HAL
archives-ouvertes.fr

# Evaluating Websites and Their Adherence to Data Protection Principles: Tools and Experiences

Amelia Andersdotter, Anders Jensen-Urstad

▶ To cite this version:

Amelia Andersdotter, Anders Jensen-Urstad. Evaluating Websites and Their Adherence to Data Protection Principles: Tools and Experiences. Anja Lehmann; Diane Whitehouse; Simone Fischer-Hübner; Lothar Fritsch; Charles Raab. Privacy and Identity Management. Facing up to Next Steps : 11th IFIP WG 9.2, 9.5, 9.6/11.7, 11.4, 11.6/SIG 9.2.2 International Summer School, Karlstad, Sweden, August 21-26, 2016, Revised Selected Papers, AICT-498, Springer International Publishing, pp.39-51, 2016, IFIP Advances in Information and Communication Technology, 978-3-319-55782-3. 10.1007/978-3-319-55783-0_4 . hal-01629154

HAL Id: hal-01629154
https://hal.inria.fr/hal-01629154

Submitted on 6 Nov 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution| 4.0 International License

# Evaluating Websites and Their Adherence to Data Protection Principles: Tools and Experiences

**Contributions to IFIP Summer School Proceedings**

Amelia Andersdotter and Anders Jensen-Urstad *

Dataskydd.net, c/o Jensen-Urstad
Alsnögatan 18, 116 41 Stockholm, Sweden
{amelia.andersdotter,anders.jensen-urstad}@dataskydd.net
https://dataskydd.net

**Abstract.** We present our two separate tools for data protection measurement and evaluation of websites. The first tool does a generic check on a single website and is openly available for any web user to use when evaluating data protection measures implemented on a website. The second tool was used to perform a more exhaustive evaluation of Swedish municipalities. The work focuses on leakages of personally identifiable information to third parties when a web visitor goes to a website, and in our accompanying website we have also identified measures that web developers could undertake, or that web visitors could request, to improve the data protection of their visitors.

**Keywords:** privacy, metrics, data protection, web tools, guidelines, policy procurement

## 1 Introduction

With the entry into effect of the new EU rules on data protection in May 2018, there is an urgent need for web developers and other actors in society to make preparations. We have developed two separate tools to identify measures which can be undertaken to improve privacy protection in public websites in Sweden. In particular, they will assist web developers and web strategists to analyse how they are currently leaking personally identifiable information (PII) to third parties, as individuals visit their websites.

Our first tool is a simple technical mechanism for private persons and web developers to evaluate the leakage of PII from websites.[1] The second tool performs more substantive measurement of PII leakage from Swedish municipal

* Co-founders Dataskydd.net. This project was realised through the kind financial assistance provided by Internetfonden/IIS over the period January-August in 2016.

[1] *See* https://webbkoll.dataskydd.net/en/ [in English and Swedish].

websites.[2] In addition, we produced technical advice on how data leakage can be mitigated.[3]

We have checked the use of web analysis tools,[4] whether a referrer policy is set,[5] the usage or absence of encrypted connections as well as inclusion of third party services (fonts, forums, weather services or text-to-voice services) or the placement of cookies (both persistent tracking cookies and functional cookies).

Additionally, we provide an element of *gamification* for Swedish municipalities, in that we developed a five-step grading system. This allows for municipalities to compare their efforts to other municipalities. With current data leakage rates being generally high, we found, however, that no municipalities obtain better than mid-level results.

This study aims to achieve utility for web designers and website developers, with an emphasis on utility for public sector institutions. While there are a number of web browser plug-ins that could be installed by web visitors to mitigate harms arising from persistent online tracking,[6] this paper is guided by the belief that privacy problems can and should be solved close to the source of the problem. We believe similar methodologies and tests could also be useful in other countries, and in so far as possible we have strived to identify cost-neutral improvements.

### 1.1 Similar tools in the Swedish context

In Sweden, prior examples of web services for monitoring private and public sector compliance with applicable law include Hitta kakor[7] and extensive work in the field of accessibility[4, guideline 1].[8] More technically inspired guidelines include efforts to improve adoption rates of encrypted connections[4, guideline 7] and DNSSEC.[9]

Additionally there have been attempts in Sweden to compile guidelines for data protection on websites[4, guideline 20] and with respect to web cookies.[10]

[2] *See* `https://dataskydd.net/kommuner/` [only in Swedish].

[3] In the right-hand column of the test-results accompanying the generic web-check on the website listed in footnote 1, or under the heading "Begrepp och tips" on the website listed in footnote 2.

[4] Google Analytics, Adobe Tealeaf, Piwik, et c.

[5] *Cf.* `https://www.w3.org/TR/referrer-policy/`

[6] Some alternatives include Cookie White List, Privacy Badger, various adblocking applications (such as Adblock Plus or uBlock Origin), Ghostery, RequestPolicy, NoScript, HTTPS Everywhere, uMatrix, Disconnect, Decentraleyes, and similar. In addition to these tools web visitors may opt to use private browsing mode, which is increasingly included by default in most major browsers.

[7] PTS, `http://e-tjanster.pts.se/internet/kakor/`

[8] *See also* `http://www.anvandningsforum.se/om/`

[9] PTS, information webpage about ongoing work to promote DNSSEC with Swedish public authorities. `https://www.pts.se/sv/Bransch/Internet/Robust-kommunikation/Atgarder/DNSSEC/`

[10] PTS-F-2005:2.

Both of these guidelines target the necessary requirements for end-user terms of service formulations given specific options implemented by the web developer.

The success rate in terms of increasing adoption rates of globally desired practises varies.

Accessibility guidelines have generally been well-received.[11] While security-oriented projects, such as the guidelines for encrypted connections and DNSSEC are ostensibly taken seriously at the national level, deployment has been slow.[12]

All of the prior guideline projects have been developed under the auspice of the Swedish national regulatory authority for telecommunications and postal services (PTS). The data protection guidelines focus on contract law, and their success rate is difficult to measure. There are no known cases where the privacy terms of a municipal website or other website have been successfully tried in court.

## 2 Methodology

We have constructed tests which are technically more convenient to monitor, and which assume an expansive view of the wordings of the legislation. What this means is that our technical monitoring of websites does not accomodate for flexibilities in the law to avoid certain types of data forwarding or data collection by referens to privacy terms. This is for a few reasons.

- End-users have been demonstrated not to reasonably have the time or capacity to read and understand such terms of services.[13]
- Having restricted ourselves to investigating municipalities in depth, we deemed it inappropriate to construct a set of tests which would assume the public sector should enter into extensive and difficult to understand agreements with their website visitors.
- , We have assumed that it is more interesting for web developers to engage with interesting technical alternatives to monitoring and tracking over engaging with the formulation of terms of services agreements.

Instead we have focused on the ability of web developers to adhere to the data minimisation principle.[14] We have also assumed that adoption of data protection enhancing measures is more likely if website owners and managers are provided with self-measurement and top-listing tools.

[11] PTS, PTS-ER-2016:19, p. 57.

[12] *Cf.* `https://dataskydd.net/kommuner/` *and* `https://www.kommunermeddnssec.se/`

[13] Aleecia M. McDonald and Lorrie Faith Cranor. The Cost of Reading Privacy Policies. I/S: A Journal of Law and Policy for the Information Society 2008 Privacy Year in Review issue *or* Lorrie F Cranor,. et al. (2014). Are they worth reading? an in-depth analysis of online advertising companies' privacy policies. Rochester, NY: Social Science Research Network *and for instance the* Terms of Service; Didn't Read (ToS;DR) project.

[14] Art 5.1 c, General Data Protection Regulation.

### 2.1 The Scoring System

We devised a five-step scoring system for municipalities which runs from E to A, with A being the highest and E the lowest score.[15]

**A** To get the highest score, a municipality should neither leak PII to third parties (Internet service providers, mobile providers, content delivery platforms, advertisers, etc.) nor collect unnecessary information for their own use. This means use of HTTPS by default, no persistent first-party cookies, no third-party requests (which implies no third-party cookies), referrer policy set to a restrictive value (i.e., *no-referrer*), setting the HTTP Strict Transport Security header (HSTS), and no insecure requests (i.e., HTTPS to HTTP).

**B** The second highest score relaxes the requirements for collecting PII for own, in-house use (such as a locally hosted web analysis tool). It requires HTTPS, no third-party requests, and no insecure requests.

**C** The middle score is obtained for those websites that either have a strong protection against PII leakage to Internet service providers (by for instance employing encrypted connections universally), or having a strong protection against PII leakage to third party web services (content delivery platforms, advertisement networks, web analysis tools, etc.): HTTPS, third-party requests (but none insecure) and third-party cookies *or* HTTP and no third-party requests. 20 municipalities out of 289 acquired this score in our latest test on August 20th 2016.[16] This can be compared with 16 municipalities obtaining this score in May of 2016.[17]

**D** The second lowest score is obtained from having partially encrypted connections (with insecure elements loaded in the visitor's browser) and making third-party requests and setting third-party cookies, *or* from using HTTP but not setting third-party cookies. 64 municipalities ended up with this score in our latest test on August 20th 2016. Two municipalities had shifted from a C score in our May 2016 test to a D score in the August 2016 test, after introducing non-encrypted elements on their websites.

**E** The lowest score, then, is when there are no data leakage protections in place: no encryption, and use of third-party cookies. In our August 2016 test, most municipalities (204 out of 289) ended up with this score.

While the scoring system is simple, we have tried not to make it biased towards any particular form of protection against PII leakage. It is, for instance, possible to get at most a C grade without using encrypted connections, but it is not possible to advance beyond a C grade without also remedying PII leakage to third parties. We chose this methodology to ensure that privacy is protected against electronic communications services as well as information society services.

Since we finalised the beta version of our in-depth tool in May 2016, we have produced scores for all Swedish municipalities on three subsequent occasions.

[15] *See* `https://dataskydd.net/kommuner/metodologi.html`
[16] *See* `https://dataskydd.net/kommuner-201608`
[17] *See* `https://dataskydd.net/kommuner-201605`

The last scores, produced in November 2016, are not accounted for in this text but are available online.[18] In Fig. 1 and Fig. 2 you will find comparisons of scores between May and August of 2016.

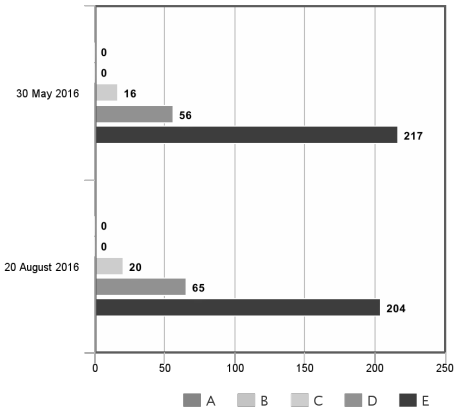


**Fig. 1.** Distribution of grades assigned by our tool for substantive measurement for all Swedish municipalities in two separate test runs performed on May 30th 2016 and August 20th 2016.

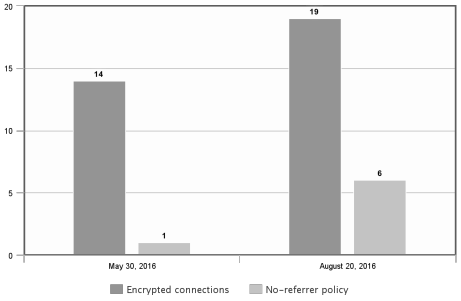


**Fig. 2.** Number of municipalities which have adopted encrypted connections and protection against referrer-leaks in May 30th 2016 and August 20th 2016 respectively. The total number of municipalities tested amounted to 290.

## 3 Technical Design

We made two tools: a generic tool for privacy checks for individuals (a web service), "Webbkoll",[19] and a more sophisticated tool for checking municipality websites (not a publicly available service).[20]

In both cases we attempt to simulate a typical user with a typical browser with default settings – e.g., with Do Not Track disabled, as that is normally the browser default – and with no particular browser extensions installed, and see

[18] *See* `https://dataskydd.net/kommuner`
[19] *Supra*, footnote 1.
[20] *Supra*, footnote 2.

what happens (requests generated, cookies set, etc.) when a certain web page is visited. This means that the tools need to run a "real", normal browser, or something as close to what an end-user would use as possible, ruling out for example web scrapers that do not execute JavaScript.

### 3.1 Generic Check: Choice of Tool for Backend/Frontend

There are a number of web privacy measurement platforms available, but the ones we found were all targeted towards researchers. We wanted to provide a simple web service that could be used by anyone to quickly check any given website, without having to install anything. There are numerous online services for checking various other aspects of websites – e.g., SSL/TLS configuration,[21] HTTP headers,[22] performance[23] – but we found none for generic privacy checks.

**Technical Choices** Since our tool was meant to be publicly available (as well as completely open source)[24] free to use by anyone at any time, and operated by ourselves on a typical Linux server, we needed something that 1) could process requests quickly, 2) could process multiple requests simultaneously, 3) could easily scale, 4) could be run on a typical low-cost VPS, and 5) would be built purely on open source components.

This called for a typical design where we would have a user-facing frontend server communicating with one or more separate backend servers (which in turn perform the actual visiting and rendering of webpages) through a REST API.

The most resource-intensive part of the infrastructure is the backend. We considered running a "real" consumer browser such as Firefox through the Selenium framework (as in [1]), but found that the overhead was too great. It should be noted that while Selenium offers more possibilities and a more stable environment, this was less of a concern here than in our later municipality test. Our online tool is not meant to be used for studies or rigorous analysis, but rather as a starting point for web developers and web visitors. Therefore we opted for "good enough" with our choice falling on PhantomJS,[25] a lightweight headless browser based on WebKit (also used by Safari; and a fork of WebKit is used by Chrome).

**Deployment** The frontend is written in Elixir, a functional language running on the Erlang VM, and uses the Phoenix web framework. When visiting the website of the tool the user is presented with a single text form field for entering the domain name or URL of any web page. When the "Check" button is clicked, the frontend does a number of things:

[21] Qualys SSL Labs SSL Server Test. `https://www.ssllabs.com/ssltest/`
[22] Securityheaders.io. `https://securityheaders.io/`
[23] Sitespeed.io (`https://www.sitespeed.io/`), Google's PageSpeed Tools (`https://developers.google.com/speed/pagespeed/`)
[24] By grant condition. Code (MIT license): `https://github.com/andersju/webbkoll`
[25] *See* `http://phantomjs.org/`

**1)** It checks whether the user is possibly a bot, and if so, rejects the request.
**2)** It makes sure the input is transformed into a proper URL – e.g., example.com becomes `http://example.com`. Since we check whether a site uses HTTPS by default, we always check the `http://` version of a site first to see whether it redirects automatically; so `https://example.com` is transformed into `http://example.com`. It is possible to visit specific pages on a domain (e.g., `http://example.com/subpage.html`); we keep the path (`/subpage.html`) but, for security reasons, no query parameters nor anything else. An input of `http://user:password@example.com/subpage.html?foo=bar` would be transformed into `http://example.com/subpage.html` before being passed on.
**3)** If the URL resulting from 2) has already been checked and is in the database, the old data is fetched and rendered. If the URL is not in the database, or if it is in the database but the user has clicked "Check again", we force a new check, and the frontend proceeds to 4).
**4)** To prevent abuse, it does some basic rate limiting: per IP (a user can only make a certain number of requests during a certain span of time) and per host (a certain host can only be queried a certain number of times during a certain span of time – this is not user-specific). If either criteria is violated, an error is returned.

Finally, if steps 1-5 completed successfully, the user's request is sent to a queue in the job handler – also on the frontend – for background processing (this handles concurrency, retries, queueing, etc). To allow for multiple backend servers, multiple queues, each having a certain backend URL tied to it, can be specified.

The job handler runs a worker to handle the user's request. The worker sends a HTTP GET request to the backend server. This request contains the URL of the webpage to visit.

The backend server runs PhearJS,[26] a server written with Node.js, handling a number of PhantomJS workers. When PhearJS receives a request from the frontend, it's passed on to one of the workers. The PhantomJS workers visits the URL and renders it, waiting a specified period of time – in our case, ten seconds – before returning the results. This is to make sure scripts have time to run. When finished, the resulting data is sent back as JSON to the frontend. This JSON contains all the request and response headers, cookies, HTML content, etc.

The JSON is decoded by the worker process and checked for errors. The worker then proceeds with processing of the data:

**1)** The final URL is noted. This is the actual page rendered, after any redirects.
**2)** From the final URL the registerable domain is extracted and noted with the help of a library that uses Mozilla's Public Suffix List.[27] This list is used by

[26] *See* `https://github.com/Tomtomgo/phearjs`
[27] *See* `https://publicsuffix.org/`

browsers to determine where cookies can be set (or not), and we use it to distinguish between first-party and third-party sites.

**3)** Cookies are split into first-party and third-party. We also count the number of unique third-party cookie domains.

**4)** From requests, third-party requests and insecure first-party requests are extracted. Request types (secure vs. insecure) are counted.

**5)** Using a HTML parser we check whether a referrer policy is set using meta referrer (a referrer policy can also be set in a Content-Security-Policy header; this is checked at a later point).

All the above is saved to the database and the status of the user's request is updated.

Meanwhile, the user is redirected to a status page where the ID of the request is in the URL. The page is reloaded automatically every five seconds using meta refresh. On every request the frontend checks the status of the page in the database—if state has changed, the user is redirected to a results or error page, otherwise the status page is shown again. While there are smoother ways to present status and do transitions, this way we avoid having to use cookies.

Finally, the user is presented with a results page. It shows the following:

- Whether HTTPS is used by default; and, if so, whether the site uses HTTP Strict Transport Security, and whether there are any insecure requests.
- Whether referrers are leaked.
- First-party cookies.
- Third-party cookies.
- Third-party requests, categorized using Disconnect's public tracker list.
- Certain HTTP headers that can be beneficial for privacy and security: Content-Security-Policy, Strict-Transport-Security, Public-Key-Pins, X-Content-Type-Options, X-Frame-Options, X-Xss-Protection.

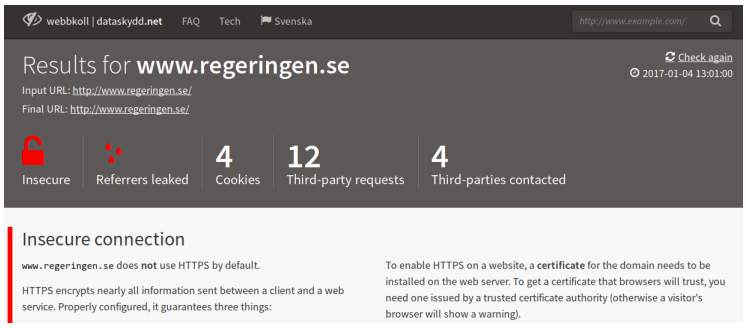


**Fig. 3.** Screenshot of part of results page for regeringen.se, the website of the Government Offices of Sweden.

We also explain what these things are and what one can do and why. Additionally, we check for certain third-party services (such as Google Analytics and Disqus) and suggest alternatives.

**Limitations** The tool is limited to what can be checked non-interactively, i..e., things that can be observed by merely loading the page – it does not perform any actions such as clicking on links.

Except for rejecting sites with invalid certificates (and checking the headers for HSTS and Public Key Pinning), it does not do any deep analysis of a site's SSL/TLS configuration – e.g., vulnerability to various attacks, forward secrecy, support for insecure protocols, etc. Currently we provide a link to Qualys SSL Labs free online service.

At first some of the tools used did not have all the necessary functionality for our purposes. We contributed some code to PhearJS to add support for returning cookies and requests,[28] and to the Elixir library PublicSuffix to add support for checking whether a domain matches a specific rule in Mozilla's Public Suffix List.[29]

### 3.2 Municipalities: Choice of Tool for Backend/Frontend

**Technical Choices** Although we had already created our online tool "Webbkoll", we opted for the web privacy measurement framework OpenWPM[1] for our municipality study. Our online tool was meant to be used by web developers and others to quickly check a website and gain ideas about possible improvements, while OpenWPM was built specifically for collecting data for privacy studies and supported more features, such as the ability to visit internal links.

**Deployment** We fed a list of the websites of Sweden's 290 municipalities into OpenWPM. OpenWPM – which uses Firefox, Selenium and a HTTP proxy – then visited each site and tried to visit up to five internal links. Firefox was configured to run without any particular extensions installed, and with Do Not Track disabled. All data (all HTTP requests and responses, cookies, etc.) was saved to a SQLite database.

We then wrote a program to enrich the database, mainly using information already contained therein, but made more easily accessible to make it easier to produce statistics and (at a later stage with the same program) generate the reports pages.[30] For example, the `http_requests` table was extended with columns for `base_domain` (shortest domain assigned to a registrant; used to determine first-party vs. third-party cookies/requests) and scheme (HTTP or HTTPS); `site_visits` was extended was extended with e.g. HSTS value, referrer policy, third-party requests, insecure requests and multiple columns for cookies (first-party and third-party profile and session cookies). Each municipality was also scored using the criteria mentioned in 2.1.

It should be noted that the results for a municipality are based not on a single page, but on the collected data from the initial page plus up to five internal pages.

[28] `https://github.com/Tomtomgo/phearjs/pull/5`
[29] `https://github.com/seomoz/publicsuffix-elixir/pull/17`
[30] Code (MIT license): `https://github.com/andersju/municipality-privacy`

In our setup OpenWPM did not save the HTML content of pages. As referrer policy can be set in both a HTTP header and in a HTML meta element, we let our post-processing tool visit the initial URL of each municipality website, get and parse the HTML and look for meta referrer.

Finally, our tool generated a static website with 1) an overview with a table containing all municipalities, sortable by score/scheme/referrer leakage/number of cookies/etc., and 2) a detailed results page for each municipality, much like the results pages produced by our web service Webbkoll.

**Limitations** While we did visit a number of internal links in this test, we are limited by what can be done in an automated fashion when we have no prior knowledge of the sites being visited. The internal links are chosen at random, and at the moment we cannot check whether it is, for example, possible to contact one's municipality in a secure way.

The `browse` command in OpenWPM 0.6.2 loads a specified URL and then tries to visit a specified number of internal links (from the same hostname as the URL) on the initial page. However, we found that no internal links were visited if the initial URL was a redirect; we thus had to use curl and a few command-line tools to process the list of municipality websites, figure out the final URL of each website, and then write those URLs to a new file that we then used as input to OpenWPM.

We found a bug that in rare instances would make OpenWPM treat external links as internal. This was reported and fixed.

## 4 Discussion

The following chapter is based on structured interviews with employees from the Swedish public sector as well as interactions with such employees that we have had since the beginning of our project. These interviews and interactions were, however, conditioned on anonymity of the individual civil servants concerned.

### 4.1 The Municipality of Enköping

We were in touch with civil servants working in the Swedish medium-sized municipality Enköping[31] since before we started our project. The municipality was, and still is, in the process of refurbishing their public-facing web environment.

The municipality lacked prior guidelines for web development beyond graphical profiles at the start of the project. The changes in the website could therefore be planned freely by responsible staff. While responsible staff had support for a data protection friendly shift from their immediate superiors in the hierarchy (the communications department), interest was more shallow in the municipal IT department. The data protection focus did, however, receive attention from

[31] A *medium-sized municipality* is a municipality with 20 000-50 000 inhabitants according to Sveriges kommuner och landsting, Kommunsgruppsindelning (2011).

the highest publicly elected official in the municipality when the new website was launched.

The focus on data protection in the development of the new website emerged only after the initial steps to change the website had already been taken[2]. Because of this, specifications which would have been useful to integrate at an earlier stage had to be appended to the specification afterwards (such as removing referrer leaks and using local analytics tools). This caused additional costs for the municipality (a one-time fee of approximately EUR 2000 for the Piwik server). While this is not a large sum, a continuing problem is the lack of qualified Piwik administrators available from subcontractors. Access to third party analytics specialists is simply higher, making it more time- and cost-efficient to use third party analytics tools.

In November 2016, the municipality has still not crossed the TLS hurdle in spite of having a pre-existing cryptographic certificate which was valid across all the municipality's domains when we first got in touch with them. The provision of municipal maps stopped working when TLS was turned on, but there is ongoing work to fix this problem. Changes in staff in summer of 2016 means that many of the planned changed are stuck, while new staff get accustomed to the work environment. The municipality is still not able to obtain a higher grade than D in the Dataskydd.net privacy web check tool.

### 4.2 Other Municipalities

After launching our municipal top list, we have noticed that more municipalities are adopting the use of referrer policy. We have also had questions from municipalities about the use of encryption.

For instance, it is still the case that some web developers fear that encryption may reduce the availability of the site (for instance, making it slower to load and requiring more server resources). While this is not supported in practical knowledge,[32], and in fact HTTP/2—which brings superior performance—in practice *requires* the use of encryption as no major browser supports unencrypted HTTP/2, it's a legacy concern that is likely to remain for some time.

Additionally, and as with many things that may require alterations in current work flows or technical tools, we have noticed that municipalities that are refurbishing their public web environments with data protection enhancements are likely to be concerned that the data protection enhancements are the cause of problems. For instance, use of referrer policy may get the blame for broken links even if, upon careful analysis, it turns out not to have been the problem. Other questions that have emerged are covered above in Sections 3.1 and 3.2.

### 4.3 Analysis

The experiences of Enköping indicates that a data protection focus becomes both cheaper and quicker if integrated from the beginning. This provides support for the utility of data protection by design.

[32] https://www.maxcdn.com/blog/ssl-performance-myth/

One municipal employee indicated that it would be helpful if some form of procedural standard was developed, equivalent to the standards for web site accessibility which have just been adopted in European law[3].

It may be assumed that the prevalence of non-third party analytics tool specialists increases if the demand for such services increase. The municipality staff also experienced that the most frequent questions they would face from other municipalities related to their experiences of Piwik. A detailed analysis of municipality websites indicates that Enköping is not alone in trying out self-hosted alternatives to Google Analytics.[33]

The environment for making changes will differ between municipalities. While Enköping experienced significant delays in ordering and installing Piwik servers and a low degree of interest in data protection from their IT department,[34] in other municipalities it is the IT department which is responsible for the design and features of the website.[35]

We have not received any feedback from private persons who've tried to use our tool to inspire changes in websites that they themselves did not manage.

## 5 Conclusions

Many of the tools under development for privacy measurement are primarily used to investigate large numbers of the most popular websites. Our work has focused on a minor subset of websites, those of Swedish municipalities, which were unlikely to be the most visited.

While our goal was to make cost-neutral suggestions for improvements in so far as possible, we found out that even nominally free ("gratuite") products or simple changes in the specifications for a municipal website may imply significant enough costs to the municipality that many parts of the municipal administration must be involved in enacting change. The experience of Enköping is that it is doable, but requires considerable effort from responsible civil servants.

The observations from the municipal civil servants are straight-forward, for instance requesting a guideline for data protection friendly developments. But such guidelines are unlikely to be adopted without significant effort. The tools we developed are not designed to make use of flexibilities in the data protection laws, but to provide simple and cheap means to maximise adherence to the principles enshrined in the General Data Protection Regulation Art. 5.

Some of our suggestions are more likely to be adopted than others: the number of Swedish municipalities using encrypted connections is increasing, as is the number of Swedish municipalities that introduce a referrer policy. We also believe that adoption of alternative analytics tools will increase, but this will depend on a few front-runners creating a demand for alternative analytics expertise.

[33] Alingsås, Arvika, Örnsköldsvik, as per Dataskydd.net:s municipal mapping 20th August 2016 (cf. footnote 16).

[34] Informal dialogue with municipal civil servant in Enköping.

[35] Informal dialogues with municipal civil servants in other cities.

We hope that our effort provide a humble starting point for future projects seeking to make data protection by default a feasible option for web developers.

## References


1. Steven Englehardt and Arvind Narayanan: Online tracking: A 1-million-site measurement and analysis. Proceedings of ACM CCS 2016. DOI: http://dx.doi.org/10.1145/2976749.2978313
2. Enköpings kommun, Förstudie ny webbnärvaro för Enköpings kommun. April 10, 2014. http://blogg.enkoping.se/webbutveckling/wp-content/uploads/sites/3/2014/04/enkoping-forstudie-nywebb.pdf
3. EU Commission, Statement by Vice-President Ansip and Commissioner Oettinger welcoming the adoption of the first EU-wide rules to make public sector websites and apps more accessible. See http://europa.eu/rapid/press-release_STATEMENT-16-3549_en.htm
4. PTS, Vägledning för webbutveckling. https://webbriktlinjer.se
5. PTS, Faktablad - Cookies och lagen om elektronisk kommunikation – PTS-F-2005:2. http://www.pts.se/upload/Documents/SE/Faktablad_Cookies_PTS_F_2005_2.pdf
6. PTS, På väg mot användbar IKT PTS slutredovisning av myndighetens delmål inom ramen för regeringens strategi för genomförande av funktionshinderspolitiken 2011-2016, PTS-ER-2016:19. https://www.pts.se/upload/Documents/SE/Dokument\%20funk/160311_PTS\%20slutrapport\%20uppf\%C3\%B6ljning\%20delm\%C3\%A5l.pdf
7. Sveriges kommuner och landsting, Kommungruppsindelning 2011. http://skl.se/download/18.5e95253d14642b207ee86e1f/1402935660165/SKL-rapport-kommungruppsindelning+2011_101020.pdf