A Framework for Measuring Organizational Information Security Vulnerability

by

Changli Zhang
B.Eng., Northwestern Polytechnical University, China, 2002
M.Eng., Northwestern Polytechnical University, China, 2005
Ph.D., Northwestern Polytechnical University, China, 2009

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Computer Science

A Framework for Measuring Organizational Information Security Vulnerability

by

Changli Zhang
B.Eng., Northwestern Polytechnical University, China, 2002
M.Eng., Northwestern Polytechnical University, China, 2005
Ph.D., Northwestern Polytechnical University, China, 2009

Supervisory Committee

Dr. Kui Wu, Supervisor
(Department of Computer Science)

Dr. Sudhakar Ganti, Departmental Member
(Department of Computer Science)

ABSTRACT

In spite of the ever-growing technology in information security, organizations are still vulnerable to security attacks due to mistakes made by their employees. To evaluate organizational security vulnerability and keep organizations alert on their security situation, in this dissertation, we developed a framework for measuring the security vulnerability of organizations based on online behaviours analysis of their employees. In this framework, the behavioural data of employees for their online privacy are taken as input, and the personal vulnerability profiles of them are generated and represented as confusion matrices. Then, by incorporating the personal vulnerability data into the local social network of interpersonal security influence in the workplace, the overall security vulnerability of each organization is evaluated and rated as a percentile value representing its position to all other organizations. Through evaluation with real-world data and simulation, this framework is verified to be both effective and efficient in estimating the actual security vulnerability status of organizations. Besides, a demo application is developed to illustrate the feasibility of this framework in the practice of improving information security for organizations.

Contents

Supervisory Committee **ii**

Abstract **iii**

Table of Contents **iv**

List of Tables **vi**

List of Figures **vii**

Acknowledgements **ix**

Dedication **x**

1 Introduction **1**

1.1 Background and Motivation 1

1.2 Related Work 3

1.2.1 Human-centered Information Security 3

1.2.2 Organizations and Cybersecurity 4

1.3 Contributions 5

1.4 Agenda 6

2 The Problem, Solution and Framework **7**

2.1 The Problem: Social Engineering Attacks 7

2.2 A Social Engineering-Based Solution 9

2.3 The Framework 11

2.3.1 The Data Model 11

2.3.2 The Calculation Procedure 12

2.4 How to Apply the Framework in Real-world 14

3 Confusion Matrix-based People Vulnerability Analysis **16**
3.1 Confusion Matrix-based Awareness Model 16
3.2 The calculation Process ($\mathscr{F}$-step) 19
3.2.1 Classifying Privacy Items 19
3.2.2 Updating the Confusion Matrices 20
3.3 The Algorithm 22

4 Graph-based Organizational Vulnerability Calculation **24**
4.1 Interpersonal Security Influence Network 24
4.2 The Network-based Calculation ($\mathscr{G}$-step) 27
4.3 The Algorithm 29

5 Evaluation **32**
5.1 Overview 32
5.2 $\mathscr{F}$-step Simulation 32
5.2.1 Data Generation 32
5.2.2 Performance Evaluation 36
5.3 $\mathscr{G}$-step Simulation 36
5.3.1 Data Generation 36
5.3.2 Performance Evaluation 38
5.4 Experiment with Real-world Data 39
5.4.1 The Dataset 39
5.4.2 Calculation Results Analysis 42
5.5 Discussion 45

6 A Demo Application **46**
6.1 Introduction 46
6.2 The Bayesian Network Model 47
6.3 Implementation and Demonstration 51

7 Conclusions **54**
7.1 Concluding Remarks 54
7.2 Future Work 55

A Additional Information **57**

Bibliography **58**

List of Tables

Table 4.1 Types of interpersonal security influence in the workplace [1] . . 25
Table 4.2 The iterating process of calculating the importance values of all employees within the local social network of interpersonal security influence for an example organization 30

Table 6.1 Definition of the variables used in the Bayesian network model . 48

List of Figures

Figure 2.1 The pattern of social engineering attacks 8
Figure 2.2 The Internet as an extension of workplaces 10
Figure 2.3 The privacy item-employee-organization model 11
Figure 2.4 The 3-step procedure for calculating organizational security vulnerability 13
Figure 2.5 A system of providing services for organizational information security monitoring and rating 14

Figure 3.1 Example Confusion matrices for 5-level privacy sensitivity . . . 18

Figure 4.1 An example of local social networks formed by interpersonal security influence in the workplace [1] 26
Figure 4.2 An example organization for demonstration of the calculation process 29

Figure 5.1 Probabilistic graphical model for generating the response matrix $\boldsymbol{R}$, the confusion matrices $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$ and the sensitivity levels of all privacy items $\{t_i\}_{1\leq i\leq I}$ 33
Figure 5.2 Examples of generating confusion matrices from *Beta* distributions given that $C = 5$ 35
Figure 5.3 Changing of inter-iteration error and the distances to ground truth over iterations for Algorithm 1 37
Figure 5.4 An example of local social networks of interpersonal security influence generated by simulation 38
Figure 5.5 Changing of error along the iterations for Algorithm 2 39
Figure 5.6 Changing of overall organizational security vulnerability along the increment of personal vulnerability for certain groups of employees 40
Figure 5.7 Average exposure rates of privacy items in the experimental dataset 41

Figure 5.8 Distribution of number of privacy items exposed among employees in the experimental dataset 41
Figure 5.9 Top 100 organizations of the biggest number of employees recorded in the experimental dataset 42
Figure 5.10 Personal vulnerability calculation results from the experimental dataset 43
Figure 5.11 Final vulnerability scores for the 100 organizations listed in Figure 5.9 44

Figure 6.1 Flow of processing for the prototype system 47
Figure 6.2 The data structure after data enhancement, consisting of the basic business email accounts, enriched people profiles and profiles of organizations 49
Figure 6.3 The Bayesian network model for predicting security vulnerability of organizations or employees 50
Figure 6.4 Python-based implementation for the demo application 52
Figure 6.5 Structure and working procedure of Django web consoles 52
Figure 6.6 Snapshots of the web page for Bayesian network-based reasoning 53

ACKNOWLEDGEMENTS

I would like to thank:

all my family members for supporting me in the low moments and going through all the difficulties with me.

Dr. K. Wu, for offering me the opportunity to study again and for his mentoring, guidance, support, encouragement, and patience during my study.

Drs. I. Traore and S. Ganti, for their willingness to serve my oral defense committee and their time and effort to help improve the thesis.

I believe I know the only cure, which is to make one's centre of life inside of one's self, not selfishly or excludingly, but with a kind of unassailable serenity-to decorate one's inner house so richly that one is content there, glad to welcome any one who wants to come and stay, but happy all the same in the hours when one is inevitably alone.

Edith Wharton

DEDICATION

To my family
and
everyone who offered the help
along the way.

Chapter 1

Introduction

1.1 Background and Motivation

Ever since the advent of the Internet, organizations like companies, institutions and governments all over the world are experiencing unprecedented growth in technologies that protect their information systems in the workplace from being compromised. However, no matter how strong the technological defence layer is, human factors are often identified by the security community as the weakest link in the security chain for an organization [2–8]. Specifically, as the Internet has become an indispensable platform for organizational activities, lack of security awareness and improper online behaviours of employees have become a major threat to the information assets in organizations [3, 9, 10]. For this reason, nowadays, many attackers have started to target employees for access to the data and services within the organizational boundaries [11–15].

Indicated by the latest report from *Ponemon Institute* [16], since July 2018, about 25% recorded security breaches in organizations over the globe are attributed to system glitches, far less than what caused by malicious and criminal attacks (51%) and human factors within workplaces (24%) combined. Particularly, in Canada, the per breach financial cost is estimated to be about CAD 5.92 million, significantly above the global average (about CAD 5.23 million), but greatly dwarfed by that of the United States (about CAD 10.92 million). The industries that are highly affected include healthcare, finance, pharmaceutics, services and high technology, with losses like business disruption, revenue losses, and customer turnover. In essence, a holistic information security approach in which human aspects in information security

management are emphasized is no doubt imperative for modern organizations [5, 7, 15, 17–19].

In this respect, just like examining technologies deployed to assess the strength of information security for an organization, it is imperative to inspect how vulnerable its security system can be by evaluating the human factors in the workplace [8, 11, 20]. This is not an easy task. For one reason, we human beings are different by nature, whereas it is widely acknowledged that the information security awareness and behaviours of employees are highly subjected to their personality traits, demographic and psychological factors, risk-taking attitudes, and even decision-making styles [12, 21, 22]. However, the effort to reveal such correlations is extremely difficult and the result could be very biased and lack generality across organizations [6], not mentioning the technical difficulties and ethical issues for collecting behavioural data of employees both in and out of workplaces. For the other reason, the workplace of every single organization can be recognized as a local social network of a small scale but with a complex topological structure [1, 4, 10]. Within each social network, employees constantly influence each other through the way they perceive security risks and deal with security issues. Therefore, how to synthesize the information security profile of individual employees into the overall security profile of the workplace is also a big challenge. Besides, in modern organizations, employees are no longer needed to be geographically co-located. As a result, the use of online communication and collaboration tools (*e.g.*, email, IM, *Skype*, *Dropbox*, *LinkedIn*, *Lync*, *etc.*) in private and business environment becomes a norm [13]. This implies that the organizations are indeed capable of observing the behaviours of their employees in the cyber-world and thus become prepared for potential security threats originated from the Internet. Following such an idea, in this thesis, we developed a framework to evaluate the information security vulnerability of organizations by analyzing the online behaviours of their employees. To be specific, this framework takes the data about how employees manage their privacy on the Internet as an indicator of their information security awareness and their capability in handling security issues. Then, based on the generated personal security vulnerability profiles, it applies network analysis methods on interpersonal security influential relationships in the workplace to rate the overall security vulnerability of each organization.

This framework can be a good addition for assessing and comparing the information security strength among organizations. The vulnerability score calculated can also provide an organization with a better understanding of its security risks,

based on which they can strengthen their defence through both technological and human-centred approaches.

1.2 Related Work

In the information security community, there are two research directions closely related to organizational information security. A majority of the R&D activities are focused on the human factors in the workplace. The other direction emphasizes the Internet where a variety of cyber-attacks targeting organizations are hosted.

1.2.1 Human-centered Information Security

In recent years, there has been a growing focus on the human and social elements in organizational information security. Many different approaches are adopted in workplaces to enhance the security awareness of employees both in and out of the organizational boundaries. Some commonly-used approaches are like setting information security rules and policies, making cyber-hygiene as part of the organizational culture, training employees with good knowledge and good act of conduct in information security, or making technologies to be more friendly to organizational users [9, 18, 19, 23, 24].

Among these studies, one of the key purposes is to understand the human and social factors that affect awareness, attitudes, and motivation of employees in dealing with information security issues. For instance, it is alleged that threat appraisal, self-efficacy, response efficacy, sanctions and neutralization behaviours all contribute to good compliance with the information security policies in the workplace [17, 25]. *Safa et al.* found that employees tend to exchange information security knowledge for the purposes like earning a reputation, gaining promotion and curiosity satisfaction [10]; *Dang et al.* justified that a local social network of interpersonal information security influence exists in each organization. In such a social network, employees are influencing each other through activities like giving security advice, providing security troubleshoot, sharing organizational updates and building trust [1, 4]. Some research looked at the correlation between personality traits and good information security behaviours in the workplace. For example, factors like risk-taking, rationality, extroversion, education, age, and even gender are found to be significant predictors for good security behaviours [3, 12, 21, 22, 26, 27].

However, research in this direction is usually conducted through surveys, questionnaires, case studies or empirical studies with a small dataset. Just as some researchers acknowledged, their findings are highly related to the samples they chose, and thus could be subjective, biased, and even contradicting [21, 22, 27]. For this reason, it is unsafe to popularize these findings as general knowledge to other organizations, especially to those with utterly different backgrounds.

1.2.2 Organizations and Cybersecurity

In this direction, many research efforts are also focused on human factors but trying to gain insight into users' risky behaviours on the Internet that put their organizations at stake. As recorded, some commonly occurred risky behaviours include using weak passwords or same passwords for different Internet services, using social insurance number as username, sharing account information with others, leaving computers or other devices logged in, downloading software with unknown source from the Internet, not installing anti-virus software or firewall, browsing infected websites, improperly disclosing personal information online, and so on [6, 7, 9, 28]. As shown in some statistics, about 49% of the research participants occasionally engaged in risky behaviours and 28% did so frequently [29]; most password used online are less than seven characters, and the passwords with more than seven characters usually contains user information, a very familiar word or a proper noun, which are all easily guessable [6]; when encountering suspicious emails, 37% of the employees would open them and click the link inside, and 13% would open the attached file; when an email appears legitimate, these numbers will rise to 42% and 30%, respectively [5].

In comparison, some studies are directly related to the various types of cyber-attacks, such as user surveillance, identity theft, phishing, viruses, spyware, trojans, and keyloggers. One type of such attacks typically targeted at organizations are called *social engineering attacks* which exploit the security vulnerability of employees to obtain confidential information or access restricted services in organizations [8, 12, 13, 20, 30, 31]. As summarized by *Krombholz et al.*, typical social engineering attacks include phishing, dumpster diving, shoulder surfing, waterholing, advanced persistent threat, and baiting. They are categorized into five types: physical, social, reverse social, technical, and social-technical [13]. What's more, an investigation conducted by *Conteh* and *Schmick* found that the major motivation behind social engineering attacks are financial gain (23%), access to proprietary information (30%),

competitive advantage (21%), revenge (10%) and even just for fun (11%), whereas the people typically targeted are new employees (41%), IT professionals (17%), clients & customers (23%), partners & contractors (12%), and top-level managers (7%) [8].

Other related research is directly aiming at social engineering attacks and other types of cyber-attacks on the Internet. For example, *Omar et al.* presented a multi-layered graph model to assess the vulnerability related to user profile against social engineering attacks [20]. *Tannous* and *Barbar* provided a fuzzy logic-based expert system model for detecting privacy vulnerability in online social networks [32]. *Mouton et al.* derived an ontology model for describing social engineering attack scenarios in a standardized format [31]. But, there is still a long way to go before all these cyber-attacks against organizational information security are effectively contained. Part of the reason lies in human nature. Just as the experiments conducted by *Cain et al.* show, many bad hygiene behaviours are rooted deeply in our daily habits, and are not easy to remove [28].

1.3 Contributions

The main contributions of this thesis include:

- **A framework for assessing the information security vulnerability of organizations based on the analysis of online user behaviours**. In this framework, the behaviour of online privacy management is taken as an indicator to estimate the personal security vulnerabilities of employees. Then, all the values of personal vulnerability for employees in each organization are synthesized into an organizational score by incorporating them into a local social network of interpersonal security influence in the workplace.

- **A confusion matrix-based model representing the profile of personal security awareness and abilities**. In this model, the content of each confusion matrix tells how a person confuses sensitive information of given security level to that of other levels. With this model, an algorithm for deriving the confusion matrices is designed after we formalized a process that optimizes the classification of privacy items and calculates the confusion matrices.

- **A graph model representing a local social network of interpersonal security influence**. This model is based on the fact that a local social network is formed in the workplace according to the interpersonal security influence among

employees. Then, a PageRank-like algorithm is designed to aggregate the personal vulnerability measures into an organizational score based on this model.

- The **evaluation** results show that this framework works well and its algorithms are efficient in generating the optimized output while being effective in approximating the ground truth of latent variables and responding the change of input data.

- A **prototype** is presented to demonstrate how the framework can be applied in practice. In this demo application, the proposed framework is used to learn the causal relationship between some indicative factors and the security status of organizations. With the learning result, we can then approximate the security vulnerability of any new organizations given its relevant information.

1.4 Agenda

The rest of this thesis is organized as follows:

In Chapter 2, after a detailed analysis of the problem and solution, a framework based on online behavior analysis is proposed.

In Chapter 3, with the support of a confusion matrix-based model for personal security profile, an algorithm is designed to generate the confusion matrices for all the employees through iterative optimization.

In Chapter 4, on top of a graph model of interpersonal security influence network, a PageRank-like algorithm is designed to calculate the security vulnerability of an organization from the personal scores of all its employees.

In Chapter 5, evaluations with real-world data and simulation-generated data are performed to verify the feasibility, efficiency and effectiveness of the framework and the related algorithms.

In Chapter 6, a prototype is implemented to demonstrate how the framework can be applied in practice in the area of organizational information security.

Finally, Chapter 7 concludes the research of this thesis and presents some future work in this direction.

Chapter 2

The Problem, Solution and Framework

2.1 The Problem: Social Engineering Attacks

In the *Oxford English Dictionary*, there are two definitions of the term "social engineering". In the first definition, it means "the use of deception in order to induce a person to divulge private information or *esp.* unwittingly provide unauthorized access to a computer system or network". That is, social engineering refers to a type of cyber-attacks that include social means in their malicious efforts and target employees for the purpose of stealing confidential data and exploiting IT services inside the organizational boundary [15].

As shown in Figure 2.1, a typical social engineering attack follows a pattern consisting of four phases: information gathering, relationship development, execution, and exploitation [20, 31]. In this pattern, the first three phases are likely to occur outside the organization border, allowing the attacker to take advantage of multiple web-based tools like web page scrapers or forum/blog aggregators. In the last step, the IT systems in the workplace typically have lost the capability to defend themselves against what seems to be authorized access, and therefore are eventually compromised.

As we can see in this pattern, personal information abundantly scattered around the Internet is the weapon for the attackers. Many modern organizations allow employees to work with their own devices both in and out of the workplace. Consequently, they frequently communicate with various online tools where plenty of sen-

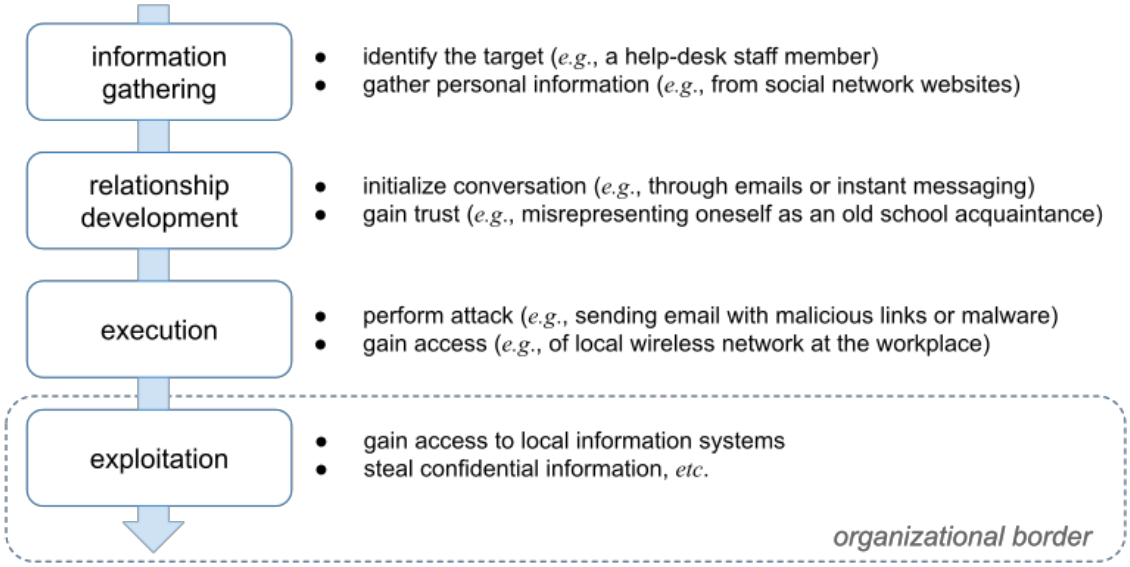


Figure 2.1: The pattern of social engineering attacks

sitive information is left. Also, for many people, it has already been a habit to work and socialize at the same time, either in the workplace or at home. As a result, the boundary between the business and the private life is blurred; the information about work and personal privacy are mixed and publicized online with very little thought of security or privacy. Once the information is gathered by some malicious parties, both the individuals and their organizations become highly vulnerable to social engineering attacks, as well as other types of security attacks.

It is not uncommon that even multinational corporations which we trust deeply and rely on a lot are falling victims to some sophisticated attacks from time to time, often with the leakage of myriads of sensitive customer information. For example, Google's internal system was compromised in 2009; the RSA security token system was broken in 2011; both Facebook and the New York Times are compromised in 2013; many PayPal customers received phishing emails and many gave the attackers private information, such as credit card number, in 2015 [13]. A very recent big attack happened in August 2019. Capital One revealed that the personal information of about 100 million bank accounts in the U.S. and about 6 million in Canada were stolen. The personal information leaked includes name, address, phone number, postal code, email address, birth date, self-reported income, *etc.* [33]. This would probably become the origin of many future attacks. In one word, security attacks,

especially social engineering attacks, are real and imminent.

2.2 A Social Engineering-Based Solution

The second meaning of social engineering in the dictionary is stated as "the use of centralized planning in an attempt to manage social change and regulate the future development and behaviour of a society". This definition grants "social engineering" a positive meaning. Although it is not directly related to information security, the definition inspires us that we can apply certain social engineering measures in the cyber-world to enhance the ability of organizations against security attacks. Following this idea, we can collect users' information from the Internet just as social engineering attacks do. But, instead of using the information to exploit the weakness of Internet users, we use it to analyze the security behaviours of employees and to understand their vulnerability against security attacks.

As shown in Figure 2.2, due to the wide usage of online tools in the workplace, the Internet can be viewed as an extension of the myriads of information systems deployed within the organizational boundaries. Almost as a daily routine, employees rely on such an extended system to work, socialize, participate in many other activities, and influence each other at the same time. For this reason, we can also think of each employee as an individual with dual identities, as an employee within the organizational border, and as an Internet user relying on many Internet services. Since each pair of the identities belong to the same person, it is reasonable to assume that they share the same personality traits, and thus should demonstrate similar security behaviours. Based on this understanding, we should be able to evaluate the security vulnerability of all the employees in the context of the Internet and then map the evaluation results to the context of the workplace to infer the security vulnerability of an organization.

On the Internet, the security status of a person can be assessed through the behaviours of privacy management Normally, the privacy data is a composition of many atomic items like name, email, hometown, phone number, friends, sexual orientation, IM screen name, business contact, and so on. Once the information is exposed online, it could potentially be harvested and used by attackers for malicious or criminal purpose. But, managing these privacy items properly is not trivial. It is nearly impossible to keep all of them secret because it's often necessary to expose some to trade for a better digital presence. Normally, sharing more privacy implies a bigger chance

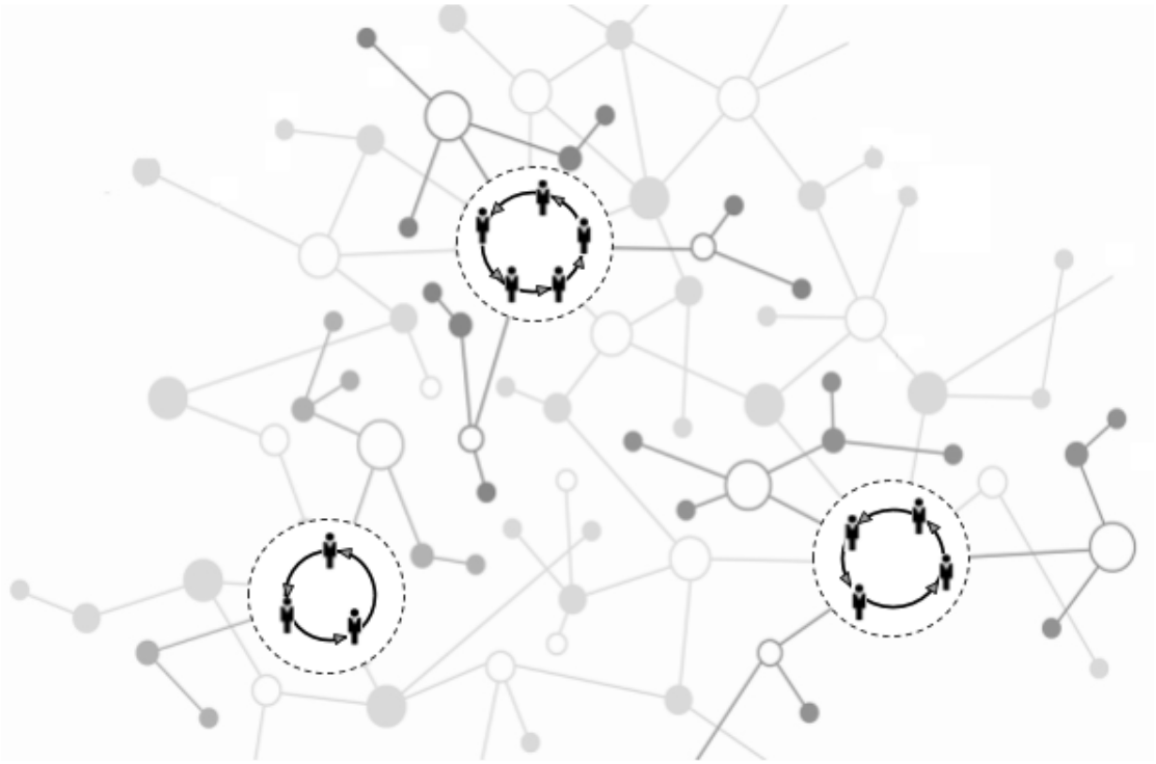


Figure 2.2: The Internet as an extension of workplaces

of getting popular, having more impact, and being easier to be located in the cyber-world [34, 35]. As a result, how people manage, either expose or keep secret, their privacy online is an ideal indicator of their information security status. By analyzing how properly the privacy items are managed, we should be able to infer the security vulnerability of a person in the context of the Internet.

Different from the traditional ways that put focus onto the workplace and use questionnaires, surveys and related empirical data to understand organizational information security, this new approach has several advantages. First, it is difficult to conduct a comprehensive survey on multiple organizations simultaneously due to the sheer difference among them. This is no longer an issue for our approach because it does not rely on the behavioural data from the workplace. Second, there is abundant data online and it is possible to collect the behavioural data for a large group of people on the Internet. This also avoids the issue for survey that many people are reluctant to answer the security-related questions. Third, our approach only investigates how each privacy item is configured and does not recording the privacy data. Therefore, it is information security-friendly for both organizations and their employees.

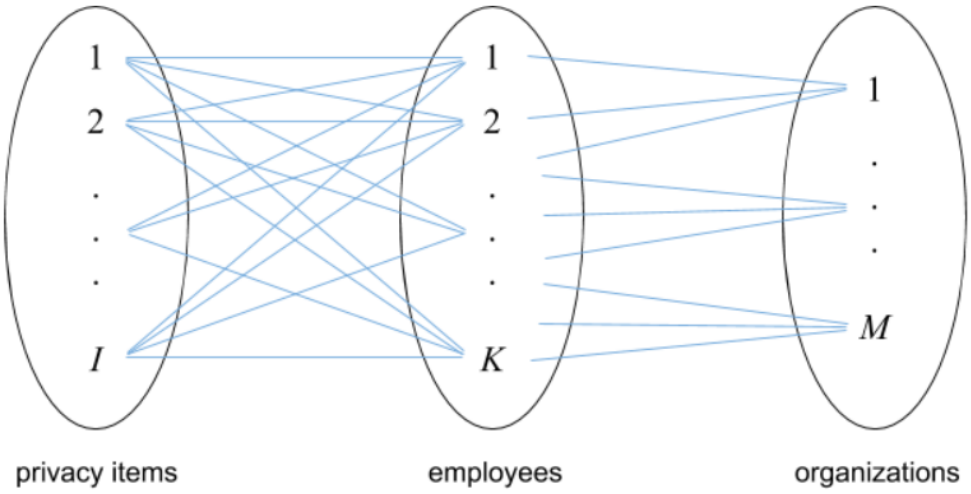


Figure 2.3: The privacy item-employee-organization model

2.3 The Framework

2.3.1 The Data Model

In the proposed framework, the behavioural data of personal privacy management is taken as an input to generate the security profile for employees, which are then synthesized into the organizational vulnerability scores by incorporating the factor of interpersonal influence in each organization.

The data model used in this framework is shown in Figure 2.3. It formalizes the data elements like privacy items, employees and organizations, as well as the mapping among them. In this model, we assume that there are in total K employees coming from M organizations. Throughout this thesis, unless otherwise specified, the variable k $(1 \leq k \leq K)$ only refers to an employee, the variable m $(1 \leq m \leq M)$ only for an organization. In accordance with the specific contexts, we will also use the terms "users" or "people" interchangeably to refer to employees, given that employees are also users of the various services on the Internet.

Similarly, suppose that there are up to I privacy items, which the Internet users need to decide whether to expose to gain the digital existence online. They can be the atomic privacy settings of a social media, a cloud service, or a mix of multiple online systems. We use numbers $1, 2, ..., I$ to identify these privacy items with the variable i $(1 \leq i \leq I)$ only used to specify a privacy item under consideration.

Using the above symbols, the user behaviours of online privacy management for all the employees can be formalized as a $K \times I$ matrix $\boldsymbol{R}$, called *response matrix*. For any k ($1 \leq k \leq K$) and i ($1 \leq i \leq I$), $\boldsymbol{R}(k,i) \in [0,1]$ stands for the probability that user k exposes her privacy related to item i on the Internet. As an example of calculating each $\boldsymbol{R}(k,i)$, suppose that we checked three online social network systems and found user k made his privacy of item i public in two of the systems, then we set $\boldsymbol{R}(k,i) = 2/3$. In addition, we denote the k-th row of $\boldsymbol{R}$ as $\boldsymbol{R}^{(k)}$, which represents the online privacy setting of user k, and the i-th column as $\boldsymbol{R}_i$, meaning all the user settings for privacy item i. Apparently, we have $\boldsymbol{R}^{(k)}(i) = \boldsymbol{R}_i(k) = \boldsymbol{R}(k,i)$.

Then, for each organization m ($1 \leq m \leq M$), a membership function $\mathcal{N}_m$ is used to identify all its employees. That is, for any employee k, $\mathcal{N}_m(k) = 1$ means that k works for m. Otherwise, we set $\mathcal{N}(m,k) = 0$. Also, a function $\mathcal{E}_m$ is used to record the interpersonal security influence in m. For any pair of employees k_1 and k_2, we have $\mathcal{E}_m(k_1,k_2) \in [0,1]^n$ ($n \in \mathbb{N}$) if both $\mathcal{N}_m(k_1) = 1$ and $\mathcal{N}_m(k_2) = 1$, otherwise let $\mathcal{E}_m(k_1,k_2) = \varnothing$. That is to say, each $\mathcal{E}_m(k_1,k_2)$ is a set where every element represents a type of interpersonal influence from k_1 to k_2 of a strength in the range $[0,1]$.

By this means, we get a directed graph $\mathcal{G}_m(\mathcal{N}_m, \mathcal{E}_m)$ to represent the local social network within organization m. Here, $\mathcal{N}_m$ and $\mathcal{E}_m$ stand for the set of nodes and the set of edges, respectively. Particularly, for any two employees k_1 and k_2 in m (that is, both $\mathcal{N}_m(k_1) = 1$ and $\mathcal{N}_m(k_2) = 1$), $\mathcal{E}_m(k_1,k_2) = \varnothing$ means that there is no directed edge from k_1 to k_2; otherwise, the values in set $\mathcal{E}_m(k_1,k_2)$ are treated as the weights of the corresponding directed edges in $\mathcal{G}_m$.

2.3.2 The Calculation Procedure

With the support of the data model above, the procedure for calculating the organizational security vulnerability from online human behaviours data is shown in Figure 2.4. This procedure contains three steps: (1) **human behaviours analysis (called $\mathscr{F}$-step)**. This step takes the response matrix $\boldsymbol{R}$ as input and generates personal vulnerability values for all employees through the means of human behaviours analysis. In Figure 2.4, we formalize the output of this step as a function v which maps each employee k into a vulnerability value $v(k) \in [0,1]$; (2) **network analysis (called $\mathscr{G}$-step)**. This step is performed for every single organization. By taking the output of $\mathscr{F}$-step, as well as the graph model $\mathcal{G}_m$, for all its employees as inputs, a vulnerability value $\mathcal{V}(m)$ is generated for organization m. Here, $\mathcal{V}$ is also a mapping

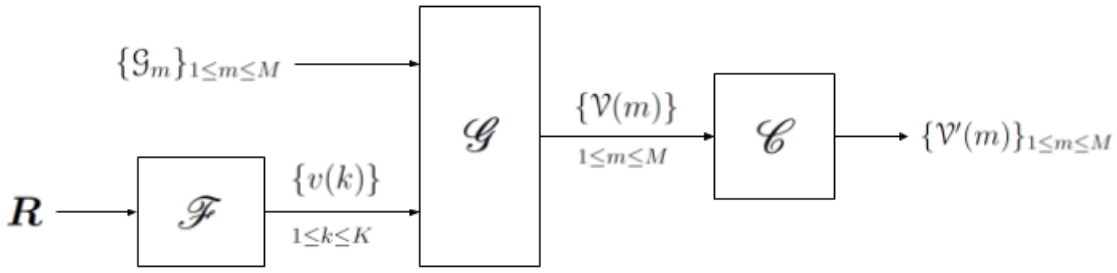


Figure 2.4: The 3-step procedure for calculating organizational security vulnerability

function, from the set of all organizations to the range $[0, 1]$; (3) **calibration (called $\mathscr{C}$-step)**. This step is to create a new mapping function $\mathcal{V}'$ from $\mathcal{V}$, providing the final organizational vulnerability ratings. That is to say, for any organization m, the final vulnerability score generated by the framework is $\mathcal{V}'(m) \in [0, 1]$.

In this process, both $\mathscr{F}$-step and $\mathscr{G}$-step contain some complicated calculations. Their details will be elaborated in Chapter 3 and Chapter 4. In comparison, $\mathscr{C}$-step is much simpler. Its purpose is to endorse a practical meaning to the final vulnerability values produced.

In practice, it is possible that the outputs of $\mathscr{G}$-step are not distributed uniformly in range $[0, 1]$. In this situation, having $\mathcal{V}(m) = 0.2$ for any organization m maybe does not mean that this organization has a good security status, having $\mathcal{V}(m) = 0.8$ also does not necessarily imply a poor security situation. Nevertheless, among the set of the $\mathscr{G}$-step outputs, one organization of a considerably bigger output than the other implies that this organization is more vulnerable in information security than the other. For this reason, we need $\mathscr{C}$-step as part of the framework to calibrate the $\mathscr{G}$-step outputs into some meaningful readings. Specifically, in $\mathscr{C}$-step, we first order the $\mathscr{G}$-step outputs in the decreasing order and then use percentile of each organization as its final vulnerability score. That is, for any organization m,

$$\mathcal{V}'(m) = \frac{|\{m' : 1 \le m' \le M, \mathcal{V}(m) \ge \mathcal{V}(m')\}|}{M} \times 100\% \tag{2.1}$$

Here, the operator $|\cdot|$ is for getting the size of a set.

For example, suppose $M = 5$ and the list of vulnerability values generated from $\mathscr{G}$-step are $\{0.2, 0.5, 0.1, 0.9, 0.05\}$, then the list of the final vulnerability readings after $\mathscr{C}$-step would be $\{0.6, 0.8, 0.4, 1.0, 0.2\}$ instead. Apparently, the percentile value for each organization can be seen as an indicator of its organizational security status compared

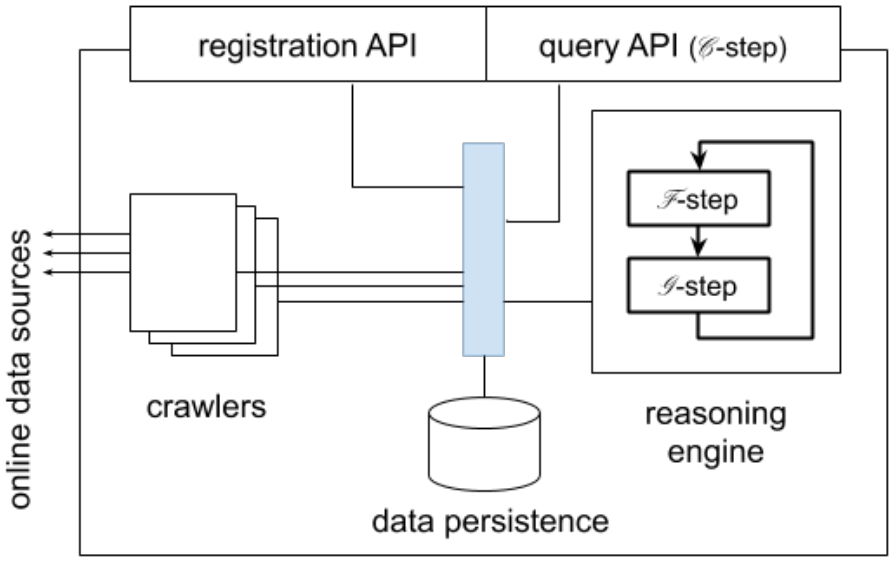


Figure 2.5: A system of providing services for organizational information security monitoring and rating

to the other organizations under consideration. In the example, the reading of the 5th organization is interpreted as that it is ranked as the top 20% organizations in terms of security condition, whereas the reading of the 4th suggests that the organization should be careful regarding its security status.

2.4 How to Apply the Framework in Real-world

Our framework can be used as a basic building block for online services that constantly monitors and rates the information security status of organizations registered. The key components for such a service is shown in Figure 2.5. Multiple web crawlers run constantly to gather and update information of employees from the Internet. An algorithm runs $\mathscr{F}$-step and $\mathscr{G}$-step periodically to take in the newly crawled data and update its knowledge base. Also, the query API is an implementation of $\mathscr{C}$-step, providing security ranks for all organizations registered for the services.

One critical requirement for applying this framework in practice is to feed it with abundant data. This requirement is somewhat similar to social engineering attacks that seek user data through illegal ways, such as cheating, setting traps or security attacks. Certainly, this should not be a choice for our framework. One optional way is

to harvest open user data by scraping web pages from the big online social media and cloud platforms. But such an approach is also controversial and is facing tremendous resistance from the service vendors nowadays. For example, last year *LinkedIn* raged a lawsuit against a company and claimed that the activity of data scraping from this company violates its user privacy and is bound to fraud and information abuse [36]. As a result, so far the safest way is to make use of the RESTful APIs provided by almost all the online platforms after getting the assent from online users who are also employees of organizations registering for above security services. Because this framework only needs some statistical data and does not collect the actual user information, getting the grant from employees should not be a big issue. In this thesis, since we only focus on the theoretical and technological aspects of the framework, we won't dive into the detail about how the data will be gathered from the Internet.

Chapter 3

Confusion Matrix-based People Vulnerability Analysis

3.1 Confusion Matrix-based Awareness Model

Confusion matrix is a concept often used in machine learning for analyzing the performance of learning algorithms, especially the ones for classification problems. Its name stems from the fact that it shows how an algorithm is confused with two classes by frequently mislabeling one as another. In such a matrix, one dimension stands for the instances in a predicted class while the other represents the instances in the actual class. This concept is also applied in many other areas besides machine learning, such as modelling a user's ability to distinguish true labels of given items in crowdsourcing [37] or modelling the run-time inter-dependency among components in cyber-physical systems [38]. Our framework follows a similar idea. It employs confusion matrices to profile the security awareness in managing personal privacy of different sensitivity levels for all the employees under consideration.

Before constructing the model of confusion matrix, here we first introduce the sensitivity levels of all the privacy items (denoted as $1, 2, .., I$) defined in Section 2.3. In cyber-security, the *sensitivity* of a piece of information is defined as the level of security risks if the information is exposed on the Internet. Similar to the way we denote the other data elements in the "privacy item-employee-organization" model shown in Figure 2.3, we also represent all the sensitivity levels as a list of numbers $1, 2, ..., C$. Here, sensitivity levels are ordinal. That is, the number 1 represents the lowest sensitivity level, whereas C represents the highest.

According to the above definition, the sensitivity of a privacy item determines how probable it will be publicized by its owner. Higher the sensitivity level, smaller the probability of exposure will be. So, we can design the coverage of all the sensitivity levels by choosing a proper exposing probability for each of them. We can generate these probability values in different ways. The naive way is to choose C values in range $[0, 1]$ in descending order and with the same interval. For example, if $C = 5$, the values chosen can be $\{0.9, 0.7, 0.5, 0.3, 0.1\}$. This means that a person would be in about 90% probability to expose a privacy item of the lowest sensitivity level, and in about 10% possibility to give up a privacy item of the highest sensitivity.

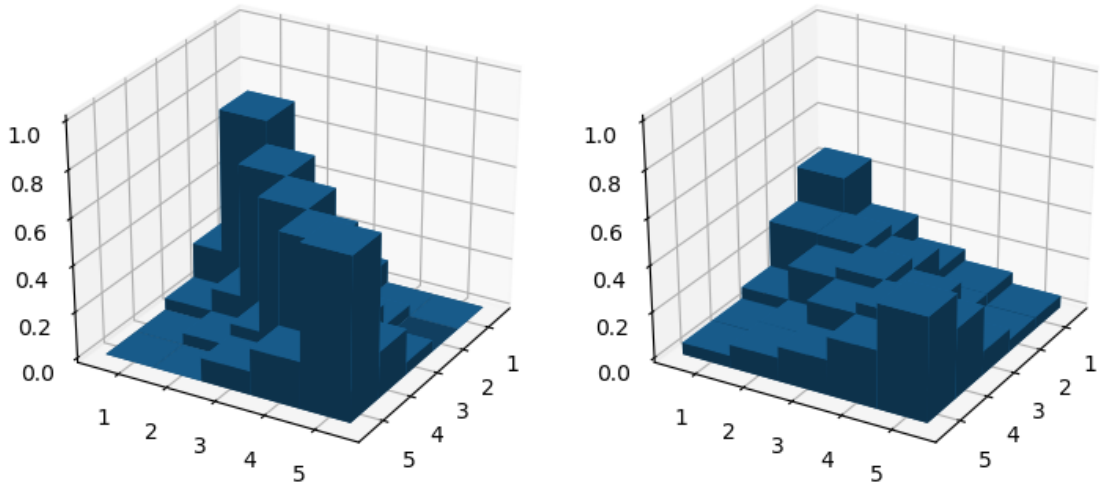
In this thesis, we adopt the idea of the Item Response Theory (IRT) model [39] and chose a more sophisticated way to generate these probability values. In specific, for any sensibility level c ($1 \leq c \leq C$), we say the probability that a person exposes a privacy item of this level is determined by the following *Sigmoid* function

$$P_c = \frac{1}{1 + \exp\left(\alpha(c - C/2)\right)} \tag{3.1}$$

where α is its controlling coefficient. For example, if choosing $C = 5$ and $\alpha = 1.0$, the probability values we get through this equation will be $\{0.88, 0.73, 0.50, 0.27, 0.12\}$. This result is very similar to the aforementioned set of values generated through the naive way.

The reason for introducing Equation (3.1) is as follows. In the IRT model, which is often used to parse data from questionnaires and tests, a similar Sigmoid function is employed to combine the difficulty of the questions and the ability of examinees together to tell how probable an examinee will correctly answer a question. In paper [39], this model is applied in privacy theory to form the probability distribution of exposing the privacy of given sensitivity by a person. The experiment in this paper showed that such a distribution models real-world dataset pretty well. Here, the Sigmoid function in Equation 3.1 follows a similar idea in generating the possibility of exposure for privacy of each sensitivity level. Specifically, similar to that in IRT model and in [39], within this Sigmoid function, c stands for the privacy sensitivity and we choose $C/2$ as the average level of privacy awareness among people.

Applying the above sensitivity levels, we define the confusion matrix of a person k as a $C \times C$ matrix and denote it as $\boldsymbol{\pi}^{(k)}$. In this matrix, the element of the t-th row and the c-th column, denoted as $\boldsymbol{\pi}^{(k)}_{t,c}$, represents the probability that k tends to confuse a privacy item of sensitivity level t (the true value) with those of sensitivity



(a) High accuracy (b) Low accuracy

Figure 3.1: Example Confusion matrices for 5-level privacy sensitivity

level c. So, all the elements of $\boldsymbol{\pi}^{(k)}$ in a whole form a profile telling how well a person handles privacy on the Internet. For instance, Figure 3.1 shows two distinct confusion matrices given that $C = 5$. Clearly, the user represented by Figure 3.1a has a better information security awareness than the person related to Figure 3.1b. The reason is that, the former's confusion matrix has higher diagonal values and thus smaller probabilities to confuse each true value with other sensitivity levels.

Also, for any true value t $(1 \leq t \leq C)$, denote the t-th row of $\boldsymbol{\pi}^{(k)}$ as $\boldsymbol{\pi}_t^{(k)}$. Then, we have

$$\sum_{c=1}^{C} \boldsymbol{\pi}_{t,c}^{(k)} = 1 \tag{3.2}$$

That is to say, each row $\boldsymbol{\pi}_t^{(k)}$ determines a *Categorical* distribution explaining how user k confuses any private information of true value t with all the other sensitivity levels (including t). So, if $\boldsymbol{\pi}^{(k)}$ is known, we can then derive the probability that k exposes her privacy (or, any privacy items defined in Section 2.3) of true sensitivity level t as

$$P_t^{(k)} = \sum_{c=1}^{C} \boldsymbol{\pi}_{t,c}^{(k)} P_c \tag{3.3}$$

Here, P_c acts as the prior probability that a privacy item of sensitivity level c $(1 \leq$

$c \leq C$) will be exposed, whereas $\boldsymbol{\pi}_{t,c}^{(k)}$ can be seen as the conditional probability given that t is known.

Besides, we can as well calculate the security vulnerability of any employee k from the corresponding confusion matrix $\boldsymbol{\pi}^{(k)}$. Specifically, suppose the coverage of any true value t among all the privacy items under consideration is ζ_t (let $\zeta_t = 1/C$ when no prior knowledge is known), the corresponding vulnerability score is calculated as

$$v(k) = \sum_{t=1}^{C} \frac{\sum_{c=1}^{C} D(t,c)\boldsymbol{\pi}_{t,c}^{(k)}}{\sum_{c=1}^{C} D(t,c)} \zeta_t \tag{3.4}$$

where $D(t,c)$ is the distance between the two sensitivity levels t and c. The simplest way is to calculate it as the absolute difference $D(t,c) = |t-c|$. Here, both ζ_t and $D(t,c)$ are used as weights. Firstly, we use $D(t,c)$ to get the intermediate value for each true value t from $\boldsymbol{\pi}^{(k)}$. Then, ζ_t is used to synthesize the intermediate values into one final value.

3.2 The calculation Process ($\mathscr{F}$-step)

Recall the $\mathscr{F}$-step in Section 2.3 which takes a response matrix $\boldsymbol{R}$ as input and outputs the personal security vulnerability measures $\{v(k)\}_{1\leq k\leq K}$. Equation (3.4) is a key part of this step. To perform the calculation, however, we need a way to: (1) reveal the values of the latent variables $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$; (2) find out the sensitivity levels of all the privacy items under consideration. These two problems are interdependent, each takes the result of the other as an input. By assuming the initial confusion matrices for all the employees, our algorithm iterates over two separate steps, *i.e.*, classifying privacy items and updating confusion matrices, until convergence or a maximum number of iterations is reached.

3.2.1 Classifying Privacy Items

In $\mathscr{F}$-step, the classification of all the privacy items into sensitivity levels is directly related to the response matrix $\boldsymbol{R}$. As a prerequisite, we assume that the confusion matrices for all the employees, denoted as $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$, are known.

Then, for any privacy item i ($1 \leq i \leq I$), the column $\boldsymbol{R}_i$ from the response matrix reveals the attitudes of all the employees towards it. With $\boldsymbol{R}_i$, we can apply the *Maximum Likelihood Estimation* (MLE) method to find out the actual sensitivity

level (or true value) of i. This can be done by maximizing the following likelihood function

$$\mathcal{L}(t) = \prod_{k=1}^{K} \left(P_t^{(k)}\right)^{\boldsymbol{R}(k,i)} \left(1 - P_t^{(k)}\right)^{1-\boldsymbol{R}(k,i)} \tag{3.5}$$

for every privacy item i across all the sensitivity levels. Here, each $P_t^{(k)}$ is determined by the confusion matrix $\boldsymbol{\pi}^{(k)}$ through Equation (3.3). This likelihood function follows the fact that in general people tend to perceive the true sensitivity level t of any privacy item i correctly. So, their decision in whether to expose i in public should maximize such a likelihood function.

To simplify the calculation, we first perform a logarithmic transformation on Equation (3.5) as follows

$$\begin{aligned} \ln(\mathcal{L}(t)) &= \ln\left[\left(P_t^{(k)}\right)^{\boldsymbol{R}(k,i)} \left(1 - P_t^{(k)}\right)^{1-\boldsymbol{R}(k,i)}\right] \\ &= \sum_{k=1}^{K} \left[\boldsymbol{R}(k,i) \ln P_t^{(k)} + \left(1 - \boldsymbol{R}(k,i)\right) \ln(1 - P_t^{(k)})\right] \\ &= \sum_{k=1}^{K} A_t^{(k)} \boldsymbol{R}(k,i) + \sum_{k=1}^{K} B_t^{(k)} \end{aligned} \tag{3.6}$$

where $B_t^{(k)} = \ln(1 - P_t^{(k)})$ and $A_t^{(k)} = \ln P_t^{(k)} - B_t^{(k)}$.

Here, since $B_t^{(k)}$ is a constant value, finding out the true value of t is the same as maximizing a new likelihood function

$$\begin{aligned} \mathcal{L}'(t) &= \sum_{k=1}^{K} A_t^{(k)} \boldsymbol{R}(k,i) \\ &= \sum_{k=1}^{K} \left[\ln P_t^{(k)} - \ln(1 - P_t^{(k)})\right] \boldsymbol{R}(k,i) \end{aligned} \tag{3.7}$$

where each $P_t^{(k)}$ is determined by Equation (3.3).

3.2.2 Updating the Confusion Matrices

Knowing the true sensitivity levels of all the privacy items is equivalent to knowing that how people will handle their privacy in general. For each person k ($1 \leq k \leq K$),

we know that the row $\boldsymbol{R}^{(k)}$ in the response matrix is the privacy setting of this person and reflects her security awareness. For estimating the confusion matrix $\boldsymbol{\pi}^{(k)}$, here we first derive a C-dimension feature vector $\boldsymbol{\lambda}^{(k)}$ from $\boldsymbol{R}^{(k)}$. It reveals the overall attitude of k towards privacy of different sensitivity levels. Assume that among all the privacy items, there are I_t items belonging to sensitivity level t and user k sets $I_t^{(k)}$ ones as public. Then, we set that $\boldsymbol{\lambda}^{(k)}(t) = I_t^{(k)}/I_t$. Apparently, each $\boldsymbol{\lambda}^{(k)}(t)$ is an estimation of $P_t^{(k)}$ in Equation (3.3) which, in turn, is closely related to the row $\boldsymbol{\pi}_t^{(k)}$ in the confusion matrix of user k.

What's more, people normally tend to perceive the sensitivity of privacy correctly. For instance, in general "age" is more sensitive than "height" and the latter is more sensitive than the city where a person lives. Thus, it is less likely for a person to expose "age" online comparing to "height", not mentioning the hometown. Even a person perceived the sensitivity level of "age" mistakenly, it is reasonable that she is more likely to confuse it to the level of "height" than to that of the hometown. That is to say, for any person k and any sensitivity level t, $\boldsymbol{\pi}_{t,c}^{(k)}$ tends to be bigger if c is closer to t.

Combining these factors together, we design the following error function to evaluate the goodness of a confusion matrix row $\boldsymbol{\pi}_t^{(k)}$

$$\xi(\boldsymbol{\pi}_t^{(k)}) = \ln\Big(\sum_{c=1}^{C} \boldsymbol{\pi}_{t,c}^{(k)} P_c - \boldsymbol{\lambda}^{(k)}(t)\Big)^2 + \varphi \ln\Big(\sum_{c=1}^{C} |t-c| \boldsymbol{\pi}_{t,c}^{(k)}\Big) \tag{3.8}$$

This error function has two parts. The former compares $\boldsymbol{\pi}_t^{(k)}$ to the feature vector entry $\boldsymbol{\lambda}^{(k)}(t)$, the latter considers how bad k confuses t with other sensitivity levels. Since the latter generates a much bigger value than the former, here we use a coefficient φ ($0 < \varphi \ll 1$) to leverage its result to be comparable in the equation. Thereby, the problem of finding a good $\boldsymbol{\pi}_t^{(k)}$ is now transformed to minimizing $\xi(\boldsymbol{\pi}_t^{(k)})$ given that all elements in $\boldsymbol{\pi}_t^{(k)}$ sum to 1, just as shown in Equation (3.2).

Here, we use Gradient Descent Algorithm (GDA) for the calculation. Firstly, by incorporating the constraint in Equation (3.2) into $\xi(\boldsymbol{\pi}_t^{(k)})$, we get

$$\begin{aligned} \xi(\boldsymbol{\pi}_t^{(k)}) = &\ln\Big[\sum_{c=1}^{C-1} \boldsymbol{\pi}_{t,c}^{(k)} P_c + \Big(1 - \sum_{c=1}^{C-1} \boldsymbol{\pi}_{t,c}^{(k)}\Big) P_C - \boldsymbol{\lambda}^{(k)}(t)\Big]^2 + \\ &\varphi \ln\Big[\sum_{c=1}^{C-1} |t-c| \boldsymbol{\pi}_{t,c}^{(k)} + (C-t)\Big(1 - \sum_{c=1}^{C-1} \boldsymbol{\pi}_{t,c}^{(k)}\Big)\Big] \end{aligned} \tag{3.9}$$

Then, for all the $\boldsymbol{\pi}_{t,c}^{(k)}$ with $1 \leq c < C$, the partial derivative is

$$\frac{\partial \xi(\boldsymbol{\pi}_t^{(k)})}{\partial \boldsymbol{\pi}_{t,c}^{(k)}} = \frac{2(P_c - P_C)}{\sum_{c=1}^{C} \boldsymbol{\pi}_{t,c}^{(k)} P_c - \boldsymbol{\lambda}^{(k)}(t)} + \frac{\varphi(|t-c| + t - C)}{\sum_{c=1}^{C} |t-c| \boldsymbol{\pi}_{t,c}^{(k)}} \tag{3.10}$$

So, we get the gradient of this error function as

$$\nabla_\xi(\boldsymbol{\pi}_t^{(k)}) = \left[\frac{\partial \xi(\boldsymbol{\pi}_t^{(k)})}{\partial \boldsymbol{\pi}_{t,1}^{(k)}}, \frac{\partial \xi(\boldsymbol{\pi}_t^{(k)})}{\partial \boldsymbol{\pi}_{t,2}^{(k)}}, ..., \frac{\partial \xi(\boldsymbol{\pi}_t^{(k)})}{\partial \boldsymbol{\pi}_{t,C-1}^{(k)}}, 1 - \sum_{c=1}^{C-1} \frac{\partial \xi(\boldsymbol{\pi}_t^{(k)})}{\partial \boldsymbol{\pi}_{t,c}^{(k)}} \right] \tag{3.11}$$

With this gradient, we can then update the confusion matrix row $\boldsymbol{\pi}_t^{(k)}$ according to the following gradient descent rule

$$\boldsymbol{\pi}_t^{(k)} \leftarrow \boldsymbol{\pi}_t^{(k)} - \kappa \nabla_\xi(\boldsymbol{\pi}_t^{(k)}) \tag{3.12}$$

where the gradient value $\nabla_\xi(\boldsymbol{\pi}_t^{(k)})$ denotes the direction of the steepest slope on the $(C-1)$-dimension hyper surface of $\xi(\boldsymbol{\pi}_t^{(k)})$ for the current value of $\boldsymbol{\pi}_t^{(k)}$; κ is the coefficient of step size controlling how fast the value of $\boldsymbol{\pi}_t^{(k)}$ is evolved.

Lastly, it's worth noting that, after getting the privacy items classification in each iteration, we only perform the updating rule in Equation (3.12) once for every confusion matrix row. This is different from the standard GDA algorithms. One reason is that the privacy items classification in each iteration is only a temporary result, it's unnecessary to find the confusion matrix row best-matching the current feature vector which is bound to change in the next round of computation. For the other reason, along the iterations for the overall calculation in $\mathscr{F}$-step, this updating rule has already been performed repeatedly.

3.3 The Algorithm

By assembling the aforementioned two processes, we design the algorithm for calculating the confusion matrices for all the employees under consideration as shown in Algorithm 1. It approximates the best confusion matrices through repeated optimization and will stop only when it reaches the maximal allowed number of iterations or the error (the difference between two adjacent iterations) becomes small enough. Within each iteration, these processes are encapsulated into two separate functions, "`privacy_item_classification()`" for privacy items classification and "`confusion_matrix_generation()`" for confusion matrices derivation.

Algorithm 1: Confuse matrix calculation algorithm (CMCA)

Input: $\boldsymbol{R}$, K, I, C

Output: $\boldsymbol{\pi}$

initialize each $\boldsymbol{\pi}^{(k)}$ as a random $C \times C$ matrix
categories $\leftarrow \{0\}_{I\times 1}$
for $j \leftarrow 1$ **to** max_iter **do**
    $\delta \leftarrow 0$
    for $i \leftarrow 1$ **to** I **do**
        categories $[i] \leftarrow$ `privacy_item_classification` $(\boldsymbol{R}_i, C)$
    end
    for $k \leftarrow 1$ **to** K **do**
        $\boldsymbol{\pi}'^{(k)} \leftarrow$ `confusion_matrix_generation` $(\boldsymbol{R}^{(k)}, C)$
        $\delta \leftarrow \delta +$ `diff` $(\boldsymbol{\pi}^{(k)}, \boldsymbol{\pi}'^{(k)})$
        $\boldsymbol{\pi}^{(k)} \leftarrow \boldsymbol{\pi}'^{(k)}$
    end
    if $\delta \leq$ min_diff **then break**;
end
$\boldsymbol{\pi} \leftarrow \{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$
return $\boldsymbol{\pi}$

For this algorithm, let's denote the maximum number of iterations as N_1. Suppose only K, the number of employees, is the scaling factor, since both functions, "`privacy_item_classification()`" and "`confusion_matrix_generation()`", iterate on each employee only once, we grade the worst-case time complexity of this algorithm as $\mathcal{O}(KN_1)$. Also, because the size of data storage for each employee is constant, we conclude that the space complexity of this algorithm is $\mathcal{O}(K)$.

Chapter 4

Graph-based Organizational Vulnerability Calculation

4.1 Interpersonal Security Influence Network

Employees are the center of an organization. Different in backgrounds and personal traits, they bond together in the workplace to form the atmosphere in the workplace which we call as organizational climate. In definition, organizational climate is the recurring patterns of behaviours, attitudes and feelings of employees that characterize life in the organization [40]. It is regarded as an accumulative result of the behavioural aspects of every individual employee that have a psychological impact on the workplace environment, such as job-satisfactory, moods, leadership, team cooperation, and so on.

As a key aspect of this climate, organizational information security is also hugely affected by the human factors in the workplace. Employees constantly share with peers their knowledge in information security, their awareness of security risks, their security attitudes and habits, as well as their moods when facing the losses after major security breaches. Especially, through the various types of interpersonal security influence, employees are even forming a local social network in the workplace [1, 4]. For instance, Figure 4.1 shows such a local social network in a large company in southeast Asia. It hosts a workforce of more than 300 employees at three offices and about 1,000 workers at two factories in multiple locations [1]. As shown in Table 4.1, there are several types of interpersonal security influence in the workplace that contribute significantly to the organizational climate of information security.

Table 4.1: Types of interpersonal security influence in the workplace [1]

Name	Description
Work advice	Employees who give work advice tend to influence the other's security behaviours as well.
Security advice	Employees who are sought for security advice tend to influence the other's security behaviours as well.
Security troubleshooting	Employees who are sought for security troubleshooting tend to influence the other's security behaviours as well.
Organizational updates	Employees who are sought for organizational updates tend to influence the other's security behaviours as well.
Trust	Employees who are trusted tend to influence the other's security behaviours as well.
Same department	Employees who work in the same department tend to influence each other's security behaviours as well.
Seniority	Employees who have higher seniority have a higher chance to influence security behaviours.

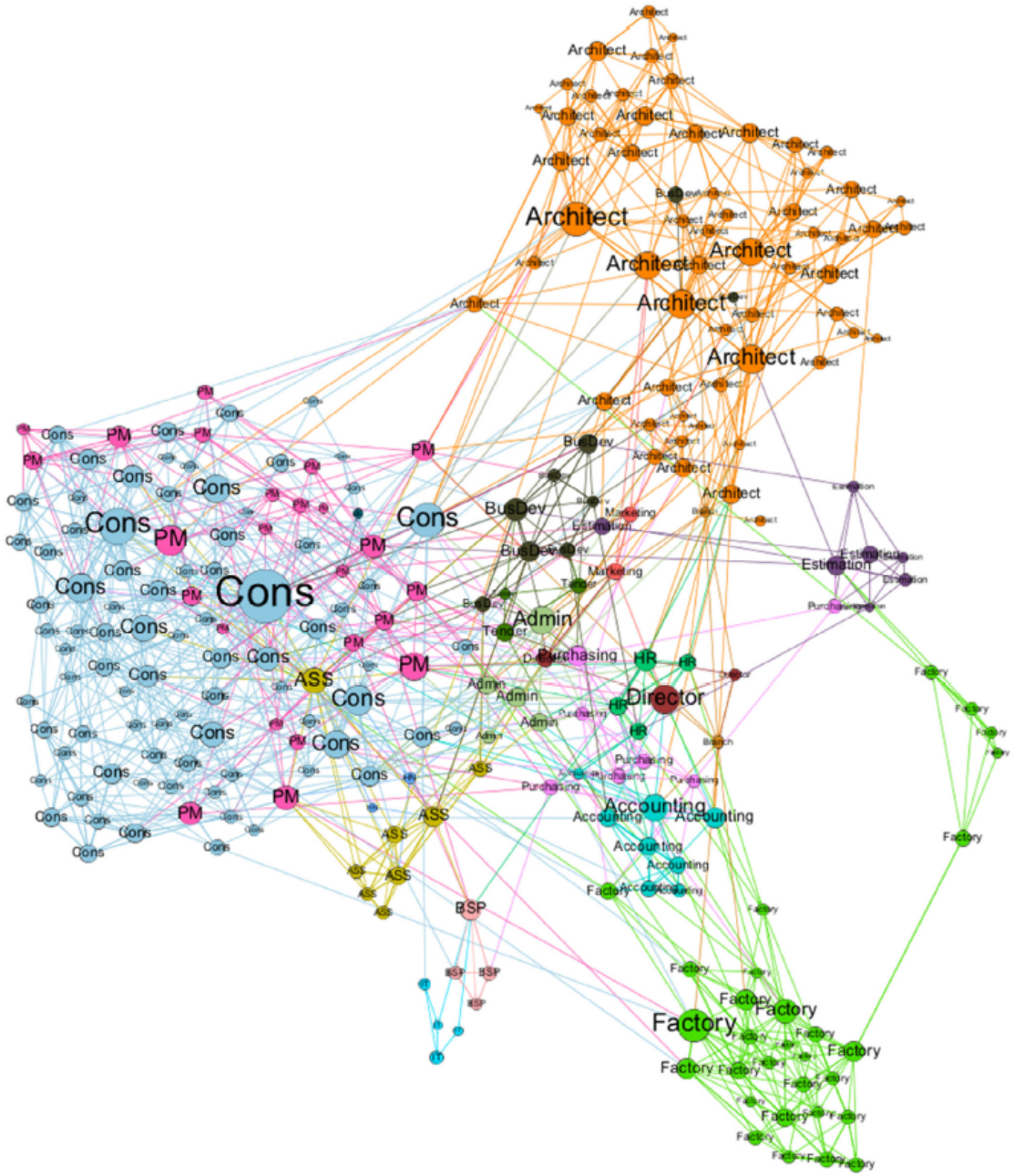


Figure 4.1: An example of local social networks formed by interpersonal security influence in the workplace [1]

In conclusion, the security behaviours of employees are capable of propagating through the local network in the workplace. For this reason, by observing the employees and tracing the paths of interpersonal influence, it is possible to estimate

how the organizational security situation evolves, as well as to assess how vulnerable the workplace security environment is, according to the human factors forming the organizational security climate. As mentioned in [4], questionnaire is a good way to uncover the interpersonal influence network for an organization. Specifically, we can ask every employee in the organization to name her colleagues from whom she has perceived each type of the security influence listed in Table 4.1. From the data collected, we can not only find out the structure of the network, but also uncover the strength of each type of the interpersonal influence for the organization.

4.2 The Network-based Calculation ($\mathscr{G}$-step)

In the framework proposed in Section 2.3, the local social network of any organization m ($1 \leq m \leq M$) is modeled as a directed graph model $\mathcal{G}_m(\mathcal{N}_m, \mathcal{E}_m)$. In the $\mathscr{G}$-step of the framework, the graph model $\mathcal{G}_m$, together with the security vulnerability scores of all the employees working in m (denoted as $v(k)$ for any employee k), is taken as the input to calculate the overall organizational security vulnerability $\mathcal{V}(m)$.

For the calculation, an idea similar to the famous *PageRank* algorithm [41] is adopted to process the interpersonal security influence among employees in the workplace. This algorithm is originally designed for the *Google* search engine to rank web documents among their search results. It assigns a numerical weight to each element of a hyperlinked set of documents and measures the relative importance of all the documents within the set. In practice, it can also be applied to any collection of entities with reciprocal quotations and references. This is exactly the case for the local social network of interpersonal security influence within each organization.

Here, let's only focus on one organization m and suppose it has K_m employees. To better explain the idea, we re-enumerate these employees as $1, 2, ..., K_m$. For any employee $k (1 \leq k \leq K_m)$, we've already acquired the personal vulnerability value $v(k)$ from the $\mathscr{F}$-step. Also, we can derive a $K_m \times K_m$ adjacency matrix $\boldsymbol{M}$ from graph $\mathcal{G}_m$. For any two employees k_1 and k_2 ($1 \leq k_1, k_2 \leq K_m$ and $k_1 \neq k_2$), we set $\boldsymbol{M}(k_1, k_2) = \max\{\mathcal{E}(k_1, k_2)\}$ as the influential factor from k_1 to k_2. That is to say, if there are several types of security influence from one employee to the other, only the strongest influence will be considered. Particularly, for any $1 \leq k \leq K_m$, let $\boldsymbol{M}(k, k) = 1$.

Then, similar to *PageRank*, we first calculate the importance of each employee based on the local social network of interpersonal security influence in organization

m. Initially, set the importance of each employee k ($1 \leq k \leq K_m$) as

$$\mathcal{I}(k) = 1 \tag{4.1}$$

In each iteration, we update the importance values of all the employees through the follow rule

$$\mathcal{I}(k) \leftarrow (1-d) + d \sum_{1 \leq k' \leq K_m}^{k' \neq k} \frac{\mathcal{I}(k')\boldsymbol{M}(k',k)}{\sum_{k''=1}^{K_m} \boldsymbol{M}(k',k'')} \tag{4.2}$$

This rule is a resemble of that in *PageRank*. There are two obvious similarities. On one hand, this equation also uses a damping factor d to prevent the importance values of some graph nodes from sinking to 0. On the other hand, it uses the ratio of influence as the weight to leverage the involvement of each current importance value in the calculation. In *PageRank*, the number of outbound edges is used instead.

When getting the importance values, we then calculate the overall security vulnerability of organization m to be the weighted average of the vulnerability scores of all its employees according to the resulted importance values. That is,

$$\mathcal{V}(m) = \frac{\sum_{k=1}^{K_m} \mathcal{I}(k)v(k)}{\sum_{k=1}^{K_m} \mathcal{I}(k)} \tag{4.3}$$

As an illustration of this process, Figure 4.2 shows an example network of a hypothetical small start-up company. This company has only 5 members: the CEO is in charge of the organizational affairs, the technical guy is responsible for the security advice and troubleshooting for all other colleagues, and everyone is trustworthy to the others.

Suppose the personal vulnerability scores of each employee after the $\mathscr{F}$-step calculation are 0.5, 0.2, 0.3, 0.6 and 0.8, respectively. The strength of security influence for trustworthy, seniority, and security advice is 0.1, 0.3 and 0.6, respectively. Then, the adjacency matrix derived from this local network is

$$\boldsymbol{M} = \begin{bmatrix} 1.0 & \mathbf{0.3} & \mathbf{0.3} & \mathbf{0.3} & \mathbf{0.3} \\ \mathbf{0.6} & 1.0 & \mathbf{0.6} & \mathbf{0.6} & \mathbf{0.6} \\ \mathbf{0.1} & \mathbf{0.1} & 1.0 & \mathbf{0.1} & \mathbf{0.1} \\ \mathbf{0.1} & \mathbf{0.1} & \mathbf{0.1} & 1.0 & \mathbf{0.1} \\ \mathbf{0.1} & \mathbf{0.1} & \mathbf{0.1} & \mathbf{0.1} & 1.0 \end{bmatrix}$$

Similar to *PageRank*, set the damping factor as $d = 0.85$. By applying recursively

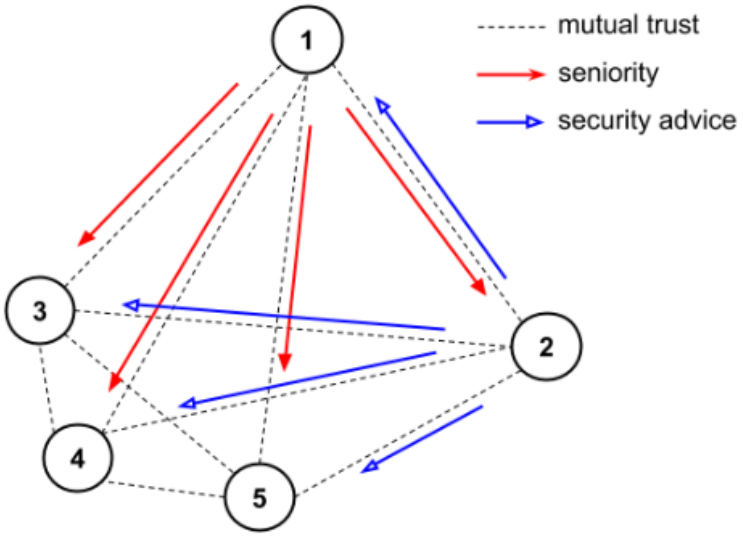


Figure 4.2: An example organization for demonstration of the calculation process

the updating rule in Equation (4.2), the importance values of all the employees will evolve through iterations as shown in Table 4.2. We can see that as the iterations proceed, the importance values converge gradually. For this example, all the values no longer change after the 15-th iteration.

For the last step, we calculate the security vulnerability value of this organization according to Equation 4.3 and the final result is

$$\begin{aligned}\mathcal{V}(m) &\approx \frac{1.036 \times 0.5 + 1.217 \times 0.2 + 0.916 \times 0.3 + 0.916 \times 0.6 + 0.916 \times 0.8}{1.036 + 1.217 + 0.916 + 0.916 + 0.916} \\ &= 0.464\end{aligned}$$

4.3 The Algorithm

The algorithm summarizing the aforementioned process is shown in Algorithm 2. It uses a loop to control the updating process. In each iteration, the updating rule, encapsulated into the function "`update()`" is performed on every employee in the organization. When a given number of iterations are performed, or no big difference is made between two consecutive iterations, this algorithm will terminate and calculate the security vulnerability measure for the given organization through the function

Table 4.2: The iterating process of calculating the importance values of all employees within the local social network of interpersonal security influence for an example organization

Iteration	$\mathcal{I}(1)$	$\mathcal{I}(2)$	$\mathcal{I}(3)$	$\mathcal{I}(4)$	$\mathcal{I}(5)$
0	1	1	1	1	1
1	1.02884615	1.23601399	0.91171329	0.91171329	0.91171329
2	1.02884615	1.23601399	0.91171329	0.91171329	0.91171329
3	1.03737897	1.21471221	0.91596961	0.91596961	0.91596961
4	1.03737897	1.21471221	0.91596961	0.91596961	0.91596961
5	1.03622334	1.21682174	0.91565164	0.91565164	0.91565164
6	1.03622334	1.21682174	0.91565164	0.91565164	0.91565164
7	1.03634853	1.21660762	0.91568128	0.91568128	0.91568128
8	1.03634853	1.21660762	0.91568128	0.91568128	0.91568128
9	1.03633555	1.21662948	0.91567832	0.91567832	0.91567832
10	1.03633555	1.21662948	0.91567832	0.91567832	0.91567832
11	1.03633688	1.21662725	0.91567862	0.91567862	0.91567862
12	1.03633688	1.21662725	0.91567862	0.91567862	0.91567862
13	1.03633675	1.21662747	0.91567859	0.91567859	0.91567859
14	1.03633675	1.21662747	0.91567859	0.91567859	0.91567859
15	1.03633676	1.21662745	0.91567860	0.91567860	0.91567860
16	1.03633676	1.21662745	0.91567860	0.91567860	0.91567860
17	1.03633676	1.21662745	0.91567860	0.91567860	0.91567860
18	1.03633676	1.21662745	0.91567860	0.91567860	0.91567860
19	1.03633676	1.21662745	0.91567860	0.91567860	0.91567860
20	1.03633676	1.21662745	0.91567860	0.91567860	0.91567860

"`average()`" which, in turn, is an implementation of Equation (4.3).

Algorithm 2: Network-based organizational security vulnerability calculation algorithm (NOSVCA)

Input: v, $\boldsymbol{M}$, K_m
Output: $\mathcal{V}(m)$

$\mathcal{I} = \{1\}_{K_m \times 1}$
for $j \leftarrow 1$ **to** max_iter **do**
  $\mathcal{I}' \leftarrow \{0\}_{K_m \times 1}$
  for $k \leftarrow 1$ **to** K_m **do**
    $\mathcal{I}'[i] \leftarrow$ `update` $(\boldsymbol{M}, \mathcal{I}, K_m)$
  end
  $\delta \leftarrow$`diff` $(\mathcal{I}, \mathcal{I}')$
  $\mathcal{I} \leftarrow \mathcal{I}'$
  if $\delta \leq$min_diff **then break**;
end
$\mathcal{V} \leftarrow$`average` $(\{v(k)\}_{1 \leq k \leq K_m}, \mathcal{I})$
return $\mathcal{V}$

In this algorithm, since the leveraging factors of all the involved importance values in Equation (4.2) can be generated beforehand from the adjacency matrix $\boldsymbol{M}$, in each iteration, we only call the function "`update()`" once for each employee. Within the function, all the employees are iterated again according to the updating rule in Equation (4.2). Thus, we conclude that the worst-case time complexity for this algorithm is $\mathcal{O}(K_m^2 N_2)$ where K_m is the number of employees in the given organization, N_2 is the maximum iterations allowed. Since the size of the adjacency matrix, $\boldsymbol{M}$, is only determined by K_m, we say that the space complexity for this algorithm is $\mathcal{O}(K_m^2)$.

Chapter 5

Evaluation

5.1 Overview

In this chapter, we evaluate the performance of the framework presented throughout the previous chapters, especially the algorithms designed for the two key components, the $\mathscr{F}$-step and the $\mathscr{G}$-step. The evaluation here is done through both simulation and real-world data based experiments.

Normally, harvesting real-world data is the first choice for evaluation. We can do this through questionnaires or field trips from targeted organizations, or through online scraping after gaining the permissions from all the employees and Internet service providers. Nevertheless, collecting real-world organizational security data is challenging, so we use simulation to break this limitation. Simulation is the imitation of the operation of a real-world process or system over time [42]. Through simulation, data is collected as if a real-world system were being observed, and then is used to estimate the measures of performance of the target system. When it's impossible to collect the real-world data, or if the real-world data collected is not enough for evaluation, simulation is no doubt a good choice to verify the analytic solutions.

5.2 $\mathscr{F}$-step Simulation

5.2.1 Data Generation

The input of $\mathscr{F}$-step, *i.e.*, the data for evaluating personal security vulnerabilities, is only the response matrix $\boldsymbol{R}$, or the data of personal online privacy setting. The content of this matrix is determined by two groups of latent variables, the personal

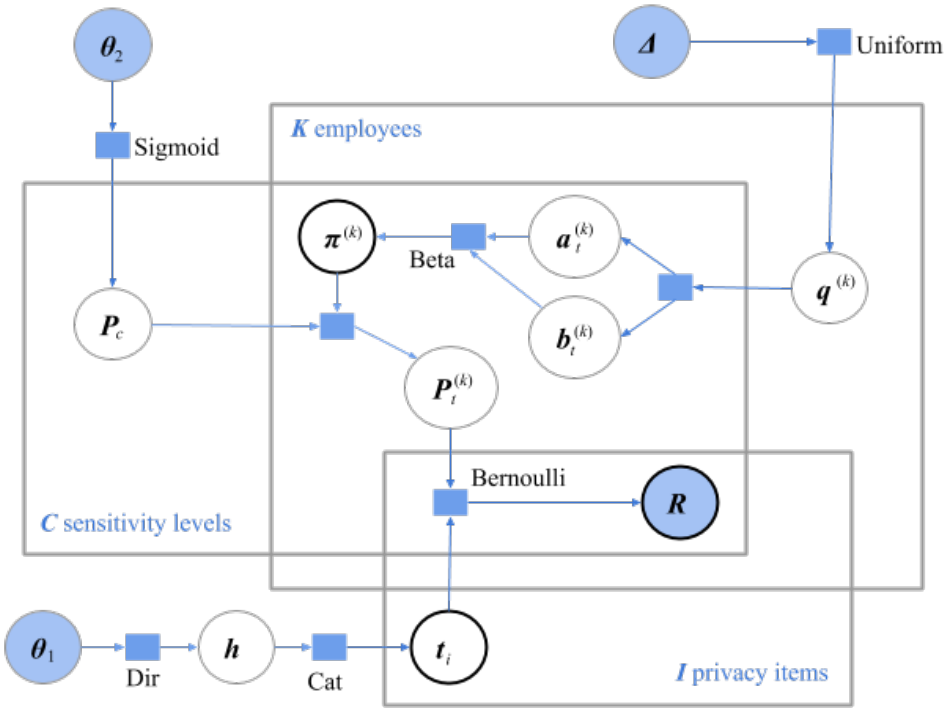


Figure 5.1: Probabilistic graphical model for generating the response matrix $\boldsymbol{R}$, the confusion matrices $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$ and the sensitivity levels of all privacy items $\{t_i\}_{1\leq i\leq I}$

confusion matrices $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$ among all the employees and the true sensitivity levels of all the privacy items $\{t_i\}_{1\leq i\leq I}$.

Figure 5.1 is the probabilistic graphical model (PGM) [43] for generating the data for $\boldsymbol{R}$, as well as the latent variables $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$ and $\{t_i\}_{1\leq i\leq I}$. There are three parts in this model: the part of privacy items shows how the true sensitivity levels of all the I privacy items are generated; that of sensitivity levels for generating a prior distribution P_c for all c that $1 \leq c \leq C$; that of employees responsible for generating the confusion matrices for all the K employees. Besides, these three plates are combined together for generating the data of the response matrix R.

Firstly, as shown in the parts of the I privacy items, we assume that the true sensitivity level, t_i, of any privacy item i is generated from a *Categorical* distribution with parameter $\boldsymbol{h}$

$$t_i|\boldsymbol{h} \sim \text{Cat}(t_i|\boldsymbol{h}) \tag{5.1}$$

where $\boldsymbol{h}$ denotes the coverage of all the sensitivity levels and is generated from a *Dirichlet* distribution $\mathrm{Dir}(\theta_1)$. In the part of the C sensitivity levels, the prior probability that a person publicizes a privacy item of category c ($1 \leq c \leq C$) is determined by a *Bernoulli* distribution with parameter P_c generated by Equation (3.1) with a controlling coefficient θ_2.

Then, in the part of employees, *Beta* distribution is used to generate the confusion matrices for all the K employees. The reason for choosing *Beta* distribution is that, by carefully controlling the shape parameters, we can generate a bell-shape distribution for each confusion matrix row and the mode of the distribution is close to the center of the the corresponding true sensitivity level. This captures the fact that people tend to perceive the true sensitivity correctly and, as a result, the levels closer to the true value are more likely to be chosen. In this part, for each employee k, a variable $q^{(k)}$ is generated from a continuous *Uniform* distribution in a given range $\Delta = [\delta_1, \delta_2]$ ($\delta_1 \gg 1$) to control the bell shape. It is then broken down into two parameters $a_t^{(k)}$ and $b_t^{(k)}$ for different (k, t) pairs ($1 \leq t \leq C$) to control the position of the mode. For example, Figure 5.2 shows two *Beta* distributions related to two different confusion matrix rows given that $C = 5$. From this example we can see that, as $q^{(k)} = a^{(k)} + b^{(k)}$ increases, the shape of the distribution becomes narrow, meaning that the person k has a better ability in perceiving the true sensitivity of a privacy items.

Thus, for each confusion matrix $\boldsymbol{\pi}^{(k)}$, we simulate its t-th row through a *Beta* distribution with parameters $a_t^{(k)}$ and $b_t^{(k)}$. That is,

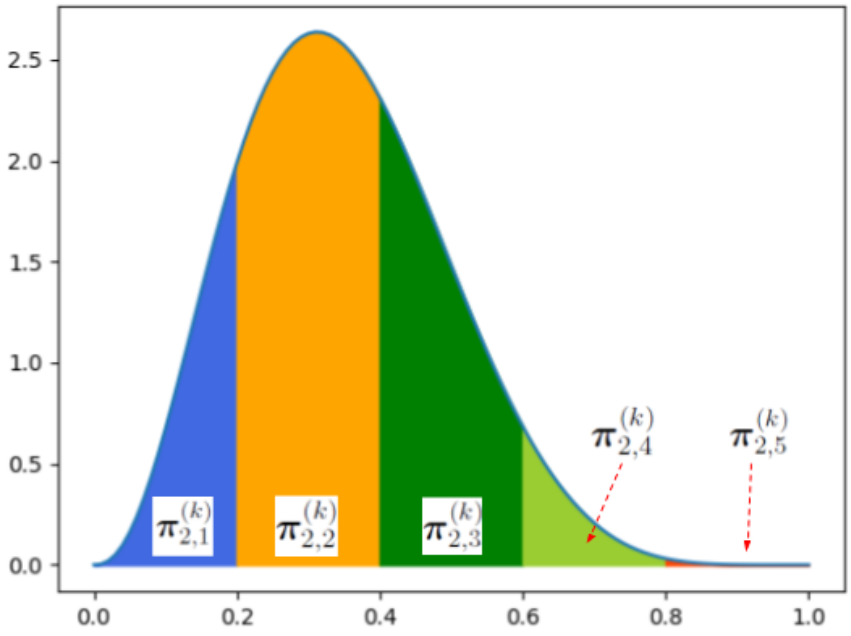
$$\boldsymbol{\pi}_t^{(k)} | a_t^{(k)}, b_t^{(k)} \sim \mathrm{Beta}(\boldsymbol{\pi}_t^{(k)} | a_t^{(k)}, b_t^{(k)}) \tag{5.2}$$

Here, it's worth noting that we only use the *Beta* distribution to generate data due to its flexibility in creating bell-shaped distributions and changing the mode. This is safe because the calculation process in $\mathscr{F}$-step, as shown in Section 3.2, does not rely on the same assumption.

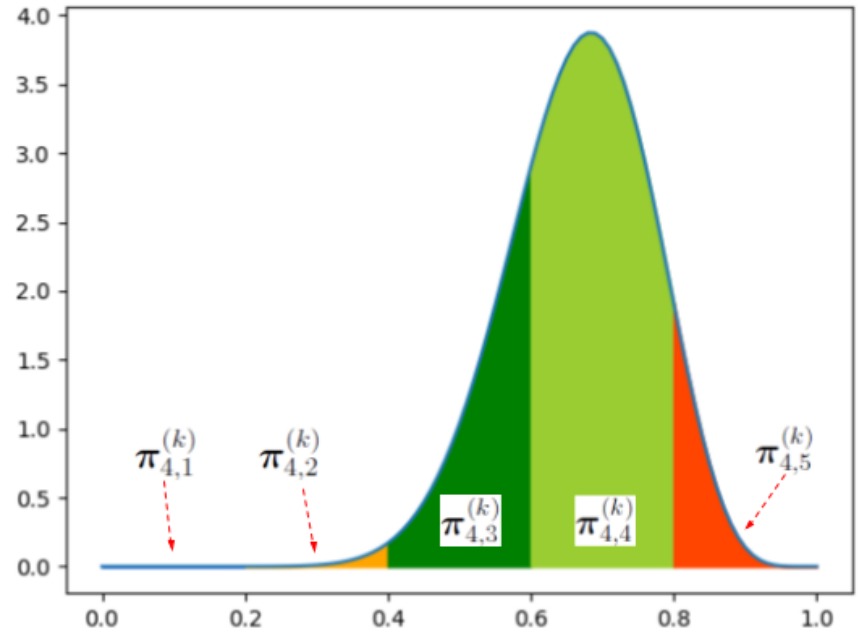
Lastly, in a combination of all the three parts, we calculate each $P_t^{(k)}$ from $\boldsymbol{\pi}^{(k)}$ and P_t ($1 \leq t \leq C$) through Equation (3.3), and then generate the final response matrix $\boldsymbol{R}$ through a *Bernoulli* distribution

$$\boldsymbol{R}_i^{(k)} | t_i, P_t^{(k)} \sim \mathrm{Bernoulli}(\boldsymbol{R}_i^{(k)} | t_i, P_t^{(k)}) \tag{5.3}$$

where $\boldsymbol{R}_i^{(k)} = \boldsymbol{R}(k, i)$.



(a) $t = 2$, $a^{(k)} + b^{(k)} = 10$



(b) $t = 4$, $a^{(k)} + b^{(k)} = 20$

Figure 5.2: Examples of generating confusion matrices from *Beta* distributions given that $C = 5$

5.2.2 Performance Evaluation

Use the simulation data generated from above model, we did three experiments for different configurations, *i.e.*, $K = 100$ and $I = 20$, $K = 200$ and $I = 30$, $K = 400$ and $I = 40$. The experimental result is shown in Figure 5.3.

As shown in Figure 5.3a, along with the iterations, the error, which is defined as the average Euclidean difference of all the confusion matrices generated in adjacent iterations, tends to decrease and converge to 0. This means that the error function we choose for Algorithm 1 is efficient in finding an optimal result.

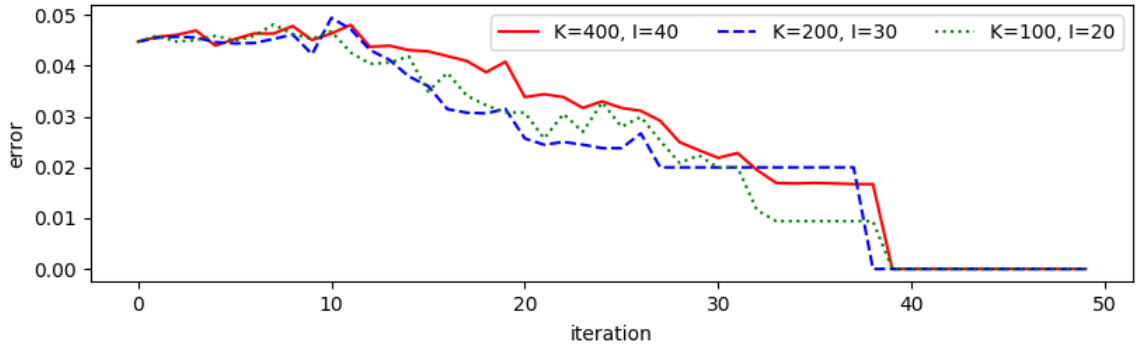
Since we also generated the ground truth of latent variables $\{t_i\}_{1\leq i\leq I}$ and $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$ through simulation, we can as well compare the intermediate results of confusion matrices and privacy items classification to their ground truth. As shown in Figure 5.3b, the difference to ground truth of $\{t_i\}_{1\leq i\leq I}$ decreases quickly and becomes stable after only a few steps. In contrast, as shown in Figure 5.3c, the difference to the ground truth of $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$ also reduces along the iterations but becomes stable gradually. Overall, this algorithm performs well in revealing the confusion matrices.

5.3 $\mathscr{G}$-step Simulation

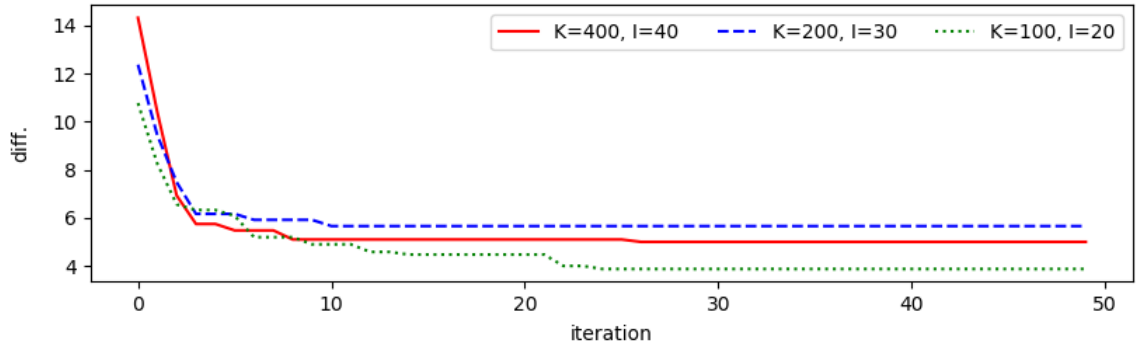
5.3.1 Data Generation

The generation of the data here follows a similar way to Section 5.2. A probabilistic graphical model is also used to generate the local social network in the workplace, represented by an adjacency matrix $\boldsymbol{M}$, and the personal vulnerability values of all the employees in the hypothetical organization.

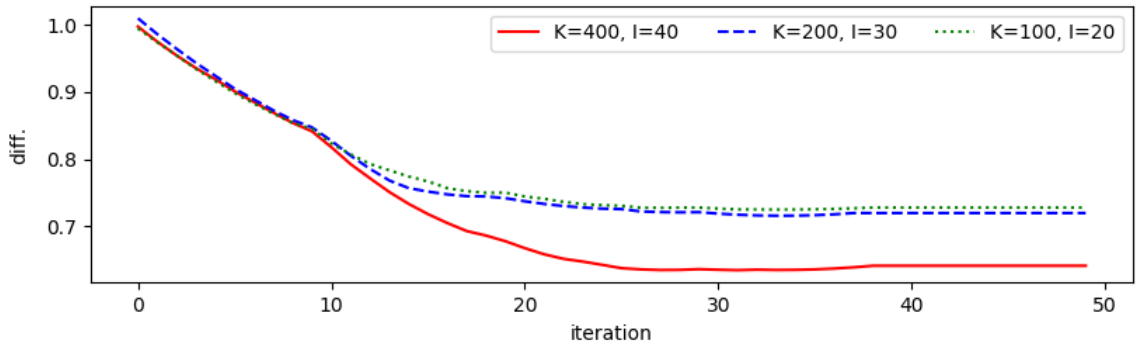
Figure 5.4 visualizes such an adjacency matrix generated by simulation. As we can see, employees are categorized into multiple departments or groups which are shown as blocks arranged diagonally. Within each department or group, employees have a strong inter-personal security influence. But, there are also somewhat weaker relationships across departments. In particular, there are two special groups. People in the first group have managerial role to other departments, and people in the second group provide technical support to all the other staff. These people are influencing strongly the security behaviours of other employees through ways like seniority, organizational information updates, security advice, and IT troubleshooting.



(a) Error over iterations



(b) Distance to ground truth $\{t_i\}_{1\leq i\leq I}$ over iterations



(c) Distance to ground truth $\{\boldsymbol{\pi}^{(k)}\}_{1\leq k\leq K}$ over iterations

Figure 5.3: Changing of inter-iteration error and the distances to ground truth over iterations for Algorithm 1

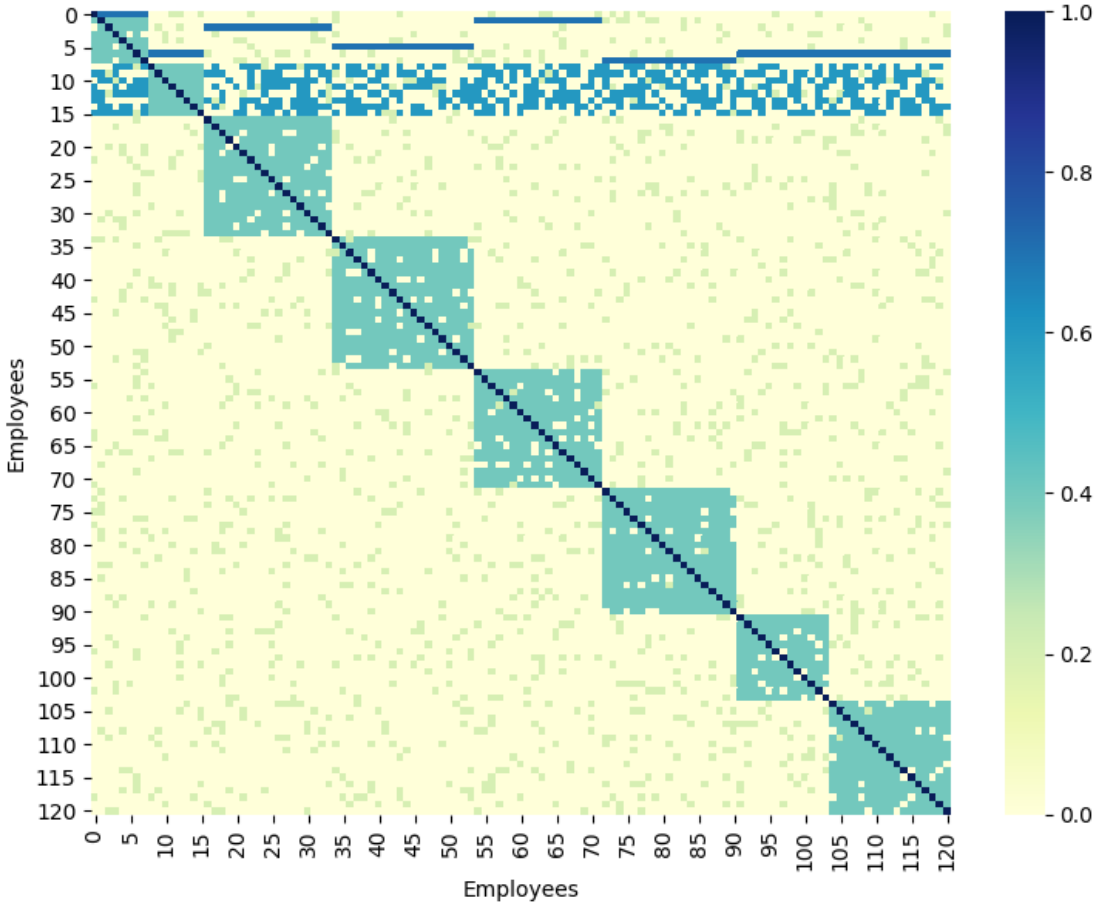


Figure 5.4: An example of local social networks of interpersonal security influence generated by simulation

5.3.2 Performance Evaluation

Based on the simulation data generated above, the resulted performance of Algorithm 2 is shown in Figure 5.5. We can see that, along the iterations the error, defined as the difference of average personal vulnerability values between adjacent iterations, decreases quickly and finally approaches 0. This means that the algorithm is very efficient in finding an optimal result.

In the experiment, we also tested the results of increasing the personal vulnerability of people in a given department or group until reaching the maximum value. Three different people groups are chosen, the group of managerial personnel, the group of people in charge of technological support, and a randomly selected department. As we

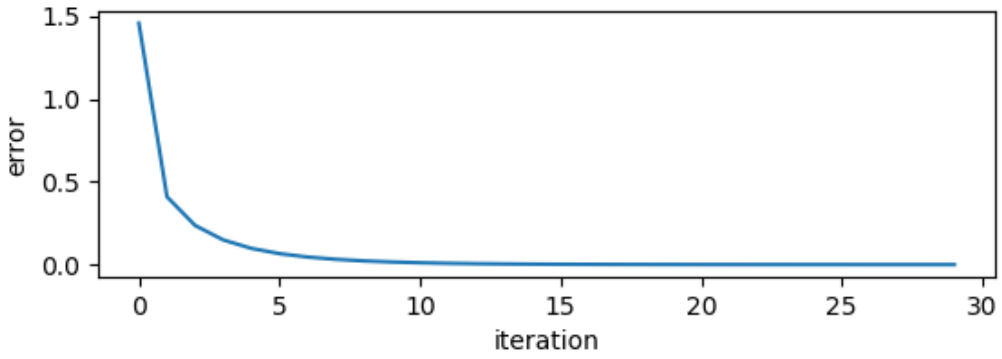


Figure 5.5: Changing of error along the iterations for Algorithm 2

can see in Figure 5.6, the overall organizational vulnerability increases monotonously as we increase the average personal values. As a proof of validity for Algorithm 2, this fits very well our perception that better behaviour of employees results in better overall security level of an organization.

Also, we can see that the curves in Figure 5.6 have different slopes and different final organizational vulnerability values. This means that the groups chosen have different influence onto the information security climate in the workplace. More experiments suggest that two types of parameters used in the simulation, *i.e.*, the size of the group and the average strength of influence the employees in each group have to other personnel, are the reasons for these differences. Bigger influence factor leads to a bigger slope, bigger group size results in a bigger final organizational vulnerability value.

5.4 Experiment with Real-world Data

5.4.1 The Dataset

As a complement, here an experiment with real-world data is presented to demonstrate the calculation process of the whole framework. The dataset used here is a set of $1,260,438$ business email accounts, from nearly $1,000$ organizations, that have been compromised after security attacks.

From this dataset, we randomly selected $10,000$ accounts and query the related user profile data from an online people search engine called *FullContact*. Among these accounts, we got the query results for $6,343$ ones. After comparison, we chose 26

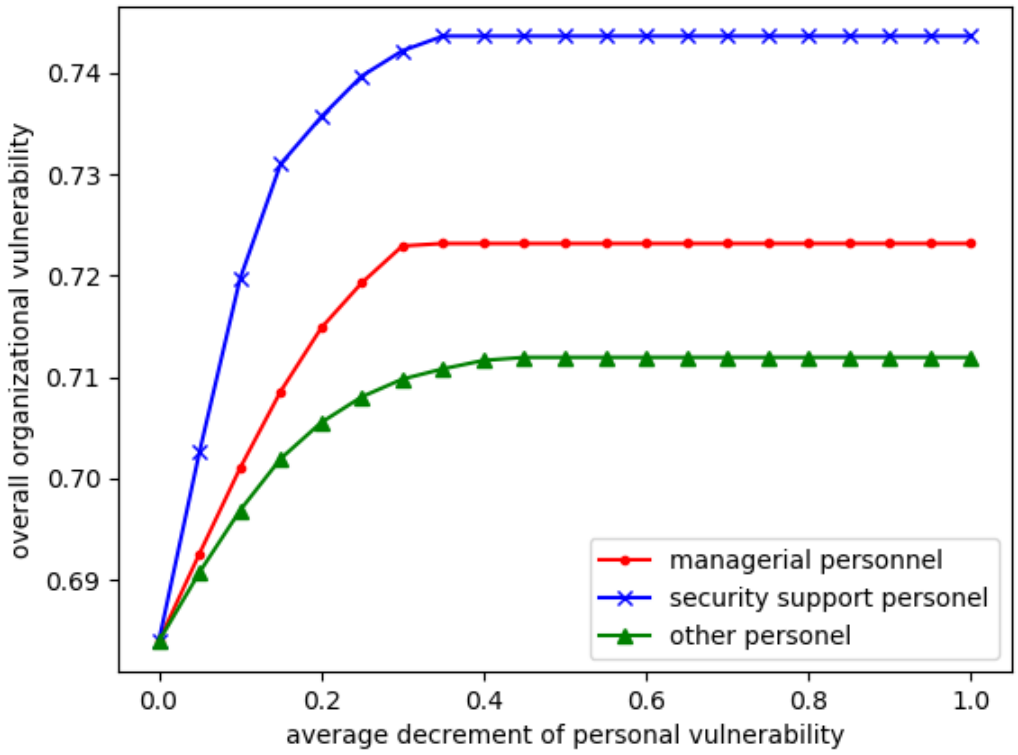


Figure 5.6: Changing of overall organizational security vulnerability along the increment of personal vulnerability for certain groups of employees

privacy items from the profile data acquired from *FullContact* for behaviour analysis. These privacy items and their average exposure among all the employees are shown in Figure 5.7. As we can see, nearly 60% of the employees choose to publicize the information of their employers, like the company name, location, and so on. In contrast, they tend to hide their accounts for social network services, especially of those related closely to their personal life.

Figure 5.8 shows the distribution of how the privacy items are exposed among all the employees. This distribution curve consists of three parts: the majority of people managed to keep their privacy secret from *FullContact*, about 10% of the people exposed 2 privacy items, the average number of exposed privacy items among the rest people is about 8.

In addition, by grouping all the email accounts according to their domains, we found that these people belong to more than 500 organizations. Figure 5.9 shows the

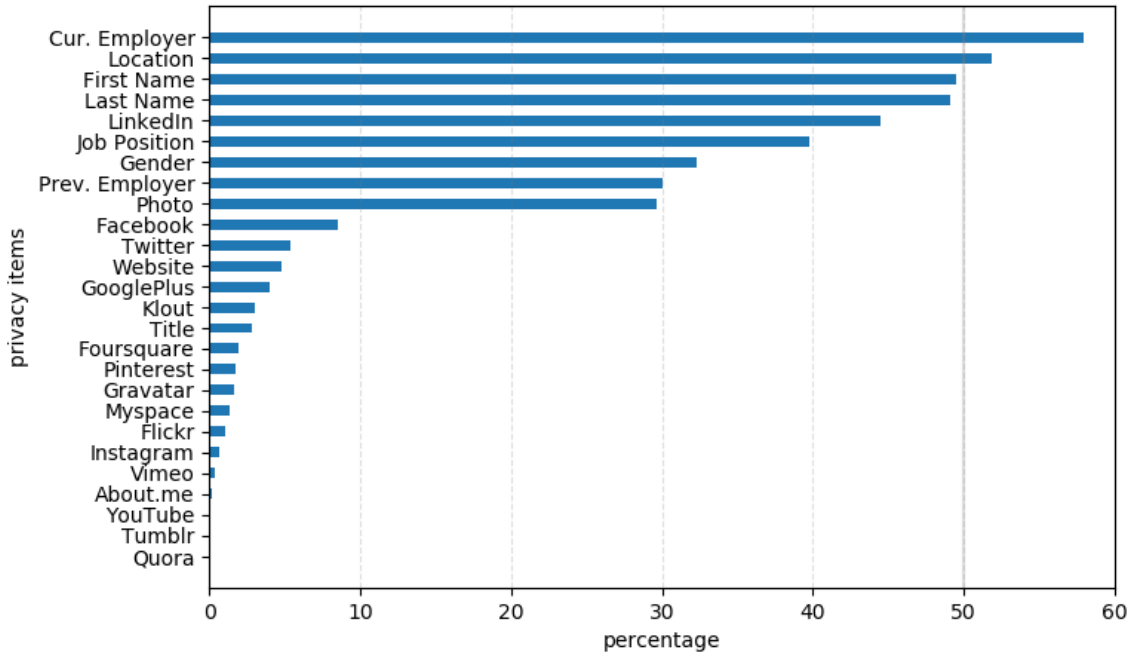


Figure 5.7: Average exposure rates of privacy items in the experimental dataset

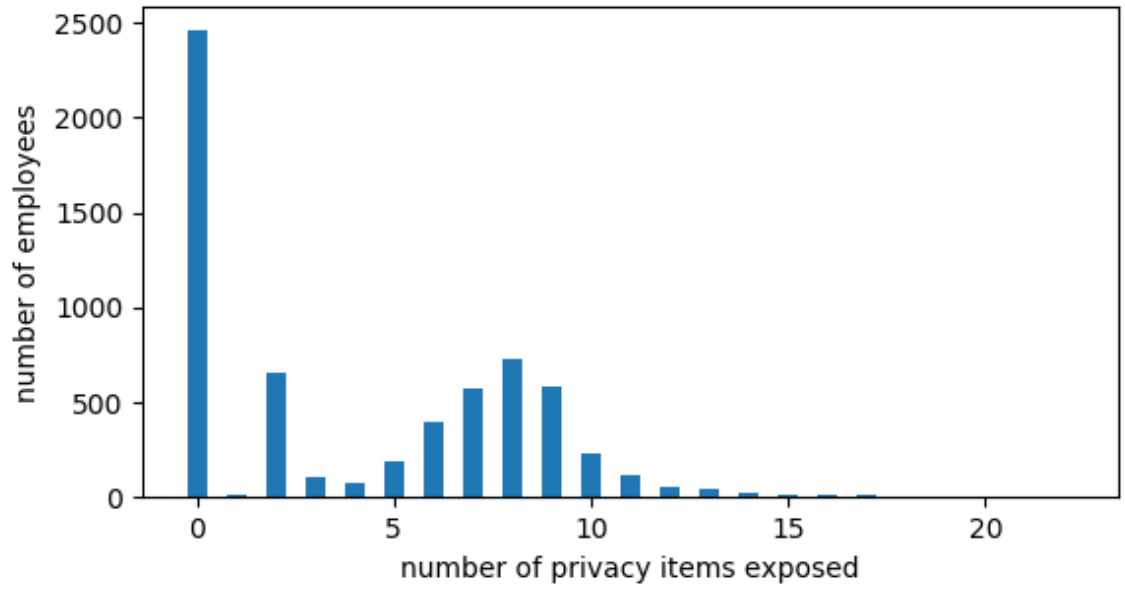


Figure 5.8: Distribution of number of privacy items exposed among employees in the experimental dataset

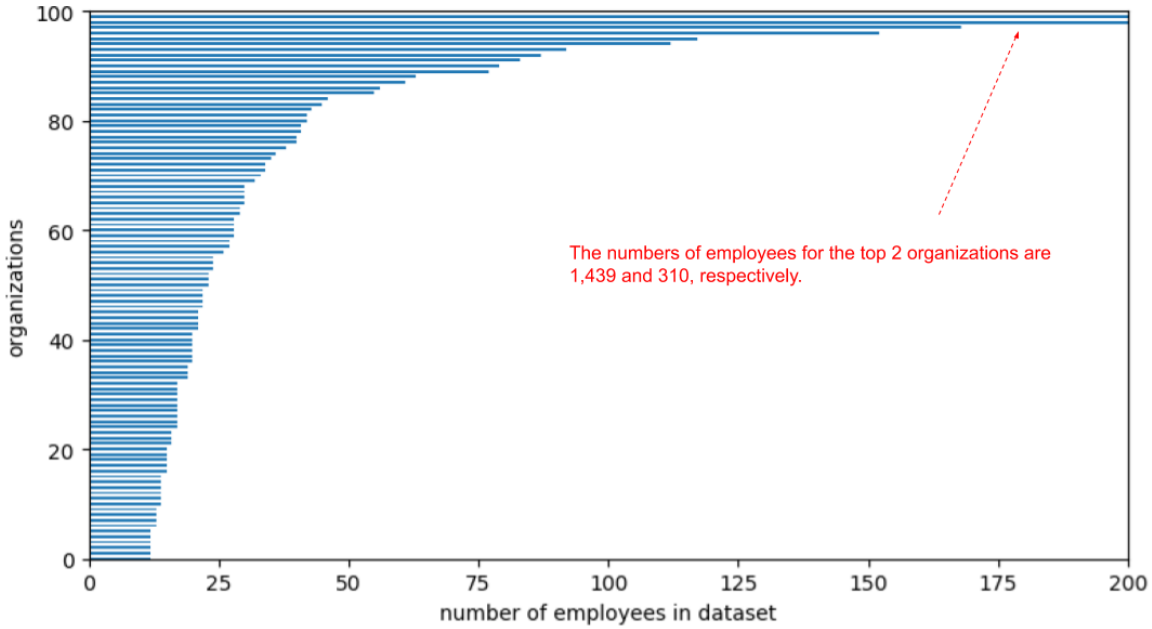
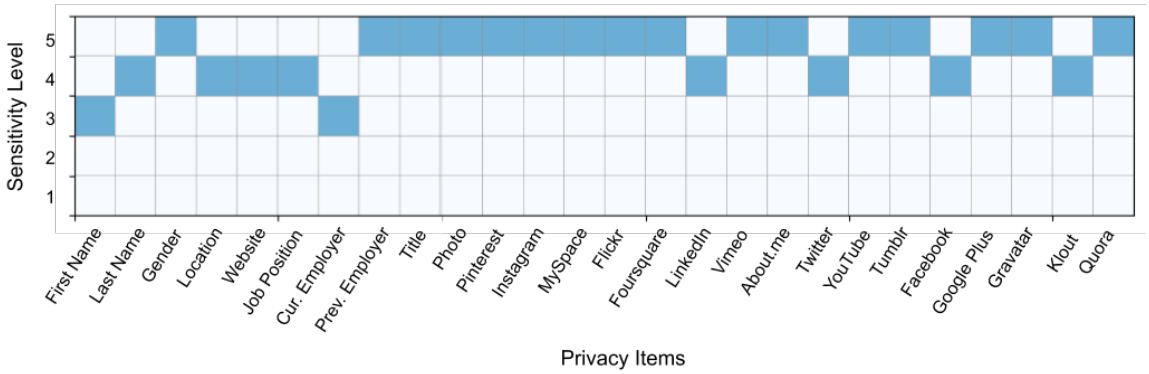


Figure 5.9: Top 100 organizations of the biggest number of employees recorded in the experimental dataset

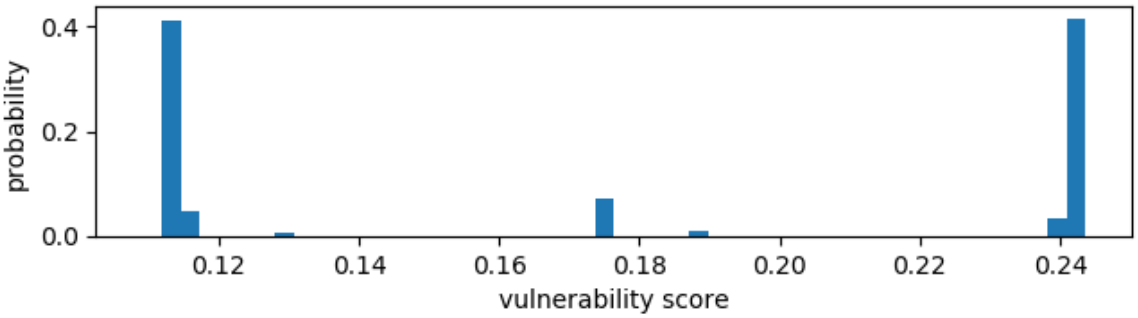
top 100 organizations that have the biggest number of employees whose email accounts are compromised. To our astonishment, the majority of these organizations are not medium or small-scaled companies, but instead top technology corporations, banks, big commodity vendors, government bodies and even military branches. For some of the top technology companies involved, our daily life has been tightly bonded to their products, like personal computers, operating systems, office automation software, smartphones and tablets, cars, daily commodities, *etc.*.

5.4.2 Calculation Results Analysis

After performing personal vulnerability calculation ($\mathscr{F}$-step), we got the results as shown in Figure 5.10. Firstly, Figure 5.10a shows the sensitivity levels found for all the privacy items during the calculation. Comparing this result to Figure 5.7, we found that both figures have a similar trend but not the same in detail. Still, from the comparison, we conclude that privacy items with bigger exposure rates tend to be less sensitive. Then, as shown in Figure 5.10b, personal vulnerability values tend to cluster into three groups. This is similar to the distribution shown in Figure 5.8 which is coincidentally also a combination of three parts.



(a) privacy items classification to the sensitivity levels



(b) Histogram for the distribution of personal vulnerability values

Figure 5.10: Personal vulnerability calculation results from the experimental dataset

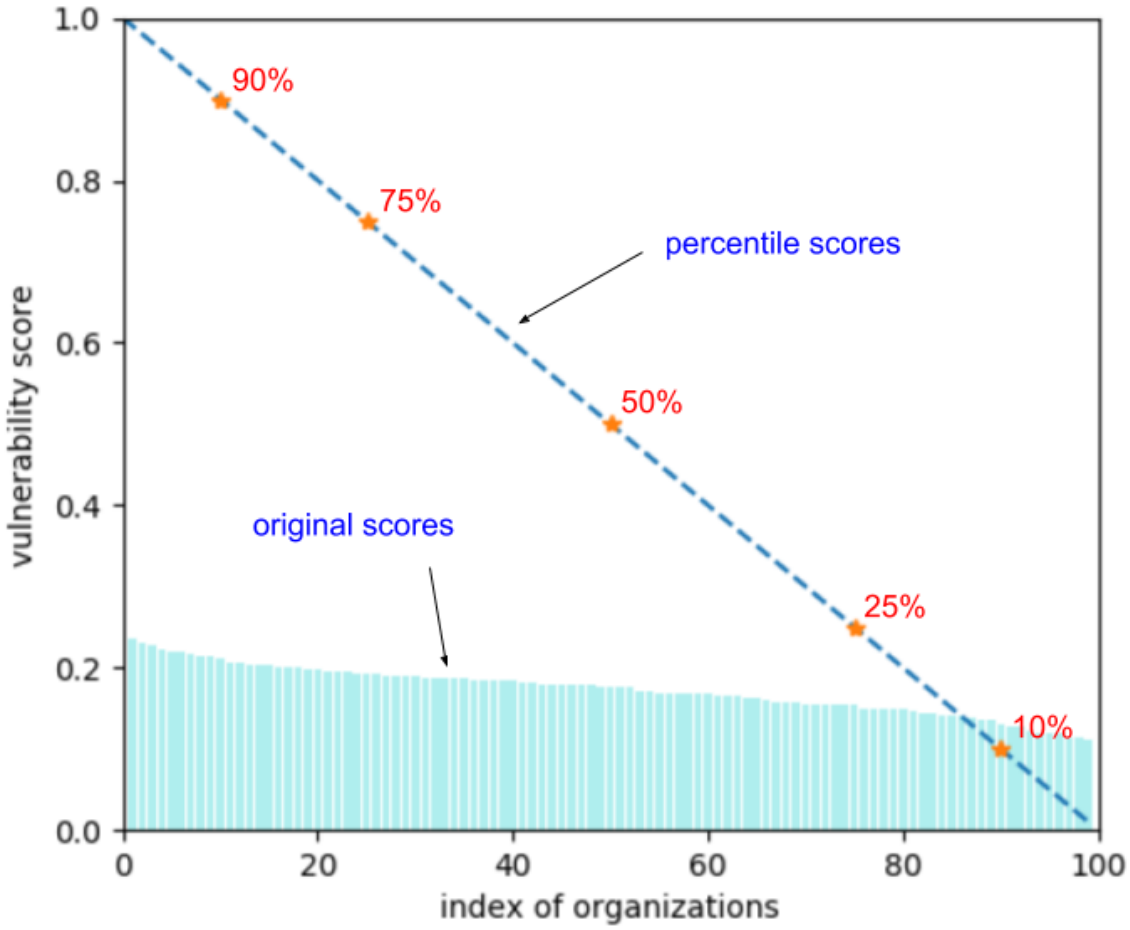


Figure 5.11: Final vulnerability scores for the 100 organizations listed in Figure 5.9

Furthermore, we then apply the calculation of $\mathscr{G}$-step and $\mathscr{C}$-step for the companies listed in Figure 5.9. Here, since we do not know about the inter-personal influential network, we simply derive the organizational vulnerability values by calculating the average personal values for all the known employees. The result is shown in Figure 5.11. The original scores, as the output of $\mathscr{G}$-step, are distributed in a pretty narrow range, approximately within $[0.1, 0.25]$. In comparison, the percentile values after the $\mathscr{C}$-step calculation is more illustrative. That is, just as discussed in Section 2.3, the calibration result is a relative position of an organization among all other organizations. It indicates how security each organization is by comparing it with the performance of the others.

5.5 Discussion

In this chapter, we used both simulation and real-world data based experiments to evaluate the performance of the framework proposed in this thesis. Admittedly, having real data is a tremendous asset to our research work. It effectively demonstrates the applicability of the framework by illustrating the calculation process in Section 5.4. In the meantime, we have to acknowledge that the dataset may not be complete. The reason is that the user profile data from *FullContact* is only an incomplete snapshot of human behaviours in online privacy management. The data may be out of date, inaccurate, or missing organizational structure information which is critical in building the local social networks for the calculation in $\mathscr{G}$-step. Also, it is possible that *FullContact* has missed the information for some people of low vulnerability, or wrongly combines the information of two different people. For this reason, the experimental results may not truly reflect the vulnerability of organizations. Our point here, however, is to develop a way for such estimation, which shows good results should the real-world data were accurate enough (like the case in simulation).

As to simulation, throughout this chapter, we found that it shows advantages in evaluation. On one hand, by using simulation, we can generate the ground truth data while accumulating the data for evaluation. As shown in Section 5.2, these ground truths play an important role in justifying the performance of our algorithms. If only with real-world data, it is almost impossible to derive these ground truth because the related variables we need are latent. On the other hand, using simulation is more flexible and the results are more general than merely using real-world data. We can easily cover different scenarios by adjusting the variables in the simulation model, like changing the size of an organization, setting the influential factors of different types of interpersonal security relationships, to name a few.

Chapter 6

A Demo Application

6.1 Introduction

As discussed in Section 2.4, a good way to put the framework into practice is to combine it with machine learning methods. One typical application scenario is to infer the security status of any new organizations based on the knowledge collected from known organizations. Specifically, suppose there is an information security company proficient in organizational information security. As it accumulates more and more security profiles of its customers, it is reasonable for this company to apply the knowledge accumulated from the previous practice for new organizations. This is a good way for this company, as well as other similar practitioners, to extend more customers and promote its products.

To demonstrate this idea, we developed a prototype system as shown in Figure 6.1. It starts from the dataset of compromised business emails we have used in Section 5.4. Instead of using all the data items, only the data of companies listed in *Fortune 1000*, a ranking system annually nominating outstanding companies in American, are selected for this system. From the website of this ranking system, a basic profile, like name, location, industry, scale, assets, *etc.*, is presented for each company nominated. Then, based on the filtered data, a machine learning engine feeds in new profile data about the companies and their employees and keeps the local knowledge base up to date. Afterward, according to the query criteria, the predicted vulnerability score of any given organizations is provided as an output.

Being the core of this prototype system, the learning engine is a combination of a group of components orchestrated to accomplish certain missions. Some compo-

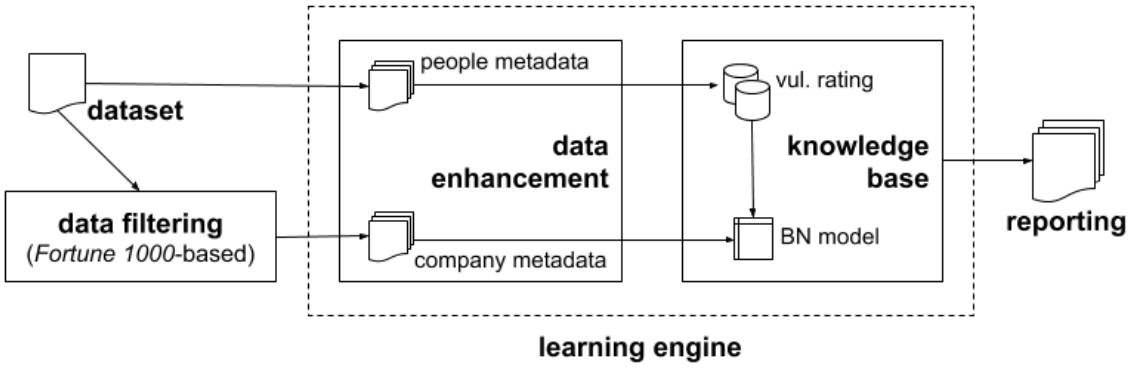


Figure 6.1: Flow of processing for the prototype system

nents are responsible for data enrichment, including the profile data, which can be retrieved from the *Fortune 1000* website, and people data, to which we have to rely on the data services of different social network systems, cloud service systems, and people search engines (like *FullContact* which we have used in Section 5.4). Other components are for knowledge management, mainly updating the personal security vulnerability values and integrating them into organizational scores. Particularly, a Bayesian network-based reasoning model [43] is included in this learning engine, for predicting the vulnerability scores of any new organizations not included in the current knowledge base.

6.2 The Bayesian Network Model

As shown in Figure 6.2, this prototype systems relies on a combination of data coming from multiple data sources. The backbone data is business email accounts, in which the domain of each email indicates the company that an employee belongs to. The backbone data is then enriched with the profiles of employees and those of organizations. Among the profile data, we are only interested in a few of the data items. As shown in Table 6.1, the data items chosen from company profiles are like the industry, scale (*i.e.*, the number of employees) and assets, the selected items from people profiles are job position and job type.

Also shown in Table 6.1, for the ease of data modelling, the values of the chosen profile items were first mapped into discrete variables. Based on these variables, we then built the Bayesian network model as shown in Figure 6.3. In this model, the

Table 6.1: Definition of the variables used in the Bayesian network model

Name	Discrete Values
Industry	1 - finance & insurance
	2 - IT & telecommunications
	3 - manufacturing & constructing
	4 - retail & consumer services
	5 - energy, oil, gas & utilities
	6 - healthcare, media, public sector, ...
Scale	1 - large ($\geq$50,000)
	2 - medium ($\geq$10,000)
	3 - small (other)
Assets	1 - big ($\geq$2M/person)
	2 - medium ($\geq$0.5M/person)
	3 - small (other)
Job Position	1 - C / VP level
	2 - director / managerial level
	3 - staff & others
Job Type	1 - management / HR
	2 - research, IT & IS
	3 - sales, marketing, support, operation, ...
Vulnerability	1 - very high ($\geq$ 90%)
	2 - high (($\geq$ 75%))
	3 - moderate ($\geq$ 25%)
	4 - low ($\geq$ 10%)
	5 - very low ($<$ 80%)

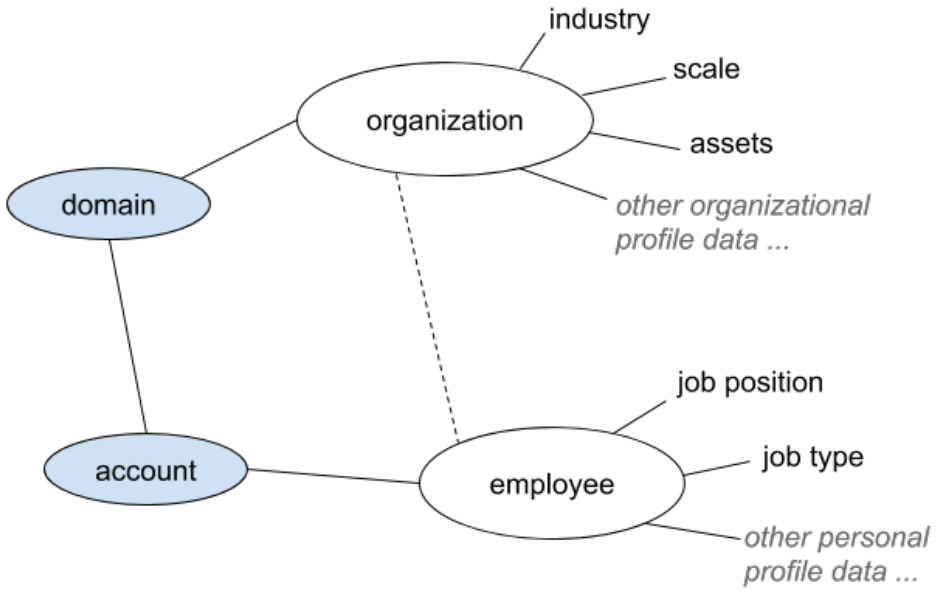


Figure 6.2: The data structure after data enhancement, consisting of the basic business email accounts, enriched people profiles and profiles of organizations

vulnerability status of an organization, denoted as a random variable V_{O}, depends on three different factors, I as the industry the organization belongs to, S as its scale, and A as its per-person assets. With this model, we can also predict the vulnerability of any employee affiliated in this organization. Denoted as V_{E}, this variable relies on V_{O} and two other factors, the job position L and the job type B of the employee.

The idea of building this model is very straightforward. Firstly, it is obvious that organizations from different industries are different in awareness and resistance to security issues due to their distinctions in technological ability, business models and industrial culture. Then, a company with more employees reasonably faces a more difficult situation in maintaining the security level of all its staff, whereas a richer enterprise is more capable to arm itself with better technologies against potential malicious attacks. For the employees in each company, it is also understandable that a person of a higher position tends to be more aware of her privacy, a person from IT department should be more skillful in protecting personal data than people from other departments.

It is very convenient to do probabilistic reasoning on this model. For example, if we've already known some facts of a company, like its industry, scale and per-person asset, the average vulnerability level of its employees can be computed as follows,

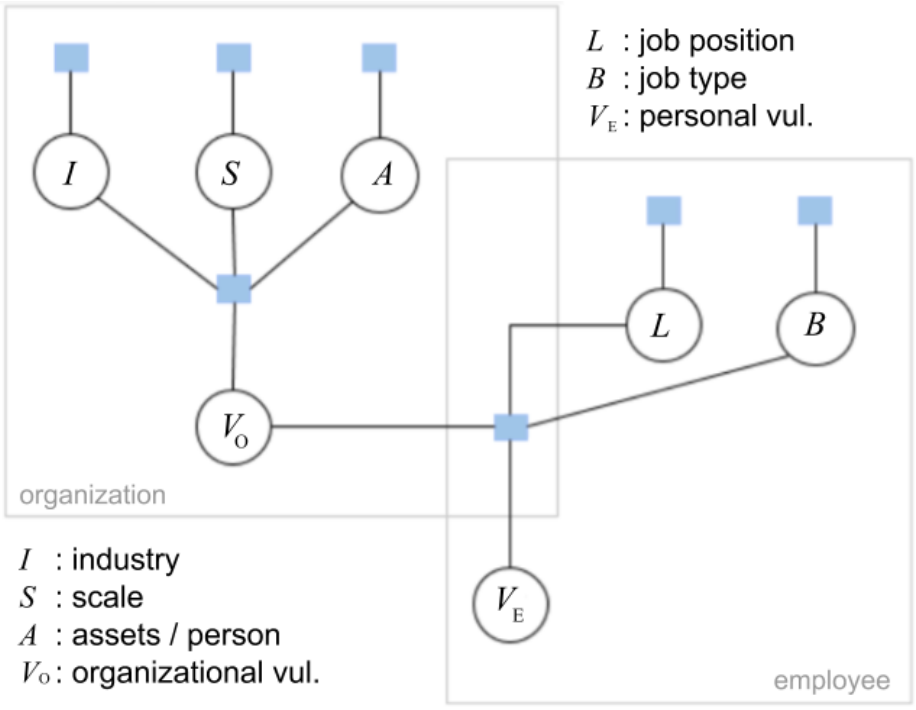


Figure 6.3: The Bayesian network model for predicting security vulnerability of organizations or employees

$$\begin{aligned} \mathrm{P}(V_{\mathrm{E}}|I,S,A) &= \sum_{V_{\mathrm{O}},L,B} \mathrm{P}(V_{\mathrm{E}}|V_{\mathrm{O}},L,B)\mathrm{P}(V_{\mathrm{O}}|I,S,A)\mathrm{P}(L)\mathrm{P}(B) \\ &= \sum_{V_{\mathrm{O}}} \sum_{L,B} \mathrm{P}(V_{\mathrm{E}}|V_{\mathrm{O}},L,B)\mathrm{P}(L)P(B)\mathrm{P}(V_{\mathrm{O}}|I,S,A) \end{aligned} \tag{6.1}$$

Even when nothing is knowing in prior, we can still do a similar reasoning. For example, for the organizational vulnerability of any company, we have

$$\mathrm{P}(V_{\mathrm{O}}) = \sum_{I,S,A} \mathrm{P}(V_{\mathrm{O}}|I,S,A)\mathrm{P}(I)\mathrm{P}(S)\mathrm{P}(A) \tag{6.2}$$

Moreover, if we know in prior the security status of an organization, we can also do the reasoning for the probabilistic distribution of its industry. The calculation is as follows,

$$\begin{aligned} \mathrm{P}(I|V_{\mathrm{O}}) &= \frac{\sum_{S,A} \mathrm{P}(V_{\mathrm{O}}|I,S,A)\mathrm{P}(I)\mathrm{P}(S)\mathrm{P}(A)}{\sum_{I,S,A} \mathrm{P}(V_{\mathrm{O}}|I,S,A\mathrm{P}(I)\mathrm{P}(S)\mathrm{P}(A)} \\ &\propto \sum_{S,A} \mathrm{P}(V_{\mathrm{O}}|I,S,A)\mathrm{P}(I)P(S)\mathrm{P}(A) \end{aligned} \tag{6.3}$$

6.3 Implementation and Demonstration

The prototype system is implemented in Python. As shown in Figure 6.4, the resulting system is a composition of several weakly-coupled subsystems. Within the system boundary, these subsystems collaborate together to achieve goals like data enrichment, probabilistic reasoning and result reporting. All these subsystems are built based on the rich selection of open-source Python packages. Specifically, a light-weight web framework, *Django*, is used to implement the web-paged console providing functions like system configuration, missions control, and output presentation. Packages *Celery* and *Redis* are adopted to build the scalable distributive tasking environment due to their strong ability in executing tasks in parallel. This is very effective in accelerating the calculation processes of data enrichment and knowledge management in our system. What's more, *Scrapy*, a web page scrawling package, together with *Splash* which is a virtual sandbox for parsing Javascript embedded in web pages, is employed to facilitate the web page analyzing tasks. Besides, *Pgmpy*, the Python implementation of utilities for probabilistic graphical modelling, is chosen for training and reasoning on the Bayesian network model.

Take the *Django*-based web framework implementation as an example. It consists of various types of Python modules, such as *Django* models, views, forms, templates, and so on. These modules work together to form an MVC (model-view-controller) structure. As shown in Figure 6.5, one function of this web framework is to control the missions of data enrichment and knowledge-base management. In this application, a *Django* model named `Mission` is used to represent missions. Each mission is controlled within three fixed status, *i.e.*, stopping, waiting (for start) and working. These states can be monitored and configured by system administrators through a web page named `console.htm`. Any user operations from this web page is then synchronized to a back-end module called `view.py`. Afterwards, the states of all the missions are read by *Celery* modules for scheduling and controlling the parallel tasks.

Finally, as a demonstration of the prototype system, Figure 6.6 shows two snap-

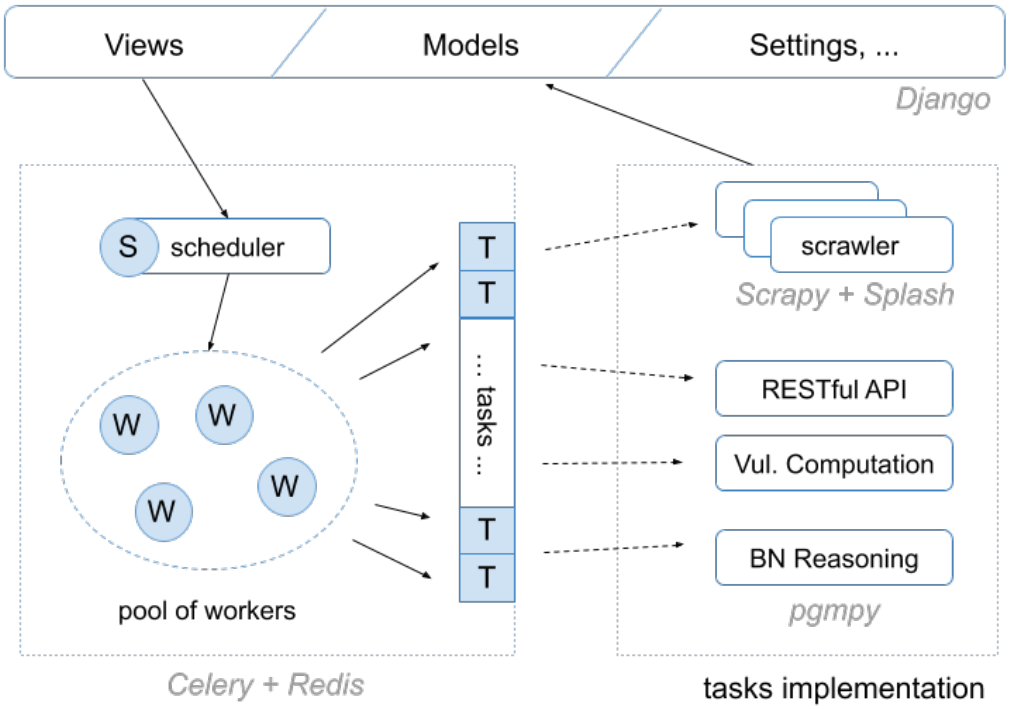


Figure 6.4: Python-based implementation for the demo application

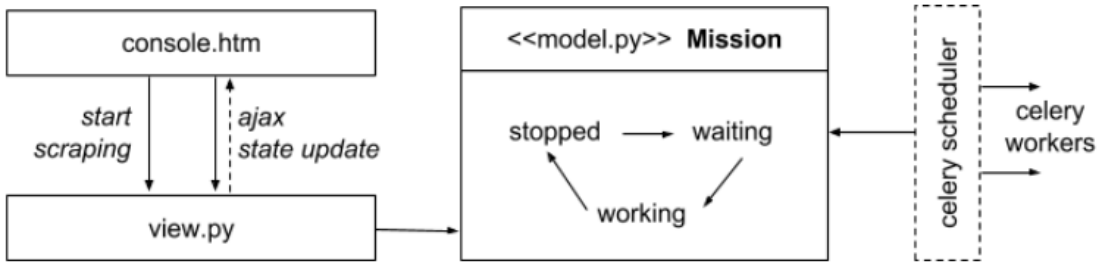
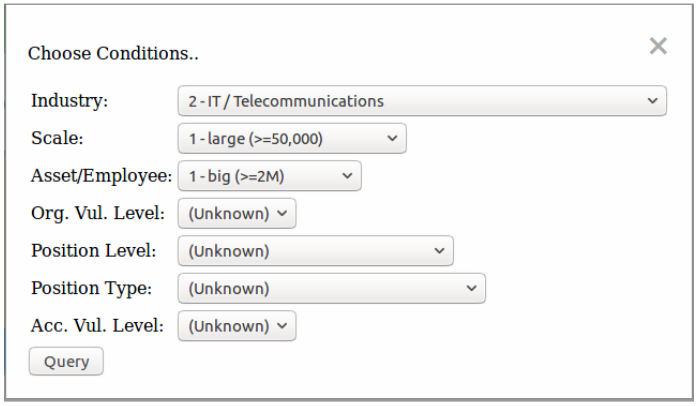


Figure 6.5: Structure and working procedure of Django web consoles

shots of the web page that visualizes the reasoning process of the Bayesian network model. In this web page, we can choose different combinations of values for the random variables, like the industry, scale and assert per person, and query the vulnerability level of an organization or a person. The underlying system workflow for this operation is as follows. After the user chooses prior information from the panel shown in Figure 6.6b, an AJAX request is sent to the server-side to trigger the reasoning of the Bayesian network model. Then, as the HTTP response is received, the

(a) Graphical illustration



(b) Random variables panel

Figure 6.6: Snapshots of the web page for Bayesian network-based reasoning

HTML5 canvas shown in Figure 6.6a is refreshed according to the returned posterior distributions.

Chapter 7

Conclusions

7.1 Concluding Remarks

Despite the ever-growing technologies that enhance the protection of sensitive information in the workplace, human factors are frequently blamed to be the weakest link in the security chain of organizations. Lacking awareness of security risks, poor ability in handling security issues, careless behaviours among employees are constantly making organizations at stake facing the various security attacks. Especially, as the biggest threat to organizations nowadays, social engineering attacks include social means in their malicious efforts and target employees to steal confident data or exploit IT services within the organizational border. In this thesis, to assess the security vulnerability of organizations, we presented a framework that takes the online behaviour data of employees as input, analyzes their security vulnearbility, and then generates a vulnerability score for each of the organizations.

In this framework, the behaviours of online privacy management are taken as an indicator of personal awareness of security risks for employees. Taking this type of behavioural data as an input, the calculation for organizational vulnerability scores consists of three steps. In the first step, by analyzing the input data, a confusion matrix is generated for each employee as the personal security awareness profile. That is, the content of such a matrix tells how a person mistakenly confuses the information of a given sensitivity level to that of other levels. In the following-up step, within each organization, the personal vulnerability values of all the employees belonging to the same organization, generated from related confusion matrices, are synthesized into an overall organizational vulnerability value according to the local social network of

interpersonal security influence in the workplace. In the last step, the scores generated from the previous step for all the organizations are sorted, the percentile value is used as the final vulnerability reading for each organization. In evaluation, experiments based on both simulation and real-world data show that this framework works pretty well in estimating organizational security vulnerabilities. Also, a prototype system shows how this framework can be applied in practice.

Comparing to many other efforts targeting workplace directly to collect data of human-side security weakness, either through surveys, questionnaires or field trips, our approach is distinct in that it is based on the user behavioural data from the Internet. This brings us several advantages. Firstly, there is plenty of data on the Internet. Such big data availability guarantees the scalability of our approach because it is open to involving more organizations and employees in the calculation. Secondly, since the data used for the assessment are of the same type, there are no longer the barriers among organizations regarding the assessment. So, we can easily integrate the assessment services for organizations of different backgrounds into one unified platform. Moreover, having the same type of data also means nondiscrimination. The traditional methods are often accused of being subjective and biased because their data may only apply to given organizations, but this is no longer an issue for our approach.

7.2 Future Work

For the research work in this thesis, we can proceed through several different directions in the future.

Firstly, it has been proven that people are often grouped into communities in online social networks. The evaluation results as shown in Figure 5.10b also demonstrate such a clustering effect. So, as an improvement of the current framework, we can group employees into communities according to the similarity of their security awareness and their abilities in dealing with security issues. In this way, we would only need a few confusion matrices in the calculation, the overall performance of the calculation can also be significantly optimized.

Secondly, one shortcoming of the current research work is the imperfection of the dataset used in the evaluation. As we discussed in Section 5.5, the data content mainly comes from a people search engine called *FullContact* and the searching results may not be accurate. So, the data may be misleading, not mentioning the fact that the

information of the local interpersonal influential network is missing. For this reason, it is necessary to collect more real-world data from both the Internet and organizations and then evaluate the performance of our framework in more depth.

What's more, as shown in Figure 2.5, a good way to bring our idea into practice is to combine it with reinforcement learning. Through this approach, the learning result (security vulnerability scores for organizations) can be updated repeatedly as more and more input data are accumulated. This is exactly the way to keep the rating result of organizational security vulnerability up to date and to better serve the needs of organizations in maintaining and improving their security defence. What we presented in Chapter 6 is a good example in this regard, but more application scenarios would be a tremendous addition in bringing the research findings of this thesis into practice.

Appendix A

Additional Information

In this thesis, we collected some user profile data from *FullContact*, an online people contact provider service, for the purpose of real-world data experiments. We hereby confirm that this research follows the privacy and ethics regulations and did not disclose any of the private data.

As stated in its website, *FullContact* "complies with the EU-US Privacy Shield Framework and the Swiss-US Privacy Shield Framework ... regarding the collection, use, and retention of personal data" and it "has certified to the Department of Commerce that it adheres to the Privacy Shield Principles" (Section 7.3, Privacy Policy). This guarantees *FullContact* to be a legal data source for our experiment.

The RESTful API of *FullContact* was used in our data collection. As said in its Terms of Use, *FullContact* grants "developers a limited, non-exclusive, non-transferable, license to ... distribute completed contact data ... to end users" with restrictions that "developer agrees not to disclose, distribute, sublicense, lease, rent, loan, resell or otherwise transfer the data ... to any third party" (Section 2, API License Addendum). In the research, we are only interested in how people dealing with personal information online and did not keep or disclose any of the data acquired. Thus, there is no privacy or ethical issue in our research.

Besides, the dataset of the compromised emails mentioned in this thesis is part of a research project at the University of Victoria. The data provider has granted us to use it for research purposes. This thesis is one of the project output and no data item in it is disclosed to the public.

Bibliography

[1] D. Dang-Pham, S. Pittayachawan, and V. Bruno, "Applying network analysis to investigate interpersonal influence of information security behaviours in the workplace," *Information & Management*, vol. 54, no. 5, pp. 625–637, 2017.

[2] J. Abawajy, "User preference of cyber security awareness delivery methods," *Behaviour & Information Technology*, vol. 33, no. 3, pp. 237–248, 2014.

[3] M. Anwar, W. He, I. Ash, X. Yuan, L. Li, and L. Xu, "Gender difference and employees' cybersecurity behaviors," *Computers in Human Behavior*, vol. 69, pp. 437–443, 2017.

[4] D. Dang-Pham, S. Pittayachawan, and V. Bruno, "Exploring behavioral information security networks in an organizational context: an empirical case study," *Journal of Information Security and Applications*, vol. 34, pp. 46–62, 2017.

[5] E. Metalidou, C. Marinagi, P. Trivellas, N. Eberhagen, C. Skourlas, and G. Giannakopoulos, "The human factor of information security: unintentional damage perspective," *Procedia-Social and Behavioral Sciences*, vol. 147, pp. 424–428, 2014.

[6] G. Öğütçü, Ö. M. Testik, and O. Chouseinoglou, "Analysis of personal information security behavior and awareness," *Computers & Security*, vol. 56, pp. 83–93, 2016.

[7] N. S. Safa, M. Sookhak, R. Von Solms, S. Furnell, N. A. Ghani, and T. Herawan, "Information security conscious care behaviour formation in organizations," *Computers & Security*, vol. 53, pp. 65–78, 2015.

[8] N. Y. Conteh and P. J. Schmick, "Cybersecurity: risks, vulnerabilities and countermeasures to prevent social engineering attacks," *International Journal of Advanced Computer Research*, vol. 6, no. 23, p. 31, 2016.

[9] A. Da Veiga and N. Martins, "Improving the information security culture through monitoring and implementation actions illustrated through a case study," *Computers & Security*, vol. 49, pp. 162–176, 2015.

[10] N. S. Safa and R. Von Solms, "An information security knowledge sharing model in organizations," *Computers in Human Behavior*, vol. 57, pp. 442–451, 2016.

[11] S. Albladi and G. R. Weir, "Vulnerability to social engineering in social networks: a proposed user-centric framework," in *2016 IEEE International Conference on Cybercrime and Computer Forensic (ICCCF)*, pp. 1–6, IEEE, 2016.

[12] L. Hadlington, "Human factors in cybersecurity; examining the link between internet addiction, impulsivity, attitudes towards cybersecurity, and risky cybersecurity behaviours," *Heliyon*, vol. 3, no. 7, p. e00346, 2017.

[13] K. Krombholz, H. Hobel, M. Huber, and E. Weippl, "Advanced social engineering attacks," *Journal of Information Security and applications*, vol. 22, pp. 113–122, 2015.

[14] S. Mamonov and R. Benbunan-Fich, "The impact of information security threat awareness on privacy-protective behaviors," *Computers in Human Behavior*, vol. 83, pp. 32–44, 2018.

[15] W. R. Flores, E. Antonsen, and M. Ekstedt, "Information security knowledge sharing in organizations: investigating the effect of behavioral information security governance and national culture," *Computers & Security*, vol. 43, pp. 90–110, 2014.

[16] Ponemon Institute, "Cost of a data breach report, 2019," tech. rep., Ponemon Institute, 2019.

[17] S. Bauer, E. W. Bernroider, and K. Chudzikowski, "Prevention is better than cure! designing information security awareness programs to overcome users' noncompliance with information security policies in banks," *computers & security*, vol. 68, pp. 145–159, 2017.

[18] S. Kassicieh, V. Lipinski, and A. F. Seazzu, "Human centric cyber security: what are the new trends in data protection?," in *2015 Portland International Conference on Management of Engineering and Technology (PICMET)*, pp. 1321–1338, IEEE, 2015.

[19] K. Renaud and S. Flowerday, "Contemplating human-centred security & privacy research: suggesting future directions," *Journal of Information Security and Applications*, vol. 34, pp. 76–81, 2017.

[20] R. Missaoui, T. Abdessalem, and M. Latapy, *Trends in social network analysis: information propagation, user behavior modeling, forecasting, and vulnerability assessment*. Springer, 2017.

[21] M. Gratian, S. Bandi, M. Cukier, J. Dykstra, and A. Ginther, "Correlating human traits and cyber security behavior intentions," *computers & security*, vol. 73, pp. 345–358, 2018.

[22] A. McCormac, T. Zwaans, K. Parsons, D. Calic, M. Butavicius, and M. Pattinson, "Individual differences and information security awareness," *Computers in Human Behavior*, vol. 69, pp. 151–156, 2017.

[23] D. Ki-Aries and S. Faily, "Persona-centred information security awareness," *computers & security*, vol. 70, pp. 663–674, 2017.

[24] E. Kolkowska, F. Karlsson, and K. Hedström, "Towards analysing the rationale of information security non-compliance: devising a value-based compliance analysis method," *The Journal of Strategic Information Systems*, vol. 26, no. 1, pp. 39–57, 2017.

[25] M. Siponen, S. Pahnila, and A. Mahmood, "Employees' adherence to information security policies: an empirical study," in *IFIP International Information Security Conference*, pp. 133–144, Springer, 2007.

[26] M. Chan, I. Woon, and A. Kankanhalli, "Perceptions of information security in the workplace: linking information security climate to compliant behavior," *Journal of information privacy and security*, vol. 1, no. 3, pp. 18–41, 2005.

[27] H. N. Chua, S. F. Wong, Y. C. Low, and Y. Chang, "Impact of employees' demographic characteristics on the awareness and compliance of information security policy in organizations," *Telematics and Informatics*, vol. 35, no. 6, pp. 1770–1780, 2018.

[28] A. A. Cain, M. E. Edwards, and J. D. Still, "An exploratory study of cyber hygiene behaviors and knowledge," *Journal of information security and applications*, vol. 42, pp. 36–45, 2018.

[29] A. R. Ahlan, M. Lubis, and A. R. Lubis, "Information security awareness at the knowledge-based institution: its antecedents and measures," *Procedia Computer Science*, vol. 72, pp. 361–373, 2015.

[30] J. M. Hatfield, "Social engineering in cybersecurity: the evolution of a concept," *Computers & Security*, vol. 73, pp. 102–113, 2018.

[31] F. Mouton, M. M. Malan, L. Leenen, and H. S. Venter, "Social engineering attack framework," in *2014 Information Security for South Africa*, pp. 1–9, IEEE, 2014.

[32] G. Tannous and A. M. Barbar, "An expert system to detect privacy's vulnerability of social networks," in *2016 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, pp. 224–230, IEEE, 2016.

[33] J. Ferreras, "Capital one data breach hits about 6 million people in canada, 100 million in u.s.," *Global News*, 2019.

[34] S. Kokolakis, "Privacy attitudes and privacy behaviour: A review of current research on the privacy paradox phenomenon," *Computers & security*, vol. 64, pp. 122–134, 2017.

[35] R. G. Pensa and G. Di Blasi, "A privacy self-assessment framework for online social networks," *Expert Systems with Applications*, vol. 86, pp. 18–31, 2017.

[36] S. Satkin, "Data scraping: theft or fair game?," *Newmeyer & Dillion LLP*, 2018.

[37] M. Venanzi, J. Guiver, G. Kazai, P. Kohli, and M. Shokouhi, "Community-based bayesian aggregation models for crowdsourcing," in *Proceedings of the 23rd international conference on World wide web*, pp. 155–164, ACM, 2014.

[38] C. Liu, S. Ghosal, Z. Jiang, and S. Sarkar, "An unsupervised spatiotemporal graphical modeling approach to anomaly detection in distributed cps," in *2016 ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS)*, pp. 1–10, IEEE, 2016.

[39] K. Liu and E. Terzi, "A framework for computing the privacy scores of users in online social networks," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 5, no. 1, p. 6, 2010.

[40] S. G. Isaksen, G. Ekvall, H. Akkermans, G. V. Wilson, and J. P. Gaulin, *Assessing the Context for Change: A Technical Manual for the Situational Outlook Questionnaire, Enhancing Performance of Organizations, Leaders and Teams for Over 50 Years*. Creative Problem Solving Group, 2007.

[41] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, vol. 30, no. 1-7, pp. 107–117, 1998.

[42] J. Banks, I. Carson, B. L. Nelson, D. M. Nicol, *et al.*, *Discrete-event system simulation*. Pearson, 2005.

[43] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.