Automating Cookie Consent and GDPR Violation Detection

Dino Bollinger, Karel Kubicek, Carlos Cotrini, and David Basin, *ETH Zurich*

https://www.usenix.org/conference/usenixsecurity22/presentation/bollinger

This artifact appendix is included in the Artifact Appendices to the Proceedings of the 31st USENIX Security Symposium and appends to the paper of the same name that appears in the Proceedings of the 31st USENIX Security Symposium.

August 10–12, 2022 • Boston, MA, USA

978-1-939133-31-1

C Artifact Appendix

C.1 Abstract

Our work in this paper consists of four separate components:

1. **The cookie consent web crawlers.** The web crawler component uses a series of Python scripts and the OpenWPM framework to gather browser cookies and associated purpose categories from websites. The output of this component is a dataset of browser cookies including category labels.
2. **The feature extraction and XGBoost classifier.** This component uses the collected dataset of cookies and transforms it into a sparse matrix representation, using all properties of a browser cookie in the process. This sparse matrix, combined with the category labels, is then used to train a decision tree model using the XGBoost algorithm. This allows us to predict purpose categories for previously unseen cookies.
3. **The GDPR violation detection scripts.** Using knowledge of the articles of the GDPR and the cookie dataset collected by the consent web crawler, these scripts identify potential GDPR violations on websites in the wild. The output of this component is a dataset of statistics detailing the prevalence of potential GDPR violations, based on 8 different methods of analysis.
4. **The "CookieBlock" browser extension.** This addon provides a privacy protection mechanism for users which automatically deletes cookies that they did not consent to. The extension uses the classifier as the central engine to decide which cookie belongs to what category. It supports Chromium-based browsers as well as Firefox.

The components have been constructed using Python 3 and JavaScript. The webcrawler in particular is based on the OpenWPM framework version 0.12.0, and must be run on Linux. For this reason, we provide an Ubuntu VM that comes with all dependencies preinstalled. We also provide a precomputed dataset of statistics and metrics which stem from our previous executions of these components, and are the datasets used for the results presented in the paper. This includes the candidate domains used for the web crawl, the complete set of performance metrics for the XGBoost classifier and the Cookiepedia baseline, as well as all statistics and data on the GDPR Violation Detection.

No specialized hardware is required to reproduce the results of the paper, but at least 8GB of RAM and 40 GB of disk space are needed. Due to the nature of the dataset collection, results may differ significantly if reproduced at a later date. Instructions on how to compare and validate the results are provided in the form of a detailed “README” document, containing a step-by-step guide detailing each part of the process. Said document also provides links to the source code release for each component.

C.2 Artifact check-list (meta-information)

- **Binary:** Cross-platform virtual machine image, containing all program components and datasets.
- **Data set:** Yes, included. The data set and VM are found at: https://doi.org/10.5281/zenodo.5838646
- **Run-time environment:** The OpenWPM crawler only runs on Linux. The other scripts and the browser extension work on Windows and Linux. An Ubuntu VM image is included.
- **Hardware:** At least 8GB of RAM needed, and approximately 40 GB of disk space. Additional CPU cores can speed up the computation, but works with a single core also.
- **Run-time state:** The results are dependent on the website content, as well as the CMP implementations, which may change over time, and are out of our control.
- **Execution:** With the complete input dataset, the web crawls alone may take between 1 and 2 weeks to complete. With a reduced dataset, the full process takes a few hours.
- **Metrics:** Accuracy, precision, recall, macro-precision, macro-recall and F1 score.
- **Output:** Printed to the console, stored in SQLite databases, JSON and log files. Expected results are included for each step of the process.
- **Experiments:** Collection of the browser cookie dataset, the training and evaluation of the classifier, the GDPR Violation detection and the generation of the extension’s classifier model can all be replicated using commands manually input by the user. We provide a detailed step-by-step guide on the process.
- **How much disk space required (approximately)?:** At least 40 GB is required for the VM. While the included datasets are much smaller than this, the data that is collected and generated may quickly take up disk space.
- **How much time is needed to prepare workflow (approximately)?:** When installing the VM image, only a few minutes. When setting the scripts up natively, at most an hour.
- **How much time is needed to complete experiments (approximately)?:** A few hours.
- **Publicly available?:** Yes, all components are publicly available on Github. Links are provided in the step-by-step guide.
- **Code licenses (if publicly available)?:** The OpenWPM crawler is GPL3 licensed. Other components are MIT licensed.
- **Data licenses:** CC by 4.0 International
- **Archived:** Yes, available at: https://doi.org/10.5281/zenodo.5838646

C.3 Description

C.3.1 How to access

The artifact is publicly available and can be downloaded as a self-contained package from:
https://doi.org/10.5281/zenodo.5838646
It includes a VM image that has all components preinstalled, as well as a README that guides the user to replicate and

reproduce the results. The document also contains links to the original repositories, should the user intend to install the scripts natively.

C.3.2 Hardware dependencies

The artifact requires no specialized hardware to run. A single core machine with 8GB of RAM and mor than 40 GB of disk space should be enough. The VM requires considerable size when set up, which is due to the libraries that are used, and because of the data collection that needs to be performed to replicate the results.

C.3.3 Software dependencies

If the VM image is used, only a virtualization product such as VirtualBox or VMWare is required. All other components should be ready to use. For native installations, some Python and Node libraries are required. The exact details are provided within the step-by-step guide included as part of the artifact.

C.4 Installation

The recommended method of setting up the artifact is to load the virtual machine image using VirtualBox. All further steps are documented in great detail within the README file of the artifact. In the interest of space, we will not repeat the steps here, and instead refer to the README.

C.5 Evaluation and expected results

First, we crawled 6M domains from a Tranco list collected on May 5th. Out of these, 30k were found to have the selected CMPs on them. From these websites, we collected a ground truth of 304k cookies with labels, which we used to train an XGBoost model with 84.4% weighted accuracy. In an analysis of the 30k websites, we found that a vast majority, namely 94.7% of them, contain at least one potential privacy violation. All the steps to reproduce these results together with the intermediate files of our results are documented in great detail within the README file of the artifact.

Note that the changes to websites content cause variance in the results. We try to document this variance below:

1. **Variance for the cookie consent web crawlers.** Within the large Tranco list, the number of websites with CMPs remains roughly the same over time. Among the more popular sites, the percentage of websites using the selected CMPs is higher, allowing the use of smaller input files. In the paper, we observed suitable CMPs on 0.63% of the Tranco 6M list (see Sections 2.1 and 2.2 of the paper). In the Master Thesis report, it was 1.6% for Tranco 1M Worldwide or 1.25% for Tranco Europe, and BuiltWith website reports the selected CMPs in over 3% of the top 1M websites. We observed on average 22 cookies with label per website, which depends strongly on the number of sub-pages visited for each site (discussed in the par. 3 of Section 2.3. of the paper). We did not measure the variance for the settings in the crawler, but the results should be consistent as long as you run the provided crawler from within EU.
2. **Variance in the XGBoost classifier.** The feature extraction is deterministic, extracting the same features with each execution. Training the model appears to be stable, as we observe a standard deviation of 0.23% in the accuracy. The model's balanced accuracy will drop from the reported 84.4% if you use a smaller training dataset. Additional standard deviations for each metric are provided in the dataset.
3. **Variance in the GDPR violation detection scripts.** The observed violations depends on website selection, but the results between the master thesis report and the paper varied by 4% for the number of websites with at least one type of violation. For individual violations this variance can be higher.