous request responses. Scans originating from three different Planet Lab locations in the same country ensured to check consistency regarding time and location of regular and direct website requests. Scans ran in sequence and thereby allowed comparisons with regard to time. If responses of these requests had been equal, it was possible to continue comparing these responses to the responses provoked from a Tor exit node from the same country.

Conducted scans requested website content with the HTTP protocol. Resulting responses delivered useful HTTP headers with HTML content of the web services. After receiving and storing HTML content and HTTP headers, the evaluation with regard to discrimination started. The evaluation of output files differentiated properties of discrimination methodologies such as blocking, CAPTCHA solving, and content length. The outcomes again determined conclusions about discrimination motivations.

In conclusion:

- We targeted the Alexa top 1 million website list for scanning
- We conducted HTTP scans from varying Planet Lab server locations
- We rescanned the target list from Tor exit nodes which were locational close to our Planet Lab servers
- We evaluated and compared scanning results with regard to client discrimination
- We discussed and reasoned about intentions of client discrimination in anonymization networks

1.3 Outline

The structure of the report follows a specific schema of scientific methods [6]. While introducing current observations about the topic of the research internship, the intruduction chapter additionally provides the hypothesis of client discrimination due to anonymization network usage. The second chapter gives background information about the topic. Afterwards the third chapter describes the methodological limitations and the approach of how to find an test new observations. After going into implementation details of the approach in the fourth chapter, the next chapter evaluates and discusses the outcomes of the scans. It thereby checks if consitency in observations exists and compares the findings to other research. The sixth related work chapter states the position of the report with regard to other similar research. Lastly the last chapter concludes the work and indicates future work trends.

Chapter 2

Background

The intention of this chapter is to introduce the main topics of the project. The following sections explain the background information with basic technical explanations without going into deep detail.

2.1 Client Discrimination

Due to the reason that application fields, motivations, and interpretations of the discrimination term drastically vary, this project breaks down the term to three main implementations and designs. The first and strongest occurence of discrimination is the form of blocking. Motivations for discriminating through blocking are clearly directed to the concerned clients. Hiding content, excluding specific user groups, and precaution of fraud are occuring motivations regarding blocking. The second and less discriminating technique of client discrimination enables the user to access websites with the drawback of an short verification. Here, techniques such as CAPTCHA solving have motivations of defending exploitation. An anonymous exploit of a web service has a higher probability compared to a regular exploit. Additionally anonymous users who conduct vicious actions find more protection behind a anonymization network and are safer with regard to blacklisting. Lastly there is the user treatment with a varying amount of content. Reduced amount of content is the weakest form of discrimination because websites are still accessible. Drawing conclusions from differences of reduced amount of responses requires the highest effort with at the same time lowest chances of successfully estimating or assuming motivations. That is why this work tries to especially find explanations for the previous two discussed forms of discrimination.

Application of discrimination techniques apply on various networking layers. Techniques such as domain fronting leverage weaknesses of discrimination measures on multiple communication layers for providing their service [9]. With regard of the scope of this project, this work focuses on apllication layer discrimination occurences. Supporting the last point, application layer protocol implementations between anonymous and regular Internet clients vary the most.

2.2 Anonymization Networks

Reasons why anonymization networks become more and more popular are the awareness of privacy through increasing government surveillance, Internet censorship, and safer browsing through anonymization. The mostly used and well known Tor anonymity network provides the core component of the investigation of this work. This network has more than seven thousand active relays [4]. Before a Tor client accesses the regular Internet it connects to and exits the Tor anonymization network. To pass a three relay circuit in the Tor network a client connects to an entry guard node, becomes redirected to the next relay, and leaves the network through an exit relay. Encryption of the application layer protocol data thereby anonymizes the client's source IP address. During the routing of the circuits, each relay decrypts and encrypts the application layer to reveal the next hop. The last relay of the circuit chain decrypts the innermost layer of encryption for sending the original data to the real destination without the source IP address of the client.

Most Tor users use the browser package of Tor which allows a minimum of three relays per circuit. The manual installation of the Tor source code on our measurement server in comparison enables more configurations. Due to reasons of faster scanning behavior this work manually creates circuits of two relays. This moreover gives the chance to configure the desired exit relay location of the circuit. There are more features of Tor network usage which are not further explained because this work only relies on the core network functionallity.

2.3 Network Scanning

Network scanning techniques distinct between active and passive scaning techniques. Whereas passive scanning techniques investigate and evaluate traffic flow data and labels, active scanning actively executes scans. Afterwards active scanning evaluates scanning responses. Therefore acive scanning produces additional network load compared

to passive scanning. This advantage of richer experience within the collected informations allows to find smaller variances caused by discrimination. Thereby the project requests target addresses more often from different locations to increase the probability of distinct reponses. Active scanning within this project relies on HTTP scans of IPv4 addresses. This ensures to gain enough data concerning the evaluation of discrimination possibilities and the comparison of reasonable data types for client discrimination for the scope of this project.

Chapter 3

Methodology

To be able to perform comparable active network scanning it is necessary to setup an elaborate approach of infrastructure usage. Scheduled interaction of network components contribute to a successful reception of the expected results. It is therefore important to define an approach which enables to pursuit the desired goals with as little complications as possible. For pointing out the dependencies of services and applications on each other while measuring for a successful scan, the description of the methodological infrastructure follows successive steps.

3.1 Approach

Initially there is the consideration that packages which use the HTTP protocol dominate most Internet traffic [12]. A first approach is to emphasise on scanning simple HTTP get requests. With this method, it is possible to cover the majority of files and with that discrimination possibilities which are requested in anonymous networks on the Internet. This moreover allows conclusions about discrimination motivations and delivers a wide range of discrimination possibilities. Also because implementations of the HTTP protocol vary and provide different request responses. A more detailed explanation of the requested data is in the data section 3.1.4 of this chapter. There, the introduction of the requested data indicates possible data fields which possibly answer the second research question which investigates the extend of differentiation of web service responses of different request practices due to anonymization network usage.

The diversity of HTTP responses caused by varying requesting locations calls for active network scanning which originates from close locations. Following this intention reduces

problems such as different website responses due to diverse geographical requesting node locations [16]. The distribution of the Tor network's exit nodes over multiple continents requires scanning possibilities from locations which are as close to the performing exit nodes as possible. As a result the use of the global overlay network Planet Lab [2] enables to run scanning activities from geographically close distributed nodes. Whereas the configurations of the Tor controller tool allow to choose specific exit nodes to perform scans, this project uses a measurement proxy tool to connect to specific Planet Lab nodes. That measurement proxy tool not only enables to conduct scans, it handles the problem of geographical coordinated network scans and with that the prevention of dynamic content.

The next subsections further explain parts of the general approach.

3.1.1 Infrastructure

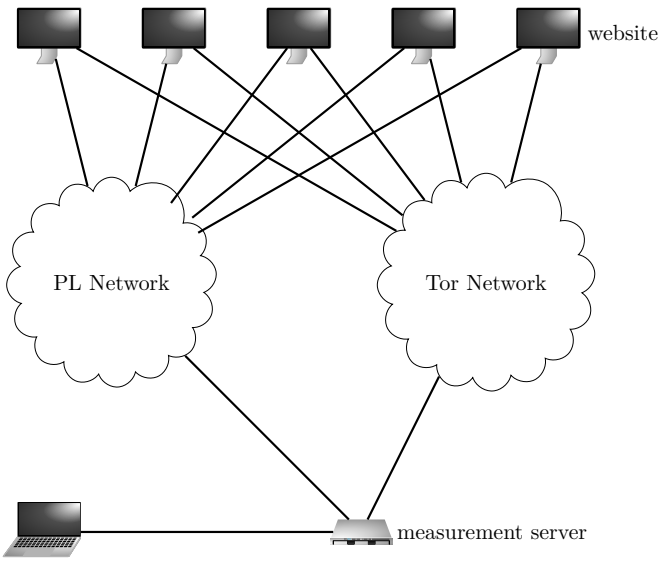

Figure 3.1: Infrastructure Setup

The set up of the infrastructure is dividable into three components. The first component is the measurement server which is located in Munich and executes the scanning script. The Tor and the Planet Lab networks make up the other two components of the infrastructure. Planet Lab is an an Internet overlay testbed with servers located in different sites across the world [22]. The Tor network anonymizes and defends their users against traffic analysis [20]. Regarding the connection between these three components, it is important to understand the abilities of the measurement server. On one side the measurement server installs the Tor source code to be capable of controlling

and configuring Tor circuits and thereby the Tor network as the second component through the local Tor controller. On the other side the measurement server uses a self developed measurement proxy software tool to establish connections to the third Planet Lab network component. The network scanning script uses both software extensions to connect to other infrastructure components. So active network scans can take the path from the measurement server with the help of the measurement proxy tool over Planet Lab nodes to the destination address of the web service. Nonetheless scanning requests can also take the path through the Tor network with the help of the Tor source code and eventually request web services from the anonymous network exit nodes.

For clarifying similarities and differences of the Planet Lab and Tor network, it can be said that both networks contain a large amount of nodes which are geographically distributed over multiple continents. Connections to the Tor network have to enter the network through an entry guard node and exit the network through an exit relay. This means that requests which traverse the Tor network have to go over Tor network circuits of at least two nodes. As it is possible to configure Tor circuits manually through the local controller, this project chooses an entry guard in germany and adds a different node of the Tor exit relay list as an exit node per scan. This ensures that every scan of requesting all targets uses a different circuit and thereby a different exit node for requesting a site. The reason for only choosing two hop circuits provides the advantage of faster scanning due to reduced relay delays. The purpose of the Planet Lab network on the other hand is to find locational and timely stable web responses requested from different geographical locations. Therefore the combination of the Planet Lab network with a simple measurement proxy tool which relocates the scanning source to the Planet Lab network nodes satifies the demands for the intentions of the project to collect comparable scans.

3.1.2 Target List

Choosing the Alexa Top 1 Million list appears to be the most suitable targeting list. This list ranks websites based on their global popularity of Alexa Toolbar users. No other service provides similar variety of important Internet addresses [7]. Advantages of choosing the Alexa Top 1 Million list are the geographical distribution of targets and the downloadable list file which the scanning program uses.

3.1.3 Direct vs Anonymous Web Requests

Since all forms of discrimination happen against regular and anonymous internet clients, the project has to evaluate the scanning results about discrimination against regular and

anonymous clients for addressing the last research question. It therefore collects web service responses from the different requesting nodes separately. These different requesting nodes are either part of a direct or non direct anonymous web requesting approach which utilizes the Tor network. In the direct web requesting case the measurement server uses Planet Lab nodes to immediately set up connections to the destination address. In the case of anonymous web requests the measurement server establishes a connection to a Tor entry guard before reaching a Tor exit node. From this exit node the web request tries to establish a connection to the final web service address. After collecting responses of direct and anonymous request, comparisons towards each client side deliver the desired results which explore the obscurities of discrimination directions.

3.1.4 Data

This project uses the RFC HTTP [8] protocol paper to find suitable comparing properties with regard to client discrimination. The IPv4 scans produce data of HTTP header information, HTML content, and error types. The gained data enables comparisons of content lengths, similaritiy matching, and content interpretations.

3.2 Limitations

Almost no research projects are free of problems which limit the pursued investigation. Similarly this report states limitations in following paragraph.

The fact of diverse website responses due to different requesting locations requires comparable web requests to originate from close locations. The continental unequal distribution of nodes of both networks limits thereby the validity of findings to with nodes well equipped continents such as Europe and the USA. Apart from that there is Planet Lab which does not support measurements of IPv6 scans even though IPv6 is the next generation internet protocol. This is the reason why this work only investigates IPv4 addresses. Another smaller limitation regarding Tor is the mandatory usage of two hop circuits which slows down scanning performance. It moreover complicates scanning settings with regard to network usage compared to a simple proxy solution.

Chapter 4

Implementation

The program and tools described in this implementation chapter aim to achieve the predefined goals. These goals have been to reveal possibilities and motivations about client discrimination, to find reasonable types of application layer data which imply discrimination, and to gain knowledge about which requesting side becomes discriminated. Therefore the main scanning program has to connect to and execute website scans using Tor and Planet Lab nodes for collecting the required data.

4.1 General Setup

Two subsections divide the general setup into Tor and Planet Lab specific settings and configurations. The first subsection refers to all settings and configurations with regard to the Tor source software. The second part explains all Planet Lab specific configurations.

4.1.1 Tor Settings and Configurations

It is necessary to install the stable Tor source code version 0.3.0.10 on the measurement server. This installation requires libevent, openssl and zlib already installed on the machine. After unzipping the Tor source code tar and switching into the Tor directory, the next step is to open the `src/or/or.h` file for chaning the line `#define DEFAULT_ROUTE_LEN 3` to `#define DEFAULT_ROUTE_LEN 2`. This ensures to have two hop circuits when circuits become built per default. At every start up Tor builds some default circuits which our scanning program has to close. This is due to the fact that

our scanning approach requires circuits with a specific exit node. The scanning program might work without changing the default route length before the Tor installation but this configuration shows that two hop cirucuits work. Now the `./configure && make` command compiles the Tor source code.

- ControlPort 9051
- CookieAuthentication 0
- ___LeaveStreamsUnattached 1
- ___DisablePredictedCircuits 1
- NewCircuitPeriod 99999999
- MaxCircuitDirtiness 99999999
- CircuitBuildTimeout 5

Before starting up Tor, it is necessary to change the parameters of the Tor configuration file to the values specified in the above listing. `CircuitBuildTimeout` specifies how many seconds Tor tries to build a circuit given a path of node fingerprints. `NewCircuitPeriod` which defines whether to periodically build a circuit after every number of seconds has the maximum possible value. This ensures that Tor does not build circuits by itself all the time. `__DisablePredictedCircuits` prevents Tor from building working 2 hop circuits. This makes sure that the scanning program is the only client building working circuits. `__LeaveStreamsUnattached` stops Tor from automatically attaching streams to already built circuits. It moreover stops Tor from creating new circuits if none are available. Furthermore the parameters `ControlPort` and `CookieAuthentication` allow a client to connect over port 9051 to the Tor controller. A connection to the Tor controller allows to configure Tor during program execution. `CookieAuthentication` set to 0 allows authentication of the Tor controller client by only sending the string message `AUTHENTICATE`. The maximum value of the `MaxCircuitDirtiness` configuration parameter prevents Tot to build new circuits in general. Using the `-f /path` flag with the path to the configuration file, Tor is able to start up with valid configurations with regard to the scanning program.

4.1.2 Planet Lab Settings and Configurations

To use Planet Lab it is necessary to register a slice online at the official Planet Lab site [23]. Afterwards it is possible to login as a user and upload the public key of the measurement server. When a user adds its own slice to other active Planet Lab nodes,

Planet Lab distributes this public key of the user to these Planet Lab nodes. This project uses a python program for the attachment of the slice to running Planet Lab nodes. The Planet Lab RPC API which the python program connected to, allows to query all active Planet Lab nodes. Moreover within the python program, the google maps python package [19] assigns locations to all active Planet Lab nodes if possible. Afterwards the program adds all successfully located Planet Lab nodes to the projects Planet Lab slice. The python program creats a text file in the end. This file contains IP adresses and country codes of specific Planet Lab nodes to which the projects slice has been added successfully.

4.2 Scanning program

In the beginning of the scanning program, the command line option parser parses any given arguments. Possible command line options are filenames of the Alexa top 1 million CSV list, the Planet Lab node list, and the output CSV list. Moreover a user is able to specify the measurement proxy timeout number and file path. Lastly the first hop Tor entryguard as well as the application protocol are options. The scanning program requires the fingerprint of the first hop Tor entryguard. HTTP and HTTPS are valid protocol options. After the command line option parsing, the program reads in two text files. The first file is the output file of the python program described in subsection 4.1.2. The second is the Alexa website list containing 1 million target addresses. Additionally the scanning program uses the Onionoo Tor network status protocol API [25] to query a JSON object. This JSON object contains all active Tor exit nodes. The function `getTorStatus` parses and returns the IP, country code, fingerprint, and host name of each exit node. A next step compares the country codes in the Tor and Planet Lab node list. If a country code of one list is in the other list, the list files keep all entries containing this country code. Otherwise the program deletes entries containing a country code of only one list file. As a step to simplify the coding, the scanning program initializes maps for the Tor and Planet Lab scanning. The keys and values of the Tor map are countries and slices. Slice entries to a specific country key represent row indices of the original Tor list file which contains this specific country. This mapping allows to loop over countries and within that loop to loop over the slices. This preprocessing step enables the implementation of the predefined measurement approach 3.1.

Next the scanning program initializes the HTTP client using the SOCKS5 protocol. This gives the measurement server the ability to make HTTP and HTTPS requests through Tor. Additionally the connection between the local Tor process and a TCP client initializes a Tor controller client using port 9051. The Tor controller connection

becomes established with the authentication string message and sets a configuration with the `setevents circ stream orconn` message. This configuration shows events which indicate a change in a relay connection. This setting adds the functionality to receive events.

Following the steps described above, the program initializes two notification channels and passes them together with the Tor controller connection into the event and background goroutines. The description of functionality of these two goroutines is in the subsection 4.3. Before eventually starting to scan, the program initializes an output writer which connects to the output file. A more extensive description about the output file is in the section 4.4. The scanning section of the scanning program consists of three for loops. The outer loop coveres all targets. The mid loop goes over all countries. The most inner loop takes all country locations in form of indices of the respective input list. This input list is either the Tor exit node or Planet Lab node list. The most inner loop shuffles the list index numbers to provoke a random selection of locations within a country. The execution of the Planet Lab scan function, Tor scan function, and again Planet Lab function follows the random index shuffling subsequently. Lastly the program flushes all written results to the output file.

The following two subsections describe the Tor and Planet Lab scan functions.

4.2.1 PL scans

The most inner scanning loop calls the Planet Lab scan function with the Planet Lab map slice, the output writer, the Planet Lab list, a target, and the scanning protocol as input parameters. First the Planet Lab scan function loops over all slice components. Inside this loop the function extracts all required informations for the scan execution and output file. Next an execution object with timeout functionality triggers the measurement proxy tool. The input of this execution object consists of the path to the measurement proxy tool, flags, and arguments for the measurement proxy tool. The measurement proxy tool subsequently executes the wget request which requests headers and content from the target through the Planet Lab node. If a request is successful the Planet Lab scanning function writes headers and content together with timestamps and the host information to the output file. Otherwise errors fill the error field of the output. Timestamps are recorded before and after the measurement proxy tool execution. Errors arise through a timeout or the measurement proxy tool. After three successful scanning locations, the function returns. Now it is time for the Tor scan function to start within the most inner loop.

4.2.2 Tor scans

The scanning program calls the Tor function with the Tor map slice, the output writer, the Tor exit node list, the Tor controller connection, the golang HTTP net client, the first channel, and the protocol option value. Similar to the Planet Lab scan function the Tor scan function loops over all slice components and quits after three successful scans. Next the Tor scan function uses the Tor controller client and the closecircuit command to close all active Tor circuits by calling the `cleanCircuits` function. The event goroutine informs the main goroutine through the first notification channel of figure 4.1 about the status of no active circuits. Thereupon the Tor controller client executes the circuit build command with the fingerprint of the specific Tor exit node. This exit node comes out of the Tor exit node list with the help of the row entry index. If a circuit is successfully built, the next channel notification reaches the main goroutine. The program is now able to execute a website request through the SOCKS5 client. It sets the request header fields equal to the measurement proxy request header fields. With an unsuccessful circuit build, the circuit failure error message represents the error field of the output entry. Similar to the Planet Lab scan function the Tor scan function writes the target, hostname, IP, location, requesting time, response time, headers, content, and errors to the output file. The output writer handles this task.

4.3 Architecture

The Tor configuration controller determines the architecture because of the two following two problem characteristics. When the scanning program requests a target via the SOCKS5 Tor client, the Tor controller client has to attach the resulting request stream event to a specific Tor circuit simultaneously. Otherwise requests result in a timeout. Moreover it is mandatory to provoke and catch Tor controller event responses in a way that the circuit building, stream parsing, and stream circuit attachment work seamlessly. These requirements lead to the following architecture:

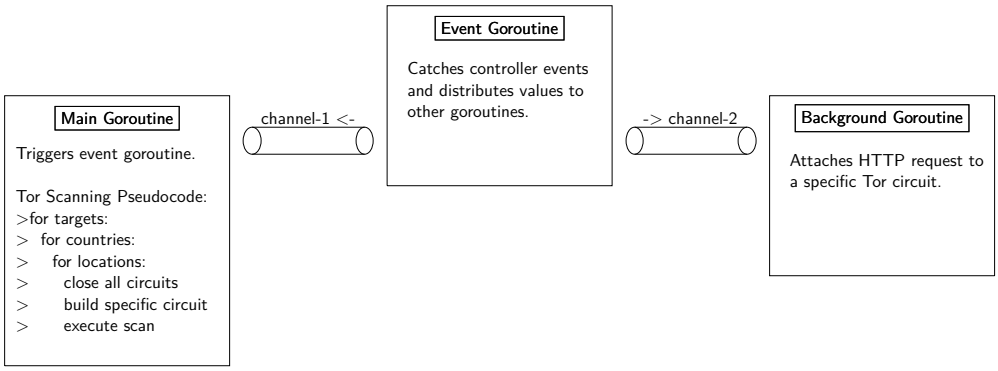

Figure 4.1: Architecture

Three goroutines allow concurrent program execution of three functions. Channels thereby enable communication and synchronization between these functions. The architecture in figure 4.1 allows communication between the main process which runs in a goroutine and the event function which runs in another goroutine. Moreover the second channel passes informations between the event and background goroutines. For instance when the event goroutine passes an argument into the first channel, the main process is able to read out this argument.

By looking at the Tor scanning pseudocode of the main goroutine in figure 4.1, the pseudocode calls the closecircuit Tor controller command which provoks controller events. The event goroutine catches these events. Hence the event goroutine calls the closecircuit controller command as long as no more circuits are active and informs the main goroutine about this status with the help of the first channel.

The next controller command from the main goroutine is the build circuit command. In the same way as before the event goroutine parses the circuit id number out of the occurring event and updates the main goroutine about a either successful und unsuccessful circuit build with the help of the first channel. Moreover it passes the circuit id number through the second channel to the background goroutine in the case of a successful circuit setup.

The HTTP request from the main goroutine causes another event consequently. This event in form of a streaming event delivers the stream id number which the event goroutine extracts. Again using the second channel the event goroutine informs the background goroutine about the new stream id number. While the event goroutine parses and discards other streaming event notifications, the background goroutine calls the attachstream Tor controller command with the Tor controller connection and at-

taches this specific stream to a specific circuit. A successful attachment results in a successful Tor website request on the main goroutine side.

4.4 Response Comparison

The row format of the resulting scanning output `result.csv` file is a CSV file with the entries `scan target`, `hostname`, `IP`, `location`, `request time`, `response time`, `headers`, `content`, and `error`. Each scan produces a new row. Thereby the scan target represents the target address. The hostname, IP, and reponse field belong to one exit node respectively. Request and response time are timestamps before and after each scan in unix time format. This layout containing the scanning results allows a comparison with regard to the projects research questions.

After data collection and with the help of the measurement server, a jupyter notebook reads in the resulting `result.csv` file. Inside the notebook, the imports of python packages numpy, json, and pandas faciliate data analysis.

Chapter 5

Evaluation

This chapter evaluates headers, content and errors of HTTP responses collected by Tor and Planet Lab scanning. Thereby the analysis of different responses answers the research questions about possibilities, motivations, the extent and subjects of client discrimination.

5.1 Evaluation Relevant Precautions

5.1.1 Server Reduction

First of all, several unreachable Planet Lab servers led to an exit node filtering on Planet Lab side. A connection trial bash script reduced Planet Lab servers to 74 working nodes. This reduced Planet Lab server list of 16 different locations produced successful requests within 4 european countries. These countries were Finland, France, Norway and Germany. Useful reponses which accurately followed the scanning approach originated from these mentioned contries. The reason for the heavy location coverage shrinkage of available Planet Lab servers was due to different error types. Planet Lab nodes had different connection settings and responded diverse to connection requests. Therefore timeout and measurement proxy errors appeared on Planet Lab scanning side during scanning. Types of measurement proxy tool errors were tunneling, authentification, name resolution and routing failures. On the other hand the errors of non available routers under specific fingerprints occured with scans through Tor exit nodes. The only positive and consecutive outcome of the server reduction was an increased scanning speed.

In contrary to working Tor connections, Planet Lab connections set the big bottleneck regarding geographical coverage of discrimination testing. Nevertheless, the smaller amount of total servers enabled to produce results which followed the defined scanning approach.

5.1.2 Data Preprocessing

For faciliating data evaluation, a jupyther notebook maps the list of rows of scanning samples of the `result.csv` file into python dictionaries. This dictionary represenation of listing 5.1 allows to evaluate IP-layer blocking thresholds more easily.

Listing 5.1: Dictionary Sample of Data Reduction

```
{'quora.com':
  {'FI': {'numb_tor_sucess': 3, 'numb_pl_success': 2,
      'numb_pl_err': 0, 'numb_tor_err': 0, 'num_tor_blocking': 0,
      'num_pl_blocking': 0},
    'FR': {'numb_tor_sucess': 3,
      'numb_pl_success': 6, 'numb_pl_err': 6, 'numb_tor_err': 2,
      'num_tor_blocking': 1, 'num_pl_blocking': 0},
    'DE': {..
```

Before inspecting the data object of listing 5.1, one point has to be considered first. The scanning approach of this project determines to scan a domain as long as there are enough successful scanning responses from both sides. This explains the numbers of the variables `numb_tor_sucess`, `numb_pl_success`, `numb_pl_err`, and `numb_tor_err`. The requirement is always to receive 6 successful Planet Lab and 3 successful Tor responses. In the case of this country object, there have been 2 Planet Lab scanning errors and 4 Tor scanning errors. The boolean variables `pl_blocking` and `tor_blocking` turn true if a specifically defined majority value of Tor and Planet Lab scans satisfy the scanning timeout delay. Figure 5.1 shows the strategy of finding the majority thresholds regarding country and requesting side. Here, the highest jump of the cumulative distribution function over timed out requests per domain defines the thresholds. These thresholds label scanning requests per location. `Request time` and `response time` of row entries of the original `result.csv` file decide thereby if a single requesting sample counts as timed out. This evaluation of timed out occurrences allows to accurately determine if multiple requests per country count as IP-level blocked. As a result boolean labels combine these results in form of the `pl_blocking` and `tor_blocking` variables and faciliate table creations.

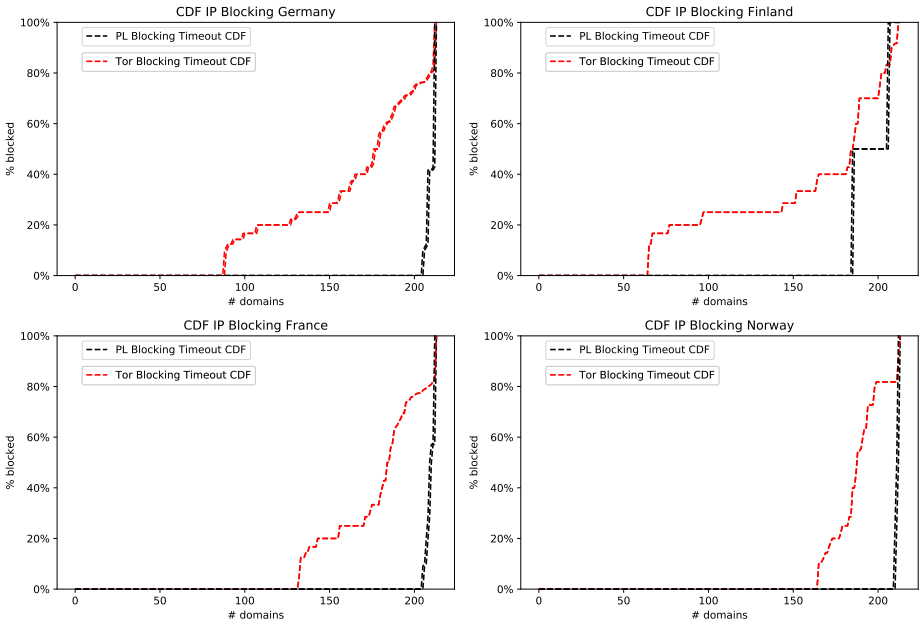

Figure 5.1: Cumulative Distribution Function of Domain Timeouts

There are two important notes to mention regarding the evaluated data in the upcoming tables in the follwing sections. First and with regard to CAPTCHA detection, the python program searches for specific content strings which indicate CAPTCHA solving. The most common CAPTCHA indicator is Google's `g-recaptcha` tag. Another example are cloudflare protected sites which show a default message of three sentences during Javascript anti-bot checking. Other variables of the content object represent average string lengths of scanning responses that contain content. A simple counting aggregation of these numbers define the values of the table 5.1. Secondly and for having a meaningful comparison of response header fields during header field evaluation, it is necessary to exclude non standardized and dynamic header fields of table A.1. Moreover the research report provides an domain overview in the supplementals section A. The domains in this table A.2 belong to the evaluation tables 5.2, 5.3, 5.5, and 5.7 respectively.

5.2 Scanning Overview

Before explaining the detection results of IP layer blocking of table 5.2, the general statistics of table 5.1 provide an overview on single scan executions. Here, table 5.1 shows a majority of Tor scans. This is due to the fact that clients continue to scan until they reach three successful responses. As there are more Tor exit nodes than Planet

Table 5.1: General Scanning Statistics

Scan Type	Scans	Success	failure	Timeout	Captcha	No Content
PL	4728 (38.54%)	3148 (25.66%)	1580 (12.88%)	84 (0.68%)	0 (0%)	857 (6.98%)
Tor	7538 (61.45%)	2325 (18.95%)	5213 (42.49%)	3468 (28.27%)	133 (1.08%)	665 (5.42%)
Total	12266 (100%)	5473 (44.61%)	6793 (55.38%)	3552 (28.95%)	133 (1.08%)	1522 (12.4%)

Lab exit nodes, it is obvious that the Tor client rescans more often in cases where web services block incoming requests. This trend moreover justifies the majority of Tor client failures. `Timeout`, `Captcha`, and `No Content` values indicate a first tendency with regard to the first research question. Clear discrimination towards the Tor client is visible. While request timeouts represent IP layer discrimination possibilities, `Captcha` and `No Content` discrimination embody application layer discrimination. Header and status codes of HTTP responses reinforce these discrimination tendencies during a more detailed evaluation in next paragraphs. The fact that the Planet Lab client receives no content within responses more frequently indicates a first remark against dominating numbers of the Tor client. Nevertheless these values might affect the Tor client with additional discriminative content and have to be treated with caution until this point. The following evaluation tables implement a domain based evaluation of scanning results. This allows a direct comparion with other related works and further clarifies these first findings.

After having a general look on single scanning statistics in table 5.1, the scanning evaluations follow two different evaluations. First, there is the investigation of IP-level blocking in section 5.3. Table 5.2 refers to this evaluation strategy. Secondly, there are a domain based evaluations of application layer properties of scanning responses described in section 5.4. Here, evaluation numbers refer to the maximum value of scanned domains. Table 5.5 represents one example of this evaluation strategy.

5.3 IP-Layer Discrimination

Table 5.2 shows IP layer blocking statistics from different locations. The investigation separates timeout resistant Planet Lab requests in the second row of the table. Afterwards, Tor client differentiation is tested on the subset of these requests. The third row of table 5.2 shows percentages between 32.43% and 78.09% of the response subset of valid Tor requests. This means that the opposing quantity of the subset is affected by either timeouts, IP level blocking, or other errors. Values between 6.19% and 16.58% of IP blocking mark higher values compared to detection percentages of 6.8% Khattak and Murdoch's work [15].

Table 5.2: IP Layer Blocking

	DE	FI	FR	NO
Targets	212 (100%)	212 (100%)	212 (100%)	212 (100%)
PL No Timeout	205 (96.69%)	185 (87.26%)	205 (96.69%)	210 (99.05%)
Tor No Timeout	87 (42.43%)	60 (32.43%)	129 (62.92%)	164 (78.09%)
Tor IP Level Blocking	34 (16.58%)	20 (10.81%)	22 (10.73%)	13 (6.19%)
PL Timeout & No Tor Timeout	1 (0.47%)	4 (1.88%)	2 (0.94%)	0 (0%)
PL Errors	7 (3.3%)	27 (12.73%)	7 (3.3%)	2 (0.94%)

Since CDF functions of figure 5.1 help to define IP blocking detection thresholds, these figures moreover show appearances of timeouts which does not label domain scans as blocked. This is due to thresholds which derive from the maximum jump within a CDF function. Nevertheless there is the assumption that other errors such as unsuccessful exit node fingerprint lookups contribute to generally small percentages of valid Tor requests within the Planet Lab valid subset. This assumption derives from the disproportionate gap between Planet Lab and Tor failures of table 5.1. It is obvious that IP layer blocking represents the strongest discrimination possibility. IP layer blocking thereby draws attention to strong discrimination motivations. The fifth entry of table 5.2 reveals Tor client only timeouts. The goal to attract anonymous users could set a motivation for this unusual treatment.

5.4 Application Layer Discrimination

The detailed evaluation of application layer discrimination through the following tables splits requesting sets the following way. Planet Lab requests which do not experience timeouts define the main subset of domains. The second row ot the application layer discrimination table shows the quantity of domains within each country. All percentages in the following tables except the percentages of `Diff Tor Status Code` of table 5.3, `Diff Tor Headers` of table 5.5, and `Diff Tor Status Code & Headers` of table 5.5 refer to the respective domain subset of each country. The percentages of these three exceptions always refer to the domain quantity of the above row. Similar to the already discussed tables, the values of the tables 5.3, and 5.5 table mark HTTP header and content as discrimination possibilities.

Table 5.3: Status Code Application Layer Discrimination

	DE	FI	FR	NO
Targets	212 (100%)	212 (100%)	212 (100%)	212 (100%)
PL No Timeout	205 (96.69%)	185 (87.26%)	205 (96.69%)	210 (99.05%)
Stable Status Code	201 (98.04%)	179 (96.75%)	200 (97.56%)	156 (74.28%)
Diff Tor Status Code	18 (8.95%)	17 (9.49%)	17 (8.5%)	10 (6.41%)
PL Errors	7 (3.3%)	27 (12.73%)	7 (3.3%)	2 (0.94%)

Table 5.3 has lower discrimination percentages than the related work of Rachee Singh et al. [24]. The numbers of this research project vary between 6.41% and 9.49% while discrimination numbers of application layer responses of Rachee Singh's work [24] 20.03% reach. On the contrary, Khattak and Murdoch's older work [15] detects 3.54% of application layer discrimination. Outcomes of discrimination occurrences of this research project are situated between these other outcomes. An explanation for this smaller variance is the scanning approach. The approach to have timely stable status codes already creates a subset the complete domain set. As a consequence the numbers of this work fall under the detections of Rachee Singh's work [24]. Nevertheless first answers to the first research question appear. Status codes deliver a possibility to discriminate. The next table 5.4 takes a deeper look into status code evaluation.

Table 5.4: Differing Status Code Counts

PL status code	Tor status code	Count
HTTP/1.1 301 Moved Permanently	HTTP/1.1 503 Service Unavailable	3
HTTP/1.1 200 OK	HTTP/1.1 403 Forbidden	7
HTTP/1.1 302 Found	HTTP/1.1 403 Forbidden	4
HTTP/1.1 301 Moved Permanently	HTTP/1.1 200 OK	9
HTTP/1.1 301 Moved Permanently	HTTP/1.1 403 Forbidden	29
HTTP/1.1 302 Moved Temporarily	HTTP/1.1 301 Moved Permanently	4
HTTP/1.1 302 Found	HTTP/1.1 301 Moved Permanently	3

When Looking at the status code table, there is the 301 moved permanently status code dominating the occurrences. The measurement proxy tool and the Tor proxy connection do not follow redirects which explain the majority of redirections. Both papers of Khattak and Sheharbano [14] and Khattak and Murdoch [15] focus and evaluate response codes with regard to the 200 OK status code. These works treat redirect status codes as unblocked requests. This work follows the approach of counting redirect codes as non discriminative measures for the client. But the 403 and 503 status codes show discriminative behavior towards clients. The 503 service unavailable response only affects the

Tor client and shows discrimination against Tor users only. The fourth column with a 301 status code on Planet Lab side and a 200 status code on Tor side indicates clear discrimination directed to Planet Lab only. This example shows discrimination towards regular clients. This is an important finding concerning the third research question. The main trend of redirect receptions comes with the scanning approach of not following redirects. In contrast to this research project, the work of Rachee Singh et al. [24] follows redirects, search, and login requests. It thereby implements the deepest response content analysis compared to this research work and Khattak and Murdoch's work [15] which evaluates 200 status codes particularly.

Table 5.5: Header Application Layer Discrimination

	DE	FI	FR	NO
Targets	212 (100%)	212 (100%)	212 (100%)	212 (100%)
PL No Timeout	205 (96.69%)	185 (87.26%)	205 (96.69%)	210 (99.05%)
Stable Headers	195 (95.12%)	179 (96.75%)	196 (95.6%)	156 (74.28%)
Diff Tor Headers	37 (18.87%)	27 (15.08%)	36 (18.27%)	24 (15.38%)
Stable Status Code & Headers	196 (95.12%)	179 (96.75%)	196 (95.6%)	156 (74.28%)
Diff Tor Status Code & Headers	43 (21.93%)	34 (18.99%)	41 (20.81%)	27 (17.3%)
PL Errors	7 (3.3%)	27 (12.73%)	7 (3.3%)	2 (0.94%)

Higher percentages of different Tor headers within a relatively large subset of Planet Lab requests indicate a higher extent of differentiation. HTTP header fields always vary up to almost 20% of compared domain samples. They are thereby in line with the discrimination findings of Rachee Singh et al.'s work [24]. The comparison of the combination of status code or header differentiation further enforces the trend. This clearly separates response header investigation from IP layer blocking and content discrimination evaluation. Percentages of blocking, CAPTCHAs, and timeouts do not cover a similarly large subset. This means that differentiation extent of header information demonstrate a very volatile behavior. As a consequence, there is the least chance to find discrimination motivations through the investigation of headers. There are to many other factors contributing to the different appearance of headers which drastically reduces the probability of unequal headers due to consistent dicriminative intentions only. In other words, there are almost no indicators about discrimination motivations within HTTP headers. The smaller percentages of status codes are more probable to reveal discriminative measures and motivations. It is very important to remember that the difference of the headers is due to the anonymous requesting client. Since the scanning approach ensures location conforming requests while preventing dynamic response behavior. The evaluation strategy makes sure that the evaluation direction towards the anonymous client clearly indicates discrimination.

The next table 5.6 extends the evaluation of header fields which vary in Tor reponses.

Table 5.6: Unequal Header Parameters

Header Field	Count
Content-Length	35
Content-Security-Policy	4
Keep-Alive	10
Server	33
Srv	8
Content-Type	28

Dominating numbers indicate `Content-Length`, `Content-Type`, and `Server` as the most volatile header fields. Interestingly these numbers point to a very strong variance in responses bewtween regular and anonymous clients. A different content type shows a very strong extent in differencing responses. The amount of such samples is rather low with regard to the total number of evaluations. Nevertheless these findings suggest very specific and accurate discrimination targeting.

Table 5.7: Captcha and Content Length Discrimination

	DE	FI	FR	NO
Targets	212 (100%)	212 (100%)	212 (100%)	212 (100%)
PL No Timeout	205 (96.69%)	185 (87.26%)	205 (96.69%)	210 (99.05%)
Significant Content Length	355.27 (72.68%)	356.12 (68.1%)	356.59 (71.7%)	360.66 (53.33%)
Significant Tor Content Length	3017.8 (70.73%)	2301.65 (60.54%)	2875.04 (71.21%)	2293.72 (67.61%)
Captcha	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Tor Captcha	13 (6.34%)	12 (6.48%)	12 (5.85%)	12 (5.71%)
PL Errors	7 (3.3%)	27 (12.73%)	7 (3.3%)	2 (0.94%)

Numbers about CAPTCHA detection and content length contribute to outcomes of the third research question. Namely CAPTCHAs only appear in Tor client responses. This shows a clear use case of a discrimination technique against one client side. Redirects deliver explanations about the fact of less Planet Lab content. The majority of temporary and regular redirects within table 5.4 together with the `No Content` values of table 5.1 indicate shorter responses. Since redirects usually contain no or very little content. The fact that redirects of popular websites follow properties such as locational personalization [16] reinforces the assumption that redirects represent regular behavior in these cases.

Summing up and regarding the first research question of discrimination possibilities and motivations, the collected responses show discrimination possibilities within response headers, response content, and web service respondence. Tor headers have the

highest extent of differentiation with regard to stable and location conforming Planet Lab response headers. Header status codes can deliver valuable information regarding client discrimination such as the 403 forbidden status code. But it is difficult to figure out intentions and motivations of web services. It is evident that both sides become discriminated respectively when looking at IP layer blocking, HTTP header, and content comparisons. Certainly the main discrimination trend addresses the Tor client side more often. This outcome meets the expectations since an anonymization network opens more ways for abuse and fraud than a regular communication network. Therefore it is obvious that web services perform precaution rather against Tor clients. This finding is conform with other research where Tor blocking percentages top blocking of regular requests.

5.5 Discussion

The evaluated findings and outcomes deliver answers to the investigated research questions. Moreover these findings extend the methodical study provided by Khattak and Murdoch's work [15] because of deeper website content investigation. Nevertheless flaws and unexpected problems invite to further discussion.

One of the most severe flaws is the attained locational coverage of scanning nodes. The coverage of only four european countries reduces on one side the significance of the research project. On the other side, working Tor exit nodes have no use in countries where discrimination is a documented fact [26]. Unfortunalty during the time of this research project, Planet Lab suffered an extended outage which took longer than a month. Planet Lab came back online on 29th May 2017. This outage might affected functionality of the network in following months. Khattak and Murdoch's solution [15] of the controlled proxy OONI network fulfil the research requirements of a distributed controlled scanning network. A strong suggestion for future research is to come up with an alternative proxy network. The delay of non availability of the Planet Lab network moreover prevents the implementation of intended tasks such as HTTPS scanning.

Even though the scanning approach excluded the possibility of comparing a locational unconform and timely unstable Planet Lab response with a Tor response, this approach slows down the scanning process per domain drastically. With 212 scanned domains of the Alexa top 1 million list, this number falls below the aimed number of scanned domains. The problem of continuing to search for three successful responses leads to an accumulation of timeouts in the case of a blocking web service. This problem leads to very long scanning delays. In contrast to the related work, the comparisons

of Tor versus non Tor responses of this research project do not contain comparisons between locational and timely distorted responses. Another problem of the related work of Khattak and Murdoch [15] of temporary unavailability of web services could not occur in the scanning approach of this work. The approach of evaluating Planet Lab responses before the comparison with Tor responses detects unavailability during Planet Lab response evaluation.

The strategy to always take the same entry guard when building a Tor circuit sometimes resulted in unavailability of this entry guard during scanning. In cases of blocking or offline behavior of the chosen entryguard a new one had to be chosen. Moreover it was necessary to sort out all scans with this type of entry guard connection error. An explanation for this entry guard behavior could be the entry guard operator's detection and countermeasure against scanning intentions and traffic. An optimization with regard to detect non working entry guards enhances scanning script reliability and robustness.

Lastly, as there was not enough time to implement concurrent scanning execution in the scanning script, a future suggestion is to improve the scanning behavior with regard to speed and faster response collection.

Chapter 6

Related Work

The intention of this chapter is to compare this work with three other scientific works which relate to the discussed topic. As a reference to this work, these three works cover the general sources of second class treatment of Internet citizens due to anonymity networks, show how user's geo-locational properties affect resulting discriminative responses, and point out the directions that discrimination can target. This comparison clearly illustrates where this work differs, equals, and extends scientific research fields of client discrimination in anonymization networks using active scans.

6.1 Second Class Treatment of Internet Citizens

The increase of popularity of anonymity networks calls for stronger analysis of effects caused by these networks. Research about anonymity networks has focused mainly on the flaws and implementation techniques of anonymity networks. Newer research especially investigates de-anonymization techniques [13]. Therefore every new development of further upgrading anonymity in these networks enjoys high attention. With less attention but with the same research importance comes the investigation of client discrimination. The importance of this research branch is obvious because HTTP which is the fundamental web service protocol is the mostly used request format in anonymity network traffic [12]. Thereby HTTP usage indicates an increase of anonymity network usage by average users. The more usage the Tor anonymity network gains by average users the more people notice discrimination treatment as Internet citizen [3]. Therefore it can be assumed that properties of anonymity networks which affect Internet users in degrading manners will receive more and more attention and investigation.

The first related work paper investigates the second class treatment of Tor users. The paper of Khattak and Murdoch [15] focuses on discrimination performed on the application layer as well as the network layer. On the network layer, port scanning of the entire IPv4 address space shows reset and dropped connections of web-accessible services. The work thereby distinguishes intentional censorship events from incidental networking failures. These conducted scans origin from active Tor exit nodes as well as non-Tor control nodes. Overall 1.3 million addresses in the IPv4 address space either block or degrade their service to Tor users.

Discrimination possibilities found at the application layer have higher relevance to our project. Here, Khattak and Murdoch's work [15] fetches home pages from Alexa's top 1,000 websites and analyzes the responses. The study of Khattak and Murdoch does not consider negative Tor discrimination. This means it counts unblocked Tor and blocked non-Tor requests as unblocked requesting pairs. It moreover finds reasons and blocking techniques concerning the question whether discrimination follows automated abuse-based or general Tor usage blocking. It methodically uncovers factors which degrade Tor users and addresses the problem of characterizing the nature of blocking of anonymity networks at scale. Considering requesting pairs web services unblock 84.4% of both requests whereas they block 6.8% of only Tor requests, 1.8% of non-Tor requests, and 7.1% of both requests. Transport layer investigations assign 0.45% of the 6.8% to only Tor blocked requests due to rejects and 2.82% due to timeouts. Reasons for blocking have the intention to automatically reduce abuse and follow government policies.

Header status code, message body, and timeout information of the HTTP response packets indicate discrimination possibilities of the first related work. This can be seen in the analysis technique which compares and categorizes scanning responses with the help of time and application relevant factors in responses.

Similar to Khattak and Murdoch's work [15] status header, message body, timeout, and error information are the factors which embody reasonable comparison possibilities in our work. This project moreover focuses more on application layer discrimination than on IP layer blocking. Apart from that, our project covers not the same target address space on application level. Here, the Alexa top 1 million list provides the targets. Another difference of our work is the exploration and evaluation of both blocked non-Tor requests and blocked Tor requests. Similar are the following limitations which affect all stated research projects similarly. The time dependent availability of web services reduces the amount of useful responses. The biggest difference is within the scanning approach. This research project unveils timely stable and locational conform web responses before comparing responses with responses collected over the Tor network. This measure makes sure that no other external factors distort the comparison of controlled

node responses and Tor node responses. Apart from that and on the contrary to Khattak and Murdoch's work [15], this project covers the investigation of deeper features of web pages such as CAPTCHAs, header status evaluation, and content evaluation.

Before finishing the project, Rachee Singh et al. [24] published another paper about the nature and dynamics of Tor exit blocking. This paper also extends the work of Khattak and Murdoch's work [15]. The extension thereby is the measurement of deeper features of websites such as blocking of login and search functionality. The work of Rachee Singh et al. [24] simultaneously performs website scanning of the Alexa Top 500 list from all Tor exit nodes and a controlled university host. Thereby they conduct front-page, search functionality, and login functionality website crawls. Results state 20.03% of website front-page loading discrimination. 3.89% of the 17.44% of search-compatible website loads shows increased discrimination. In addition, 7.48% of the 17.08% of the login-compatible website loads shows increased discrimination. Similar to the work of Rachee Singh et al. [24], the research project also performs deeper feature detection of website responses. On the contrary the project inspects the first response in more detail and does not follow redirects or even login or search loads. Another difference is scanning execution from a controlled network for receiving comparable responses. For another clarification regarding scanning strategies, Khattak and Murdoch's [15] short time experiment scans Alexa URLs from all Tor nodes and one controlled node for a period of five days. The other scanning strategy collects results over a year. Thereby paired Tor and non-Tor scans collect data with the help of Exitmap and Stem on Tor side and the Open Observatory of Network Interference (OONI) network on non-Tor side.

6.2 Location Based Discrimination

The second related work proves that geolocation properties of website queries matter. The paper of Kliman-Silver et al. [16] investigates the question if location based personalization cause differences in search results. It therefore queries websites with identical requests of controversial topics from different locations. The Jaccard Index and the Edit Distance enable to extract reasonable information for discrimination out of the HTTP responses. The Edit Distance calculates the number of necessary additions, deletions, and substitutions for rebuilding identical responses. The Jaccard Index represents a value indicating overlap between the queried responses. Another point is that the second related work uses identical queries with the exact same GPS coordinates from 50 different Planet Lab machines across the US. This measure proves the dependence of search results on GPS coordinates rather than IP addresses.

Findings of the work show that location-based personalization causes more differences in search results than any other feature. High average Jaccard indications on political and controversial search request topics from national, state, and county areas differ from local search request topics which have a remarkable lower Jaccard Index. Consistency show the results of the Edit Distance with high values on local search request topics and very low numbers for political and controversial search subjects. State and national originating requests thereby provoke additional varying numbers. Especially local search request topics have 2 times higher Edit Distances when send with state and national granularity. Consequently requests with county granularity have an higher Jaccard Index.

According to these findings the usage of the Planet Lab network allows to reduces the location based differences between controlled and Tor exit nodes. Our work moreover compares equal responses with responses provoked through the Tor network. But before doing so, our project filters and extracts locations with low variation of response content with the help of Planet Lab similarly.

6.3 Angle of View on Discrimination

The third and most comprehensive related work of Khattak and Sheharbano [14] analyzes user side discrimination as well as publisher side discrimination. User side censorship blocks or discriminates the user's online communication. In the case of publisher side discrimination, web services or publisher refuse to respond based on certain properties of the user's request. The authors use Tor and with adblocking software equipped users as examples of publisher side discrimination victims. Characterization of the practices of the study derive from requests of the Alexa top 5 thousand list. Results state that on average 3.67% of Alexa's top 1K websites respond with a non-200 status header message when visited through a Tor exit node. Also 6.7% of Alexa's top 5K websites conduct anti-adblocking discrimination.

The analysis and exploration of the publisher side censorship is the main objective of our study. As our work does not cover such wide fields as the last related work, it gives a more extended view on user discrimination details. Because the usage of the Alexa top 1 million list and finer comparison characteristics further improve client discrimination evaluations.

Chapter 7

Conclusion

The strategy to provoke timely stable and location conforming HTTP requests is dominated by the interaction of single infrastructure components. Non availability of exit nodes thereby results in a fast reduction of locational coverage. Next to infrastructur setup, the scanning approach has to prevent all external factors which could influence discrimination detection. Strictly following these requirements enables to collect website responses which show anonymous client directed discrimination.

Detailed analysis of reponse fields indicate discrimination not only against anonymous clients. Discrimination in HTTP responses affect anonymous and regular clients. The exploration of the data reveals discrimination possibilities in IP layer blocking, HTTP header, and content implementation. The extent of differentiation of responses varied between from equal to responses containing varying content types. CAPTCHAs, reduced or no content, and differing status codes mark implementation differences in affected responses. Discrimination percentages of up to 18% in application layer responses support outcomes of recently published related research of Rachee Singh et al. [24]. IP-layer blocking percentages on the contrary reveal with up to 16.58% increasing IP-layer blocking numbers compared to older research percentages of 3.54% of Khattak and Murdoch [15]. Strongly varying responses do not reveal discrimination motivations. Anonymous internet clients mark the majority of discriminated users while discrimination against regular users exists.

This work extends Khattak and Murdoch's work [15] with a deeper analysis of HTTP responses. Moreover it ensures to detect discrimination with the help of a specifically developed scanning approach which ensures location conforming and timely stable requesting pairs. The strategy enables clear identification of discrimination directed

to anonymous clients. The work of Rachee Singh et al. [24] implements a more detailed HTTP response feature analysis by investigating login, search, and front-page properties. It does not implement location conforming requesting pairs and focuses on application layer discrimination only.

The following last subsection 7.0.1 highlights the research limitations and provides future work suggestions.

7.0.1 Future Work

This section suggests future research and further optimization with regard to the main points revealed in the discussion section 5.5 and limitations of section 3.2 of the project.

The continental unequal distribution of Planet Lab network nodes and the inability of IPv6 Planet Lab requests leaves space for alternative proxy networks. Better support for IPv6 and ICMP can lead to better measurements. But unfortunately neither Tor nor Planet Lab provide support. Further investigations with proxy networks which provide functionality of the described capabilities would allow to extend client discrimination detection. Another measure of revealing more detailed and specific discrimination occurrences is the approach to follow HTTP redirects. Further investigation of application layer features such as searches, content contributions and logins can detect increasing or decreasing discrimination implementations.

The improvement of the scanning approach could provoke consecutively closer Tor and measurement proxy requests. An implementation suggestion is a global data structure which keeps track of all built Tor circuits. Similar to the implementation of the project [15]. Following this intention means to build and store all Tor relay circuits before HTTP request execution. Measurement proxy and Tor request could become executed consecutively closer due to the missing circuit build timespan in between the scans. Advantages of this implementation reduce architectural complexity. But drawbacks such as unforeseeable teardown of circuits through any circumstances of this suggested implementation question if this implementation approach leads to a more robust technique of catching dynamic content. However, backup circuits could help to solve these problems. Apart from that, the following scanning script suggestions will extend this research project. Requesting HTTP messages with different User-Agent headers allows to imitate and thereby cover more internet clients. Impersonation of more clients leads to a wider client base for discrimination detection. Another open option is the improvement of the scanning script with regard to concurrent scanning. This would enable faster scanning and reduce the probability of detecting dynamic content in website reponses.

Lastly further investigations about data collection and evaluation could improve the findings about the differences present in responses of web services wich allow discrimination. Due to the fact that performing successful scans took more time than expected, this project did not collect and investigate HTTPS certificates which potentially deliver discrimination indications. Additionally, following request redirects went beyond the scope of this project and mark a future research field.

Chapter A

Supplementals

Table A.1: Excluded Header Fields during Header Field Comparisons

Date	X-Timer	Referer
Location	X-Request-Guid	X-Frontend
Expires	X-C	X-Served-By
Set-Cookie	X-Li-Pop	X-Cache
Referrer-Policy	X-Via	X-Varnish
Age	X-Amz-Cf-Id	Cache-Control
Timing-Allow-Origin	X-Dropbox-Request-Id	Last-Modified
Request-Id	X-Cache-Hits	P3P
X-Client-Ip	X-Ac	X-Akamai-Transformed
X-Response-Time	X-Connection-Hash	X-Netflix.instance-Status
X-Server-Id	X-Aspnet-Version	Xxn
X-Farmid	X-Api-Version	X-Dis-Request-Id
X-Request-Id	X-Fb-Debug	X-Xss-Protection
X-Cacheable	X-Li-Proto	X-Vserver
X-Officefd	X-Frame-Options	X-Instart-Request-Id
X-Correlationid	X-Usersessionid	X-Powered-By
X-Readtime	X-Li-Uuid	X-Fastly-Request-Id
X-Content-Digest	P3p	Eagleeye-Traceid
X-Msedge-Ref	X-Dns-Prefetch-Control	Cf-Chl-Bypass
Via	Cf-Ray	X-Varnish-Cache

Table A.2: Discriminationg Domains Overview

Tor IP Level Blocking	Diff Tor Status Code	Diff Tor Headers	Tor Captcha
360.cn	amazon.co.uk	9gag.com	adf.ly
adexchangeprediction.com	amazon.de	amazon.co.uk	amazon.co.uk
alipay.com	amazon.in	amazon.de	amazon.de
amazon.co.uk	bongacams.com	amazon.in	amazon.in
amazon.in	chaturbate.com	apple.com	chaturbate.com
askcom.me	craigslist.org	chaturbate.com	coccoc.com
baidu.com	getmyads.com	detik.com	getmyads.com
cctv.com	mediafire.com	diply.com	ntd.tv
chaturbate.com	netflix.com	getmyads.com	openload.co
china.com	ntd.tv	google.com	porn555.com
china.com.cn	openload.co	huffingtonpost.com	thepiratebay.org
chinadaily.com.cn	porn555.com	netflix.com	txxx.com
csdn.net	thepiratebay.org	ok.ru	upornia.com
detail.tmall.com	txxx.com	pixnet.net	yandex.ru
deviantart.com	upornia.com	quora.com	
github.io	weibo.com	steamcommunity.com	
google.com.eg	wittyfeed.com	washingtonpost.com	
googleusercontent.com	yandex.ru	weibo.com	
hao123.com	X-Dns-Prefetch-Control	yelp.com	
huaban.com	Cf-Ray	zhihu.com	
jd.com	yelp.com	xhamster.com	
list.tmall.com		xinhuanet.com	
onlinesbi.com			
qq.com			
roblox.com			
savefrom.net			
service.tmall.com			
sina.com.cn			
so.com			
sohu.com			
soso.com			
steamcommunity.com			
steampowered.com			
taobao.com			
tianya.cn			
tmall.com			
twimg.com			
weibo.com			
xinhuanet.com			
youth.cn			
zhihu.com			

Chapter B

List of acronyms

API	Application Programming Interface.
AS	Autonomes System.
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart.
CDF	Cumulative Distribution Function.
DNS	Domain Name System.
HTML	Hypertext Markup Language.
HTTP	Hypertext Transfer Protocol.
HTTPS	Hypertext Transfer Protocol Secure.
IPv4	Internet Protocol Version 4.
IPv6	Internet Protocol Version 6.
JSON	JavaScript Object Notation.
OONI	Open Observatory of Network Interference Network.
RPC	Remote Procedure Call.

Bibliography

[1] Alex Biryukov et al. "Content and popularity analysis of Tor hidden services". In: *Distributed Computing Systems Workshops (ICDCSW), 2014 IEEE 34th International Conference on*. IEEE. 2014, pp. 188–193.

[2] Brent Chun et al. "Planetlab: an overlay testbed for broad-coverage services". In: *ACM SIGCOMM Computer Communication Review* 33.3 (2003), pp. 3–12.

[3] Roger Dingledine. "A call to arms: Helping Internet services accept anonymous users". In: *Tor Project Blog* (2014).

[4] Roger Dingledine, Nick Mathewson, and Paul Syverson. *Tor: The second-generation onion router*. Tech. rep. DTIC Document, 2004.

[5] Lucas Dixon, Thomas Ristenpart, and Thomas Shrimpton. "Network Traffic Obfuscation and Automated Internet Censorship". In: *IEEE Security & Privacy* 14.6 (2016), pp. 43–53.

[6] Gordana Dodig-Crnkovic. "Scientific methods in computer science". In: *Proceedings of the Conference for the Promotion of Research in IT at New Universities and at University Colleges in Sweden, Skövde, Suecia*. 2002, pp. 126–130.

[7] Steven Englehardt and Arvind Narayanan. "Online tracking: A 1-million-site measurement and analysis Draft: July 11th, 2016". In: ().

[8] R Fielding et al. *RFC 2616-HTTP/1.1, the hypertext transfer protocol*. 1999.

[9] David Fifield et al. "Blocking-resistant communication through domain fronting". In: *Proceedings on Privacy Enhancing Technologies* 2015.2 (2015), pp. 46–64.

[10] Glenn Greenwald and Ewen MacAskill. "NSA Prism program taps in to user data of Apple, Google and others". In: *The Guardian* 7.6 (2013), pp. 1–43.

[11] John Holowczak and Amir Houmansadr. "CacheBrowser: Bypassing chinese censorship without proxies using cached content". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2015, pp. 70–83.

[12] Markus Huber, Martin Mulazzani, and Edgar Weippl. “Tor HTTP usage and information leakage”. In: *IFIP International Conference on Communications and Multimedia Security*. Springer. 2010, pp. 245–255.

[13] Rob Jansen et al. *The sniper attack: Anonymously deanonymizing and disabling the Tor network*. Tech. rep. DTIC Document, 2014.

[14] Sheharbano Khattak. *Characterization of Internet censorship from multiple perspectives*. Tech. rep. University of Cambridge, Computer Laboratory, 2017.

[15] Sheharbano Khattak et al. “Do you see what i see? differential treatment of anonymous users”. In: *Network and Distributed System Security Symposium*. 2016.

[16] Chloe Kliman-Silver et al. “Location, location, location: The impact of geolocation on web search personalization”. In: *Proceedings of the 2015 ACM Conference on Internet Measurement Conference*. ACM. 2015, pp. 121–127.

[17] Peter Kublickis. *System and methods for a micropayment-enabled marketplace with permission-based, self-service, precision-targeted delivery of advertising, entertainment and informational content and relationship marketing to anonymous internet users*. US Patent App. 11/118,998. 2005.

[18] Karsten Loesing. *Counting daily bridge users*. Tech. rep. Technical Report 2012-10-001, Tor Project, Oct. 2012. https://research.torproject.org/techreportscounting-daily-bridge-users-2012-10-24.pdf, 2012.

[19] Google Maps. *Python Client for Google Maps Services*. `https://github.com/googlemaps/google-maps-services-python`. Accessed: 2017-04-26. 2017.

[20] Damon McCoy et al. “Shining light in dark places: Understanding the Tor network”. In: *International Symposium on Privacy Enhancing Technologies Symposium*. Springer. 2008, pp. 63–76.

[21] Marti Motoyama et al. “Re: CAPTCHAs-Understanding CAPTCHA-Solving Services in an Economic Context.” In: *USENIX Security Symposium*. Vol. 10. 2010, p. 3.

[22] Larry Peterson et al. “Planetlab architecture: An overview”. In: *PlanetLab Consortium May* 1.15 (2006), pp. 4–1.

[23] *PlanetLab Europe*. `https://www.planet-lab.eu/`. Accessed: 2017-04-26.

[24] Rachee Singh et al. “Characterizing the Nature and Dynamics of Tor Exit Blocking”. In: (2017).

[25] Inc. The Tor Project. *Tor Metrics Onionoo*. `https://metrics.torproject.org/onionoo.html`. Accessed: 2017-08-24.

[26] Philipp Winter and Stefan Lindskog. “How china is blocking Tor”. In: *arXiv preprint arXiv:1204.0447* (2012).