Politecnico Di Torino

Master’s Degree in Computer Engineering

Development of a tool for the automated analysis and reporting of personal data transfers to non-EEA domains

Supervisor
prof. Antonio Vetrò

Candidate
Lorenzo Laudadio

Academic Year 2022/2023

Summary

The thesis project which we describe in this document is divided in two main parts.

The goal of the first part was to develop a software tool which could help users identify illegal HTTP requests directed to non-EEA (European Economic Area) domains while navigating the web and then generate claims which could be submitted to the Italian supervisory authority, namely *"Garante per la Protezione dei Dati Personali"*.

The goal of the second part was to extend the work by creating a set of automatic scripts which could automatically navigate through websites taken from a list and identify illegal request directed to non-EEA domains, and then run an analysis on the Italian public administration websites by using those scripts, possibly building a visual representation of the obtained results.

First of all, we focused on the legal aspects, to understand when a certain data transfer from the European Economic Area to an external country is illegal. We identified the GDPR articles which regulate personal data transfers to third countries. In particular, article 45 of the GDPR specifies that the European Commission has the power to issue an *adequacy decision* about a country which is outside the EU/EEA. When such a decision is issued, it means that personal data can flow freely to that specific third country. Articles 46 and 49 specify other cases in which data can be transferred, i.e. when additional safeguards are available or on the basis of some derogations.

We considered as a specific case study the EU to US data transfers, since the past adequacy decisions about the US have been invalidated multiple times from the Commission of Justice of the European Union (CJEU), and recently another adequacy decision has been adopted for the data transfers to the US under the new EU-US Data Privacy Framework. This decision implies that, according to the European Commission, the US ensures an adequate level of protection for personal data transferred from the EU to US companies under the new framework. Several doubts have arisen after the decision, since it is the third time that an adequacy decision for the data transfers to the US is issued, but it seems that there is no substantial change in the internal US law from the last times, in particular for what concerns the mass surveillance on European citizens. In fact, section 702 of the FISA Amendments act of 2008, gives the Attorney General and the Director of National Intelligence the power to allow targeting of non-US citizens which are believed to be located outside the United States. This specific act was considered incompatible with the Charter of Fundamental Rights of the European Union, and therefore there are grounds to believe that the CJEU may invalidate the adequacy decision this time as well.

After considering the legal scenario, we moved to the development of the application, which was written in JavaScript using the Electron cross-platform framework. For the

development we adopted the best practices from the software engineering world. We first formally defined all the requirements, the architecture and components, and then we started with the implementation. All the implementation choices have been discussed within this thesis, with their pros and cons. Basically, our application consists of a GUI which allows the users to navigate the web, a component which captures and analyzes the HTTP traffic, in order to detect requests addressed to domains belonging to companies located in countries for which no adequacy decision has been issued, and creates a HAR (HTTP Archive) log, and another component which builds the claim. To keep a serverless approach, we rely upon an internal *blacklist* which contains all the third-countries domains. The GUI and the various components can communicate thanks to the Electron Inter-Process Communication channels. We decided that the HAR format was the best choice for storing logs, despite its limitations. The software was tested both by MonitoraPA developers and by end users.

We believe that the value of this software lies in the fact that it is targeted to non-advanced users, which will have the possibility to check easily and independently whether their personal data are being transferred to third countries, without the need for a server. We made a comparison with other applications, but we did not found any software with the same characteristics (ease of use, serverless approach, open source).

Then we built *minos-cli*, the set of scripts targeted to analysis and visualization, by employing a mix of different languages and libraries. From the logical point of view, the approach was quite the same as for the GUI version: we record the network traffic, create a HAR log and analyze the HTTP requests to detect the illegal ones. The difference here is that everything is automatic, i.e. there is no explicit interaction of the user with the webpages, which are known at the beginning. Moreover, since the goal was to use these scripts to run a mass analysis, we had to make some implementation choices to reduce the time and space overhead. For example, we chose not to store all the HAR object in separate files, but we just kept a single file with the list of all the illegal requests, as having one separate file per each website would have required a lot of space and a much longer time of analysis.

By using the tools we had just built, we ran the analysis on a set of 22890 public administration websites from Italy, and we computed some statistics on the results. The reason why we are targeting the public administration is quite simple: unlike private companies, here we have a free access list of all the organizations involved which is updated daily.

We found many entities not respecting the GDPR articles on personal data transfers: more than 15% of the entities were detected making requests to domains of companies located in countries lacking an adequacy decision.

We divided the entities by category, and we discovered that most of the violations occurred in the categories *Comuni e loro Consorzi e Associazioni* (i.e. Italian municipalities) and *Istituti di Istruzione Statale di Ogni Ordine e Grado* (i.e. Italian schools), with respectively more than 1100 and more than 600 violations. This result is rather alarming, considering that these kind of entities process very sensitive data (citizens and scholars' data).

Then we grouped the requests that we found by domains and domain groups, to have an idea of which services are the most requested. We found that the most popular domain is `.amazonaws.com`, which is associated to cloud computing services offered by Amazon, with almost 6000 requests, i.e. more than 41% of the total number of requests. In the second place we have `.fontawesome.com`, an icon library and toolkit, with about 1000 requests (~7%), and then all the others, with less than 1000 requests each.

With 41.97% of requests, Amazon holds the record as the most popular company. Then we have Google, with 17.03%, and Fonticons, with 12.90%. The most requested services are cloud computing (42.22%), cdn (26.52%) social media (7.15%) and icons (7.14%).

These results highlight a discrepancy between the GDPR goals and the behaviour of the PA's websites, which still keep transferring personal data to third countries without the proper legal basis.

Contents

«Who watches the Watchmen?»
- Alan Moore, Watchmen (1986-1987)

Chapter 1

Introduction

1.1 Data is the new currency

The technological progress of humanity in the last few decades has been completely oriented towards the concept of information.

In today's digital and connected world, data is the new currency. Everything is built upon data, and everything can be turned into data, from the mathematical models which underlie the meteorology, to the performance of an automobile's braking system. Most importantly, now more than ever, identities of people can be turned into data.

We have several tools available for generating and sharing our personal data (sometimes unwillingly). These pieces of information are collected and processed, and contribute to forming what is called the *digital identity*. Privacy is the ability of an entity to control and possibly conceal these information about itself.

1.2 Privacy nowadays

Privacy is a topic which has become increasingly more important in recent years, in which two of the biggest data-scandals of all time occurred. In 2013 we have witnessed Edward Snowden's revelations [1] about the mass surveillance programs conducted by the US government with the help of big tech companies [2], which have been the focus of public debate for many time. Then, in 2018, it was the turn of the Facebook - Cambridge Analytica scandal [3], when the world discovered that personal data belonging to millions of Facebook users have been collected without their consent by the British company Cambridge Analytica, and that they have been used to influence people's political opinions. This is the evidence that your personal data can tell very much about you.

1.2.1 Users' perception

Even if superficially almost all of us seem to be concerned with privacy, not everyone seems to be aware of its rights and duties. A 2020 survey by the European Union for Fundamental Rights reveals that 41% of the European users does not want to share any personal data with private companies, but only 22% always read terms and conditions when using online services, while only 51% are aware that they can access their personal data held by organizations [4].

The lack of awareness may be caused simply by disinformation, but also from deeper and more subtle psychological phenomena. The study *Facebook and Online Privacy: Attitudes, Behaviors, and Unintended Consequences* shows that the gratifications of using Facebook tend to outweigh the perceived threats to privacy [5]. This is often also related to the so-called *third-person effect*, that occurs when people expect mass media to have a greater effect on others than on themselves.

1.2.2 Companies and organizations

If users lack self-awareness, at the same time companies and organizations are not performing very well when it comes to complying with the privacy regulations.

The GDPR Enforcement Tracker [6], which keeps a list of the fines that data protection authorities within the EU have imposed under the EU General Data Protection Regulation, shows that after five years of GDPR the number of fines in Europe is constantly increasing. Moreover, Italy is in second place for number of fines in Europe, while it is in fourth place for total sum of fines.

In the meantime new privacy issues and threats are arising, especially with the ongoing march of all the AI-related technologies, which can use personal data to analyze, forecast and influence human behavior.

In the case of companies, the failure to apply the privacy principles may be due to a lack of knowledge, but in some other cases it can also be voluntary. Many companies in fact profit from selling users' personal data, therefore they may be willingly collecting and processing user data by using methods which are non compliant with the GDPR principles.

1.3 A respectful use of personal data

In this complex scenario, we cannot rely only on people and organizations' awareness of privacy, and here is why we need regulations. At the same time, experience says that regulations are not applied if there is not enough awareness. Both of them are necessary.

We are not against the personal data collection and processing by the companies, we simply believe that those collection and processing should be carried out respectfully.

In this thesis work we want to focus on a specific privacy problem: the personal data transfers from the European Economic Area (EEA) to countries which are not subject to an adequacy decision. The GDPR prohibits personal data transfers to those countries, since in most cases their internal law is not in line with the Charter of Fundamental Rights of the European Union. Therefore, such transfers should be considered disrespectful to the EU citizens.

We will first give an overview of the legal context in which we are. Then we will present the development of our software tool which is able to detect unlawful data transfers. At the end, we will show the results of an analysis on personal data transfers from the Italian public administration websites to the countries which are outside the EEA.

Chapter 2

Motivation

This chapter presents all the legal background necessary to understand how transfers of personal data to non-EU and non-EEA countries should be protected according to the GDPR.

Special attention is given to the data transfers to the US, since most of the hosts considered during the analysis are from the US, and there is a legal back-and-forth between the EU and US which has been going on for years and now seems to have reached a new turning point, with the approval of the Data Privacy Framework.

2.1 GDPR

The General Data Protection Regulation 2016/679 ("GDPR") is the EU regulation which addresses data protection and privacy in the EU and EEA (Europan Economic Area), and the transfers of personal data outside the EU and EEA [7]. We will now clarify some important concepts which are useful to understand the legal context in which we find ourselves.

2.1.1 Art. 4 - Personal Data

The Article 4 of the GDPR presents some definitions which are used throughout the regulation. In particular, the GDPR defines *personal data* as *any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person.* As you can see the definition of personal data is quite wide. There is no doubt that the IP address can be considered as "personal data", since it can lead to the identification of a natural person. Therefore, when we make a request to a certain domain we are indeed sharing personal data with it.

2.1.2 Art. 28

The Article 28 of the regulation states that when the data processing has to be done by processors on behalf of the data controller, the controller should choose only processors which provide *sufficient guarantees to implement appropriate technical and organizational measures* such that processing meets the requirements of the GDPR and ensures the protection of the rights of the data subject [8]. Also, the data processor should treat personal data only when there is a documented instruction of the controller, even if data is transferred to third countries.

2.1.3 Arts. 45, 46, 49

Chapter V of the GDPR, instead, regulates data transfers of personal data to third countries or international organizations [9].

The Article 45 states that the European Commission has the power to decide whether a country outside the EU/EEA offers an adequate level of protection for personal data. Such a decision is called an *Adequacy Decision*, and it implies that personal data can flow freely from the EU and EEA to third countries without any additional protection [10] [11].

If such a decision is missing, data can still be transferred under certain conditions: the controller or processor shall provide appropriate safeguards, and enforceable data subject rights and effective legal remedies for data subjects shall be available (art. 46) [12].

Ultimately, in the absence of an adequacy decision and the appropriate safeguards, a data transfer could take place based on some derogations for specific situations, for example if *the data subject has explicitly consented to the proposed transfer, after having been informed of the possible risks of such transfers for the data subject due to the absence of an adequacy decision and appropriate safeguards*, or if *the transfer is necessary for important reasons of public interest* (art. 49) [13].

In general, any data transfer which does not fall under one of these cases is not compliant with the GDPR, and therefore is illegal.

This is a focal point: Minos, the software which was developed in the context of this thesis, needs a blacklist of hosts to analyze the HTTP requests sent from the client and to check whether those request are legitimate or not. If the destination host is present within the blacklist, then it is not legitimate. The blacklist was compiled by MonitoraPA [14] by including several hosts from countries for which an adequacy decision was not taken, that have not been proven to provide adequate additional safeguards and for which the derogations are not applicable. For more details on how the blacklist was compiled see: Section 2.4.

2.2 Data transfers to the US

2.2.1 Safe Harbor and Schrems I

According to the 1995 Data Protection Directive (95/46/EC), applicable within the EU, personal data can only be transferred to third countries which are deemed to take adequate data protection measures.

In 2000, the EU, together with the US government, created the so called *US-EU Safe Harbor Framework*. Any US company taking part in the Safe Harbor Framework was considered to have an adequate data protection level. To join the US-EU Safe Harbor, a company must self-certify that it complies with seven principles and related requirements: *notice, choice, onward transfer, access, security, data integrity* and *enforcement*.

After some time, several reports from the EU verified and testified the lack of transparency and respect of the Safe Harbor principles, from companies which self-certified them [15].

Several years later, in 2013, the US computer intelligence consultant Edward Snowden revealed the existence of several global mass surveillance programmes run by the US intelligence services [16]. This gave birth to the so-called *Datagate* scandal.

In the same year, the Austrian lawyer and privacy advocate Max Schrems filed a complaint against Facebook Ireland Limited with the Irish Data Protection Commissioner, arguing that the US did not offer adequate protection against surveillance and asking to suspend the data transfer. Facebook relied on the US-EU Safe Harbor Framework as the legal basis for personal data transfers, but the Snowden disclosures showed that US organizations and entities were not obliged to respect the rules on data protection when these could collide with the US national security requirements and of public interest.

The Irish High Court referred the case to the Court of Justice of the European Union (CJEU). In 2015, the CJEU invalidated the Safe Harbor decision with the judgment C-362/14 ("Schrems I") [17], since any *legislation permitting the public authorities to have access on a generalized basis to the content of electronic communications must be regarded as compromising the essence of the fundamental right to respect for private life, as guaranteed by Article 7 of the Charter of Fundamental Rights of the European Union.* Another important point which comes out from the judgment is that EU citizens had not the same rights and defensive measures of the US ones for what concerns personal data protection: they had no possibility to pursue legal remedies in order to have access to personal data relating to them, or obtain the rectification or erasure of such data.

2.2.2 Privacy Shield and Schrems II

On 15 April 2016 the European Parliament and Council of the European Union adopted the GDPR. Some months later, on 12 July 2016, the European Commission approved a new agreement called *EU-US Privacy Shield* (Decision 2016/1250), which aimed at replacing the previous US-EU Safe Harbor invalidated by the CJEU. This decision, in practice, asserts that the level of protection ensured by the US is essentially equivalent to the one offered in the EU by the GDPR.

Among other improvements, such as stronger obligations on US companies and stronger monitoring and enforcement by the US Department of Commerce and Federal Trade Commission, Privacy Shield also provided a new mechanism called the *Privacy Shield Ombudsperson* to possibly facilitate the processing of and response to requests relating to the possible access for national security purposes by US intelligence authorities to personal data transmitted from the EU to the US. In practice, the Privacy Shield Ombudsperson is a senior official within the US Department of State who is independent from US intelligence services.

The Article 29 Working Party, an advisory body which was instituted with the 1995 Data Protection Directive (and later dismissed), gave its opinion on the Privacy Shield, stating that it would bring *significant improvements compared to the Safe Harbor decision* [18], but it would not solve three main problems: the data erasure, the significant amount of collected data, the lack of privacy guarantors.

Once again, in 2020, the EU-US Privacy Shield was invalidated by the CJEU with the judgment C-311/18 (“Schrems II”) [19] on much the same grounds of the Schrems I. The Court stated that the restrictions on the personal data protection imposed by the internal US law, which gives access and use rights to the US public authorities, result in limitations on the protection of personal data which do not meet the requirements of the EU law, and therefore the US could not be deemed to offer adequate protection to the personal data of EU citizens. Moreover, the legislation does not offer valid rights to the data subjects against the US authorities. The Ombudsperson mechanism was deemed to be non compliant with Article 47 of the Charter of Fundamental Rights (CFR) of the European Union, which states that everyone whose rights and freedoms guaranteed by the law of the Union are violated has the right to an effective remedy before a tribunal which should be *impartial* and *independent*.

With regards to the US domestic law, the FISA 702 [20], would permit *the Attorney*

General and the Director of National Intelligence to authorize jointly [...] the surveillance of individuals who are not United States citizens located outside the United States in order to obtain 'foreign intelligence information', and provides, inter alia, the basis for the PRISM and UPSTREAM surveillance programmes. In the context of the PRISM programme, Internet service providers are required [...] to supply the NSA with all communications to and from a 'selector' (e.g. an email address or phone number)*, some of which are also transmitted to the FBI and the Central Intelligence Agency (CIA).* Those surveillance programmes were considered to be not "proportionate" - according to the principle of proportionality, as expressed in the Article 52 of the Charter of Fundamental Rights of the European Union - by the CJEU.

The FISA 702, jointly with the EO 12333 [21], was considered to be in conflict with the GDPR [7]. Because of them, an equivalent level of protection could not be guaranteed.

Therefore, the Privacy Shield decision is invalid and could not be used as legal basis for personal data transfers to the US.

2.2.3 The Data Privacy Framework

After the abolition of the Data Privacy Shield, different actors from both the EU and the US stressed the need for a new agreement. It is important to underline the effects of the Russian invasion of Ukraine, which was used by the US government to put pressure on the EU on sharing personal data. Meanwhile, organizations have been keeping transferring personal data from the EU to the US, while national EU regulators had mostly ignored it, putting their hope in a new deal which would solve the privacy issues [22].

The US Department of Commerce and the European Commission developed a new framework called the Data Privacy Framework (DPF). This framework is based on a mechanism which closely resembles the Safe Harbor and Privacy Shield ones: the eligible US-based organizations willing to take part in the DPF program have to self-certify their compliance to certain data protection obligations. You can check the current list of entities at the *dataprivacyframework.gov* website [23].

On the 10th of July 2023, the European Commission adopted the adequacy decision about the EU-US Data Privacy Framework. This important decision establishes that the level of protection for personal data offered by the Data Privacy Framework principles is essentially equivalent to the one of the European Union.

After this decision new concerns have arisen from Max Schrems and its non-profit *noyb*: they declared that the "*New Trans-Atlantic Data Privacy Framework (is) largely a copy of "Privacy Shield". [...] We now had 'Harbors', 'Umbrellas', 'Shields' and 'Frameworks' - but no substantial change in US surveillance law. [...] Just announcing that something is 'new', 'robust' or 'effective' does not cut it before the Court of Justice. We would need changes* *in US surveillance law to make this work - and we simply don't have it."* [24].

Specifically, the US has refused to reform FISA 702, which in practice negates reasonable privacy protections to non-US persons. Moreover, FISA 702 will have to be prolonged by the end of 2023, given a clause in the US law.

The *noyb* organization declared on its official website that it would challenge the decision. Moreover, when the new system will be implemented by the companies, people will have the possibility to bring a challenge with Data Protection Authorities or Courts. *It is not unlikely that a challenge would reach the CJEU by the end of 2023 or beginning of 2024 - noyb* says.

2.3 Garante per la Protezione dei Dati Personali

Chapter VI of the GDPR deals with the supervisory authorities. In particular, Article 51 states that *each Member State shall provide for one or more independent public authorities to be responsible for monitoring the application of this Regulation, in order to protect the fundamental rights and freedoms of natural persons in relation to processing and to facilitate the free flow of personal data within the Union.* Such an authority is called *supervisory authority.*

The *Garante per la Protezione dei Dati Personali* (GDPD), also known as Garante della Privacy, is the Italian supervisory authority, which was established in 1996.

The powers of the supervisory authorities, listed in GDPR Art. 58, include the possibility of obtaining any useful information from the controller or the processor, ordering the controller or processor to comply with the data subject's requests to exercise his or her rights, imposing limitations on the data processing, etc. We considered this specific article for writing the claim template which is used in Minos (see: Chapter 3).

2.4 The blacklist

As we said before, Minos internally relies on a blacklist of domains to check whether requests are compliant or not with the GDPR. This blacklist is a focal point within our thesis, therefore we think it is important to discuss the way it is composed.

The blacklist was compiled between 2022 and 2023 by MonitoraPA. Most of the domains included within the list were suggested in the issues of the MonitoraPA project on GitHub by Federico Leva, an Italian developer and activist [25].

When I contacted Federico he was happy to contribute to this thesis with his knowledge. So I asked him how he found the domains he suggested within the MonitoraPA issues, and he replied saying that he just checked the list of the most popular services in

Italy according to *builtwith.com*, a website which monitors web technology trends and provides rankings and comparisons [26].

The list is not complete and it can change at any time. It just includes the most popular hosts from non-EEA countries. Most of them are from the US, but there are also some from China and Russia.

2.5 Validity of this thesis work

The fact that the new Data Privacy Framework has been released may raise some doubts about our thesis work, since most of the domains we considered in the blacklist are from the US (and at the time of writing some of them have already certified their compliance to the DPF). We want to dispel these doubts right now, before we get into the heart of the thesis.

First of all, when we started this thesis on April 2023, the adequacy decision on the DPF was not already taken, therefore the entire work is based on this fact.

Second, even if the decision on the DPF has been taken, many companies from the US have not yet self-certified their compliance.

Third, the blacklist does not only include domains from the US, but also from other countries for which an adequacy decision has not been taken, and the list can be easily modified to exclude US hosts. Additionally, Minos does not automatically report to the Italian Garante per la Privacy; it is up to the user to submit the claim to the Garante, if he or she believes that its rights have been violated. Minos is just a tool.

Moreover, less than a month passed between the decision on the DPF (10/07/2023) and the moment when we started with the analysis on the Italian PA (30/07/2023), therefore it is reasonable to suppose that many analyzed websites relied on non-EEA service providers even before the adequacy decision. In fact, it is unlikely that all the bad entities (see Subsection 3.2.1) identified during the analysis waited for the adequacy decision to use the services of those providers.

In conclusion, we believe that the validity of this work is not influenced by the adequacy decision on the Data Privacy Framework.

Chapter 3

Minos

One of the main tasks of this thesis work was to develop a software tool targeted at non-advanced users for navigating the web while analyzing and reporting HTTP requests directed to non-EEA domains.

We will present here, quite formally, the architecture and development of our software, Minos, by using some tools proper to software engineering. We will highlight our implementation choices, discussing the possible alternatives. We will also introduce *minos-cli*, the mass-analysis targeted version of Minos, which has been used to produce the results of Chapter 4. At the end, we will compare our software with other existing tools with similar goals.

The name *Minos* comes from Dante Alighieri's Divine Comedy, where Minos is presented as the one who sits at the beginning of the Hell, judging the sins of each soul and assigning it to its appropriate punishment. The idea is that our software should behave like a "judge" of the sins committed by the websites that do not comply with the GDPR.

3.1 MonitoraPA

MonitoraPA defines itself as a community of activists which focus on privacy-related problems and defend the digital rights of their fellow citizens [27]. Starting from 2022, MonitoraPA operates an Automatic Distributed Observatory on the Italian PA. The software tool which they use is available on their GitHub repository [28].

The development of Minos was carried out with the collaboration of MonitoraPA. They defined the specifications and outlined the user interaction. They provided support during the development by answering to questions, testing parts of the code and sharing their opinions, as well as data sources, references, etc.

Most importantly, MonitoraPA compiled the blacklist of *bad hosts* which is used to individuate *bad requests* (see: Subsection 3.2.1).

3.2 Requirements

The software development followed the basic Software Engineering principles.

The specifications from MonitoraPA were translated into functional and non-functional requirements with the help of various tools, such as a graphical mockup of the application, an activity diagram, etc. For the mockup we used the Pencil open source GUI prototyping tool [29].

3.2.1 Glossary

Herein we present some important definitions that will be used in the rest of the document:

- **Bad host:** A host which is present within the blacklist compiled by MonitoraPA. To be more precise, we could use the term *domain*, or *subdomain*, in place of *host* - a domain is usually associated to more hosts - but for our present discussion this difference will not matter that much.

- **Bad group:** (sometimes also *group*) A group of bad hosts, which usually - but not always - gives the name to the company behind the hosts contained within it. In some cases, the name of a group corresponds to the name of a service offered by the company. *Example*: the bad domains *.adobe.com*, *.fyre.com*, *.demdex.com* all belong to the same group of hosts *adobe*.

- **Bad request:** A request sent from a website to an URL which matches against one of the bad hosts. To see how the matching is performed please see Subsection 3.3.1.

- **Bad entity:** A website for which at least one bad request has been identified.

3.2.2 Activity Diagram

Due to the highly interactive nature of the software product, I think that the best way to describe it is through an activity diagram. A possible activity diagram representing the user interaction is reported on Figure 3.1.

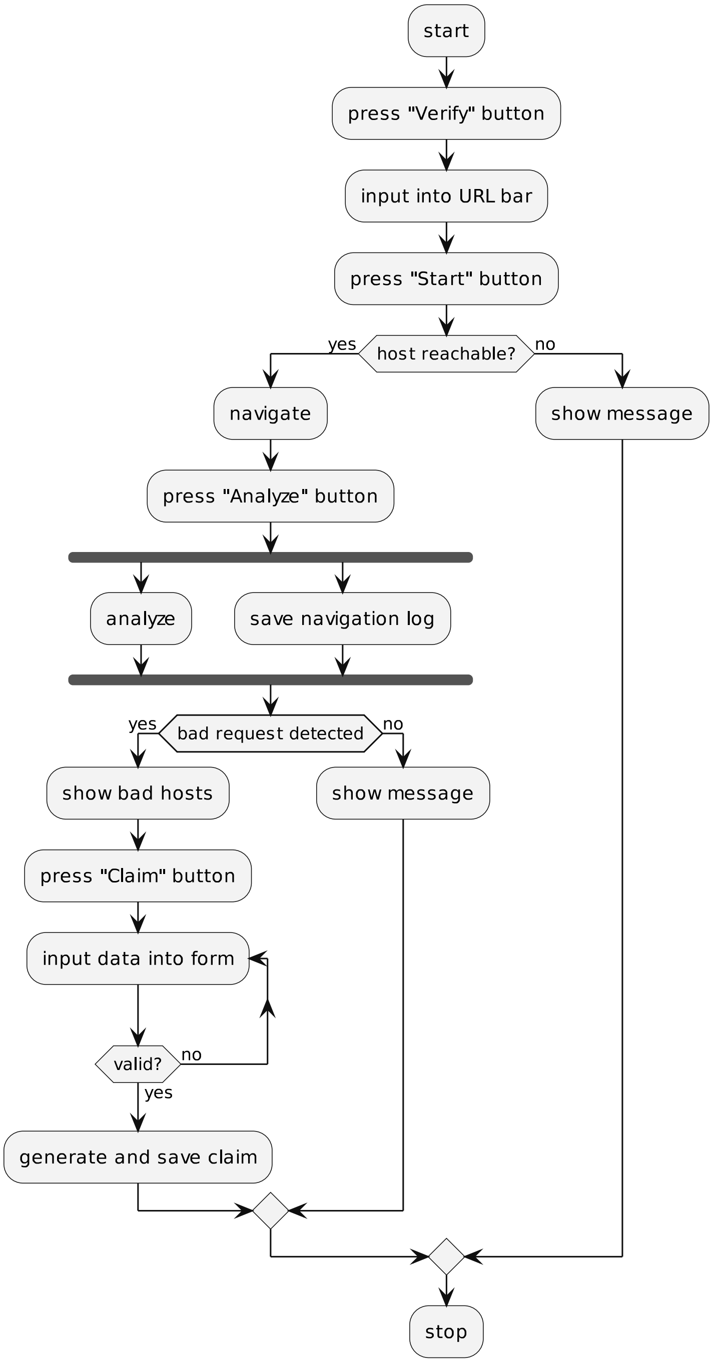

Figure 3.1: Activity Diagram

3.2.3 Functional Requirements

The following list reports the functional requirements which have been identified during the requirements engineering phase:

1. **Navigation:** The user should be able to navigate the web
   1.1 The user should visualize the start screen with a button at the bottom, containing the label *Verify a Website*.
   1.2 After clicking the button, the user should visualize a window with an input bar at the top left and a *Start* button at the top right.
   1.3 The user should be able to input text in the input bar.
   1.4 The user should be able to press the *Start* button in order to start the navigation.
   1.5 If the host is unreachable, the user should visualize an error message and quit.
   1.6 If the host is reachable, the user should be able to continue the navigation.
   1.7 The user should be able to click the *Analyze* button in order to continue with the analysis step.

2. **Analysis:** The user should be able to analyze the data which have been collected during the navigation in order to possibly identify *bad hosts*.
   2.1 The user should visualize a dialog which lets him save the log of the navigation, possibly with a different name.
   2.2 If no bad hosts have been detected, the user should visualize an info message and quit.
   2.3 The user should see the list of the bad hosts which have been detected.
   2.4 The user should be able to press the *Claim* button in order to continue with the generation of the claim.

3. **Reporting:** The user should be able to generate a claim directed to the Italian Data Protection Authority (namely Garante per la Protezione dei Dati Personali).
   3.1 The user should input its data into a multiple-page form. The navigation from a page to the next one should be possible only if all data within the current one are present.
   3.2 The user should visualize a dialog which lets him save the claim as a pdf file, possibly with a different name.

3.2.4 Non-functional requirements

- **Portability:** the software should run on Windows, Linux, MacOS.
- **Usability:** a medium user which knows how to navigate the web should learn how to use the application within max 15 minutes.

3.2.5 Design requirements

Here we list some design constraints which were specifically defined by MonitoraPA:

- **Serverless:** the application should not depend upon a server to work; everything should be contained into a single executable.
- **Proxyless:** the application should not use a proxy to work.
- **Minimize dependencies:** try to minimize the dependencies as much as possible.
- **Simple and minimal codebase:** the code should be easy to understand and maintain; try not to use syntactic tricks, readability comes first.

3.3 Architecture and components

The architecture of the application heavily depends on the Electron process model [30].

We chose to keep the discussion about components at a fairly high level: we will describe the software components from a logical point of view. However, the boundaries between them, within the code, may be more blurred.

An Electron app consists of two kind of processes: the *main* process and the *renderer* processes.

The *main* process runs in a `node.js` environment, it can use the `node.js` API, require modules, manage windows, etc. It also has access to an extended set of custom APIs to interact with the operating system.

The *renderer* process, as the name says, is responsible for rendering web content. There is one renderer process per each browser window. The renderer process does not have access to the *node.js* environment.

We also have the possibility to write *preload scripts*, which acts like a sort of bridge between the *main* and *renderer* processes, and allows them to communicate.

3.3.1 Main process components

Windows and views management

One of the goals of the main process, as said before, is to manage windows. A window is represented by the Electron class *BrowserWindow*. Our application will have a single window containing two browser views.

A *BrowserView* is an Electron component which basically consists of a rectangle of free space. Each view has its own lifecycle and content, and can be moved and resized, so the idea was to use two different views: one to display local content (*localview*) and the other one to display remote content (*webview*). So for example, while the user is navigating, the local view shows the top bar and the other one shows the web content.

The main process will create, resize and configure the two browser views.

Navigation

The navigation is performed by calling the `loadURL` method on the `webView. webContents`.

We want to point out that the browser allows navigating to a single URL per navigation session: when the user has input the URL in the top-bar and clicked *Start*, the URL cannot be changed anymore. This choice is not random: in fact, we imagine that such a product will be run by users to test specific websites, and check whether they respect or not GDPR. So we suppose that the user will navigate to a specific website that he knows, then generate the report and close the browser. Maybe at a later time he will need to check another website, so he will run again the software and navigate to that website, and so on.

It does not make sense, in our opinion, to have a general purpose web browser to accomplish this task, and here is why Minos is not built as a generic web browser.

Network traffic capture

At the beginning, the idea was to exploit the Electron *WebRequest API* to extract data from the web requests. It turned out the WebRequest API does not allow to retrieve enough data to generate a HAR file (see the discussion on the HAR format: Section 3.4).

Eventually, the choice fell on the *Chrome Debugger API* [31]. The idea is simple: the Chromium-based web browsers offer a set of APIs called *Chrome DevTools Protocol* (CDP), which can be used to accomplish several tasks. In particular, we are interested in the Network APIs [32], which make it possible to capture and process network traffic.

The Electron framework internally relies on a Chromium engine to render web pages, therefore we can exploit the CDP with the instance of Chromium which runs inside Electron. Moreover, Electron offers a convenience API to exploit this protocol: the *Debugger API* [31].

After the user clicks the button, the main process attaches the debugger to the webview, in order to be able to capture the network traffic.

HAR object creation

Now that we have collected the web requests, we have to bag everything inside an HAR object, that will be exported to a file.

To create the HAR object in memory, we relied upon the *Chrome HAR Capturer* [33], an interesting software project which basically allows to capture traffic from a Chromium instance and translate it into a HAR file. Network events are mapped to HAR fields.

Since we already solved the problem of capturing the traffic, we only took and modified the part related to the CDP-to-HAR translation.

Analysis

Now that we have recorded the network traffic, it is easy to identify the bad requests. We just need to check if the `request.url` field of the entries within the `har.log.entries` array matches against one of the bad hosts in the blacklist.

To identify the bad hosts, a *longest matching* strategy is used: if an URL matches against more bad hosts, only the longest one is taken.

Furthermore, we decided to intercept the requests as soon as they are produced. This means that we do not expect the HAR object to be available, but rather we wait for the event `Network.requestWillBeSent` to happen and we extract the URL from the request - the same URL that will be put into the `request.url` field of the HAR object.

Claim generation

After the user has input its data, the main process has to generate a pdf file containing the claim. To generate the pdf file the *pdfkit* [34] library is used.

To keep a modular approach and make it configurable, the text of the document is divided into different sections that can be configured within the `assets/document.json` file.

3.3.2 Renderer process components

We tried to keep everything as modular as possible. Since we have two browser views, we will also have two renderer processes.

The renderer process associated to the *webview* is very simple: it just has to display content from the web.

The renderer process associated to the *localview*, instead, must display local content. Right after the creation of the view, an `index.html` file is loaded. This html file contains all the GUI components: the top bar, the buttons, the report section, the form section, etc. These components are hidden and shown as needed from the javascript code. The html file is shipped along with a css file, which defines the appearance of all the components.

The html file loads a `renderer.js` file which accomplishes several tasks:

- Define event listeners on DOM elements.
- Load text labels from the `strings.js` file into the corresponding DOM elements.
- Allow navigation between form pages, validating the data step by step (i.e. you can only go on if data within the current page are valid).
- Show and hide components on demand.
- React to main process events and generate events directed to the main process.

3.3.3 Inter-Process Communication

In Electron we can use the `ipcMain` and `ipcRenderer` modules to define custom communication channels. The communication channels must be exposed via the `preload` script. We can think of communication channels as events which may carry additional info.

Within the code we defined 11 different communication channels, that we represent with a sequence diagram[1] in Figure 3.2.

We will now explain the meaning of each of them:

- **start:** When the user is ready to start the navigation it clicks on the *Start* button, and this event is generated from the renderer process to tell the main process that the web navigation should start. From now on, the user will not be able to change the URL in the top bar with the keyboard.
- **change-url:** Whenever the web content location changes, we want the top-bar URL to change. This event tells the renderer process to update the interface.
- **navigation-fail:** If the navigation to a given URL fails, this event is generated to tell the renderer process to display an error message.

[1] The actual names of the communication channels may differ from the ones in the sequence diagram.

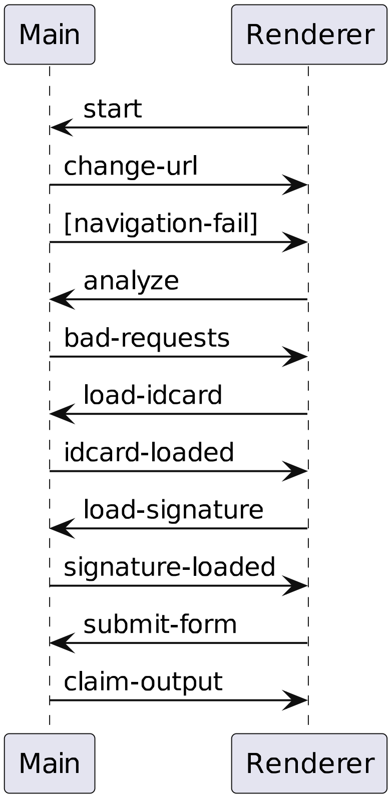

Figure 3.2: Sequence Diagram

- **analyze:** When it is done navigating, the user presses the *Analyze* button; the renderer process alerts the main, which generates the HAR file.
- **bad-requests:** Right after the HAR file creation, the main process sends this event to the renderer process, along with all the bad requests detected. The renderer process displays the bad requests.
- **load-idcard:** Ask the main process to create a dialog for uploading the ID card photo.
- **idcard-loaded:** Tell the renderer process that an ID card photo has been uploaded.
- **load-signature:** Ask the main process to create a dialog for uploading a signature photo.

- **signature-loaded:** Tell the renderer process that a signature photo has been uploaded.
- **submit-form:** Send all the form data collected from the renderer to the main process. The main process will generate the claim document with all the info.
- **claim-output:** Tell the renderer process that the user has saved the claim to an output file.

3.4 The HAR format

A separate discussion is needed to examine the HAR format, that MonitoraPA chose as the output format for log files.

3.4.1 Introduction

The HTTP Archive Format (HAR) is a JSON-based format which is used to export detailed performance data about the loaded web pages in a browser. The HAR specification was published by the W3C Web Performance Working Group as draft [35]. The last specification was in 2012, and then it was abandoned. Figure 3.3 shows the basic structure of a HAR object.

```
{
    "log": {
        "version" : "1.2",
        "creator" : {
            "name": "creator",
            "version": "1.0"
        },
        "browser" : {},
        "pages": [...],
        "entries": [...]
    }
}
```

Figure 3.3: HAR sample

3.4.2 Advantages

Diffusion

HAR is not a standardized format: it means that any tool can or cannot implement it. However, it is the closest thing to a standard that we have by now. Even though it has been abandoned, it is still used by many web browsers and network analysis tools. This means that the Italian Data Protection Authority should have the possibility to use standard existing tools to examine the logs sent by the user.

3.4.3 Disadvantages

Huge overhead

Since the HAR format must record the whole traffic, it must include all the information about every request and response, included the response and request bodies, timing information, etc. This results in a huge overhead, considering that we're only interested in the requested URLs.

To show the impact of this overhead, we examined about 4000 HAR files generated by Minos. We computed the average overhead as follows: first, we computed the size in bytes of the whole HAR file, and the size in bytes of the rows from the file containing the key "url", then we divided the total size by the url-rows size to get the overhead for the specific file; the average overhead is computed as the sum of the single file overheads divided by the number of HAR files. The resulting number is 46.17. This means that if, for example, a certain website makes only one request to an url of length 100 bytes, the resulting HAR file will be almost 4600 bytes long, i.e. ~4.5kB.

Privacy issues

As specified in the HAR draft itself,

> *The HAR format may contain privacy & security sensitive data and the user agent should find some way to notify the user of this fact before it transfers the file to anyone else.*

There is no reason for us to handle this privacy and security sensitive data, since we're only interested in request URLs.

Unclear implementation choices

Lastly, some implementation choices in the HAR format are rather obscure [36]: some fields in particular were defined ambiguously.

At the same time, many browser do not follow the specification strictly. This is due to the fact that, as we said before, the specification is not a standard, therefore browsers are not obliged to implement it literally.

For example, the HAR format defines, within the `log` object, an array of `pages`. However, it is not fully clear from the specification nor from the vocabulary [37] what a *page* actually is. Regarding the pages array, the vocabulary says:

> *This object represents list of exported pages. There is one <page> object for every exported web page.*
>
> - - HAR Vocabulary

Intuitively, we could think as a page as "what gets loaded in a browser tab", however the various browsers may adopt different strategies for exporting pages: if you navigate with a Chromium-based or Firefox browser, and you export the traffic, you will get a HAR file which only contains a single page, and all the entries - which in turn represent requests - will be associated to this page.

For our purposes, this is not a big problem: our browser allows navigating to a single URL per navigation session, so the `page` array within the HAR object will contain a single page associated to that specific URL.

Moreover, the pages array is marked as optional in the specification, but a HAR file missing this field will not be imported by Firefox and Chromium-based browsers.

3.4.4 Alternatives

We considered two possible alternatives to the HAR format:

- A custom JSON-based format, which would have included only the fields of interest from the HAR format. This solution requires to specify the custom format and possibly implement a viewer/analyzer for this specific format to be shipped along with the log files.

- A csv file which contains the requests performed by the analyzed website, along with all the necessary data to interpret the log.

3.5 Testing

The software was first tested by MonitoraPA developers. In the first days they found some bugs which were quickly corrected.

Then *User Acceptance Testing* [38] was adopted: the software was released in an alpha version, so that the users would have been able to run it, identify bugs and report them to the developers.

Minos is tested on Arch Linux, Windows, Debian 12.1.

From now on the development and maintenance will be carried out by MonitoraPA developers.

3.6 Minos-cli

minos-cli is a set of JavaScript and Python scripts with the goal of running a massive automatic analysis on the Italian PA websites (the results are presented within Chapter 4), compute some statistics and present the results with some sort of visualization.

This scripts are not targeted to end users, they have only been used by us to collect and analyze data. Therefore, the development of these scripts did not follow formal steps.

However, we will still try to give an idea of the rationale behind the implementation choices. We will present all the scripts, their features, options, configurations, etc. In particular, we will follow the flow that brings us from the input datasets to the output visualizations.

3.6.1 join-entities-categories.py

`join-entities-categories.py` is a Python script aimed at joining “tables” from the input csv datasets.

This script links the data of the entities with the names of the categories and produces a new file with all the data in one place.

For a more in-depth discussion about the datasets please see Section 4.1.

3.6.2 minos-cli

Now that we have the list of entities along with their websites, we can contact them and record the network traffic.

The `minos-cli` script first reads all the entities from the `entities.csv` file, and it filters out all the entities that have already been processed (we store them into a `done.csv` file) and the ones that have an empty website URL. Then it uses a modified version of the *Chrome Har Capturer* (CHC) module (see Subsection 3.3.1) to contact the websites and produce HAR objects.

While the original CHC module runs the capture on all the input websites and then it produces a global HAR object, our modification allows it to produce one distinct HAR object per website, so that we can identify the bad requests as soon as they happen.

As before, to identify a bad request we can read the `entry.request.url` field of the entries within the `har.log.entries` array and check whether it matches against one of the bad hosts. If it does, we append one row to the `bad-requests.csv` file, which links the entity *IPA code* to the request's *URL*, plus some additional info. After processing an entity, we write its code to the `done.csv` file, with an optional error message if we could not reach the website.

`minos-cli` accepts two optional command line options: the `group-size` option, which defines how many websites should be contacted within the current analysis, and the `parallel` option, which defines the number of parallel threads to spawn.

Note that, as before, we need a running instance of a Chromium-based web browser to exploit the CDP protocol. Since we are not in an Electron environment anymore, we need to run the browser separately and set the remote debugging port number to the one specified by the Chrome Har Capturer module (defaults to `9222`). Here is why we also wrote a convenience script (`run.sh`) which checks whether an instance of a Chromium-based web browser is running before running the `minos-cli` script.

3.6.3 check-file-consistency.py

We run this script to check the presence of consistency errors within the input and output files: `entities.csv`, `done.csv` and `bad-requests.csv`. In particular, the script checks:

- That each *IPA code* within the `done.csv` file is actually present in `entities.csv`.
- That each *IPA code* within the `bad-requests.csv` file is actually present in `entities.csv`.

3.6.4 analyze.py

This script takes as input all the `.csv` files from the previous stages (`entities.csv`, `done.csv` and `bad-requests.csv`) and computes several statistics on data, which are saved to a `stats.json` file. The computed statistics and results are presented in Chapter 4.

3.6.5 check-stats-consistency.py

We run this script to check the internal consistency of the `stats.json` file. For example, we check that the total number of requests is equal to the total number of bad requests reported is equal to the number of bad requests grouped by category, and so on.

3.6.6 visualize.py

This script produces a graphical representation of the statistics. To generate the pictures the `matplotlib` library was used. The generated graphics will be presented and commented in Chapter 4. We would like to point out that we managed to make the data visualization inclusive by using different levels of hue, saturation and lightness, which ease the readability by people with color blindness.

3.6.7 Performance

At the beginning we were saving one HAR file per each entity. Needless to say, this resulted in a huge overhead, both from the filesystem occupancy point of view (see the discussion on HAR format disadvantages: Subsection 3.4.3) and from the analysis speed one. Here is why we adopted the `csv` files strategy to store data.

Disk Occupancy Overhead

We first generated ~4000 HAR files, one per entity. The filesystem occupancy was huge: 1.1 GB.

With the `csv` files, to represent data for the same number of entities, we reduced the occupancy to just 1099 kB, i.e. the 0.1% of the HAR files occupancy.

Analysis Time Overhead

The `csv` files choice not only impacts on the disk occupancy, but also on the analysis speed.

If we try to compute the statistics on the original ~4000 HAR files, the time spent elapsed during the analysis is about 21 seconds.

Instead, if we run the same analysis on the `csv` files, we only need about 0.16 seconds to generate the statistics, which is almost imperceptible from the user point of view.

This is obviously due to the filesystem access overhead. We must consider that when we run the analysis on 4000 separate HAR files we have to open the files, read the content in memory and close the files 4000 times, and this results in a very huge number of system calls.

3.7 Comparison with similar software

There exist several tools which promise “anti-tracking protection”, “GDPR compliance checking”, and so on. We will now present the most interesting ones that emerged from our research and compare them with our software, highlighting the differences and similarities. We also tested these services against one of our blacklisted domains (namely `yandex.com`, a Russian domain). We will also report the results that we obtained.

3.7.1 Privacy Badger

Privacy Badger is one of the most downloaded anti-tracking extension for browser. Published by the Electronic Frontier Foundation (EFF) [39], it works by sending the `Global Privacy Control` signal [40] and the `Do Not Track` signal [41] to websites; if they ignore the signals, Privacy Badger will block them in future. Therefore, the extension does not contain a "blacklist", but rather it only blocks domains if they are observed collecting unique identifiers after the signals are sent.

It is clear that Privacy Badger does not focus on detecting non-EEA domains, but rather on trackers specifically. It makes no distinction between trackers in different countries.

The extension is theoretically serverless, however the web browser will still contact the EFF’s CDN (Fastly) to fetch some resources which ensure that some assets used by the application are always fresh even if there has not been a new Privacy Badger release in a while.

When we tested Privacy Badger with the `yandex.com` website, it did not report any issue. This is probably due to the fact that maybe `yandex.com` could be respecting the Global Privacy Control and Do Not Track signals, even if, as said before, the domain is from a country for which an adequacy decision has not been taken.

3.7.2 Blacklight

Blacklight by The Markup [42] is a web application which emulates the user interaction on a web page and tries to detect possible privacy violations. It claims to be able to identify seven different types of privacy violations:

- Third-party cookies
- Ad trackers
- Key logging

- Session recording
- Canvas fingerprinting
- Facebook tracking
- Google Analytics "Remarketing Audiences"

The tool works by opening a headless browser and visiting the URL homepage as well as additional randomly selected web pages from the same website.

Also in this case we have an application which focuses on privacy issues in general, not on non-EEA data transfers specifically. Being an online web application it relies on a server.

When we tested Blacklight on `yandex.com`, we received messages informing us about the trackers, third-party cookies and keystrokes/clicks monitoring, but none of them warning us about an non-EEA data transfer.

3.7.3 ImmuniWeb Security Test

ImmuniWeb Security Test [43] is an AI-based online tool for security assessment. It allows running some security tests on a given website.

Among the others, ImmuniWeb also offers the EU GDPR Compliance Test. Some checks are performed: Privacy Policy, TLS Encryption, Cookie Protection, etc. However, no checks on non-EEA data transfers are performed.

When we ran the ImmuniWeb Security Test on `yandex.com` we only found some issues regarding Website Security, Cookie Protection and Cookie Disclaimer. Again no warning messages about data transfers to non-EEA countries.

Being closed-source, we also do not know anything about the methodology adopted by ImmuniWeb.

3.7.4 2gdpr

2gdpr [44] is another web application which allows testing for GDPR compliance. Unlike the other tools, 2gdpr also claims to be able to detect personal data transfers to non-adequate countries.

Unfortunately, when we performed the analysis with the `yandex.com` website, the scanner did not found any issue related to personal data transmission outside the EEA. We repeated the test with other blacklisted domains, but still no warnings.

2gdpr, like ImmuniWeb, is a closed-source application, so we are unable to interpret the results. However, we can make some hypotheses. The `complianz.io` website offers one possible interpretation [45]: online testers - they say - do not work region-based, and some of them have their servers based in non-regulated regions.

3.7.5 webXray

webXray [46] is an open source Python tool for analyzing webpage traffic and content, extracting legal policies and identifying the companies which collect user data. It was developed by Timothy Libert, a privacy and security researcher.

Like Minos, the tool exploits the Chrome DevTools Protocol. It allows to scan a list of webpages and to generate several reports: the details on percentages of sites tracked by different companies and their subsidiaries, the most frequently occurring third-party domains, the most-frequently occurring third-party requests, etc.

However, it does not focus on specific GDPR violations, even if in theory it could be used to build a tool which does.

webXray is not available for Windows, and it runs from a command line interface, therefore it is not targeted to novice users. Unfortunately, at the time of writing, the official repository is no longer available on Github.

3.7.6 OpenWPM

OpenWPM is an extensible web privacy measurement framework which makes it easy to collect data for privacy studies on a scale of thousands to millions of websites. It is built on top of Firefox, with automation provided by Selenium. It includes several hooks for data collection.

OpenWPM is able to record and store in a database several web events, such as HTTP requests, HTTP responses, HTTP redirects, JavaScript method calls, etc.

OpenWPM is a very powerful and advanced tool which focuses on performance, but just like webXray it does not target a specific GDPR violation.

3.7.7 Summary

Name	non-EEA data transfers checking	Anti-tracking	Server-based	Open source
Minos	Yes	No	No	Yes
Privacy Badger	No	Yes	No	Yes
Blacklight	No	Yes	Yes	Yes
ImmuniWeb	No	No	Yes	No
2gdpr	Yes*	No	Yes	No
webXray	No	Yes	No	Yes
OpenWPM	No	Yes	No	Yes

**2gdpr claims to be able to identify data transfers to inadequate countries, however our tests showed that it fails to detect such transfers in more than one case. Therefore it could not be considered as a valid competitor.*

Chapter 4

Analysis on the Italian PA

As part of this thesis work, after the development of Minos, we decided to carry out a mass analysis of the data transfers from the Italian public administration (PA) websites to Countries for which an adequacy decision was not already taken[1], by using the *minos-cli* set of scripts.

Within this chapter we want to present the results of the analysis, highlighting the relationships between them, and trying to give some possible interpretations and explanations to results.

4.1 The datasets

4.1.1 The sources

All the data used within the analysis has been downloaded from the *OpenData IPA* website [47]. The OpenData IPA is a freely available database which is stored, produced and updated by the Italian public administrations. Data is updated every day.

We relied upon two datasets: `enti.csv` and `categorie-enti.csv`.

The `enti.csv` file contains all the personal data of the entities which we considered for the analysis. Each entity is identified by the unique identifier `Codice_IPA`. In Figure 4.1 we represent all the columns of the `enti.csv` dataset in JSON format. The column names are quite self-explanatory, however, we are interested in the following fields:

- **Codice IPA:** unique entity identifier.
- **Denominazione ente:** the name of the entity.

[1] Since the thesis work started before the EU-US Data Privacy Framework was released, the analysis obviously includes data transfers directed to the US.

- **Codice categoria:** the code of the category the entity belongs to.
- **Sito istituzionale:** URI of the website of the entity.

The categorie-enti.csv file contains all the data about the categories of the entities. In Figure 4.2 we list all the columns of the categorie-enti.csv dataset, in JSON format. Some names are not so clear at a first glance, however we are only interested in the following two fields:

- **Codice categoria:** the unique identifier of the category.
- **Nome categoria:** the name of the category.

The join-entities-categories.csv, that we presented in Subsection 3.6.1, produces a new entities.csv file, which is used as input for the analysis, by joining the two input datasets using the Codice_categoria field as join-key, and selecting only the fields of interest (Codice_IPA, Denominazione_ente, Codice_categoria, Sito_istituzionale, Nome_categoria).

4.1.2 Data quality

Before diving into the analysis, we present some data quality metrics that we computed on data and we consider important.

Completeness

As a measure of completeness we computed the number of missing fields within the entities.csv file. It turns out all the missing values are within the website: 755 over 22890 websites are missing, i.e. 3.29% of all the websites.

Uniqueness

Another important metric on data is the uniqueness. We took as measure of the uniqueness the number of duplicate websites. We computed an absolute number of 197 duplicate entries over 22890, i.e. 0.86% of all the websites.

Dirty data

Dirty data is data which is unusable because it is inaccurate, incomplete or inconsistent. The way we define dirty data depends on the object of analysis.

In our case, we can define dirty data as all the hostnames which contain errors and therefore they could not be contacted. We may have two kinds of errors:

```
[
    "Codice_IPA",
    "Denominazione_ente",
    "Codice_fiscale_ente",
    "Tipologia",
    "Codice_Categoria",
    "Codice_natura",
    "Codice_ateco",
    "Ente_in_liquidazione",
    "Codice_MIUR",
    "Codice_ISTAT",
    "Acronimo",
    "Nome_responsabile",
    "Cognome_responsabile",
    "Titolo_responsabile",
    "Codice_comune_ISTAT",
    "Codice_catastale_comune",
    "CAP",
    "Indirizzo",
    "Mail1",
    "Tipo_Mail1",
    "Mail2",
    "Tipo_Mail2",
    "Mail3",
    "Tipo_Mail3",
    "Mail4",
    "Tipo_Mail4",
    "Mail5",
    "Tipo_Mail5",
    "Sito_istituzionale",
    "Url_facebook",
    "Url_linkedin",
    "Url_twitter",
    "Url_youtube",
    "Data_aggiornamento"
]
```

Figure 4.1: `enti.csv` fields

```
[
    "Codice_categoria",
    "Nome_categoria",
    "Tipologia_categoria",
    "SFE",
    "UTD",
    "NSO",
    "AOO",
    "UO"
]
```

Figure 4.2: `categorie-enti.csv` fields

- The URLs which are obviously wrong (e.g. “`about:blank`”, “`/`”, etc.). We will call these *badly formatted URLs*.
- The URLs which may seem correct but they are wrong because of typos (e.g. if you write “`http://www.poliso.it`” in place of “`http://www.polito.it`”). It is clear that to identify such kind of error, one should know the original name of the website. We will call these *misspelled URLs*.

Unfortunately there is no way to check programmatically whether a given hostname is valid or not without contacting the corresponding host.

During the analysis we collected the errors that occurred when we contacted the websites. Among all the errors there is one that is a symptom of a badly formatted URL: `Error: Cannot navigate to invalid URL`. We checked against the `entities.csv` file, and it turned out that this kind of error is always associated to a badly formatted URL. The total number of websites for which this kind of error was thrown is 8 over 22890, which results in a very tiny percentage: just 0.03% of all the URLs. This statistic should not be considered exhaustive, since badly formatted URLs could lead to other type of errors (for example, in some cases we experienced the `Error: net: ERR_NAME_NOT_RESOLVED` error due to badly formatted URLs).

In conclusion there is no way to identify all the dirty entries a priori: even if you check the data one by one, you may spot all the badly formatted URLs, but you still will not be able to catch all the misspelled URLs, since you do not know the original name of the website. However, even if we are not able to give a true estimate of the impact of dirty data on our statistics, we can still report the error percentage.

4.2 The statistics

4.2.1 Unreachable websites

As we said before, during the network traffic recording, we saved not only the successful requests, but also the ones that received no response due to errors.

Before moving on to the statistics on requests and entities we want to provide a visual feedback of the impact that these errors have on the results.

In Figure 4.3 we compare the number of unreachable websites with the number of reachable websites. Please note that the unreachable websites include both the websites which were missing and the ones that could not be contacted due to errors. We highlighted them with different colors.

The number of unreachable hosts represents 14.83% of the total number of websites, which is a quite big value.

Since we collected also the error types, we can also identify the most common errors: we show them in Figure 4.4.

The most common error is the `net::ERR_NAME_NOT_RESOLVED` error. From our discussion on Dirty Data (see: Section 4.1.2) it should be clear that this kind of error could happen both for badly formatted URLs and for misspelled URLs. As we can see, the impact of these errors is quite large if compared to the total number of errors.

4.2.2 Bad entities vs good entities

The first important statistic we would like to report is the comparison between good and bad entities (Figure 4.5).

The bar plot shows that only 84.57% of all entities is possibly compliant with the GDPR.

4.2.3 Bad entities grouped by category

We are not only interested in the absolute number of bad entities. We also would like to tell in which area the violations occurred. In Figure 4.6 we show the top 10 categories in which bad entities were detected.

In the first place we have the category *Comuni e loro consorzi e associazioni*. Entities belonging to this category represent the Italian municipalities. We did not found any specific article or research possibly explaining this result, however we have found some articles that show a general trend towards widespread violations of the GDPR by Italian municipalities. In 2019, a report by Federprivacy showed that at the time 47% of the

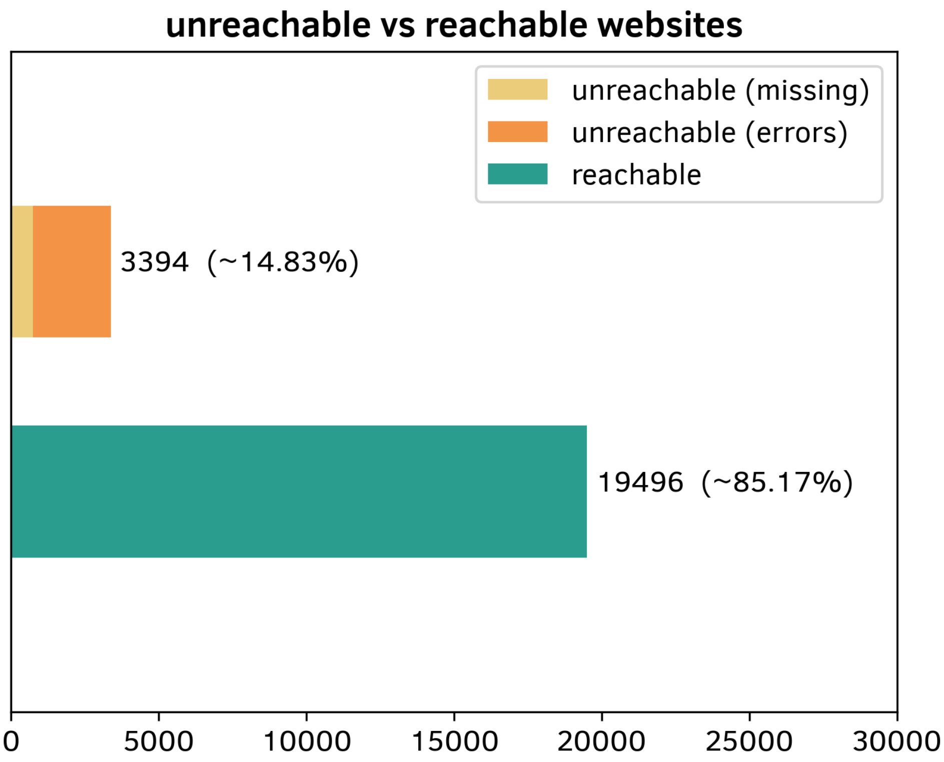

Figure 4.3: Unreachable vs reachable websites

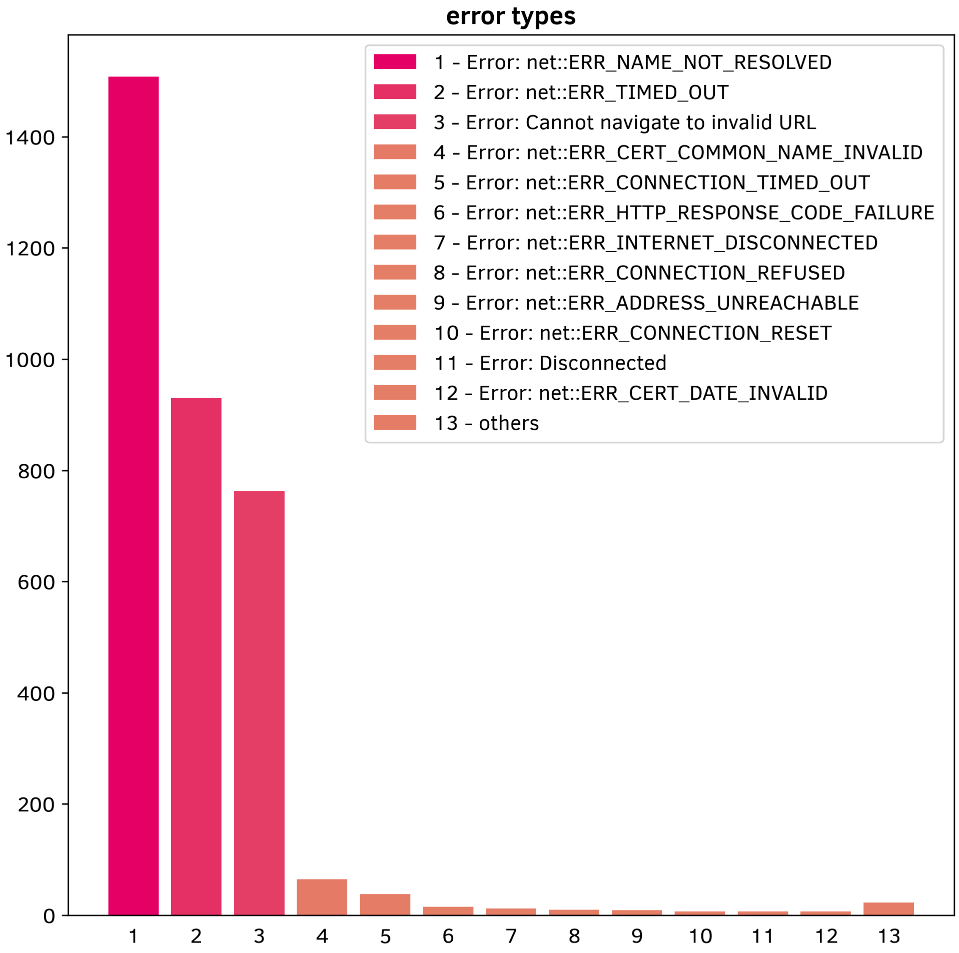

Figure 4.4: Error types

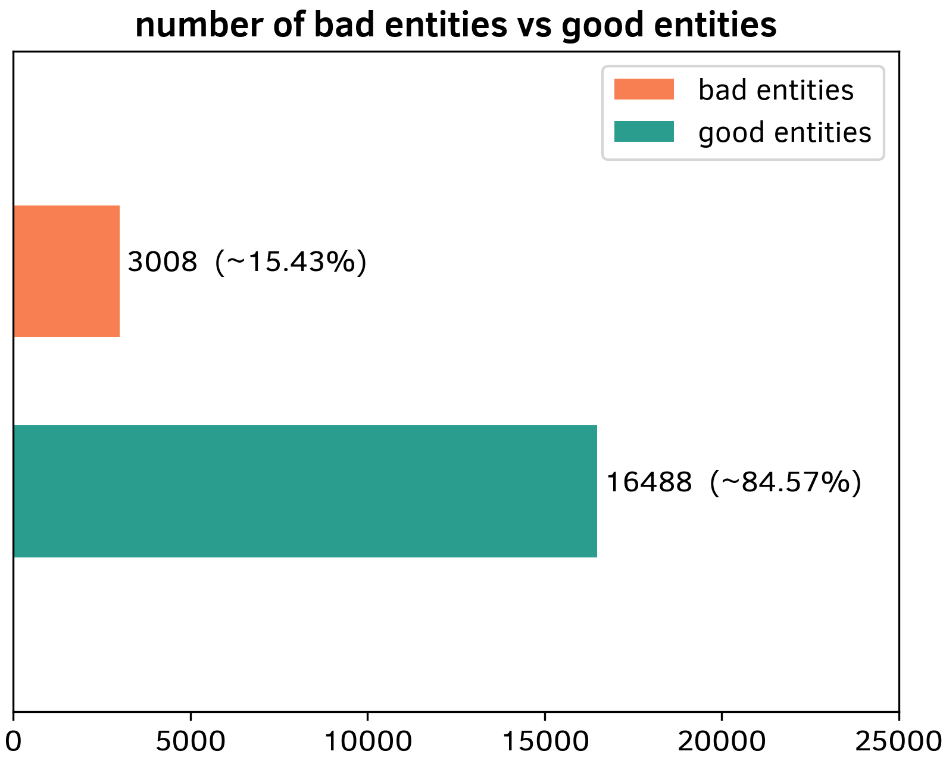

Figure 4.5: Bad entities vs good entities

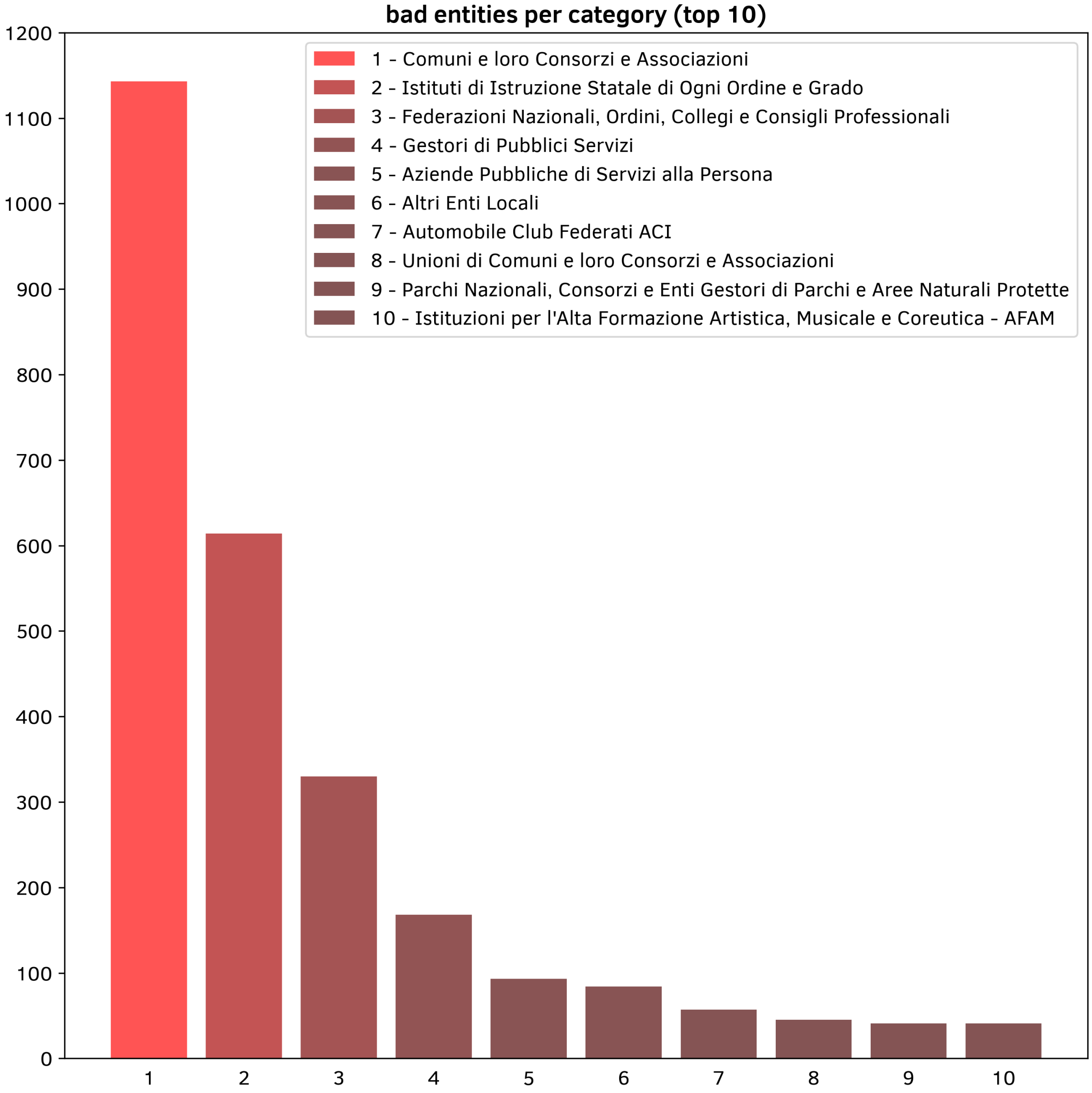

Figure 4.6: Bad entities per category (top 10)

Italian municipalities did not provide `https` access to their websites, while 36% did not specify the contact details of the Data Protection Officer, although the regulation was in force since 2015 [48]. Moreover, between 2020 and 2022, the Italian Garante per la Privacy, sanctioned several municipalities for unlawful processing or diffusion of citizens' personal data [49] [50] [51]. In general, Italian municipalities seem to have difficulty complying with the GDPR.

Another interesting point is the presence, in second place, of the category *Istituti di istruzione statale di ogni ordine e grado* (i.e. educational institutions). Several sources have reported an increase in the use by Italian educational institutes of services offered by the US big tech companies in recent years, in particular after the COVID-19 global pandemic. As Paolo Monella says, in its article «*Education and GAFAM: from awareness to responsibility*» [52]:

> *Since the outburst of the COVID-19 pandemic, school and college remote teaching in Italy was based almost exclusively on the infrastructures and platforms developed by 'big tech' multinationals, such as Google (G-Suite for Education), Microsoft (Teams) and Zoom. This raises issues concerning student personal data protection that have been largely underestimated in the public discourse.*

Even if many free and open source teaching tools exist, Italian schools mainly rely on external service providers from the US. In March 2021 the "Commissione Bilancio" of the Senate of the Italian Republic has approved an amendment which funds the UNIRE project, which includes, among other things, the institution of a Ministry of Education structure for the operation of integrated digital teaching and the provision of services related to those activities. At the time of writing, the project is not yet operational, therefore many schools still do not have the necessary support to offer their digital services without recourse to the US big tech companies.

One interesting fact: Paolo Monella cites the Politecnico di Torino, which during the COVID-19 emergency relied only on open technologies for teaching activities, as a positive example of digital independence.

4.2.4 Bad requests grouped by domain, group, company

We will now try to look at the data from another point of view: we will not focus on the sources of the requests (i.e. the entities) but rather on the destinations (i.e. the domains).

First of all, we grouped the domains per number of requests received. Figure 4.7 shows the absolute values, while Figure 4.8 shows the percentages.

Then we considered the domains with more requests, the ones in Figure 4.7, and from the domains we moved to their groups. The groups we considered are listed in

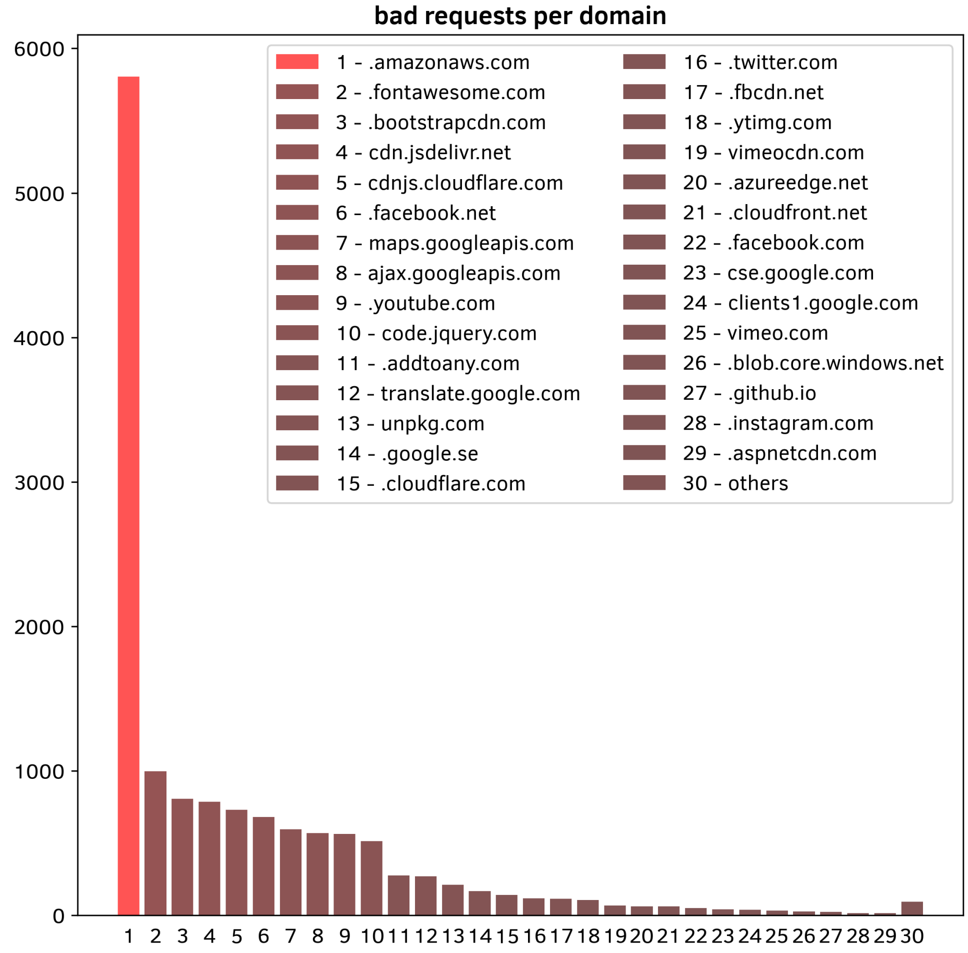

Figure 4.7: Bad requests per domain

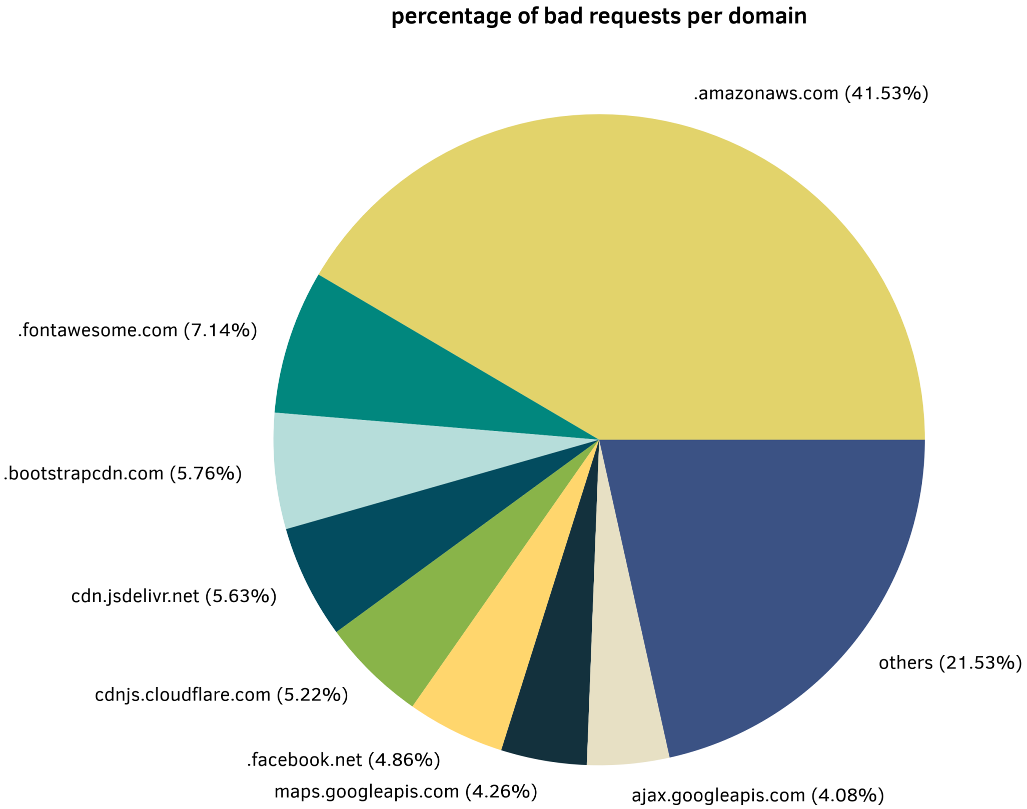

Figure 4.8: Percentage of bad requests per domain

Figure 4.9, where per each group you can read the list of domains which belong to that group. We used this list to produce Figure 4.10 and Figure 4.11 where requests are grouped by domain groups (again, absolute values and percentages).

At the end, we took the domain groups with more requests and traced the relative companies these groups belong to. We produced the JSON file in Figure 4.12 which contains, for each company, the list of groups owned by the company and the country in which the company is based.

To retrieve the names and countries of the companies behind the domain groups, we used a mix of different sources, namely Wikipedia, *opencorporates.com* [53], the *ipinfo.io* IP Geolocation API service [54] and the *webXray domain owner list*, a big hierarchical list which collects some third-party domains on the web and the owning companies [55].

```
{
    "Intuition Machines": ["hcaptcha"],
    "Adobe": ["adobe"],
    "Amazon": ["aws", "cloudfront"],
    "Fonticons": ["fontawesome"],
    "Facebook": ["facebook"],
    "Google": ["youtube", "googlemaps", "googlehostedlibraries", "googletranslate", "googlesearch", "googletagmanager"],
    "Cloudflare":["cdnjs", "unpkg", "cloudflarecdn", "turnstile"],
    "StackPath": ["jquery"],
    "AddToAny": ["addtoany"],
    "Twitter": ["twitter"],
    "Vimeo": ["vimeo"],
    "Microsoft": ["azure", "microsoft"],
    "Fastly": ["jsdelivr", "fastly"]
}
```

Figure 4.12: Companies

We used the JSON file to produce the two figures 4.13, and 4.14 in which requests are grouped by company. Please note how all the companies are from the US.

To explain these results we have to understand what are the services offered by these companies.

At the first place we have Amazon. As you can see from figures 4.7, 4.8, 4.9 and 4.11, most of the request (41.53%) are addressed to `.amazonaws.com` and the *aws* group. Amazon Web Services (AWS) is a subsidiary of Amazon which offers on-demand cloud computing services. Its main competitors are Microsoft Azure and Google Cloud Platform. Various

```
{
    "aws": [ ".amazonaws.com" ],
    "fontawesome": [ ".fontawesome.com", ".bootstrapcdn.com" ],
    "cdnjs": [ "cdnjs.cloudflare.com" ],
    "jsdelivr": [ "cdn.jsdelivr.net" ],
    "googlemaps": [ "maps.googleapis.com" ],
    "googlehostedlibraries": [ "ajax.googleapis.com" ],
    "jquery": [ "code.jquery.com" ],
    "addtoany": [ ".addtoany.com" ],
    "googletranslate": [ "translate.google.com" ],
    "unpkg": [ "unpkg.com" ],
    "google": [ ".google.se" ],
    "cloudflarecdn": [ ".cloudflare.com" ],
    "twitter": [ ".twitter.com" ],
    "youtube": [ ".youtube.com", ".ytimg.com" ],
    "vimeo": [ "vimeocdn.com", "vimeo.com" ],
    "azure": [ ".azureedge.net", ".blob.core.windows.net" ],
    "cloudfront": [ ".cloudfront.net" ],
    "facebook": [ ".facebook.com", ".facebook.net", ".instagram.com", ".fbcdn.net" ],
    "googlesearch": [ "cse.google.com", "clients1.google.com" ],
    "microsoft": [ ".github.io", ".aspnetcdn.com" ]
}
```

Figure 4.9: Groups and their domains

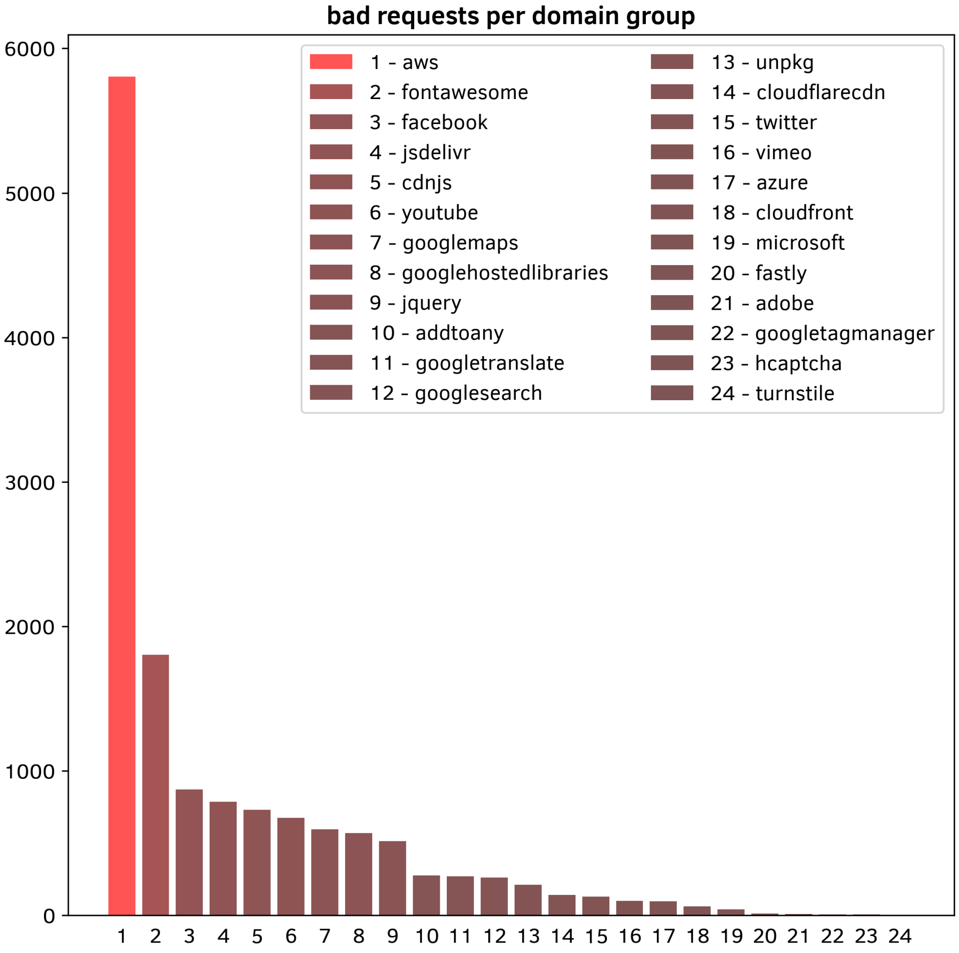

Figure 4.10: Bad requests per domain group

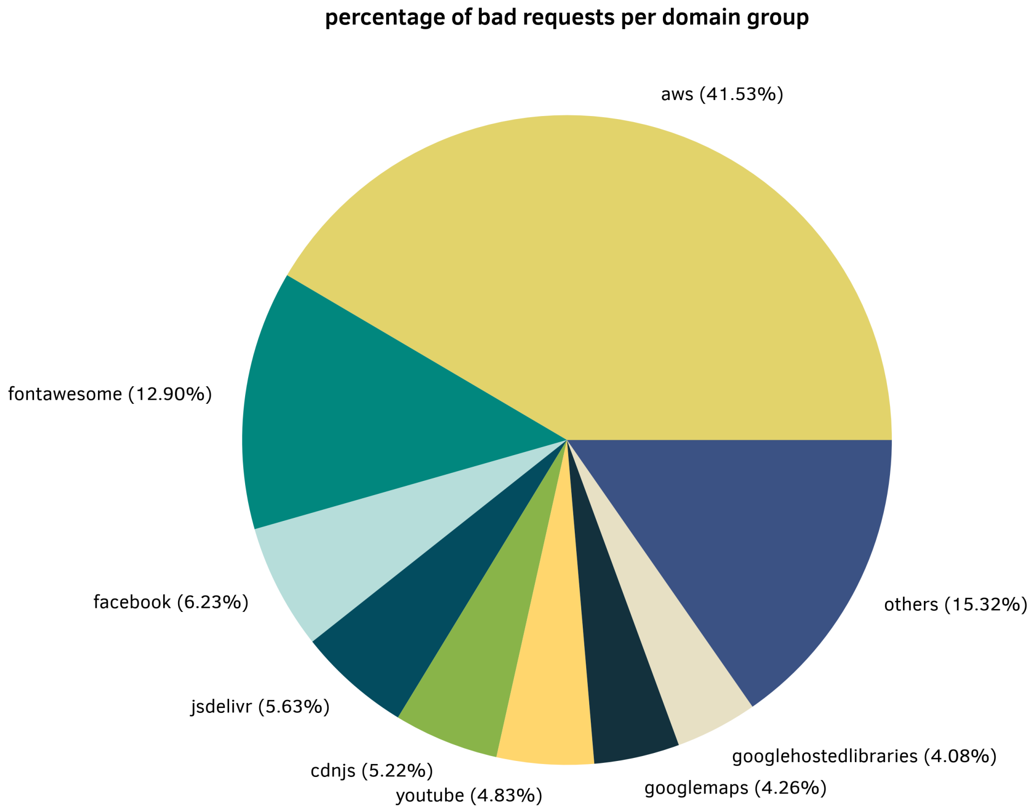

Figure 4.11: Percentage of bad requests per domain group

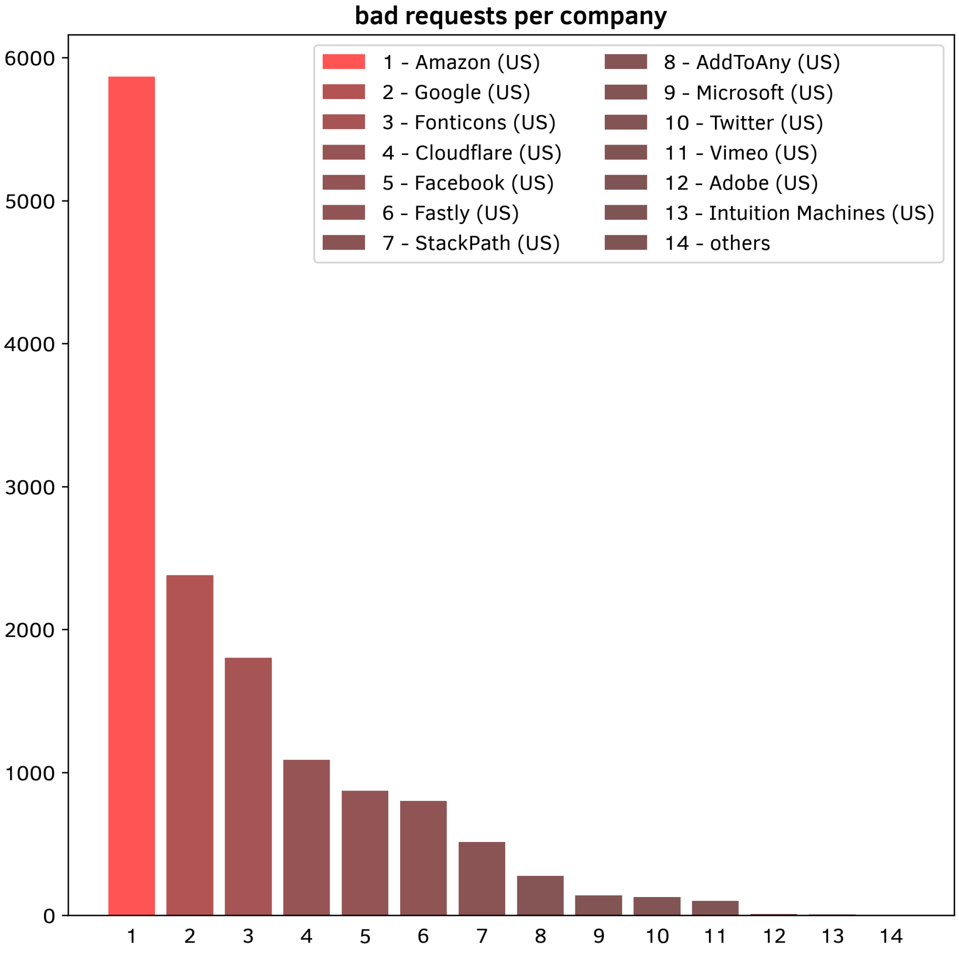

Figure 4.13: Bad requests per company

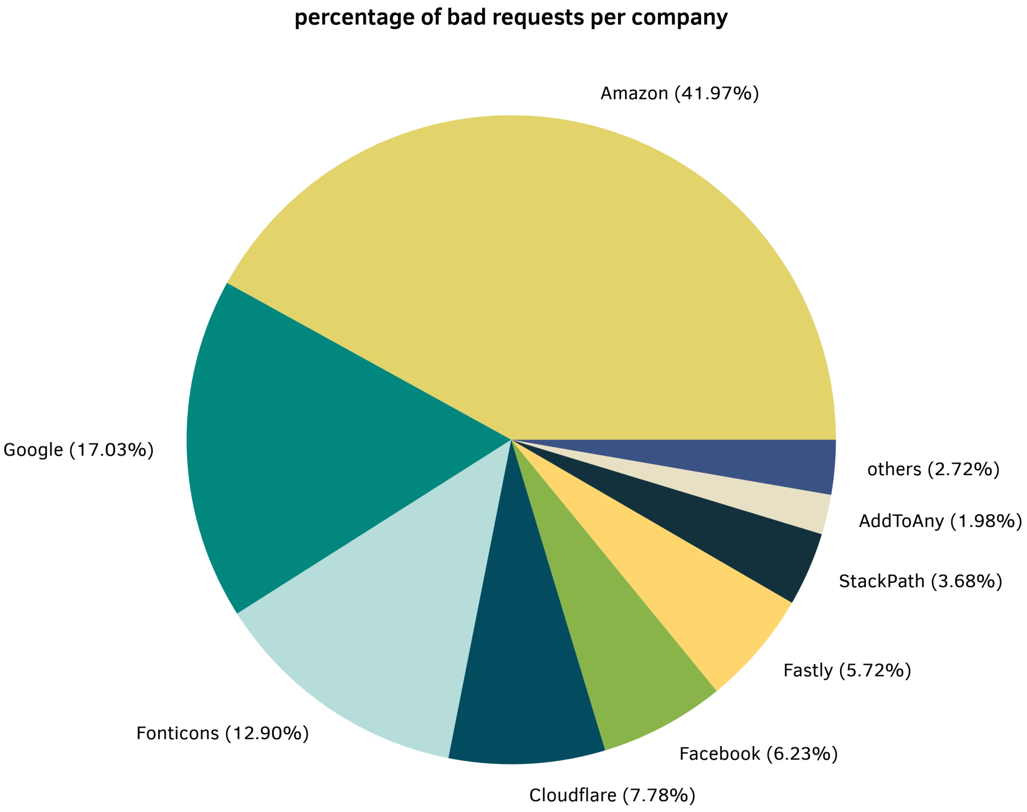

Figure 4.14: Percentage of bad requests per company

online sources report AWS as the leading provider of cloud services globally [56]. Its popularity seems to be due to several factors:

- AWS is the oldest big cloud service provider on the market: it was launched publicly in 2006.
- AWS offers a wide range of cloud services, including computing, storage, databases, machine learning, analytics, Internet of Things (IoT), and more. The competitors only offer a subset of those services.
- AWS offers competitive pricing and a pay-as-you-go model, making it accessible to a wide range of users and businesses.

Even if there exist free and open source alternatives to some of the AWS cloud services [57] [58], not all the services are covered, so organizations and companies seem to prefer Amazon for its completeness and simplicity, while the self-hosted deployment instead requires advanced knowledge and skills, as well as high deployment costs, especially in the early stages of the business.

In addition to self-hosted solutions there are other viable alternatives: the *european-alternatives.eu* website lists several cloud service providers from the EU [59]. However, the numbers are still very low when compared to those of AWS.

Moreover, by 2029, Amazon has planned an investment of 2 billions euro in Italy, and there is a section of the AWS website specifically targeting public administrations [60]. In the end, it is not surprising that Italian PA entities make use of these services.

For what concerns the other entries within the plots, the data is in line with what we could expect. Right after Amazon we have Google, another popular big tech giant which offers a full series of services: maps, web search, website development tools, multimedia hosting, etc.

All the other companies offer mostly content delivery networks (e.g. Fastly), social networking and multimedia (e.g. Facebook), libraries (e.g. Fonticons) or other general services (e.g. Adobe).

To better understand which are the most required services by the entities, we retrieved the service types associated to the top bad domains (Figure 4.15), and we grouped the requests by using the type of service as key. The result is shown in Figure 4.16 and Figure 4.17.

As you can see from the chart most of the requests are for cloud computing services (42.22%) and content delivery networks (26.52%).

Once again, the conclusion is the same: Italian PAs still seem to prefer big service providers from the US rather than self-hosted solutions and EU providers. This could mean a lack of knowledge, but also a lack of infrastructures.

```
{
  "cloud computing": [ ".amazonaws.com", ".azureedge.net",
      ".blob.core.windows.net", ".azurewebsites.net", ".azure.net" ],
  "icons": [ ".fontawesome.com" ],
  "cdn": [ ".bootstrapcdn.com", "cdn.jsdelivr.net", "cdnjs.cloudflare.com",
      "ajax.googleapis.com", "code.jquery.com", "unpkg.com", ".cloudfront.net",
      ".aspnetcdn.com", "fastly.net" ],
  "social media": [ ".facebook.net", ".twitter.com", ".twimg.com",
      ".fbcdn.net", ".facebook.com", ".instagram.com", ".fbsbx.com" ],
  "maps": [ "maps.googleapis.com" ],
  "video": [ ".youtube.com", ".ytimg.com", "vimeocdn.com", "vimeo.com",
      ".youtube" ],
  "translation": [ "translate.google.com" ],
  "search engine": [ ".google.se", "cse.google.com", "clients1.google.com",
      ".google.it", ".bing.com", ".google.com.co", ".google.com.pa",
      ".google.com.bo" ],
  "web hosting": [ ".github.io", ".githubusercontent.com" ],
  "various": [ ".adobe.com", ".cloudflare.com", ".googletagmanager.com",
      ".addtoany.com", ".omtdrc.net", ".skypeassets.com" ],
  "captcha": [ "challenges.cloudflare.com", ".hcaptcha.com" ]
}
```

Figure 4.15: Types of services and relative domains

4.3 Limitations

The results that we obtained should not be considered as 100% accurate, but rather as an underestimate of the reality. Within this section we will explain why.

4.3.1 Dynamic components loading

As we said before, the *minos-cli* script needs a running instance of a Chromium-based web browser to navigate the websites. The script spawns a new tab of the browser which is immediately closed after the connection to the website is established (i.e. when the `Network.loadingFinished` event occurs). This is an automatic process, i.e. it does not require user interaction.

In some cases this is not enough to detect the presence of bad domains within the website, since many requests only occur when certain components of the webpage are

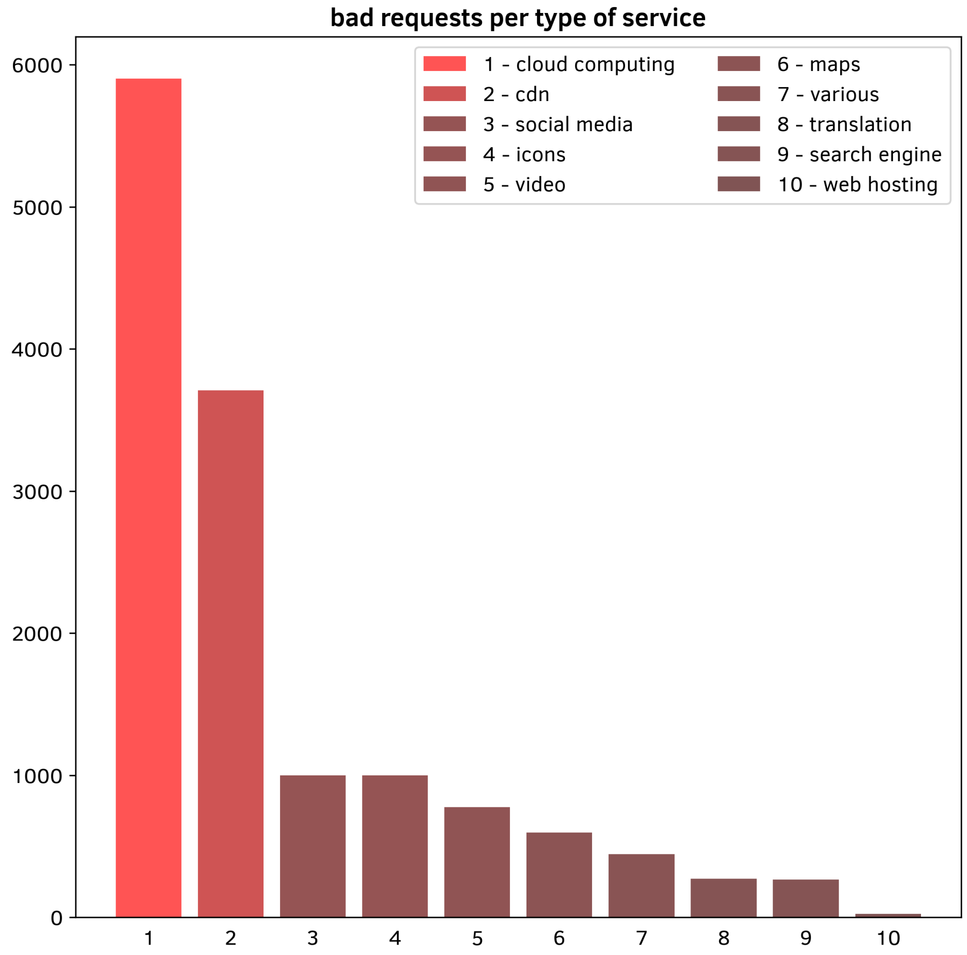

Figure 4.16: Bad requests per type of services

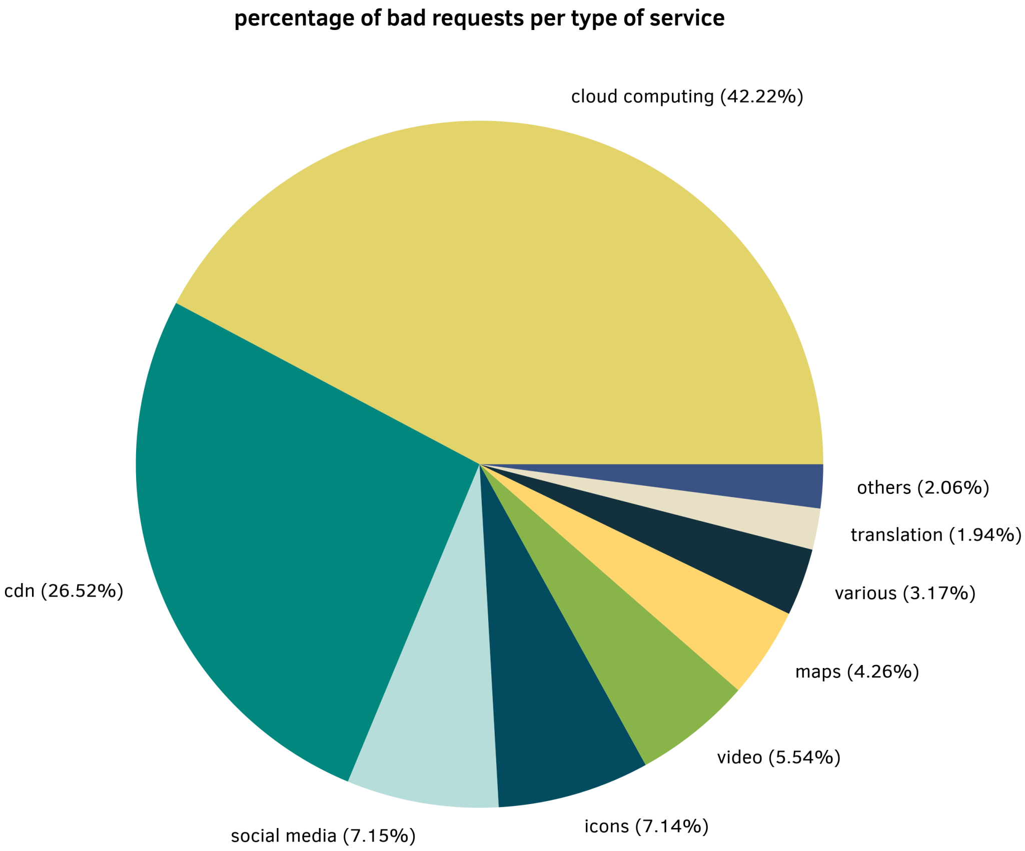

Figure 4.17: Percentage of bad requests per type of service

loaded, and these components are dynamically loaded when the user actually interacts with the webpage.

We experienced this behavior with some websites: the *minos-cli* script did not spot the presence of bad domains within the website, whereas the GUI version did. In particular, requests directed to the `yandex.com` Russian domain were detected while navigating the website of one of the entities within the dataset. Here is why we cannot exclude the presence of other domains besides those from the US.

Obviously, due to the huge number of websites to be analyzed, it would not have been possible to run the same analysis by hand, as this would have taken much longer. However, a good compromise could be the usage of browser automation tools like Selenium [61].

4.3.2 Homepage-only analysis

Moreover, it should be remembered that we are not analyzing the entire website, but just the home page. Different pages within the website hierarchy could send requests to different domains.

Analyzing the whole website would require a mix of web scraping techniques that go beyond the scope of this thesis work.

4.4 Similar works

Automated GDPR compliance assessment for cross-border personal data transfers in android applications [62] is a very interesting work in which the researchers focused on the world of mobile applications.

Very similarly to what we have done for the websites of the Italian PAs, they presented and applied an automated method for assessing the compliance of Android application with the GDPR requirements for cross-border personal data transfers. They discovered that nearly half of the examined applications sending personal data were potentially non-compliant with GDPR requirements.

We can draw some interesting insights from the research:

- The analysis of the privacy policies of the applications.
- webXray [46] and the webXray domain owner list [55] to analyze an classify the domains.
- *ipinfo.io* [54] IP Geolocation API service to detect the location of the servers receiving the app's connections.

- A man-in-the-middle (MITM) proxy to intercept the network traffic.

Chapter 5

Conclusion and future works

In this work, we focused on the problem of personal data transfers from the European Economic Area to external countries not subject to adequacy decision.

We first presented the legal scenario in which we are, highlighting the GDPR rules which define the behaviours to be adopted when making personal data transfers from the EEA to third countries. We explained what an adequacy decision is, why it can be used as a prerequisite for data transfers, and what are the alternatives. We gave particular attention to data transfers to the US, since they are very important at this specific time, especially in the light of the recent adequacy decision about the EU-US Data Privacy Framework.

Subsequently, we explained the stages that led to the development of Minos, a software tool which allows non-expert users to navigate the web, recognize illegal requests from Italian websites to non-EEA domains and generate complaints which can be submitted to the Garante della Privacy. We showed in detail all the development steps, from the requirements analysis to the implementation, by discussing all the choices we made, with their pros and cons. We also introduced the cli version of Minos, which consists of a set of analysis-oriented scripts. We described each of the scripts, their inputs and outputs, the libraries they are based on, their performances, etc. At the end we proposed a comparison with similar software based on different parameters, in which we concluded that, at the time of writing, there are no equivalent applications with the same structure of Minos.

Finally, we showed the results of the analysis that we ran on the Italian PA websites. We found that all the domains with at least one request were from the US, among which Amazon domains are by far the most popular. In general, cloud computing services are the most requested ones, followed by cdn, social media and icon library services. We discovered that the categories of entities with the most violations are municipalities and schools. After giving some measurements of data quality, we used bar and pie charts to display the statistics that we computed on the datasets, jointly with some possible explanation of our results. In particular, we tried to give an explanation of Amazon Web Services' popularity, and we identified three distinctive strengths that distinguish it from its competitors: first, AWS is the oldest big cloud service provider on the market; second, AWS offers a wide range of services and products; three, AWS offers a competitive price and a very accessible payment model. At the end we discussed the limitations of our approach, and we offered a comparison between our work and a similar one.

After nearly three years the Privacy Shield was declared illegal, many Italian PA entities still have difficulties in complying with the GDPR. Most of the requests analyzed are for essential services, such as cloud computing and content delivery networks, which in many cases require lots of computational power and big infrastructures. Even if some alternatives exist in the EU, numbers are still low, and the self-hosting does not seem a popular alternative, since it requires advanced knowledge and, at least at the beginning, larger investments. Some measures oriented towards the digital independence have been announced, like the UNIRE project, which involves the institution of a Ministry of Education structure for the operation of integrated digital teaching and the provision of services related to those activities. However, it seems that there are currently no particular incentives for the PA's organizations to move away from the big US service providers.

In conclusion, we would like to present some interesting insights for possible future works:

- Even if Minos is currently able to save cookies in the HAR file, there are no controls on them. A possible improvement would be to check the domain field of the cookie against the hosts in the blacklist and report them to the user, possibly also detecting whether they are installed without the user's explicit consent.

- We could use browser automation tools like Selenium [61] to simulate the user interaction with the web pages, and see if that produces more requests to blacklisted domains. This could possibly improve the results of the analysis.

- Article 13 of the GDPR specifies all the information that each entity collecting personal data should provide. In particular, the controller shall provide *where applicable, the fact that the controller intends to transfer personal data to a third country or international organization and the existence or absence of an adequacy decision by the Commission, or in the case of transfers referred to in Article 46 or 47, or the second subparagraph of Article 49(1), reference to the appropriate or suitable safeguards and the means by which to obtain a copy of them or where they have been made available.*.

An interesting extension of the analysis work could be to check whether the websites are correctly including this information into their privacy policy, and whether the data collection and processing is actually in line with what is stated.

- We could make a list of alternative service providers from the EU (including for example, but not only, the ones indicated by *european-alternatives.com*) and check how many websites from the Italian PA are relying on their services, to detect not only the bad entities, but also the good ones, and then make a comparison.

Bibliography

[1] https://www.theguardian.com/world/2013/jun/09/edward-snowden-nsa-whistleblower-surveillance, *Edward Snowden: the whistleblower behind the NSA surveillance revelations.*

[2] https://www.nytimes.com/2013/06/08/technology/tech-companies-bristling-concede-to-government-surveillance-efforts.html, *Tech Companies Concede to Surveillance Program.*

[3] https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election, *Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach.*

[4] European Union Agency for Fundamental Rights. https://fra.europa.eu/en/news/2020/how-concerned-are-europeans-about-their-personal-data-online, *How concerned are Europeans about their personal data online?*

[5] Bernhard Debatin, Jennette P. Lovejoy, Ann-Kathrin Horn, and Brittany N. Hughes. Facebook and Online Privacy: Attitudes, Behaviors, and Unintended Consequences. *Journal of Computer-Mediated Communication*, 15(1):pages 83 (2009). ISSN 1083-6101. doi:10.1111/j.1083-6101.2009.01494.x.

[6] https://www.enforcementtracker.com/?insights, *GDPR Enforcement Tracker - Statistics.*

[7] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32016R0679&qid=1691485867121, *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance).*

[8] The European Parliament and the Council of the European Union. Processor general obligations. *Regulation (EU) 2016/679*, page 49 (2016).

[9] https://commission.europa.eu/law/law-topic/data-protection/reform/rules-business-and-organisations/obligations/what-rules-apply-if-my-organisation-transfers-data-outside-eu_en, *What rules apply if my organisation transfers data outside the EU?*

[10] The European Parliament and the Council of the European Union. Transfers on the basis of an adequacy decision. *Regulation (EU) 2016/679*, page 61 (2016).

[11] https://commission.europa.eu/law/law-topic/data-protection/international-dimension-data-protection/adequacy-decisions_en, *Adequacy decisions - How the EU determines if a non-EU country has an adequate level of data protection.*

[12] The European Parliament and the Council of the European Union. Transfers subject to appropriate safeguards. *Regulation (EU) 2016/679*, page 62 (2016).

[13] The European Parliament and the Council of the European Union. Derogations for specific situations. *Regulation (EU) 2016/679*, pages 64–65 (2016).

[14] https://monitora-pa.it/, *Monitora PA - Osservatorio Automatico Distribuito sulla PA.*

[15] https://web.archive.org/web/20060724174359/http://www.ec.europa.eu/justice_home/fsj/privacy/docs/adequacy/sec-2002-196/sec-2002-196_en.pdf, *The application of Commission Decision 520/2000/EC of 26 July 2000 pursuant to Directive 95/46 of the European Parliament and of the Council on the adequate protection of personal data provided by the Safe Harbour Privacy Principles and related Frequently Asked Questions issued by the US Department of Commerce.*

[16] https://en.wikipedia.org/wiki/Global_surveillance_disclosures_(2013%E2%80%93present), *Global surveillance disclosures (2013–present).*

[17] Court of Justice of the European Union. Judgment C-362/14 (“Schrems I”). *https://eur-lex.europa.eu/* (2015). https://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX:62014CJ0362.

[18] Article 29 Data Protection Working Party. https://web.archive.org/web/20160413231340/http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2016/wp238_en.pdf, *Opinion 01/2016 on the EU – U.S. Privacy Shield draft adequacy decision.*

[19] Court of Justice of the European Union. Judgment C-311/18 (“Schrems II”). *https://eur-lex.europa.eu/* (2020). https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:62018CJ0311.

[20] https://en.wikipedia.org/wiki/FISA_702#Section_702:_Non_U.S._persons, *FISA 702: Non U.S. persons.*

[21] https://en.wikipedia.org/wiki/EO_12333, *Executive Order 12333.*

[22] https://www.politico.eu/article/us-eyes-breakthrough-on-data-dispute-with-eu-biden-visit-privacy-shield-ukraine/, *US eyes breakthrough on data dispute with EU as Biden visits Brussels.*

[23] https://www.dataprivacyframework.gov/s/, *Welcome to the Data Privacy Framework (DPF) program.*

[24] Max Schrems. European Commission gives EU-US data transfers third round at CJEU. *noyb.eu* (2023). https://noyb.eu/en/european-commission-gives-eu-us-data-transfers-third-round-cjeu.

[25] https://federicoleva.eu/en/, *Federico Leva’s website.*

[26] https://builtwith.com, *BuiltWith.*

[27] https://monitora-pa.it/faq.html, *MonitoraPA - FAQ.*

[28] https://github.com/MonitoraPA/monitorapa/, *MonitoraPA - GitHub.*

[29] https://pencil.evolus.vn/, *Pencil Project.*

[30] https://www.electronjs.org/docs/latest/tutorial/process-model, *Electron Process Model.*

[31] https://www.electronjs.org/docs/latest/api/debugger, *Debugger API.*

[32] https://chromedevtools.github.io/devtools-protocol/tot/Network/, *Chrome DevTools Protocol - Network APIs.*

[33] https://github.com/cyrus-and/chrome-har-capturer, *Chrome HAR Capturer.*

[34] https://pdfkit.org/, *PDFKit.*

[35] https://w3c.github.io/web-performance/specs/HAR/Overview.html, *HTTP Archive (HAR) format - Historical Draft* (2012).

[36] https://indigo.re/posts/2020-10-09-har-is-clumsy.html, *Why HAR format is clumsy.*

[37] https://www.w3.org/community/bigdata-tools/files/2017/10/HAR_Spec_TO_HAR_Vocabulary.pdf, *HAR Vocabulary.*

[38] https://en.wikipedia.org/wiki/Acceptance_testing#User_acceptance_testing, *Wikipedia - User Acceptance Testing.*

[39] https://eff.org, *Electronic Frountier Foundation.*

[40] https://globalprivacycontrol.org/, *Global Privacy Control.*

[41] https://www.eff.org/issues/do-not-track, *Do Not Track.*

[42] https://themarkup.org/blacklight, *The Markup - Blacklight.*

[43] https://www.immuniweb.com/websec/, *ImmuniWeb Security Test.*

[44] https://2gdpr.com/, *2gdpr.*

[45] https://complianz.io/why-online-privacy-testing-tools-are-not-accurate/, *Complianz - Why online privacy testing tools are not accurate.*

[46] https://webxray.org/, *webXray.*

[47] https://indicepa.gov.it/ipa-dati/, *OpenData IPA.*

[48] https://www.federprivacy.org/informazione/societa/privacy-il-47-dei-siti-dei-comuni-italiani-e-a-rischio-hacker, *Privacy, il 47% dei siti dei comuni italiani è a rischio hacker.*

[49] https://www.garanteprivacy.it/web/guest/home/docweb/-/docweb-display/docweb/9440025, *Ordinanza ingiunzione nei confronti del Comune di Greve in Chianti.*

[50] https://www.garanteprivacy.it/home/docweb/-/docweb-display/docweb/9587053, *Ordinanza ingiunzione nei confronti di Comune di Palermo.*

[51] https://www.garanteprivacy.it/web/guest/home/docweb/-/docweb-display/docweb/9789899, *Ordinanza ingiunzione nei confronti di Comune di Afragola.*

[52] Paolo Monella. Education and GAFAM: from awareness to responsibility. *Umanistica Digitale*, 5(11):page 27–45 (2021). doi:10.6092/issn.2532-8816/13685.

[53] https://opencorporates.com, *Open Corporates.*

[54] ipinfo.io/products/ip-geolocation-api, *ipinfo - IP Geolocation API.*

[55] https://github.com/PrivApp/webXray_Domain_Owner_List, *webXray Domain Owner List.*

[56] Synergy Research Group. https://www.srgresearch.com/articles/huge-cloud-market-is-still-growing-at-34-per-year-amazon-microsoft-and-google-now-account-for-65-of-all-cloud-revenues, *Huge Cloud Market Still Growing at 34% Per Year; Amazon, Microsoft & Google Now Account for 65% of the Total.*

[57] https://github.com/guenter/aws-oss-alternatives, *Open Source Alternatives to AWS Services.*

[58] https://en.wikipedia.org/wiki/OpenStack, *OpenStack - Wikipedia.*

[59] https://european-alternatives.eu/category/cloud-computing-platforms, *European Alternatives to AWS.*

[60] https://aws.amazon.com/it/stateandlocal/, *AWS Cloud per la Pubblica Amministrazione statale e locale.*

[61] https://www.selenium.dev/, *Selenium - Browser automation tool.*

[62] Danny S. Guamán, David Rodriguez, Jose M. del Alamo, and Jose Such. Automated GDPR compliance assessment for cross-border personal data transfers in android applications. *Computers & Security*, 130:page 103262 (2023). ISSN 0167-4048. doi: https://doi.org/10.1016/j.cose.2023.103262.