Journal of
Cybersecurity and Privacy

MDPI

Article

The Privacy Flag Observatory: A Crowdsourcing Tool for Real Time Privacy Threats Evaluation

Vasileios Vlachos [1,2,*], Yannis C. Stamatiou [2,3] and Sotiris Nikoletseas [2,4]

[1] Department of Economics, University of Thessaly, GR-54124 Volos, Greece
[2] Computer Technology Institute and Press "Diophantus", GR-256504 Patras, Greece
[3] Business Administration Department, University of Patras, GR-26504 Patras, Greece
[4] Computer Engineering and Informatics Department, University of Patras, GR-26504 Patras, Greece
* Correspondence: vsvlachos@uth.gr

Abstract: Instilling good privacy practices to developers and users appears to be a difficult and daunting task. The World Wide Web encompasses a panspermia of different technologies, commercial and open source APIS, evolving security standards and protocols that can be deployed towards the implementation of complex, powerful, web applications. At the same time, the proliferation of applications and services on all types of devices has also increased the attack surface for privacy threats. In this paper, we present the Privacy Flag Observatory, a platform which is one of the main tools produced by the Privacy Flag EU funded research project. The goal of this initiative is to raise awareness among European citizens of the potential privacy threats that beset the software and services they trust and use every day, including websites and smartphone applications. The Privacy Flag Observatory is one of the components that contributed to a large extent, to the success of the project's goals. It is a real-time security and privacy threat monitoring platform whose aim is to collect, archive, analyze and present security and privacy-related information to the broader public as well as experts. Although the platform relies on crowdsourcing information gathering strategies and interacts with several other components installed on users' devices or remote servers and databases, in this paper, we focus on the observatory platform referring only cursorily to other components such as the mobile phone add-on.

Keywords: cybersecurity; cybercrime; ICT security; social network analysis; security tools

check for updates

Citation: Vlachos, V.; Stamatiou, Y.C.; Nikoletseas, S. The Privacy Flag Observatory: A Crowdsourcing Tool for Real Time Privacy Threats Evaluation. *J. Cybersecur. Priv.* **2023**, *3*, 26–43. https://doi.org/10.3390/jcp3010003

Academic Editor: Giorgio Giacinto

Received: 26 December 2022
Revised: 13 January 2023
Accepted: 18 January 2023
Published: 29 January 2023

1. Introduction

The World Wide Web, or WWW, is over 30 years old [1]. During these years, its purpose and scope have changed dramatically. From a purely research tool deployed by small communities of scientists, it has become a ubiquitous complex technological ecosystem that permeates every aspect of people's professional and personal lives. Consequently, the technologies that support rich, multimedia, web content and powerful mobile applications have advanced accordingly to cover all activities and needs. However, the initially static HTML web pages were soon not adequate even for the most primitive web experience. Thus, web standards have evolved to integrate advanced complex content on the server-side as well as on the mobile code which is executed on users' browsers to create highly interactive content.

In contrast, on the negative side, the adoption of these complex web technologies has exposed users to numerous privacy and security risks. As a general rule, the more advanced a technology is, the more exploitation paths it may contain that can be taken into advantage for malicious purposes. Two such major privacy risks, for instance, that have been made possible by the complexity of modern web applications are *user tracking* and *user fingerprinting*.

The situation appears to be more serious for users of *mobile* devices. With the advent of low-cost smartphones, web access from mobile devices has surpassed the corresponding one from Personal Computers [2]. The wide adoption of lightweight computing systems with "smart" capabilities and fast Internet connectivity has created a diverse, massive, ecosystem of hundreds of millions mobile devices and IoT equipment [3]. Despite the strict security and privacy requirements for mobile devices, a non-negligible number of manufactures and users does not comply with them. For instance, although a significant number of smartphone models and portable devices has been discontinued and left unsupported, their users continue to use them despite known vulnerabilities [4]. To make things worse, prior to the enactment of the *General Data Protection Regulation* (GDPR), the incentive for application developers to explicitly address security and privacy issues *by design* and not as an afterthought was quite limited.

Today, data breach incidents are alarmingly rising in number and severity. Additionally, there is a global tendency for people to create virtual digital "selves" and online profiles, mostly on information-leaking social media platforms, rich in personal (and possibly private) information. Thus, there may be significant privacy and security risks even by having a device only connected to the Internet without any further action. Social media activity, saved passwords, sensitive medical information and even visited places or websites can provide valuable information about people. Such type of personal information is highly valued in the digital underworld and is traded by hackers. What is more important is that such personal data trading may take place even without the intervention of hackers or malicious groups, simply by the data brokers as the notorious Cambridge Analytica scandal demonstrated [5]. This and other similar data trading scandals brought to light clearly demonstrate the fragility of users' privacy as well as the necessity to educate users about their rights to their personal data.

To enable users to defend themselves against this gloomy situation and develop the necessary privacy awareness, the main objective of the Privacy Flag EU research project was to propose and develop a novel awareness approach which combines the "wisdom of the crowd", i.e., crowdsourcing, with the power of automated privacy and security analysis tools. In this context, a set of methodologies and corresponding tools were implemented to assess the security and privacy protection mechanisms of websites and mobile phone applications (please see https://privacyflag.eu/, accessed on 30 November 2022, for more information about the project as well as how to obtain and install its tools).

A central component of the Privacy Flag tool set is the *Privacy Flag Observatory* whose goal was to raise security awareness and help users enhance their knowledge on good privacy protection practices. That was achieved through the assignment to each website or mobile application of a specific score that reflects the level of privacy protection they provide. The Privacy Flag Observatory assembles the general trends of compliance with the recommended best privacy and security practices in cybersecurity.

2. Related Works

The need for a systematic monitoring of cyber attack trends is well documented in numerous works (see, e.g., [6]). A large body of academic publications, as well as a significant number of technical reports and white papers from governmental agencies and the private sector, focuses on the application of technical, legal and human safeguards to enhance citizens' privacy and security in the cyberspace [7]. As the scope of digital security and privacy is very broad, in our work, we focus on monitoring platforms and, in particular, observatories for collecting, archiving, analyzing and presenting security and privacy-related information to the broader public. Our aim is to raise awareness and promote best practices. Therefore, our work's goals differ from the majority of real-time security monitoring platforms that are described as *Security Operation Centers* (SOCs). SOCs operate in the context of Managed Services from various cybersecurity vendors and aim at highlighting imminent threats, issuing early warnings and taking the appropriate measures when needed. Their goal, however, is not to evaluate the existing applications with respect to appropriate privacy and security metrics in order to educate and protect users in the long term (see [8,9]). Thus, the Privacy Flag observatory, as all the other observatories, is not designed to detect ongoing cyberthreats, although it can, but, rather, to follow the trends on the adoption of privacy and security aware technologies.

Closest to our work is the Mozilla Observatory (https://observatory.mozilla.org, accessed on 30 November 2022), which analyzes websites for various security issues. The Mozilla Observatory focuses on the use of cryptographic technologies for real-time data encryption in the web. Additionally, this observatory performs various other checks including cookies, Cross-Origin Resource Sharing (CORS), Content Security Policy (CSP) and X-XSS-Protection. However, it does not take into account the risk of obsolete and deprecate technologies while it does not contain modules to assess the mobile applications which access the website. Nevertheless, the Mozilla Observatory is a mature and robust framework which has assessed millions of websites and helped more than 300,000 web administrators to address various privacy and security issues in their websites. Another important observatory is the Qualys SSL Pulse. It performs very analytical and focused investigation of SSL/TLS protocols of websites and can identify possible vulnerabilities that lead to attacks such as BEAST, Heartbleed, POODLE and CRIME, which will be discussed later in more detail. The Qualys SSL Pulse provides detailed statistics and summary reports with respect to the usage of the TLS protocol [10].

The Alexa company was known for its extensive data collection on every aspect of web activity. One of the outcomes of this undertaking was the accumulation of a vast dataset, known as the *Alexa Million*, containing technical data on the encryption methods of hundreds of thousands of websites. Although this dataset is no longer updated, it is still available in several repositories. Together with the Mozilla Observatory and the Qualys SSL Labs, the Alexa dataset has helped numerous researchers in obtaining the necessary data for their investigations (see, e.g., [11–13]). Furthermore, this dataset has also facilitated several high-impact research works (see [12]).

Another remarkable effort was the design of the Cyber Threat Observatory for Computer Emergency Response Teams (CERT) [14]. This observatory combines several useful components in a unified framework. These elements include security advisories, Common Vulnerabilities and Exposures (CVEs), Indicators of Compromise (IoCs), social media aggregation and even chat/messaging functionality for direct communication among CERT members. Obviously, the Cyber Threat Observatory is designed for the IT security personnel of the Computer Emergency Response Teams. Thus, it provides much more advanced and sophisticated information an average user could use. On the other hand, as its development and operational level are not covered in detail, it is difficult to assess it in more depth with respect to its potential and scope.

Another more focused and, therefore, more promising approach is presented in the *e-Privacy Observatory*. The e-Privacy Observatory shares some similarities with the Privacy Flag Observatory we discuss in this paper, but it emphasizes more the fingerprinting and tracking threats rather than the overall cybersecurity risks. At the time of writing of this paper, the e-Privacy Observatory is not online and, thus, it is difficult to evaluate its capabilities [15].

The Privacy Flag project has also attempted to address the problem of privacy leakage in users' mobile devices. Android applications are granted or denied access to various systems based on the "group permission model". The limitations of this approach are well documented [16,17]. To mitigate these issues, researchers have proposed different solutions. The Personalized Privacy Assistant (PPA) utilizes machine learning to create privacy profiles based on users' personal preferences which are extracted through a questionnaire [18]. The PrivacyFlag smartphone application, on the other hand, requests from users to rank different types of privacy-related information, such as the location, phone call history, access to camera and microphone, etc. The purpose of this step is to capture the variety of privacy sensitivities of different users and score the privacy level of each application accordingly. The PPA is more advanced as it can provide recommendations and it helps users to understand the specific privacy issues of each application. The results of this work are encouraging, although it might be difficult to be adopted on a larger scale as it requires supervisor level access rights. Finally, a highly visual and attractive approach is presented in the form of a graphical information system using personal examples to improve risk communication [19]. In that system, the users can see visual representations and examples of the types of data that are requested by an application. Thus, they can make more informed security and privacy related decisions. Nonetheless, the complexity of the android permission model requires further investigation. In this respect, the Mobile Security Behavior Observatory we propose is a positive step towards understanding mobile users' behavior [20]. When these data are made available, it would be useful to compare them with the findings of the Privacy Flag Observatory.

3. The Privacy Flag Observatory Architecture

The overall Privacy Flag architecture is presented in Figure 1. The black framed subsystem is the Privacy Flag Observatory and its components, on which we focus on this paper. Its main objective is to inform users, developers, stakeholders and researchers on the level of adoption of good privacy and security practices, but also to issue warnings when insecure, obsolete or deprecated technologies are used. The Privacy Flag platform allows people to monitor and control their privacy with a user-oriented approach supported by a friendly interface and interactive menus. Furthermore, interested stakeholders can observe the level and rate of compliance with privacy-preserving practices and technologies for the most popular websites.

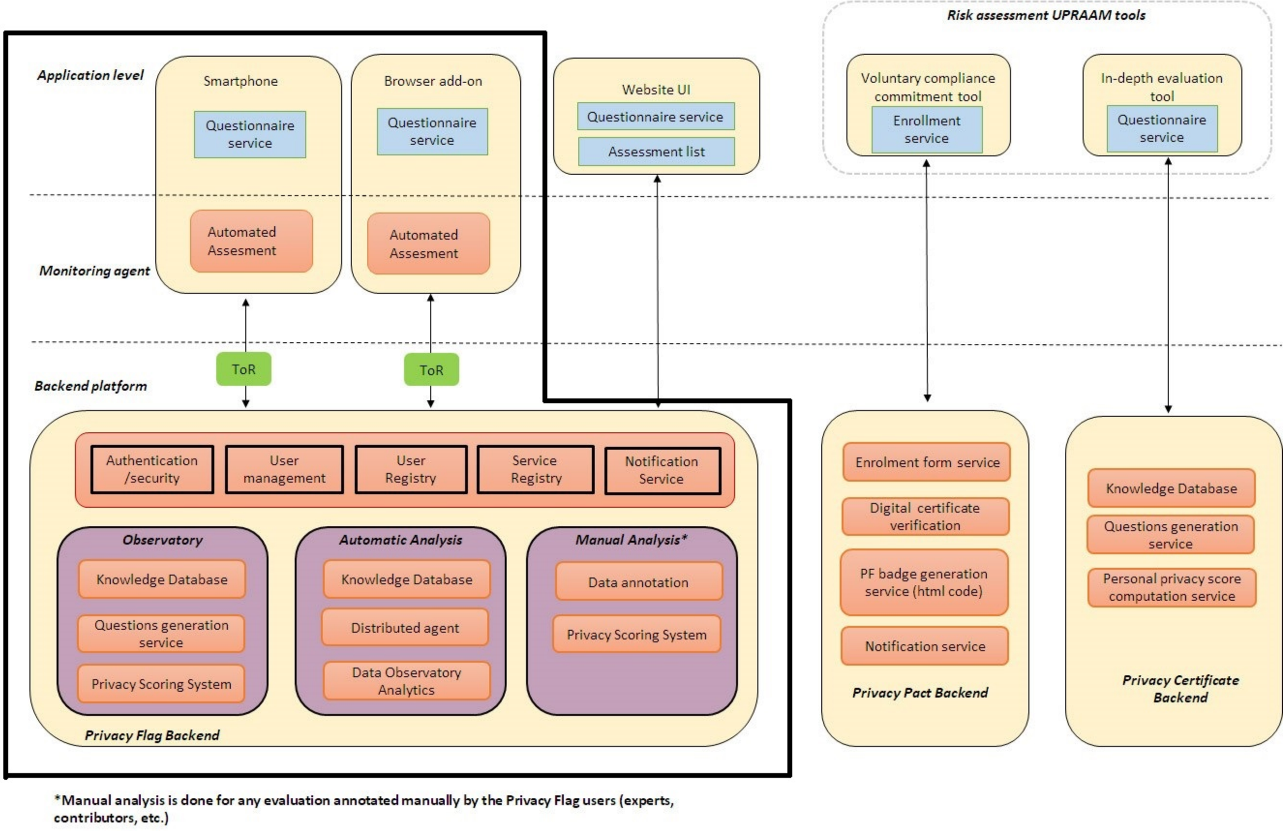


Figure 1. Overall platform architecture.

Focusing on the framed part of the architecture in Figure 1, the Privacy Flag Observatory includes, as it is shown at the top left-hand corner of the figure, a *smartphone application* and a browser *web add-on*, both operating on the users' devices. These tools are implemented as *Distributed Agents (DAs)* that run both on the user's side and the server. Two types of DAs exist. One is the Privacy Flag Web add-on which is installed on the user's browser. Each time a user visits a new website that has not been previously evaluated, a set of automated security and privacy checks is initiated. A part of these checks is performed by the Privacy Flag Web add-on locally, on the user's device, while the rest of the checks are executed remotely by and on the Privacy Flag Server. The reason for this decision is technical, based on the nature of each specific test. Some tests can be executed, more conveniently, on the Privacy Flag server, while others are easier to execute on the user's device. From an architectural perspective, we tried to offload, as much as possible, the computational effort of the Privacy Flag Web add-on to the Privacy Flag Server in order to achieve minimal impact on the users' devices. Therefore, the Privacy Flag Server is extensively used to perform the site evaluations.

The completion of this evaluation step is user-transparent and does not require human intervention. A visual notification indicates whether evaluation feature is turned on so that users can deactivate or reactive it at any time. The evaluation results are *anonymously* transmitted and stored in the Privacy Flag Database to be included in the analysis and extraction of the general privacy and security trends. The users can also choose to manually assess the website by answering ten short questions related to the information they received regarding the privacy policy and personal data management.

Similarly, a Privacy Flag smartphone application or Privacy Flag SmartApp has been developed to assess mobile applications. The Privacy Flag SmartApp evaluates the installed applications on the mobile device, based on the number and the level of permissions that each application requires to operate.

Finally, after the collective assessment and the corresponding analysis on the Privacy Flag Server, a color-based code scheme indicates the privacy status of each website or mobile application. It is important to clarify that the Privacy Flag SmartApp cannot detect malicious applications. Instead, it highlights the applications that require access to different sensitive subsystems of the mobile device. Therefore, these applications constitute a significant privacy risk and should be treated with caution.

The exact nature of all the checks will be discussed in detail later in this paper. However, it is important to emphasize that due to space limitations it is not possible to extensively analyze all the subsystems of the Privacy Flag platform. Therefore, this work is focused on the Privacy Flag Observatory alone.

A central component of the Privacy Flag Observatory architecture, as shown in Figure 1, is the *P*rivacy Flag Database which contains submitted and deduced assessments provided by users and the distributed agents. All Privacy Flag subsystems communicate with this database, which contains up-to-date information resulting from the automated security checks and the users' responses. At the same time, the Privacy Flag Observatory publishes online analytics, in text as well as in graphical forms. The goal is to provide useful information to the public and the experts with respect to the security and privacy risks that stem from the adoption of problematic practices—see Figure 2.

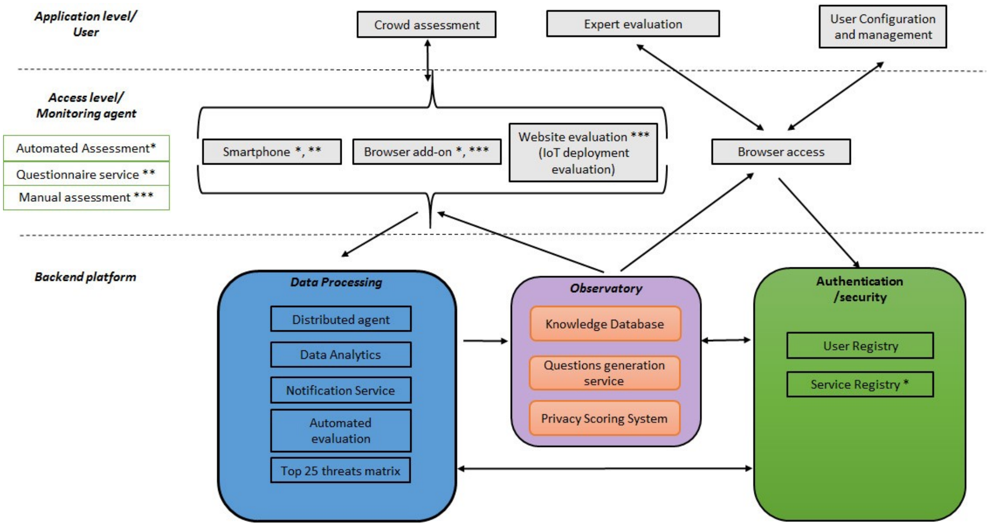


Figure 2. The position and role of the Observatory in the overall assessment processes.

Among the main components of the Privacy Flag Observatory are the following: the *Privacy Scoring System*, the *Questions Generation Service* and the *Knowledge Database*, which accumulates the crowdsourced data provided, voluntarily, by participating users (see also Figure 2). Each time a user visits a site, the distributed agents running on the Privacy Flag Web add-on and the Privacy Flag Server perform, together, several privacy and security checks, without the user's intervention. These checks rely on the Privacy Scoring System which encompasses the *Top 25 Threat Matrix* created and maintained by the Privacy Flag participants as a "living document" which is updated in real time. This living document contains some of the most dangerous privacy and security risks which are described in suitable formats called *threat descriptors*. In Table 1 we see an example of such a threat descriptor for a concrete threat which also contains information as it is stored by the Privacy Flag Observatory.

Table 1. An example Thread Descriptor.

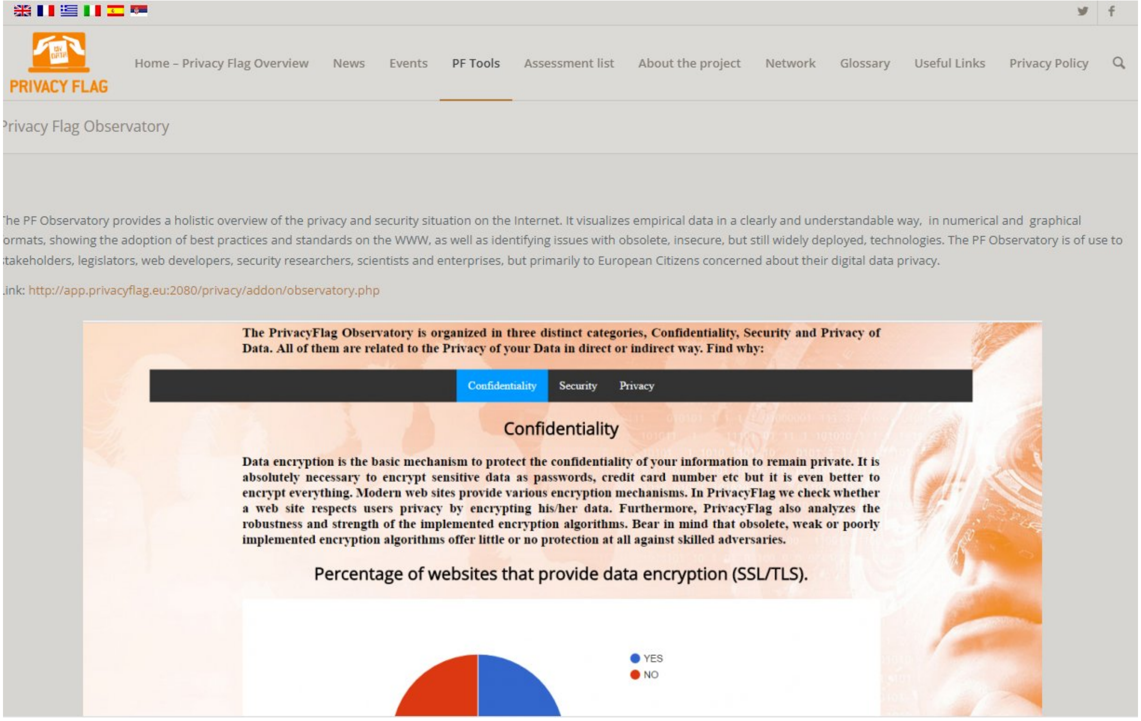
Threat Descriptor	
Name	**Does the Website Use Certificate Pinning? (HTTP Public Key Pinning)**
Threat	**Website Impersonation**
High-level Description	HTTP Public Key Pinning (HPKP) is a security mechanism which allows HTTPS websites to defend against impersonation attacks in which attackers deploy misissued or fraudulent certificates. For example, attackers might compromise a certificate authority (i.e., the entity that issues soft authentication certificates for websites) and then misissue certificates for any domain. To defend against this risk, the web server can provide a list of "pinned" public key hashes. Thus, on subsequent connections, web browsers expect that server to use one or more of those public keys in its certificate chain.
Threat Category	Confidentiality of Communications
Implementation Details	This threat is implemented as a backend script that takes the URL as input and reads the HTTP headers. If the "public-key-pins" header exists, the "max-age" is well configured (i.e., greater than 30 s) and there is a valid SHA256 hash for the public key, then the script returns true, otherwise, it returns false.
Return Value	True/False

The Privacy Flag Observatory currently handles 25 threats using the appropriate detection mechanisms. In the end, the results of the Top 25 Threat Matrix based analysis are stored, anonymously, in the Privacy Flag Database for further processing, evaluation and visualization. The global trends of the privacy technologies are presented, in graphical and text formats, on the Privacy Flag Observatory and are made available to any interested party.

The significant advantage of the Privacy Flag Observatory is that due to its crowdsourcing implementation, it analyzes websites and mobile applications which are popular among 'real' users. Therefore, it is much more accurate than other alternatives based on static web indexes or web crawlers that analyze, indiscriminately, any existing website. Thus, the Privacy Flag Observatory can indeed help identify the progress on the deployment of secure cryptographic algorithms and privacy protection mechanisms on *popular* websites. Additionally, it can pinpoint obsolete and insecure web technologies which are still, unfortunately, in wide use.

4. Implementation and Operation of the Observatory

The Privacy Flag Observatory evaluates mobile applications and services with respect to threats related to three distinct categories: *Confidentiality*, *Security* and *Privacy of Data*. In Figure 3, we see the initial page of the platform's website.



Privacy Flag Observatory

he PF Observatory provides a holistic overview of the privacy and security situation on the Internet. It visualizes empirical data in a clearly and understandable way, in numerical and graphical ormats, showing the adoption of best practices and standards on the WWW, as well as identifying issues with obsolete, insecure, but still widely deployed, technologies. The PF Observatory is of use to takeholders, legislators, web developers, security researchers, scientists and enterprises, but primarily to European Citizens concerned about their digital data privacy.

ink: http://app.privacyflag.eu:2080/privacy/addon/observatory.php

The PrivacyFlag Observatory is organized in three distinct categories, Confidentiality, Security and Privacy of Data. All of them are related to the Privacy of your Data in direct or indirect way. Find why:

Confidentiality

Data encryption is the basic mechanism to protect the confidentiality of your information to remain private. It is absolutely necessary to encrypt sensitive data as passwords, credit card number etc but it is even better to encrypt everything. Modern web sites provide various encryption mechanisms. In PrivacyFlag we check whether a web site respects users privacy by encrypting his/her data. Furthermore, PrivacyFlag also analyzes the robustness and strength of the implemented encryption algorithms. Bear in mind that obsolete, weak or poorly implemented encryption algorithms offer little or no protection at all against skilled adversaries.

Percentage of websites that provide data encryption (SSL/TLS).

Figure 3. The initial page of the platform's website.

In what follows, we will discuss these categories and the operation of the observatory.

4.1. Confidentiality

The need for the deployment of strong cryptographic algorithms has become widely evident since Phil Zimmerman introduced the *Pretty Good Privacy* application and helped millions of people to securely communicate and exchange information. Thus, it is now a widely agreed upon fact that the strong encryption of sensitive data is the minimum requirement for every web service and website.

At the same time, recent initiatives, such as the *Let's Encrypt* program (https://letsencrypt.org, accessed on 24 October 2022), promote the encryption of all web communications. To this end, a number of organizations offer free digital certificates for website encryption. The Privacy Flag Web add-on extracts valuable information regarding the cryptographic measures that are in use by websites. The evaluation's scope ranges from the most essential confidentiality mechanisms, such as the encryption of the traffic, to more advanced techniques that are deployed to minimize various sophisticated cryptanalytic attacks.

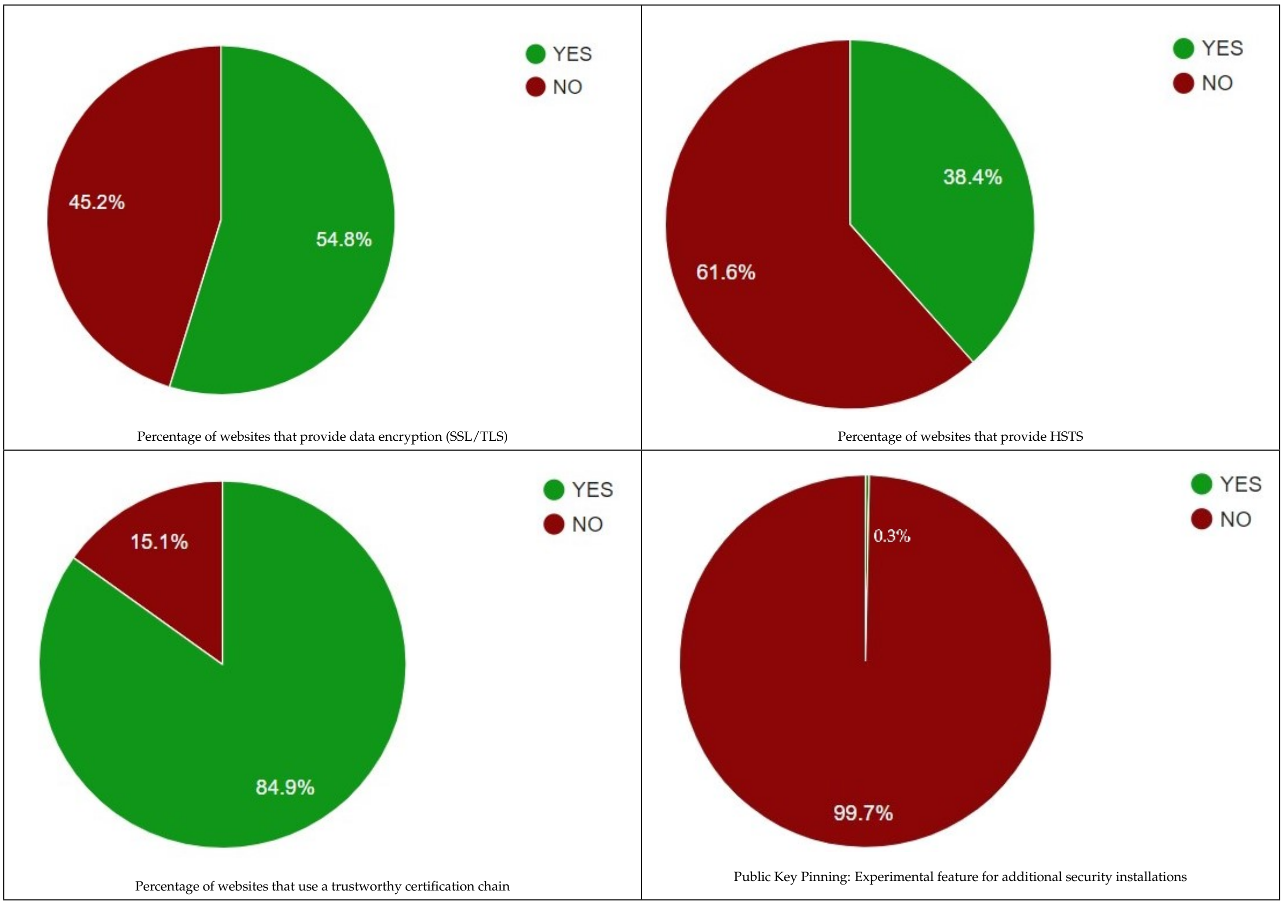
The following checks are performed during the interaction of a Privacy Flag user with a website:

- Encryption of the of traffic: The encryption of the traffic between a user's computer and the web server is the most essential step to enforce data confidentiality. The Privacy Flag Observatory computes the percentage of websites that deploy encryption. However, not all deployed encryption mechanisms provide adequate protection against modern cryptanalytic techniques [21]. In particular, obsolete cryptographic suites such as SSLv3, TLS 1.0 or earlier, no longer ensure sufficient security protection [22,23]. Earlier versions of the TLS protocol, for instance, are associated with a list of known attacks such as the BEAST attack [24]. Moreover, the obsolete SSL protocol has enabled the POODLE and FREAK exploits [25], while some implementations of the TLS compression procedures can be abused by the CRIME and BREACH attacks [26,27]. Similar security issues arise in various vulnerable implementations of the OpenSSL cryptographic software library, which are still in widespread use.

Despite the deployment of newer versions of the TLS protocol, which are secure, their integration into past implementations of the OpenSSL create paths to several attacks. The Heartbleed bug, for instance, demonstrated that several web applications can be compromised [28]. However, even a non-secure cipher is a much better approach than the transmission of sensitive information in plaintext. Even the less robust cryptographic algorithms often require significant effort to bypass their encryption. On the contrary, plaintext can be intercepted with minimal effort using well-documented open source tools, without requiring a high level of expertise. Therefore, we decided to accept *all* cryptographic algorithms and highlight the importance of data encryption as a means to achieve confidentiality and privacy for all Internet users. Our main objective is, thus, to encourage users to visit websites using the HTTPS protocol instead of the insecure HTTP protocol.

- Use of the HSTS protocol: Several of the shortcomings of the HTTPS protocol have been addressed with the *HTTP Strict Transport Security* or *HSTS* protocol [29]. The websites that have adopted this enhancement can protect their users more effectively. In particular, the HSTS can neutralize the *protocol downgrade attacks*. It is also very effective against *Man-In-The Middle (MITM)* attacks [29]. However, the protocol has some limitations as the user must have had accessed, previously, a website using the HSTS within a trusted network and the HTTPS connection mode. In this case, the browser will enforce HTTPS connection throughout the whole communication session. Otherwise, if the initial connection is made using the standard HTTP protocol in an insecure network, it is possible that an eavesdropper can intercept the initial request and redirect the traffic to a malicious website. Furthermore, the HSTS protocol can significantly reduce the risk of SSL stripping attacks. However, as expected, the HSTS is not a panacea. Sophisticated attacks, such as *BEAST* and *CRIME*, cannot be eliminated, but the deployment of HSTS lowers their success rate. Finally, HSTS should be enabled and supported both on the client and server sides.
- Use of a trustworthy certificate chain: A website should have a *valid* and *trusted* certificate. The process of validating the certificate is based on a chain of trust that links back to a *Root Certification Authority (CA)* trusted by the user's browser [30]. Websites that use *self-issued*, *expired* or *non-recognizable* certificates from unknown or not trusted CAs [31,32] are not suitable for web applications that handle sensitive data and content. On the other hand, a self-issued certificate might be a better alternative to the deployment of the simple HTTP connection which offers no protection at all, allowing all information to be transmitted as plaintext. A compromising solution may be to use certificates, such as the ones provided by the *Let's Encrypt* initiative, which are *free encryption certificates*. However, such certificates have limited support and lifetime compared to the commercial certificates [33]. The adoption of valid and trusted certificates is a key aspect in the efforts to increase the security of the Web.
- Public key pinning: *HTTP Public Key Pinning (HPKP)* is a security mechanism which allows HTTPS websites to defend against impersonation attacks [34]. These attacks are based on malformed or invalid, i.e., fraudulent, certificates [35]. For example, attackers might compromise a Certificate Authority and then issue fraudulent certificates for any domain. To defend against this threat, the web server can provide a list of "pinned" public key hashes. In this way, in subsequent connections, web clients will expect the server to use one or more of those public values (keys) in its certificate chain [36].

In Figure 4, we see examples of how this information is provided in graphical form (pie charts) by the platform.



Percentage of websites that provide data encryption (SSL/TLS)

Percentage of websites that provide HSTS

Percentage of websites that use a trustworthy certification chain

Public Key Pinning: Experimental feature for additional security installations

Figure 4. Example of the graphical information provided for the confidentiality section.

4.2. Security

It is a fact that some old, legacy, applications are still in use today by organizations and users alike. Usually, these applications contain less secure technologies and, thus, they are more susceptible to security issues. The reason for that is that they are use obsolete security mechanisms that do not meet current security and privacy standards. Thus, it would be beneficial if such applications could be detected and, subsequently, be either updated or removed.

The Privacy Flag Observatory focuses on the following well known and widely present obsolete technologies with less than adequate, or non-existing, security properties. On the other hand, new and quite untested, in large scale, technologies can also extend the attack surface. The Privacy Flag platform identifies and monitors some of the new powerful capabilities of HTML5 standard in order to assess them. The complete list of the technologies with security implications which is tracked by the Privacy Flag Observatory is the following:

- Flash: It was once the most commonly used multimedia content player. Most websites delivered interactive content almost exclusively for the Flash player. Unfortunately, the Flash protocol has been ranked highly as a major source of security risks [37]. Therefore, today, most websites avoid using Flash in favor of new multimedia codecs. Thus, although it is not always possible to refrain from using Flash, users should try to use websites with the more secure, native, HTML5 video players. The percentage, as identified by the Privacy Flag Web add-on, of the websites currently using the risky Flash codec is presented on the Privacy Flag Observatory.
- HTML5 APIs - Web Audio API: The HTML5 Web Audio is a very useful technology for capturing and storing sound streams from various audio input sources as well as the devices' microphones. Naturally, care should be taken to protect users from unautho-

rized recording or eavesdropping [38] of their audio streams and their environment. Thus, this specific recording functionality should be used with utmost caution. The percentage of websites that provide potentially risky access to the microphone is displayed on the Privacy Flag Observatory.

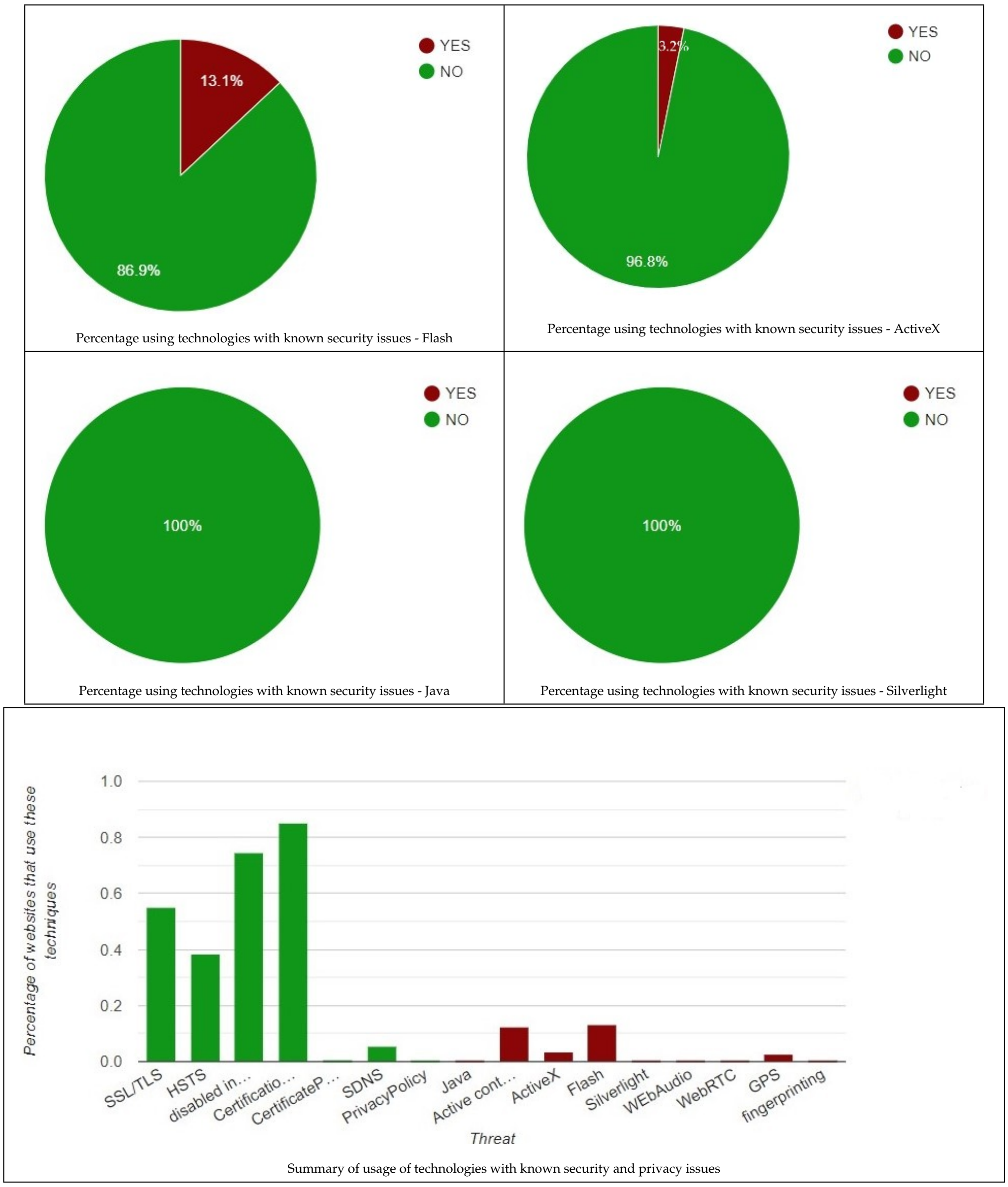
- HTML5 APIs—WebRTC: It is a very effective mechanism for providing real-time communication, but it is also used by hackers to intercept sensitive information or deanonymize users [39,40]. Nonetheless, this is a promising and useful technology, but whenever privacy is absolutely necessary, WebRTC should be avoided. The percentage of websites that use potentially privacy threatening WebRTC communication sessions is presented on the Privacy Flag Observatory.
- ActiveX: It is an obsolete Microsoft technology supported only by older Internet Explorer browsers. ActiveX components can be used to build complex scripts to automate several tasks. ActiveX normally operates from the website directly on the users' devices. As a consequence, many serious security issues may arise (see [41,42]). Microsoft has disabled ActiveX on the recent versions of the Internet Explorer browser, but older versions still support it for legacy web applications. The percentage of websites that use the highly insecure ActiveX components is highlighted on the Privacy Flag Observatory.
- Java Applets: A very popular programming language, Java has been used since the earliest days of the web to develop powerful web applications known as *Java Applets*. Due to the many vulnerabilities that Java has suffered during the past years, it is not considered a good practice, from a security perspective, to incorporate Java Applets in webpages [43,44]. Most web browsers support deprecated Java Applets in a way or another, but a limited number of web business applications still require Java Applets to function properly. The percentage of websites that contain Java Applets is depicted on the Privacy Flag Observatory.
- Silverlight: It is a Microsoft technology based on the .NET framework. It is used for the development of highly interactive applications which enrich user experience [45]. As any middleware, .NET with direct access to a user's computer can give rise to security risks [46]. If not absolutely necessary, it should be avoided. The percentage of websites that are based on the Silverlight framework are shown on the Privacy Flag Observatory.

A set of comprehensive diagrams provided by the Privacy Flab Observatory, shown in Figure 5), summarizes the adoption trends of these technologies (SSL/TLS, HSTS, insecure SSL/TLS, Certification Chain, Certificate Pinning, SDNS, Privacy Policy, Java, Active content documents, ActiveX, Flash, Silverlight, WebAudio, WebRTC) by the websites which have been visited by the Privacy Flag Users.

4.3. Privacy

Most websites use several mechanisms to store information regarding users' preferences. This information should be related only to users' viewing and browsing experience and not to their personal data or their overall browsing activity. Privacy Flag analyzes various user tracking mechanisms deployed by different websites. The following two checks are automatically performed by the Privacy Flag Web add-on, although they are only partially implemented, since this functionality became a standard feature in all modern browsers.

- Average number of coolies per site: It is useful to have a good estimate of the average number of cookies per site, since an unusually large number of cookies in a website may be an indication of privacy risks.
- Use of potentially risky types of cookies: Although most cookies are not dangerous, some types of cookies such as super cookies, zombie cookies, evercookies and LSO (Local Shared Objects) are persistent and very difficult to remove [47,48]. Unfortunately, their reliable detection requires much more effort in comparison with the standard or the third-party cookies. Therefore, this feature was not implemented in the Privacy Flag Observatory.



Percentage using technologies with known security issues - Flash

Percentage using technologies with known security issues - ActiveX

Percentage using technologies with known security issues - Java

Percentage using technologies with known security issues - Silverlight

Summary of usage of technologies with known security and privacy issues

Figure 5. Example of the graphical information provided for the security section.

4.4. Mobile Applications Permissions

Each Android application is associated with a list of access permissions that it needs to operate. These are organized into *permission groups*. However, there is a list of permissions that are considered more threatening and privacy invasive (see [49]) than the rest. Privacy Flag analyzes the permissions which each installed application has requested in an Android environment and evaluates them with respect to the privacy risks they represent. The Privacy Flag SmartApp categorizes the installed programs using a code color, with green indicating a privacy-friendly, orange a privacy-neutral and red a privacy-threatening application. The outcome of this assessment is based on the total number of permissions that are required from each application. All requested permissions are not treated equally but they are ranked, according to their importance, by the users themselves using a crowdsourced Borda counting scheme [50]. If a mobile application requires advanced permissions to function properly, the responsibility lies with the user to decide whether they should continue to use the specific software. On the other hand, the Privacy Flag SmartApp may reveal applications that request, with no appropriate justification, *elevated* permissions to operate. As an example, one may consider the case of a flashlight application. If this application requires access to specific privileges such as the right to read the user's contacts, obviously, a serious discrepancy is present with evident privacy implications. In any case, the privacy scores are anonymously transmitted to the Privacy Flag Server to extract the general trends. The Privacy Flag Observatory can, subsequently, calculate how often each permission is used in mobile applications, which is indicative of the privacy risks that modern mobile devices pose to their owners. The following statistics are presented in the Privacy Flag Observatory, together with a short explanation of their importance.

- The percentage of evaluated applications that use permissions that belong to the Camera group: If an application has access to the device's camera, it can take pictures with or without the user's knowledge. For applications that are related to image editing or social networking as well as other communication tools, it is normal to require such access to provide the full experience to the users. On the other hand, it is a a very serious privacy violation incident if an application takes pictures without the user's knowledge and explicit consent [51].
- The percentage of evaluated applications that use permissions which belong to the Contacts group: Personal contacts on a mobile device can be accessed by applications upon appropriate user authorization. Software that can handle calls, e-mails or social media are expected to require permission to use this information. Yet, again, a malicious application might gain knowledge about users' personal and professional relations and, thus, endanger their privacy.
- The percentage of evaluated applications which use permissions that belong to the Calendar group: The calendar application helps users organize meetings and set up task reminders. As the calendar application has a complete knowledge of a user's schedules, tasks and plans, such as meetings with other people, attending events and visiting places, it is important that this information remains private, unless it is required otherwise.
- The percentage of evaluated applications that use permissions which belong to the Location group: By allowing an application to access a user's location, it can extract, accurately, all the mobility patterns and habits of a user, e.g., the path that the user follows commuting to work during the day or the places the user visits for recreational purposes. Therefore, the software can reveal detailed information about the places users frequent, i.e., where they live, work and travel. An application might need this information to help users optimize their daily mobility plans, to suggest nearby shops, restaurants and bars. However, location-related information is considered sensitive and, therefore, it should be adequately protected [52].
- The percentage of evaluated applications that use permissions which belong to the Microphone group: Accessing the microphone implies that it is possible, for an application, to capture all discussions and sounds in the proximity of the user's mobile phone. This is entirely normal for applications that provide real-time communication capabilities, but it can also be very risky since a malware can turn a mobile device into a powerful spying machine [53].
- The percentage of evaluated applications that use permissions which belong to the Phone group: A very limited number of legitimate applications that provide real-time communication capabilities might need to access a mobile phone's telephony subsystem. A malware, however, may use this functionality for initiating and receiving calls towards spying on users or calling premium toll numbers [54].
- The percentage of evaluated applications which use permissions that belong to the Sensors group: Smart devices are equipped with a variety of sensors to enable them to monitor the mobile phone's motion and orientation as well as various environmental parameters and conditions. If an application has access to the data of these sensors, it is possible to infer users' behavior patterns and launch privacy breach attacks related to users' physical activity and perform user profiling and tracking [55].
- The percentage of evaluated applications that use permissions that belong to the SMS group: Only a limited number of applications should require access to the SMS

functionality, since the exchanged messages usually contain private and, sometimes, sensitive information. Therefore, if an application needs to use this functionality on the user's behalf should clearly state the purpose of doing that and receive the user's consent. Otherwise, the information contained in the exchanged messages may become available to, perhaps, malicious third parties. In addition to this, malware may send SMS messages to premium toll numbers and, thus, increase a user's mobile phone bill [56].

- The percentage of evaluated applications that use permissions which belong to the Storage group: If an application accesses the external (e.g., SD card) storage of a mobile device, it can read, write or modify the user's documents, photographs and data. This can lead to privacy violation if the user stores private or sensitive information in the mobile device's external memory [57]. Of course, for maintenance applications that need to periodically organize the contents of the mobile phone, access to the external storage should be granted.

The Privacy Flag Observatory summarizes the findings of the Privacy Flag SmartApp in a set of detailed graphs. A limited number of them, due to space limitations, are shown in Figure 6. However, all the graphs are available in the Privacy Flag Observatory website (http://app.privacyflag.eu:2080/privacy/addon/observatory.php, accessed on 30 November 2022).

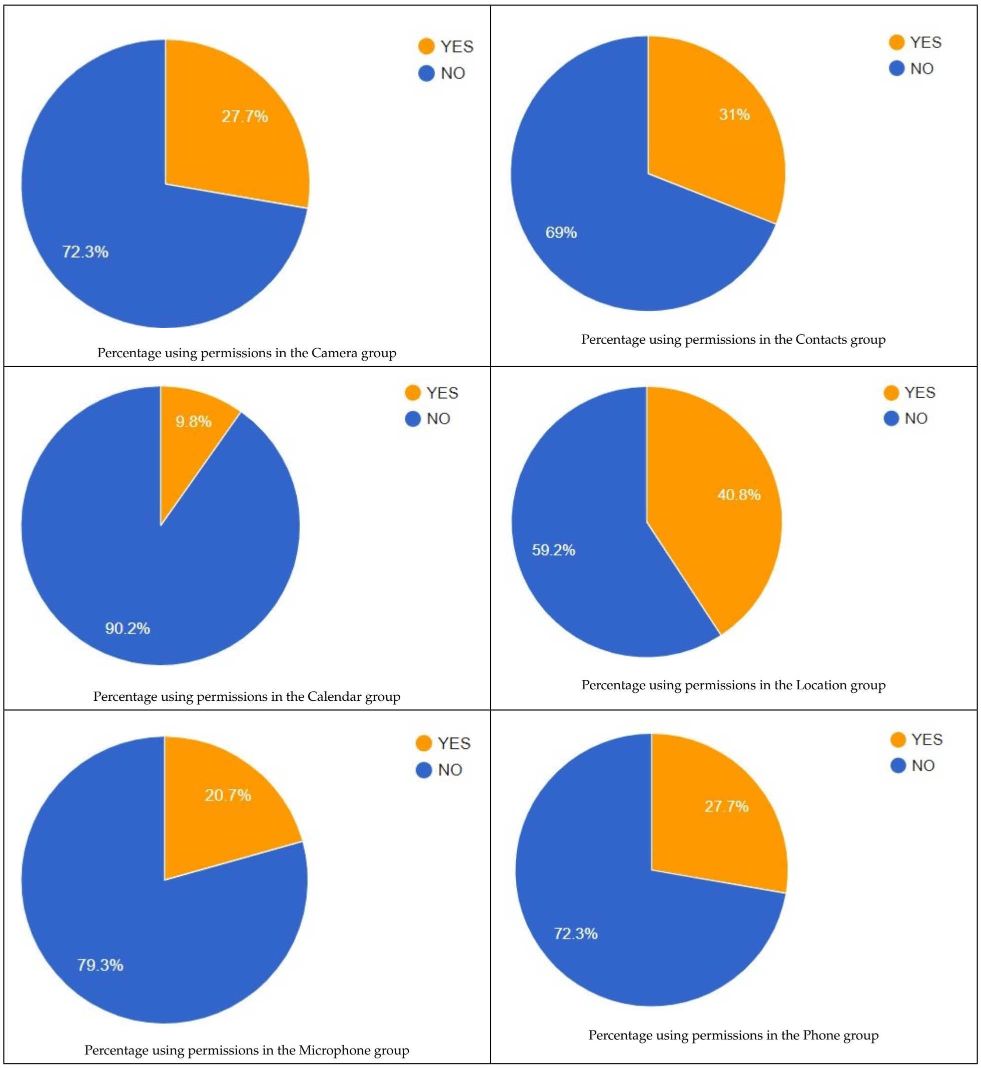


Figure 6. Example of the graphical information provided for the mobile applications permissions section.

Additionally, in Figure 7, we see an important information item provided by the platform, namely, the number of applications using permissions in each dangerous group of mobile phone components.

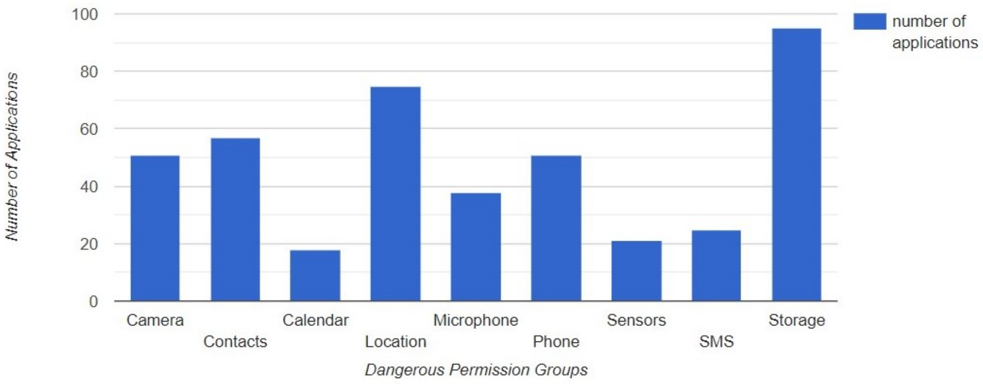


Figure 7. Number of applications using permissions in each dangerous group.

5. Discussion

The Privacy Flag project was an EU-funded research project whose goal was to raise awareness of data privacy and cybersecurity. The Privacy Flag platform, the main outcome of the project, utilized crowdsourcing intelligence for evaluating websites and smartphone applications to assess their privacy and security protection measures. The results of these automated evaluation checks have also proven useful in monitoring the overall status of the Internet in terms of the adoption of good security and privacy practices. The number of the actively participating users, during this effort, was well over 300. The volunteers have installed the Privacy Flag Web add-on and the Privacy Flag SmartApp and, therefore, have helped us to focus on the websites and applications that they frequently use. The key findings of this research can be summarized as follows:

- Cryptography: The majority of the websites use encryption, which is a positive finding (the corresponding percentage is about 54.8%). Various initiatives that offer certificates at a low or no cost at all (e.g., Let's Encrypt), have reinforced the adoption of encrypted communications on the Internet. Even so, the vast majority of the certificates originated from a trusted source (84.9%). The HSTS protocol is utilized in about one-third of the websites (38.4%). More advanced and secure techniques, however, such as the Public Key Pinning, are practically non-existing, as they were identified on only in the 1.3% of the websites.
- Legacy Technologies: Obsolete technologies are the relics of the first and second generations of the web. At those times, the trend was to overcome the limitations of the HTML protocol by developing new frameworks which could be executed in the web browsers as Web Plugins. The implications of this decision, however, were devastating in terms of privacy and security. This tendency allowed the execution of powerful applications on the users' computers, a concept known as *mobile code*. The obvious problem with this approach was that it was difficult to enforce effective sandboxing techniques to eliminate the risks of harmful actions. Gradually, but steadily, these technologies were deprecated and expelled from all modern browsers. Yet, some websites still rely on them. At the time of this research, the participating users did not encounter websites that required Java Applets (%0) or Sivlerlight extensions (%0). On the other hand, 13.1% of websites were based on Flash, which was more or less expected given the wide popularity of the particular Adobe tools. Counter-intuitively, ActiveX Controls which had been long abandoned were found on a small percentage of websites (3.2%).
- Modern Protocols: As discussed earlier, the latest version of the HTML protocol, HTML5, provides some powerful capabilities which might have security implications.

None of the websites that Privacy Flag users interacted with had enabled them (0%) at the time of the research.

- Mobile Applications: The number of permissions required by each mobile application is indicative of the privacy risks that may arise. Smartphone applications that request access to many different sensitive subsystems of the mobile devices represent a far greater threat than applications with minimum access requirements. This metric was effective in identifying potential risky smartphone applications with the Privacy Flag SmartApp tool, but the overall statistical findings in the Privacy Flag Observatory cannot be directly translated to privacy recommendations. Nonetheless, these findings are significant in the sense that they demonstrate that modern applications require a considerable number of permissions to function properly.

The findings of the Privacy Flag project highlight the developing trends towards the adoption of more secure and privacy-friendly technologies. Cleartext data transmission over the network is less frequent and in the foreseeable future, it is expected to vanish. More robust protocols, such as the HSTS, that can enforce proper encrypted communications, are gaining traction. On the other hand, the most effective mechanisms, such as the Certificate Pinning, are practically nonexistent in terms of user acceptance. The evolution of software technologies and web standards are helping users to abandon the obsolete and insecure model of the inclusion of external plugins to the browsers. We were unable to identify websites that still relied on Java Applets or the Silverlight plugins. A non-negligible 13.1% share of websites were using Adobe's Flash to display interactive multimedia content. Given the extremely poor security record of this framework, it is rather disconcerting that it is still in use, albeit in declining numbers. A very small percentage of 3.2% of websites have not removed ActiveX controls, which is very surprising given the fact that ActiveX components are among the oldest and least secure mobile code technologies. One way to address most of these issues is to adopt new standards that have been developed to be "secure by design", such as the new HTML5 standard. Nonetheless, the powerful functionality of the HTML5 protocol must be very cautiously implemented to minimize the risks of abuse.

6. Concluding Remarks and Future Work

Privacy Flag promoted a new risk detection and privacy awareness approach based on the constant monitoring of the adoption of good security practices by the "crowd". Usually, privacy risk detection and prevention are mostly of a centralized and static nature, under the control of cybersecurity companies that provide to SOCs security management services. This approach, however, has a number of disadvantages: (a) users rely on the credibility and trustworthiness of these companies; (b) commercial cybersecurity services can be expensive for the average user; (c) it is not easy, even for large security companies, to locate, log and analyze *all* possible security and privacy breaches; (d) many users are suspicious of the data collection policies of companies; and (e) many security breaches that users encounter are neglected and not shared with other users.

The Privacy Flag approach aims to address these issues. The Privacy Flag Observatory cannot replace but should be used complementary to all traditional protective applications, such as Intrusion Prevention Systems (IPS), AntiViruses (AV), Firewalls, etc. The first release of the Privacy Flag platform tools is available on the Privacy Flag website (https://privacyflag.eu, accessed on 30 November 2022).

As more users participate in the Privacy Flag platform, along with the necessary fine-tuning of the heuristic algorithms, the accuracy of the crowdsourcing assessment mechanism will be increased and the graphical information provided by the observatory will be enhanced. In general, crowdsourcing systems can provide satisfactory results only when the number of their participants exceeds some thresholds. The challenge of attracting new users exists not only for the Privacy Flag Observatory, but for every crowdsourcing platform. Therefore, there is an inherent need to support crowdsourcing projects as an alternative to commercial cybersecurity solutions. The aim of our work was to signify the need for a more decentralized strategy against digital privacy threats. Of course, this will require significant investment to support the large numbers of volunteering users. The Privacy Flag Server has been successfully stress-tested with scenarios of up to 5000 concurrent connections. Nonetheless, a full-scale development would require a partial redesign of the system and investment in more powerful equipment. Complimentary to the traditional privacy threats, the proliferation of IoT devices creates additional security and privacy issues. The Privacy Flag approach provides some very basic and preliminary functionality for identifying and pinpointing IoT device issues. The goal is to rank each IoT device according to its privacy impact and inform, accordingly, users. Since this work is at a very initial stage, it was not discussed in this manuscript.

Another important issue is the composition of the monitored threats. The Privacy Flag project was initiated some years ago. Since then, the threat landscape has evolved considerably. A whole range of new cyber attacks has appeared and therefore, the threat indicators of the Privacy Flag Observatory should be updated accordingly. On the other hand, it is still useful to monitor the effectiveness of cyber attacks against known and obsolete technologies. The extension of the Privacy Flag Observatory with novel and sophisticated privacy threats will facilitate the comparison between old and new attack vectors. Such an analysis can help the scientific community to understand the effectiveness of the existing cybersecurity mechanisms and develop the best strategies to defend against privacy-invasive practices.

Author Contributions: All authors have contributed equally to all aspects of the paper. All authors have read and agreed to the published version of the manuscript.

Funding: EU funded project Privacy Flag.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We thank the anonymous referees for their valuable comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Berners-Lee, T. *Universal Resource Identifiers in WWW: A Unifying Syntax for the Expression of Names and Addresses of Objects on the Network as Used in the World-Wide Web*; RFC 1630: Washington, DC, USA, 1994. [CrossRef]
2. Kim, S.J.; Viswanathan, V.; Lee, H.M. Platform war vs. platform synergy? A longitudinal analysis of media substitution between personal computers and mobile devices. *J. Broadcast. Electron. Media* **2020**, *64*, 65–88. [CrossRef]
3. Singh, S.; Singh, N. Internet of Things (IoT): Security challenges, business opportunities & reference architecture for E-commerce. In Proceedings of the 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), Delhi, India, 8–10 October 2015; pp. 1577–1581.
4. Mulliner, C.; Oberheide, J.; Robertson, W.; Kirda, E. Patchdroid: Scalable third-party security patches for android devices. In Proceedings of the 29th Annual Computer Security Applications Conference, Austin, TX, USA, 4–8 December 2013; pp. 259–268.
5. Isaak, J.; Hanna, M.J. User data privacy: Facebook, Cambridge Analytica, and privacy protection. *Computer* **2018**, *51*, 56–59. [CrossRef]
6. Kaur, J.; Ramkumar, K. The recent trends in cyber security: A review. *J. King Saud-Univ.-Comput. Inf. Sci.* **2021**. [CrossRef]
7. Alagheband, M.R.; Mashatan, A.; Zihayat, M. Time-based gap analysis of cybersecurity trends in academic and digital media. *ACM Trans. Manag. Inf. Syst.* **2020**, *11*, 1–20. [CrossRef]
8. Sundaramurthy, S.C.; Case, J.; Truong, T.; Zomlot, L.; Hoffmann, M. A tale of three security operation centers. In Proceedings of the 2014 ACM Workshop on Security Information Workers, New York, NY, USA, 7 November 2014; pp. 43–50.
9. Jacobs, P.; Arnab, A.; Irwin, B. Classification of security operation centers. In Proceedings of the 2013 Information Security for South Africa, South Africa, 14–16 August 2013; pp. 1–7.
10. Ristić, I. SSL/TLS Deployment Best Practices. 2012. Available online: https://www.ssllabs.com/downloads/SSL_TLS_Deployment_Best_Practices_1 (accessed on 22 October 2022).
11. Lavrenovs, A.; Melón, F.J.R. HTTP security headers analysis of top one million websites. In Proceedings of the 2018 10th International Conference on Cyber Conflict (CyCon), Tallinn, Estonia, 29 May–1 June 2018; pp. 345–370.

12. Felt, A.P.; Barnes, R.; King, A.; Palmer, C.; Bentzel, C.; Tabriz, P. Measuring {HTTPS} adoption on the web. In Proceedings of the 26th USENIX Security Symposium (USENIX Security 17), Vancouver, BC, Canada, 16–18 August 2017; pp. 1323–1338.
13. Libert, T. Exposing the hidden web: An analysis of third-party HTTP requests on 1 million websites. *arXiv* **2015**, arXiv:1511.00619.
14. Kaufhold, M.A.; Basyurt, A.S.; Eyilmez, K.; Stöttinger, M.; Reuter, C. Cyber Threat Observatory: Design and Evaluation of an Interactive Dashboard for Computer Emergency Response Teams. In Proceedings of the ECIS, Crete, Greece, 18–24 June 2022.
15. Douha Prieto, I. Analysis, Detection and Classification of Web Tracking Techniques. Master's Thesis, Universitat Politècnica de Catalunya, Barselona, Spain, 2021.
16. Calciati, P.; Kuznetsov, K.; Gorla, A.; Zeller, A. Automatically Granted Permissions in Android Apps: An Empirical Study on Their Prevalence and on the Potential Threats for Privacy. In Proceedings of the 17th International Conference on Mining Software Repositories, Seoul, Republic of Korea, 29–30 June 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 114–124.
17. Gibler, C.; Crussell, J.; Erickson, J.; Chen, H. AndroidLeaks: Automatically Detecting Potential Privacy Leaks in Android Applications on a Large Scale. In Proceedings of the Trust and Trustworthy Computing, Vienna, Austria, 13–15 June 2012; Katzenbeisser, S., Weippl, E., Camp, L.J., Volkamer, M., Reiter, M., Zhang, X., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 291–307.
18. Liu, B.; Andersen, M.S.; Schaub, F.; Almuhimedi, H.; Zhang, S.A.; Sadeh, N.; Agarwal, Y.; Acquisti, A. Follow My Recommendations: A Personalized Privacy Assistant for Mobile App Permissions. In Proceedings of the Twelfth Symposium on Usable Privacy and Security (SOUPS 2016), USENIX Association, Denver, CO, USA, 22–24 June 2016; pp. 27–41.
19. Harbach, M.; Hettig, M.; Weber, S.; Smith, M. Using Personal Examples to Improve Risk Communication for Security & Privacy Decisions. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA, 26 April–1 May 2014; CHI '14, pp. 2647–2656. [CrossRef]
20. Yamada, A.; Tanaka, S.; Sawaya, Y.; Kubota, A.; Matsuda, S.; Matsumura, R.; Umemoto, S.; Christin, N.; Nakajima, J.; Crichton, K.; et al. Mobile Security Behavior Observatory: Long-term Monitoring of Mobile User Behavior. USENIX ATC'20: 2020 USENIX Conference on Usenix Annual Technical Conference, Berkeley, CA, USA, 15–17 July 2020.
21. Meyer, C.; Schwenk, J. SoK: Lessons Learned from SSL/TLS Attacks. In Proceedings of the Information Security Applications, Jeju Island, Republic of Korea, 19–21 August 2014; pp. 189–209.
22. Meyer, C.; Schwenk, J. Lessons learned from previous SSL/TLS attacks-a brief chronology of attacks and weaknesses. *Cryptol. Eprint Arch.* **2013**.
23. Eldewahi, A.E.; Sharfi, T.M.; Mansor, A.A.; Mohamed, N.A.; Alwahbani, S.M. SSL/TLS attacks: Analysis and evaluation. In Proceedings of the 2015 International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering (ICCNEEE), Khartoum, Sudan, 7–9 September 2015; pp. 203–208.
24. Sarkar, P.G.; Fitzgerald, S. Attacks on ssl a Comprehensive Study of Beast, Crime, Time, Breach, Lucky 13 & rc4 Biases. 2013. Available online: https://www.isecpartners.com/media/106031/sslattackssurvey.pdf (accessed on 20 March 2021).
25. Fogel, B.; Farmer, S.; Alkofahi, H.; Skjellum, A.; Hafiz, M. POODLEs, More POODLEs, FREAK Attacks Too: How Server Administrators Responded to Three Serious Web Vulnerabilities. In Proceedings of the Engineering Secure Software and Systems—8th International Symposium, ESSoS 2016 Proceedings, London, UK, 6–8 April 2016; Caballero, J., Bodden, E., Athanasopoulos, E., Eds.; Springer: Berlin, Germany, 2016; Volume 9639, pp. 122–137. [CrossRef]
26. Karakostas, D.; Zindros, D. Practical new developments on BREACH. *Black Hat Asia* **2016**.
27. Gluck, Y.; Harris, N.; Prado, A. BREACH: Reviving the CRIME attack. *Unpubl. Manuscr.* **2013**.
28. Durumeric, Z.; Kasten, J.; Adrian, D.; Halderman, J.A.; Bailey, M.; Li, F.; Weaver, N.; Amann, J.; Beekman, J.; Payer, M.; et al. The Matter of Heartbleed. In Proceedings of the Internet Measurement Conference, Vancouver, BC, Canada, 5–7 November 2014; pp. 475–488.
29. Hodges, J.; Jackson, C.; Barth, A. Http Strict Transport Security (hsts). 2012. Available online: http://tools.ietf.org/html/draft-ietf-websec-strict-transport-sec-04 (accessed on 22 October 2022).
30. Yee, P. Updates to the internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile. *RFC 6818* **2013**.
31. Fu, Y.; Wang, Q.; Lin, J.; Sun, A.; Lu, L. Exploring the Security Issues of Trusted CA Certificate Management. In Proceedings of the Information and Communications Security, Virtual Event, 6–9 September 2021; Gao, D., Li, Q., Guan, X., Liao, X., Eds.; Springer International Publishing: Berlin, Germany, 2021; pp. 384–401.
32. Kent, S. Evaluating certification authority security. In Proceedings of the 1998 IEEE Aerospace Conference Proceedings (Cat. No.98TH8339), Snowmass, CO, USA, 28 March 1998; Volume 4, pp. 319–327. [CrossRef]
33. Aas, J.; Barnes, R.; Case, B.; Durumeric, Z.; Eckersley, P.; Flores-López, A.; Halderman, J.A.; Hoffman-Andrews, J.; Kasten, J.; Rescorla, E.; et al. Let's Encrypt: An Automated Certificate Authority to Encrypt the Entire Web. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, Association for Computing Machinery, New York, NY, USA, 11–15 November 2019; CCS '19, pp. 2473–2487. [CrossRef]
34. Petrov, I.; Peskov, D.; Coard, G.; Chung, T.; Choffnes, D.; Levin, D.; Maggs, B.M.; Mislove, A.; Wilson, C. Measuring the Rapid Growth of HSTS and HPKP Deployments. Available online: http://www.cs.umd.edu/content/measuring-rapid-growth-hsts-and-hpkp-deployments (accessed on 10 October 2022).

35. Buchanan, W.J.; Helme, S.; Woodward, A. Analysis of the adoption of security headers in HTTP. *IET Inf. Secur.* **2018**, *12*, 118–126. [CrossRef]
36. De los Santos, S.; Torres, J. Analysing HSTS and HPKP implementation in both browsers and servers. *IET Inf. Secur.* **2018**, *12*, 275–284. [CrossRef]
37. Buhov, D.; Rauchberger, J.; Schrittwieser, S. FLASH: Is the 20th Century Hero Really Gone? Large-Scale Evaluation on Flash Usage & Its Security and Privacy Implications. *J. Wirel. Mob. Netw. Ubiquitous Comput. Dep. Appl.* **2018**, *9*, 26–40.
38. Mavroudis, V.; Hao, S.; Fratantonio, Y.; Maggi, F.; Kruegel, C.; Vigna, G. On the privacy and security of the ultrasound ecosystem. *Proc. Priv. Enhancing Technol.* **2017**, *2017*, 95–112. [CrossRef]
39. Tian, Y.; Liu, Y.C.; Bhosale, A.; Huang, L.S.; Tague, P.; Jackson, C. All your screens are belong to us: Attacks exploiting the html5 screen sharing api. In Proceedings of the 2014 IEEE Symposium on Security and Privacy, San Jose, CA, USA, 18–21 May 2014; pp. 34–48.
40. Loreto, S.; Romano, S.P. Real-time communications in the web: Issues, achievements, and ongoing standardization efforts. *IEEE Internet Comput.* **2012**, *16*, 68–73. [CrossRef]
41. Hayes, S. Java and activeX: Background and risks to the business. *Comput. Fraud. Secur.* **1998**, *1998*, 9–12. [CrossRef]
42. Hopwood, D. A comparison between java and activeX security. *Netw. Secur.* **1997**, *1997*, 15–20. [CrossRef]
43. Špiláková, P.; Jašek, R.; Schauer, F. Security risks of java applets in remote experimentation and available alternatives. *Appl. Math. Comput. Sci. Eng.* **2014**. Available online: http://www.europment.org/library/2014/varna/bypaper/AMCSE/AMCSE-23.pdf (accessed on 20 October 2022).
44. Niinimaki, P.; Markkanen, P.; Kajava, J. Java applets and security. In Proceedings of the Databases and Information Systems, 3rd IEEE International Baltic Workshop, Tallinn, Estonia, 16–19 June 1998; pp. 125–136.
45. Suresh, J.K. Comparative Analysis of Security and Accessibility of Silverlight XAML with Other User Interface. *Int. J. Comput. Electr. Eng.* **2009**, *1*, 1793–8163.
46. Kontaxis, G.; Antoniades, D.; Polakis, I.; Markatos, E.P. An empirical study on the security of cross-domain policies in rich internet applications. In Proceedings of the Fourth European Workshop on System Security, Salzburg, Austria, 10 April 2011; pp. 1–6.
47. Verleg, P.; van Eekelen, M.; Vranken, H. Cache Cookies: Searching for Hidden Browser Storage, Bachelor's Thesis, Radboud University, Nijmegen, The Netherlands, 2014.
48. Saito, T.; Koshiba, R. Examination and Comparison of Countermeasures Against Web Tracking Technologies. In Proceedings of the International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, Asan, Republic of Korea, 1–3 July 2019; pp. 477–489.
49. Souppaya, M.; Scarfone, K. Guidelines for managing the security of mobile devices in the enterprise. *NIST Spec. Publ.* **2013**, *800*, 124.
50. Emerson, P. The original Borda count and partial voting. *Soc. Choice Welf.* **2013**, *40*, 353–358. [CrossRef]
51. Wu, L.; Du, X.; Fu, X. Security threats to mobile multimedia applications: Camera-based attacks on mobile phones. *IEEE Commun. Mag.* **2014**, *52*, 80–87. [CrossRef]
52. Wernke, M.; Skvortsov, P.; Dürr, F.; Rothermel, K. A classification of location privacy attacks and approaches. *Pers. Ubiquitous Comput.* **2014**, *18*, 163–175. [CrossRef]
53. Petracca, G.; Sun, Y.; Jaeger, T.; Atamli, A. Audroid: Preventing attacks on audio channels in mobile devices. In Proceedings of the 31st Annual Computer Security Applications Conference, Los Angeles, CA, USA, 7–11 December 2015; pp. 181–190.
54. Hwang, S.; Lee, S.; Kim, Y.; Ryu, S. Bittersweet adb: Attacks and defenses. In Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, Singapore, 14–17 March 2015; pp. 579–584.
55. Raij, A.; Ghosh, A.; Kumar, S.; Srivastava, M. Privacy risks emerging from the adoption of innocuous wearable sensors in the mobile environment. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Atlanta, GA, USA, 10–15 April 2011; pp. 11–20.
56. Tu, G.H.; Li, C.Y.; Peng, C.; Li, Y.; Lu, S. New security threats caused by IMS-based SMS service in 4G LTE networks. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 1118–1130.
57. Penning, N.; Hoffman, M.; Nikolai, J.; Wang, Y. Mobile malware security challeges and cloud-based detection. In Proceedings of the 2014 International Conference on Collaboration Technologies and Systems (CTS), Minneapolis, MN, USA, 19–23 May 2014; pp. 181–188.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.