A study on Information & Cyber Security System

Author: Mrs,D.Renukadevi

Asst.Professor

Department of B,Com CS

Sri Ramakrishna College of Arts and Science

Coimbatore

Abstract

Cyber-security systems, which protect networks and computers against cyber-attacks, are becoming common due to increasing threats and government regulation. At the same time, the enormous amount of data gathered by cyber-security systems poses a serious threat to the privacy of the people protected by those systems. To ground this threat, we survey common and novel cyber-security technologies and analyze them according to the potential for privacy invasion. We suggest a taxonomy for privacy risks assessment of information security technologies, based on the level of data exposure, the level of identification of individual users, the data sensitivity and the user control over the monitoring, and collection and analysis of the data. We discuss our results in light of the recent technological trends and suggest several new directions for making these mechanisms more privacy-aware. CCS Concepts: • Security and privacy → Privacy protections; Malware and its mitigation; Intrusion detection systems; Information flow control; Firewalls; • Networks → Network privacy and anonymity;

Key Words : Information security, privacy, system monitoring, network surveillance, privacy-preserving methods

INTRODUCTION

In recent years, governments and corporations have increasingly relied on cyber-security systems to protect against increasing threats on networks, devices, and organizational and personal information. These systems prevent adversaries from breaking into networks and devices, from sabotaging digital activity, and from accessing private information. At the same time, by monitoring networks and computing devices, cyber-security systems ultimately affect individuals' privacy. Systems in domains such as intrusion detection, malware detection, data leakage prevention, and phishing identification regularly monitor network traffic, device use, and personal communications. In many cases, the monitoring system can trace the identities of users and access sensitive information. For instance, many enterprise cyber-security systems monitor IP addresses that can be easily traced back to a particular individual. Moreover, the user's device identification on mobile devices is often accessed by cyber-security applications. Therefore, while cyber-security mechanisms protect individuals from attacks from hackers and other third-party adversaries, they also create new vulnerabilities for privacy violation from the entity that runs the cyber-security system. These vulnerabilities can be realized if the security systems themselves are compromised, if insiders make use of this information, or if the personal data are used contrary to the expectations of end users.

The increasing threat of computer attacks and the intrusiveness of cyber-security mechanisms present policymakers and technology developers with the difficult challenge of balancing security risks against privacy and civil liberties concerns (Tene 2014; Landau 2014). The fact that many national cyber-security policies require the sharing of the detailed information of attack logs and other types of information necessitates an urgent understanding of the privacy risks related to cyber-security (Sales 2013; Nolan 2015). Privacy concerns are among the reasons why employees switch to their personal devices (e.g., smartphones and portable computers) to perform workrelated activities (Pfleeger et al. 2014) and home-users turn away from some anti-virus applications (Warkentin and Willison 2009). Therefore, understanding and solving privacy threats is crucial, as those threats can reduce the acceptance and usage of cyber-security systems by organizations and individuals, leading to increased number of threats for everybody.

2. CYBER-SECURITY MECHANISMS

The aim of cyber-security is to protect networks, computers, programs, and data from attacks and unauthorized access. This section first introduces cyber attacks and provides the language for describing cyber attacks and cyber-security systems. The second part of the section proposes a categorization of cyber-security mechanisms that will be helpful when considering their impact on privacy.

2.1 Classification of Cyber Attacks

A first dimension for classifying an attack is the goal of the attack. This is often related to the way an adversary monetizes the attack (e.g., by stealing information and selling it to advertisers or criminals). Overall, the attack goals fall into one of the following categories (Lala and Panda 2001): (1) stealing information, such as data on a device, media files, and user credentials; this action is usually performed by spyware malware; (2) tracking user information, i.e., monitoring users' sensitive data (e.g., locations, activities, or health-related data); this action is usually achieved using mobile malware; (3) taking control of a system, as is done by Trojan, botnet, and rootkit (Graziano et al. 2016).

2.1.1 Hardware Attacks.

At the hardware level, we find attacks that include manufacturing backdoors, gaining access to memory, and hardware tampering. The common goal of these attacks is twofold: modifying the hardware to access sensitive information and creating a backdoor (Tehranipoor and Koushanfar 2010) (e.g., install an invisible program in the hardware circuit) that can be used to regain access to the compromised machine. Such hardware attacks can be applied to several types of devices, such as network appliances, surveillance systems, and industrial control systems.

2.1.2 Network Attacks.

Network attacks can target the network protocol or the network device software, and their goal is either the denial of service or hijacking a network connection to steal sensitive data. Specifically, frequent attacks using vectors at the network layer are Denial of Service (DoS) (Schweitzer et al. 2016), IP spoofing (Thang and Nguyen 2016), and man in the middle attacks (Desmedt 2011).

2.1.3 Application Attacks.

At the application level, phishing and client-side web attacks are the most common attack vectors, according to the main security market players (e.g., Symantec (2015b)). These attacks target applications such as e-mail services and browsers, since they are the most exposed to the Internet. Regarding attacks through email, phishing is a form of fraud in which the attacker tries to gather sensitive information, such as credentials and credit card numbers by impersonating a reputable entity or person via email, IM or other communication channels (Fette et al. 2007; Ma et al. 2009). Many application-level attacks make use of social engineering techniques that use humans to compromise systems, manipulating them into carrying the attack through deceit (Krombholz et al. 2015). A common example of client-side web attacks is Cross-Site Scripting (XSS), which consists of injecting client-side script code (e.g., JavaScript) into web pages. Such injected code could be used for different purposes, such as to bypass access control or to force a user to execute some actions on a remote website on behalf of the attacker. A large number of application level attacks can be categorized as malware (Lanzi et al. 2010). Malware is any malicious software that an attacker manages to run on the target computer. It is used to gather sensitive information, to gain access to private computer systems, or to perform massive attacks. Malware is defined by its malicious intent, acting contrary to user requirements. Malware can be classified into several categories depending on the design goal. The most common malware categories are mobile malware, botnets, spyware (which transmit personal communications), ransomware (which encrypt a victim's data and force victims to pay to decrypt it), and banking malware (Symantec 2015b). Different techniques are used to install malware on a target system. For example, mobile malware is usually installed via SMS, via unofficial application repositories, or by exploiting vulnerabilities of the OS. Once the malware is installed, it can perform several malicious actions, such as stealing information (in this case, it is also called spyware) or tracking user actions.

2.2 Classifying Cyber-Security Technologies

In this subsection, we propose a classification of cyber-security technologies. It is important to note that commercial products do not necessarily have a direct mapping in our classification system, since they often package different protection mechanisms under the same name (e.g., anti-virus), which are possibly offered both as standalone and client-server architectures and for different ecosystems. In the following subsections, we describe four classification categories, and in Section 2.2.5, we present the classification of well-known cyber-security solutions.

2.2.1 System Architecture.

Protection systems are software packages that are designed to be deployed according to a specific architecture. The three main architectures are standalone,centralized client-server, and collaborative architectures. The first architecture (standalone) is an architecture in which the cyber-security mechanism is installed only on the local machine to be protected. Such a configuration can be found in the first generation of anti-virus products (Cristalli et al. 2016), where the system performs the entire detection task on the local machine without passing data across the network. The second architecture (centralized client-

server) is composed of a client, which is usually installed on the system to be protected, and a centralized server that runs the detection algorithm. This architecture is often adopted by contemporary anti-virus systems when, for example, it has to check whether some visited web domains are malicious or not. The client sends the URL of a particular machine, and the server replies based on its blacklist. The last architecture (collaborative) is implemented as a distributed system, possibly following a peer-to-peer paradigm. It is often adopted by network detection systems such as Snort (Roesch et al. 1999), where sensors are localized on different network nodes and cooperate with each other using a correlation algorithm to determine anomalies/attacks on the monitored network. Recent examples include Worminator, a collaborative intrusion detection system based on encoding threats using Bloom filters (Locasto et al. 2005; Vasilomanolakis et al. 2015a), and other works based on hidden Markov random field (Xie et al. 2016) and autonomic and self-organizing hive-like collaboration (Korczynski et al. 2016).

2.2.2 Type of Detection.

Defense mechanisms can operate at the same three levels defined for the attack model (hardware, network, and application) and can be broadly classified in two main categories: anomaly-based detection, which learns the routine behavior of a user or application and tries to capture anomalies, i.e., the deviations from the routine behavior (Garcia-Teodoro et al. 2009; Continella et al. 2017), and signature-based detection, which tries to characterize the generic behavior of an attack as a signature and then monitors the system, detecting an attack when the signature is observed. There are two main approaches for the last category: the automatic approach builds the signature by using behavioral analysis (system calls, function calls, etc.), while the manual approach requires security experts to explicitly construct the signature by specifying the malicious behavior (Cannady 1998)

2.2.3 Type of Data.

Security systems can also be distinguished based on the type of data that their detection algorithm processes. For example, a network intrusion detection system such as Snort (Roesch et al. 1999) analyzes network packets at different network protocol levels, while a host intrusion detection system analyses system call operations performed by an application running on a host. We classify the data used by security technologies into three main categories: (a) application data, (b) file data, and (c) network data. The first category includes both system calls performed by applications and application level data exchanged on the network. For system calls and function libraries, some mechanisms look only at the call itself, while others also inspect the specific parameters of the call; similarly for HTTP protocol requests or emails, some mechanisms look only at the header (e.g., for HTTP, they look only at the request line, i.e., GET and POST commands), while others also inspect the body (e.g., the data being posted with an HTTP request). In the second category (file data), we consider the files that are inspected to ensure that they do not hide a security threat. The most relevant ones are Microsoft Office documents, PDF documents, media files (video, pictures, etc.), and executable files. In the third category (network data), we haveinformation contained in low-level network packets. Technically, low level refers to levels below application. Different mechanisms may inspect both the packet header and the contained data or the header only.

2.2.4 Ecosystems.

Another dimension that we use to classify security systems is associated with the ecosystems in which a detection mechanism can be applied. In particular, we can apply defensive mechanisms in three main ecosystems: enterprise, mobile devices, and IoT. The enterprise ecosystem represents the typical organization infrastructure, which is composed of locally connected PCs, servers, and network devices, but can also be extended to the use of private and public cloud and web technologies. The mobile devices ecosystem is composed of personal devices typically used in mobility (e.g., smartphones and tablets). The Internet of Things ecosystem is just emerging, but it is already posing serious security concerns. It includes IP-enabled devices (e.g., netcams and smart appliances) as well as sensor networks synchronized to IP-enabled hubs.

3. Data Exposure

To analyze the privacy risk posed by each technology, we need to understand which entities have access to the data and the context in which data exposure occurs. There are two main factors influencing data exposure: One is related to the system software architecture that defines the data flow. The other is related to how data in transit, data at rest, and data in use are protected. Spiekermann and Cranor define network centricity as the "degree to which a user's system relies on a network infrastructure to provide a service, as well as the degree of control a network operator can exercise over a client's operations" (Spiekermann and Cranor 2009). A higher network centricity means that the data are more accessible and controllable by external entities, such as data collectors, service providers, location servers, and cloud infrastructure operators. Figure 2 depicts typical network centricity models: (1) a standalone topology in which a user has full control over a standalone client; (2) a centralized topology that rests on a centralized server that monitors data or communications; (3) an external client-server third-party provider that monitors a network by using the cloud; (4) a collaborative topology that carries out monitoring by using a decentralized architecture.

3.1 Level of Identification

It is well known that the simple removal of explicit identifiers from released data is not sufficient to enforce anonymity (Samarati 2001). Indeed, in several cases, an adversary may re-identify the data respondents by joining part of the released data, called Quasi-Identifiers (QIs), with available background knowledge. Consider, for instance, the release of a medical record including personal data such as birth date, gender, and home town. Even if the record does not include the patient's name or social security number (SSN), an adversary having access to a personal information registry may easily limit the set of candidate respondents to those people matching the personal information in the record. In the worst case, the adversary can uniquely re-identify the respondent. Several privacy notions have been proposed in the literature for measuring the level of reidentification of released data, and algorithms have been devised to enforce those notions in different domains (release of database records, transaction data, statistics, etc.). Perhaps the most known privacy notion in databases is k-anonymity (Sweeney 2002): a record is k-anonymous if it can be

associated with a set of at least k possible respondents. This privacy notion can be enforced by generalizing QI values such that each record belongs to a group (called a QI-group) of at least k records having identical values for the QI attributes. Assuming that each individual is the respondent of at most one record, each record can be associated with at least k individuals (i.e., the respondents of that record's QI group). Hence, the value k is intended to measure the level of protection from identity disclosure. However, this approach was later shown to have several drawbacks, since the distribution of sensitive values associated with a group of k individuals (even if undistinguishable) has a relevant impact on the privacy risk of revealing the value associated with a specific individual (Li et al. 2007) (e.g., in the case where the same sensitive value is associated with all individuals).

3.2 Data Sensitivity

As noted above, an adversary can perpetrate a privacy violation from either side of the sensitive association. While k-anonymity provides some level of protection against re-identification, it does not always prevent privacy violations. Indeed, if all the records in a QI-group share the same value for a released sensitive attribute, the adversary can reconstruct the sensitive association between each respondent of the records in that group and the sensitive value. Based on the above weakness of k-anonymity, various privacy notions have been proposed to protect both sides of the sensitive association. In particular, l-diversity (Machanavajjhala et al. 2006) ensures that the records in a QI-group have sufficiently diverse values for the private attribute. Hence, the value l can be used to measure the level of protection from both identity and attribute disclosure. While l-diversity considers syntactic diversity among sensitive data, a further notion, named t-closeness (Li et al. 2007), has been introduced to offer additional protection from attribute disclosure based on the semantics of sensitive data. Anonymity is not enough to ensure privacy in all situations relevant to cyber-security. For example, some services may require authentication with a real identity for billing or for accountability. In these cases, privacy protection techniques need to focus on sensitive attribute disclosure. Obfuscation via generalization is one of the most popular techniques. Generalizing location data, for example, is based on the observation that location information becomes less sensitive when it is less precise, i.e., knowing that an individual is in Manhattan on a given day is usually less sensitive than knowing that an individual is at the address of a particular political party at the time a campaign speech is given (Mascetti et al. 2014).

3.3 Level of User Control

If identified or identifiable personal information is collected, then Fair Information Practices point to mechanisms of user control as policy-based mechanisms that can mitigate privacy threats (Wright and De Hert 2011). Specifically, privacy impact assessment methods describe how notices and choices can inform users about data practices and provide meaningful controls to the user (Oetzel and Spiekermann 2014; of the Australian Information Commissioner 2014). Moreover, for technologies to be perceived as trusted, they are required to provide users with the ability to view a comprehensive or a short privacy policy that includes information about the collection, analysis, use, processing, exposure, and transfer of personal data (Pollach 2007). However, it is very challenging to design supportive technological interfaces that provide

users appropriate ad hoc notices regarding data collection and use choices. According to Spiekermann and Cranor, meaningful and timely information "can be offered with minimal disruption by positioning notices at the point in an interaction where they are most relevant, by providing information in a format that succinctly conveys the most important information, and by limiting notices to situations that are most likely to raise privacy concerns" (Spiekermann and Cranor 2009). Beyond notices, we can measure the level of control that technologies give their users in specifying preferences and have them applied to current and future situations. To analyze the level of user control, we look at the ability of a technology to interact with a user. While the actual notice and choice relies on the actual implementation of the technology, a preliminary condition for applying them is the underlying interaction model

4.Impact of the System Architecture

The specific software architecture of a system determines, among other things, how data are transferred between different entities and which process each entity should execute. Hence, the architecture is directly related to the potential data exposure to different parties. Standalone mechanisms, which are installed on a user's device (e.g., computer, smartphone, or wearable) and operate only locally, provide access only to the user (Symantec 2016c; Portokalidis et al. 2010) and therefore have low network centricity. Most systems, however, funnel data to a centralized server (Cheng et al. 2007; Symantec 2016b; Roesch et al. 1999; Paxson 1999; Portokalidis et al. 2010). As explained in Section 3.1, the actual level of data exposure depends also on how data are protected while stored, while in use (processing) and when in transit. While most defense mechanisms protect data in transit, low protection is usually offered for data at rest and in use. The exposure of unprotected personal data to multiple parties carries a privacy risk. This happens not only in the case of untrusted parties, but also in the case of attacks to these parties, as well as in the case of transfer of part of this data to other parties not explicitly involved in the cyber-security architecture. For instance, governments currently require organizations to share cyber-security information with the government and through collaborative exchanges, information that often includes personal information (Sales 2013).

4. Impact of Monitored Data

The possibility of anonymization in cyber-security is tightly related to the type of monitored data. Each type of data provides different levels of protection for users, ranging from data that is fully identified to data that can be re-identified but only with some significant effort. It is important to note that we have not found a cyber-security technology that claims that it can provide full anonymization to its users, given the existing re-identification body of knowledge. Several systems, such as phishing detectors, spam detectors, and mobile protection systems, have direct access to straightforward identifying information, such as a user's name, email address, phone number, or email content (sent and received). To protect privacy, these systems require complex mechanisms that can obfuscate identifying details to restore anonymity (Di Castro et al. 2016). In many cyber-security mechanisms, the monitored data do not directly contain the identity of users but are sufficient for re-identifying users. For example, many types of malware detection systems and anti-virus systems require ongoing access to system calls (including their parameters), which often contain

information about usernames, passwords, and other types of information that can be used to identify a user. Web monitoring systems access application-level protocols, such as HTTP and SMTP, all of which regularly contain information that can be used to re-identify users. For instance, HTTP header information can reveal personally identifiable information, such as email addresses typed in a web form (Starov et al. 2016) or the specific browser being used (Nikiforakis et al. 2013). Similarly, mobile protection systems access information that can be used to re-identify users, including the configuration of mobile devices (Kurtz et al. 2016)

Analysis Summary The analysis reveals that different aspects of cyber-security systems have different impacts on possible privacy breaches due to the exposure of identifiable personal data. Despite the analysis being quite involved and dependent on the specific design choices of a system with respect to different dimensions, a high-level consideration emerges: The system architecture has the most impact on data exposure, the type of data being inspected influences identification and sensitivity, and the ecosystem impacts user control (e.g., mobile devices may easily provide alerts, while IoT devices may not have a direct interface) and usually contains particular data sources that can be integrated with the some information to re-identify users. It is important to consider that certain characteristics of cyber-security technologies may significantly influence others. For example, anomaly-based detection is usually performed in a centralized client-server architecture, since a server has more computational resources and can improve the accuracy of the model by considering data from multiple systems. Hence, based on our analysis, privacy risks due to correlated cyber-security properties will lead to combined risks..

5.CONCLUSIONS

The taxonomy presented in this article suggests a way to classify and analyze the privacy implications of cyber-security defense systems. We find that almost all cyber-security technological categories require some access to personal sensitive information, under reasonable assumptions of re-identification and computing power. Our analysis reveals that evaluating the privacy risks involved in using a cyber-security system requires more than the system's generic technological category; it is necessary to classify them in terms of the dimensions identified in Section 2 and to carefully consider their impact on privacy risks, as discussed in Section 4. Since different privacy-preserving techniques have been proposed to mitigate specific privacy threats (e.g., anonymization and obfuscation for decreasing the sensitivity of the personal data), we believe that the identification of the privacy risks involved in a specific aspect of a cyber-security technology can offer guidance not only in choosing one technique over another but, more importantly, in designing more privacy-aware cyber-security technologies with little or no compromise with regard to their effectiveness in protecting from cyber attacks. For policymakers, this analysis can be used to guide the regulation, checks, and design requirements that follow the development of a technology. This is particularly important against the backdrop of legislation and policies that set the cyber-security requirements of government agencies and companies. This analysis can also serve as a framework for analyzing the trade-off between the risk that cyber-security systems protect against and the privacy risk that is imposed by the systems themselves. Because our analysis shows that users can be identified in nearly all

cybersecurity systems, we argue that policies should emphasize the embedding of privacy protections and controls in cyber-security requirements

REFERENCES

1.Jagdish Prasad Achara, Gergely Acs, and Claude Castelluccia. 2015. On the unicity of smartphone applications. In Proceedings of the 14th ACM Workshop on Privacy in the Electronic Society. ACM, 27–36.

2. Italian Data Protection Authority. 2016. Processing of personal data of employees by e-mail and other work tools. Retrieved from http://www.garanteprivacy.it/web/guest/home/docweb/-/docweb-display/docweb/5408460.

3. Andreas Kurtz, Hugo Gascon, Tobias Becker, Konrad Rieck, and Felix Freiling. 2016. Fingerprinting mobile devices using personalized configurations. Proc. Privacy Enhanc. Technol. 2016, 1 (2016), 4–19.

4. Bingdong Li, Jeff Springer, George Bebis, and Mehmet Hadi Gunes. 2013. A survey of network flow applications. J. Netw. Comput. Appl. 36, 2 (2013), 567–581.

5. Bingdong Li, Jeff Springer, George Bebis, and Mehmet Hadi Gunes. 2013. A survey of network flow applications. J. Netw. Comput. Appl. 36, 2 (2013), 567–581.