ASONAM '19: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining August 2019, Pages 863–870
https://doi.org/10.1145/3341161.3343520

An Empirical Study of Security Culture in Open Source Software Communities

Shao-Fang Wen, Mazaher Kianpour and Stewart Kowalski

Faculty of Information Technology and Electrical Engineering

Norwegian University of Science and Technology, Norway

{shao-fang.wen, mazaher.kianpour, stewart.kowalski}@ntnu.no

Abstract: Open source software (OSS) is a core part of virtually all software applications today. Due to the rapidly growing impact of OSS on society and the economy, the security aspect has attracted researchers' attention to investigate this distinctive phenomenon. Traditionally, research on OSS security has often focus on technical aspects of software development. We argue that these aspects are important, however, technical security practice considering different social aspects of OSS development will assure the effectiveness and efficiency of the implementation of the tool. In this empirical study, we explore the current security culture in the OSS development phenomenon using a survey instrument. By performing a security cultural analysis with six dimensions: attitude, behavior, competency, subjective norms, governance and communication, this paper provides an in-depth insight into its influence on participants' security behaviors and decision-making. Measurements of security culture and the corresponding issues that need to be addressed in OSS communities were defined and discussed.

Keywords: open source software; security culture; software security

1. Introduction

Open source software (OSS) is based on the principle that software programs should be shared freely among users, giving them the possibility of introducing implementations and modifications [1, 2]. OSS is released under license in compliance with the Open Source Definition as articulated by the Open Source Initiative (also known as the OSI). To create and sustain OSS, numerous technical and non-technical individuals interact with collaborating peers in online communities of practice [3-5]. The activities that these communities perform are usually called OSS projects. This development culture includes hundreds of thousands of distributed programmers voluntarily producing, sharing, and supporting their software with no monetary compensation for their efforts. Because of the low-cost software solutions, and the openness and real collaboration of the software development process, OSS has become an increasingly popular choice instead of closed source (proprietary) software: About 80% of companies run their operations on OSS [6], and 96% of applications utilize OSS as software components [7].

Due to the rapidly growing impact of OSS on society and the economy, the security aspect has attracted researchers' attention to investigate this distinctive phenomenon. As a result, numerous security practices for secure OSS development have been provided [8]. However, OSS vulnerabilities are being found at an increasing pace, nearly doubling from 2017 [9]. From a literature review of OSS security research using a socio-technical analysis [10], Wen [8] found that only 16% of papers talked about the social sectors of OSS security (cultural, structural, legal, managerial, and operational), and he concluded that existing software security practices have limitations in supporting secure OSS development. Because OSS in the socio-technical context is broader than the technical definition [11], technical security practices that consider different social aspects of OSS development will assure the

effectiveness and efficiency of the implementation of the tool [8, 10]. This can be viewed as a necessary condition within a security management framework, as the two aspects are equally important [12].

There is still a dearth of empirical research on the social study of OSS security. Thus, this study intended to complement the research by empirically investigating the social and cultural aspects of OSS security. As Zeitlyn [13] pointed out, we need to better understand the culture of the OSS movement and the corresponding social norms that regulate people's behavior. Culture has strongly influenced the formation of many security means in an organization, such as security policy, information ethics, security training, and privacy issues [14, 15]. Security culture can also support all organizational activities in such a way that security becomes a natural aspect of the daily activities of every individual [15, 16]. By exploring the current security culture in OSS communities, we can start to understand the influence of security on participants' security behaviors and decision making. Then we can evaluate what changes would influence security in a positive way, so that we can make realistic and practical suggestions.

The paper is organized as follows. After the introduction, in section 2, we present a review of the literature on OSS communities and security culture. In section 3, we present our research framework of this study. Section 4 describes the research methodology. We present the results of this study in section 5. In section 6, we discuss the results. We present the limitations of this study and the conclusion in sections 7 and 8, respectively.

2. Literature Review

2.1. OSS Communities

OSS is predominantly characterized by clan control, which is based on common values and beliefs [17], or clan- and self-governance [18], based on self-monitoring [19, 20]. In the OSS community, individuals interact with collaborating peers to solve a particular software problem and exchange ideas [21]. They work in geographically distinct locations of the world, and rarely or never meet face-to-face [22]. In OSS communities, social and technical interaction primarily occurs in a networked mediated computing environment populated with web browsers, a mailing list, a discussion forum, instant messaging programs, and other software development tools, such as version control systems, compilers, and bug tracking systems [11]. In this context, cooperation among members of OSS communities is maintained through an elaborate infrastructure that almost exclusively uses web technologies [1]. A strong culture and group behavior have been developed in connection with the community, enabled by the Internet [23].

The structure of OSS communities is fundamentally different from that of traditional project organizations of proprietary ("closed source") software development. Traditional software development projects tend to coordinate software development work through the organizational hierarchy and centralized planning [24], or they implement security control mechanisms, including behavior- or output-based control [25]. Unlike traditional organizations, OSS communities do not have a formal organizational structure, and projects in these communities are not dictated by formal plans, schedules, and deliverables [26, 27]. The organizational challenges faced by OSS development are considerable, because the project must deal not only with the software engineering problems faced by a development team but also with the complexity of coordinating the efforts of a geographically distributed base of volunteers working on the software [28]. Moreover, proprietary software projects pay experts to come up with high-quality solutions, which is not necessarily true of open source projects, which rely on the motivation and personal interests of individual developers [29].

An OSS community has a unique structure depending on the nature of the system and its member population. In general, the initial OSS developer maintains a lead role, and is responsible for the governance and coordination process [30]. The project leader, or the core team, usually partitions

the software development tasks into manageable modules, and has participants choose what to work on according to their interests. The OSS development model allows developers to integrate with non-technical members to form a broader, more transparent community [31]. In this context, users and developers coexist in a community where the software grows and expands based on personal needs and benefits [32]. These benefits include fun, reputation, learning, enjoyment, and peer recognition [33]. Membership in the community is fluid; current members can leave the community, and new members can join at any time [26]. Consequently, individual ownership of products is not apparent in OSS communities; instead, recognition of expertise is important. Community members believe in shared risks, shared rewards, and shared ownership [34].

2.2. Security Culture

Security culture is the set of values, shared by everyone in an organization, which determine how people are expected to think about and approach security, and is essential to an effective personnel and people security regime [15]. Many researchers have defined security culture and identified its importance in organizations. Dhillon [35] defined security culture as "the whole of human attributes, such as behaviors, attitudes, and values which may contribute to the protection of all kinds of information within a certain organization." Schlienger and Teufel [14] defined security culture as "all socio-cultural measures that support technical activity methods, so that information security becomes a natural aspect in the daily activity of every employee." Martins and Eloff [36] defined security culture as the perceptions, attitudes, and assumptions that are accepted and encouraged by employees in an organization in relation to information security. Ngo et al. [37] suggested that security culture is the accepted behavior and actions of employees and the organization as a whole, as well as how things are done in relation to information security. In short, security culture is the way our minds are programmed to create different patterns of thinking, feeling, and actions for providing the security process [38].

Security culture covers social, cultural, and ethical measures to improve the security-relevant behavior of organizational members, and is considered a subculture of organizational culture [2]. This culture is recognized in the security community and scientific literature as one of the most important foundations of organizational security. Security culture is based on the interaction of people with information assets, and the security behavior they exhibit within the context of the organizational culture in the organization [39]. Security culture involves identifying the security-related ideas, beliefs, and values of the group, which shape and guide security-related behaviors [40]. The importance of creating a security culture within organizational settings arises from the fact that the human dimension in information security is always considered the weakest link [14, 41, 42]. The results of numerous surveys suggest that people's attitudes and lack of awareness of security issues are among the most significant contributors to security incidents [43]. If appropriate security culture is neglected, individuals will not develop habitually secure behavior or take the initiative to make better decisions when problems arise. Therefore, the creation of security culture is necessary for effective management of information security.

3. The Research Framework

In this section, we elaborate characteristics of security culture that can be adopted in the context of OSS development. Based on a systematic review and a synthesis of relevant publications on security culture and information gathered from numerous pilot studies, we identified six dimensions of security culture: attitude, behavior, competency, subjective norms, governance, and communication.

3.1. Attitude

ASONAM '19: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining August 2019, Pages 863–870
https://doi.org/10.1145/3341161.3343520

Attitude is an important factor that influences humans' emotions (how you feel or what you believe) and behavior [44]. Specifically, attitude can also refer to the degree to which a person has a favorable or unfavorable feelings about an object [45]. The object can be an event, person, thing, place, idea, or activity. Other commonly used descriptions include behavior that is liked or disliked, desirable or undesirable, good or bad, or behavior that is viewed positively or negatively [46]. Chia [47] asserted that in a good security culture, individuals of the organization not only feel responsible but also have a sense of ownership about security. Unless they believe that security is important, people are unlikely to work securely, irrespective of how much they know about security requirements. Attitudes give a strong indication of individuals' disposition to act. For this study, attitude can be seen as OSS participants' feelings and emotions about the various activities that pertain to software security. Aspects include participants' belief (value) about security, responsibility for software security in the community, and positive thinking and perception of security requirements.

3.2 Behavior

The notion of behavior is based on what individuals do and relate to actual or intended activities [48]. According to Cox, Connolly, and Currall [49], human behavior is crucial in ensuring an efficient environment for information security. Essentially, security behavior is performed by individuals who are governed by instructions and requirements when using computer resources, but the way people think, believe, and subsequently, appreciate the organization of security affect how they behave [50]. Thus, security behavior can be seen as a function that frames the way that organizational actors collectively construct the meaning of different experiences of security tasks. The importance of participants' behavior in software security management cannot be ignored. In the context of software development, security behavior includes the use of security technologies, adoption of secure coding practices, and compliance with organizations' security policies. Subsequently, risk-taking is another important component of security behavior, when the people involved in the design and/or operation of a system fail to perceive some set of conditions that might arise and cause the security of the system to be compromised [51]. People adjust their risk-taking behavior toward their "comfortable" level of risk (i.e., their "secure" level of risk).

3.3 Competency

Competency is defined as the underlying human characteristic that distinctly affects superior job performance in real-life and context-specific situations [52]. This characteristic is the collection of underlying knowledge and skills, which potentially enables some individuals to meet demands more effectively than others [53]. Competency, therefore, provides the potential capability to be skilled in relation to a specific goal or job task. To improve job performance and satisfaction, competency has been widely used to match employees to jobs by matching the competencies of a person to the job requirements [54, 55], which causes individuals to feel that their behavior will not have any bad consequences. In the domain of software security, competency can be defined as software engineers' knowledge level and skills in protecting their software from a wide range of threats to software security, and with the ability to apply knowledge and skills productively (effectiveness). Having adequate competency regarding software security is a prerequisite to performing any software development task securely. Therefore, security competency may be regarded as an important factor to cultivate in security culture as the first line of defense in information security effectiveness.

3.4 Subjective Norms

A subjective norm is a person's belief about what people think about him or her should be done [56]. We recognize the term, subjective norms, as describing "directed normative relationships

between participants in the context of an organization" [57]. Norms are a powerful means of regulating interactions among autonomous agents [58]. What is perceived as normal behavior in social settings has a strong influence on what is considered acceptable behavior in an organization, and what is not [59], independent of what the rules or formal policies dictate. Individuals are influenced by both—messages about expectations and the observed behavior of others [60]. For security culture, subjective norms represent a combination of perceived expectations of relevant individuals or groups along with intentions to comply with security-related tasks. It regards what is right and wrong regarding information security, involvement in organizational communication processes, and awareness of security policies. If the group considers information security an important and serious problem, then it is more likely that the individuals within that group will value and follow the security policies. Conversely, if risk-taking is accepted within the group, then it is likely that greater risks will be taken. Failing to meet this expectation may incur a sanction against the offender. For example, members in OSS projects are often expected to follow a coding convention. Failure to adhere to this obligation may result in the code being rejected by the community. The level of intention toward a secure action is higher if the person has a positive attitude about and a subjective norm for the behavior [56].

3.5 Governance

Governance refers to the processes involved in developing and enforcing policies and norms for a given community or organization with the aim of structuring some set of activities [61]. Security governance is the means by which one controls and directs an organization's approach to security [61]. Security governance provides a framework in which the decisions made about security actions are aligned with the organization's overall business strategy and culture [62]. Thus, security governance is about decision making per se, which is concerned with setting directions, establishing standards and policies, and prioritizing investment and implementation. Effective security governance must provide mechanisms that enable managers to allocate expertise and responsibilities accordingly [62, 63]. It requires roles and responsivities of security tasks, defined policies, implementation, and oversight mechanisms. In growing and maintaining an OSS project, people, such as the core contributors/maintainers, leaders, and community managers, must develop guidelines for writing and documenting code, implementing rules about licensing and distribution, determining methods for evaluating contributions to the project, and providing venues for like-minded users to communicate and build working, trust-based relationships (e.g., Slack channels and discussion forums) [64].

3.6 Communication

Communication, in simple terms, can be considered an interactive process of sending and receiving messages among individuals, groups, and organizations, including some form of feedback [65]. DeVito [66] defined communication as an act: "Communication refers to the act, by one or more persons, of sending and receiving messages that are distorted by noise, occur within a context, have some effect, and provide some opportunity for feedback." Clear, open, effective communication can create a sense of transparency in the organization, which builds trust between levels of employees. As Adams and Sasse [67] pointed out, insufficient communication with individuals in the organization "causes them to construct their own model of possible security threats and the importance of security and these are often wildly inaccurate." It is imperative that security has an internal voice in the form of broadcasting channels, ensuring policies, procedures, and relevant breaking news items are universally and regularly communicated. In the present study, communication refers to the methods OSS participants use to communicate security information within a community, information transferring facilities, codification, and personalization information. Developers and users of an OSS

project do not all necessarily work on the project in proximity. They require electronic means of communications. Internet resources have the advantage of providing the community with an information infrastructure for sharing codification materials of software development in the form of hypertext, video, and software artifact content indexes or directories. Personalization communication has the inherent flexibility of transmitting tacit knowledge, and allowing for discussions and sharing interpretations that may lead to the development of new knowledge [68].

4. Research Methodology

This research adopted a quantitative approach to investigate the security culture in OSS communities. Quantitative research methods such as conducting surveys and the validation of research frameworks ad questionnaires have been greatly applied in the information security discipline [69, 70]. Organizations can use survey instruments to study information security behavior in general [71]. The use of an OSS participant survey was deemed appropriate in this study, as the survey enables clear, direct, and objective answers to the questions presented to the respondents. For the purpose of this study, a self-administered web-based survey was used to collect individual-level perception data from participants in OSS projects.

4.1 Instruments

The survey instrument used in this study was the outcome of an iterative process of checking and refinement. We developed a questionnaire based on the six dimensions defined in section 3. The primary measurement items and the corresponding questions are summarized in Table 1. Some survey questions were inspired by existing studies, while others were created specifically to suit the research context of this study. Each item in the questionnaire was measured on a five-point Likert scale ranging from strongly disagree (1) to strongly agree (5).

Table 1. Security culture dimensions and corresponding survey questions.

Dimension	Items	Question
Attitude	Value	• I believe software security is an important factor in achieving project success.
	Responsibility	• Software security is important to my work in software development.
	Positivity	• The requirements for software security do not interfere with my ability to get the job done.
Behavior	Acts	• I make the software components behave in a secure manner despite unexpected inputs or user actions.
	Compliance	• I adhere to the security principle and secure coding practices.
	Risk-Taking	• When I do my work, I assume that the software might be misused to reveal bugs that could be exploited maliciously.
Competency	Knowledge	• I know the principles and best practices for secure software development.
	Skills	• I can quickly identify specific coding errors or security vulnerabilities while examining the code base.
	Effectiveness	• I can apply methods or techniques adaptive to my project to prevent exploits against vulnerabilities.
Subjective Norms	Trust	• I believe the community can govern the security of the software products.
	Supportiveness	• Members of the community help each other solve security issues.
	Expectation	• I am encouraged to work securely by members of the community.
Governance	Expertise	• There is a security team (or at least one member) who deals with software security for the project.
	Policy	• The project has a general policy for software security management (vulnerability reporting, security testing, auditing, etc.).

ASONAM '19: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining August 2019, Pages 863–870
https://doi.org/10.1145/3341161.3343520

	Implementation	• The project has implemented secure coding practices (coding style, library, API, etc.).
Communication	Infrastructure	• There are dedicated communication channels (web page, mailing list, forum, etc.) related to security subjects in the community.
	Codification	• It is easy for me to find specific security information in the community.
	Personalization	• I know where to go for advice related to a software security issue in the community.

4.2 Data Collection

Samples for the empirical study were randomly collected from participants in OSS development projects, available on GitHub. GitHub is an online database of OSS projects. Users and potential contributors can access information about projects, and download current versions of the software being developed. As of June 2018, GitHub reported more than 30 million users [72] and 57 million repositories [73], making it the largest host of source code in the world [74].

The anonymous questionnaires were sent via e-mail to a list of OSS participants at the beginning of December 2017, and the data collection period lasted four months. Of the 321 questionnaires returned, 67 were excluded, because the respondents did not participate in an OSS community. In total, 254 respondent questionnaires were used for the final analysis. Table 2 shows demographic information about the sample, including gender, age, and seniority in the community, and the product categories of the projects.

5. Data Analysis

5.1 Respondent Demographics

Table 2 describes the general demographic information of the 254 respondents, in terms of gender, age, educational background. Nearly 90% of respondents were male, while there were only 9 female respondents. A large body of participants, that is 80%, was between 20 and 40 years old, and with a bachelor's degree (72.4%). Figure 1 shows the top 10 fields that the respondents' majors or anticipated majors. In the survey questionnaire, respondents were allowed to indicate more than one fields if applicable. As the figure indicates, about 65% of respondents have been educated in the academic disciplines of computer and information sciences. In terms of characteristics of OSS communities, the largest group of seniority in the community was 47.6% of the total, with between 3 and 5 years of experience, and the 254 respondents were from various product profiles and horizons.

Table 2. General demographic characteristics of the respondents (n = 254)

Item	Category	Frequency	Percentage
Gender	Male	228	89.8%
	Female	17	6.7%
	Prefer not to say	9	3.5%
Age	< 20	9	3.5%
	20–30	114	44.9%
	31–40	91	35.8%
	41–50	31	12.2%
	> 50	3	1.2%
	Prefer not to say	6	2.4%
Education	High school degree or equivalent (e.g. GED)	3	1.2%
	Associate degree (e.g. AA, AS)	12	4.7%
	Bachelor's degree (e.g. BA, BS)	184	72.4%

Master's degree (e.g. MA, MS, MEd)	53	20.9%
Professional degree (e.g. MD, DDS, DVM)	2	0.8%
Doctorate (e.g. PhD, EdD)	3	1.2%

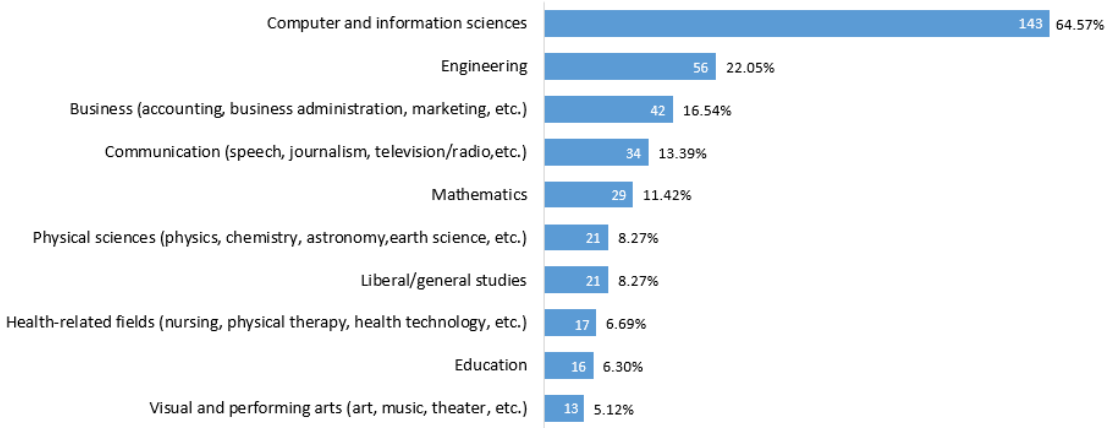

Figure 1. Top 10 fields that the respondents' majors or anticipated majors

Table 3. OSS Characteristics of the respondents (n = 254)

Item	Category	Frequency	Percentage
Seniority in the community	< 6 months	4	1.6%
	6 months to 1 year	34	13.4%
	2–3 years	95	37.4%
	3–5 years	121	47.6%
	> 5 years	98	38.6%
Product Category	Browser, Content management	30	11.8%
	Database, File system	29	11.4%
	Security, Firewall, Anti-virus, Encryption	24	9.4%
	Development framework	23	9.1%
	Education, eLearning, knowledge management	19	7.5%
	Communication (email, chatting, messaging)	19	7.5%
	Gaming, Entertainment	16	6.3%
	Healthcare	16	6.3%
	AI, Machine learning	13	5.1%
	Enterprise (finance, logistics, manufacturing)	12	4.7%
	Operating system	12	4.7%
	Word processing, Text editor	7	2.8%
	Retail & E-Commerce	7	2.8%
	Geospatial, Astronomy	5	2.0%
	Social media	4	1.6%
	Others	18	7.1%

5.2 An Overview of the Security Culture Scores

The mean scores of the security culture dimensions are plotted as a radar chart with six axes (Figure 2). As depicted in the chart, Attitude is the only dimension that reaches a mean value at the degree of 4.00. The respondents overwhelmingly reported a positive attitude toward software

security. More concerning, however, is the evidence that a significant minority of respondents were unwilling or unable to put this positive attitude into practice. The mean value of participant-reported behavior is 3.90, showing that the behavior of OSS participants is at a mild level of maturity, but still, on average, insecure. The mean score for Competency is 3.72, indicating that, on average, the respondent communities faced moderate to serious in equipping relevant security knowledge and skills. The Subjective Norms aspects were not well developed, as the mean score was 3.74. Notably, this study revealed there was very weak security governance to support security culture (mean = 3.28), suggesting that an insufficient complement to security expertise, as well as limited establishment and implementation of security policies. Last, Communication of security information in the OSS communities studied was, on average, very weak (mean = 3.28). Communication is the least developed dimension in security culture, as the mean is the lowest of all six dimensions.

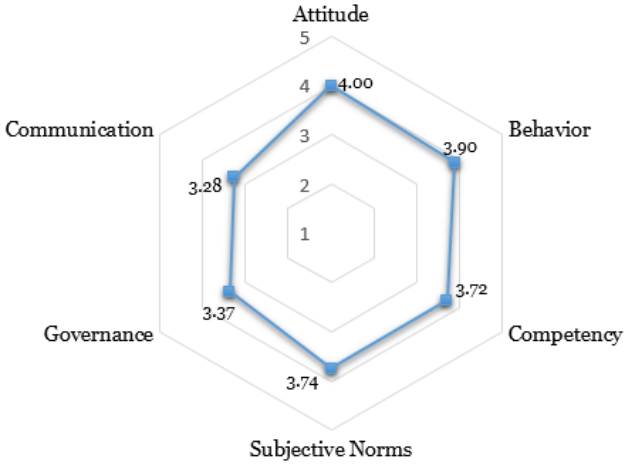

Figure 2. The mean score of security culture dimensions

5.3 Attitude

The results (Table 4) show that the vast majority of respondents (90%) held the value that security was an important factor in achieving project success. This could be a result of high-profile vulnerabilities and security incidents of OSS in recent years, which have generated a lot of adverse publicity for OSS development. Despite acknowledging the value of security for the project, only 56% of respondents agreed that software security was important to their work in the community, and a quarter of the survey population held a neutral position while answering this question. In addition, the mean score was statistically significantly low (3.67) in this dimension. OSS participants were still skeptical about the obligation to "build security in," as part of their jobs or roles. They had an inadequate understanding of how individual actions contribute to the security of the software system as a whole. In addition, we found that a third of respondents (disagree and neutral) felt security might interfere with their ability to get the job done. The result indicated that OSS participants viewed security as something that was necessary to their projects, but at times, also expressed their perception in the conflict between the security requirements and how they were used to writing code. Thus, the respondents shifted responsibility for software security to the community or public.

Table 4. Descriptive analysis of the Attitude dimension

Item	Frequency (Percentage)					Mean
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	

ASONAM '19: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining August 2019, Pages 863–870
https://doi.org/10.1145/3341161.3343520

Value	2 (1%)	8 (3%)	16 (6%)	76 (30%)	152 (60%)	4.45
Responsibility	11 (4%)	25 (10%)	76 (30%)	67 (26%)	75 (30%)	3.67
Positivity	12 (5%)	22 (9%)	54 (20%)	65 (26%)	101 (40%)	3.87

5.4 Behavior

We found that most respondents agreed about secure coding behavior. As the results reveal in Table 5, 70% of respondents agreed with the following statement about security acts: "I make the software components behave in a secure manner despite unexpected inputs or user actions." Similarly, nearly three out of four (74%) reported that they complied with secure coding policies in their work. However, in the two questions, the proportion of neutral responses was relatively high (20% and 17%, respectively). In addition, a minority group (nearly 10%) actively disagreed with the two statements about secure acts and compliance. The two groups of people (neutral and disagree) totaled nearly one-third of the survey population, which presents notable issues for OSS security. Most OSS participants might primarily focus on their immediate goals that usually involve functional requirements and performance, instead of security. In addition, the further result showed 38% of respondents performed risky behavior at a certain level in secure software development. They were likely to skip policies or bypass them to make their job easier, unaware of the potential damage, thinking that attackers would not be interested in their applications, or that their company was not big enough to be a target for attacks.

Table 5. Descriptive analysis of the Behavior dimension

Item	Frequency (Percentage)					Mean
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	
Acts	5 (2%)	19 (7%)	51 (20%)	94 (37%)	85 (33%)	3.93
Compliance	4 (2%)	18 (7%)	44 (17%)	96 (38%)	92 (36%)	4.00
Risk-Taking	6 (2%)	25 (10%)	66 (26%)	80 (31%)	77 (30%)	3.78

5.5 Competency

Worryingly, fewer than two-thirds of respondents, that is, 66% (in Table 6), said they had knowledge about general principles and best practices for secure software development, and only 65% said they had relevant skills for identifying specific security errors in code repositories. In addition, more than one-third of respondents (34%) did not agree with the following statement: "I can apply methods or techniques that adapt to my project to prevent exploits against vulnerabilities." The issues of OSS participants' lack of security competency mostly resulted from the fact that they come from various academic disciplines (as shown in Figure 1), and might not have formal college-level security training. Thus, a lot of confusion remained in participants' minds about what was secure code and what the project wanted. This confusion forced them to take risks based only on their personal experience, without fully considering the project's requirements and priorities.

Table 6. Descriptive analysis of the Competency dimension

Item	Frequency (Percentage)					Mean
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	
Knowledge	9 (4%)	23 (9%)	55 (22%)	102 (40%)	65 (26%)	3.75
Skills	7 (3%)	28 (11%)	54 (21%)	103 (41%)	62 (24%)	3.73
Effectiveness	12 (5%)	26 (10%)	48 (19%)	110 (43%)	58 (23%)	3.69

ASONAM '19: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining August 2019, Pages 863–870
https://doi.org/10.1145/3341161.3343520

5.6 Subjective Norms

The degree to which OSS participants trusted their community in the governance of software security was relatively high in the dimension of Subjective Norms. Nearly 80% of respondents conveyed their trust of their communities' security governance (Table 7). This result implies that OSS projects relied on the communities' management and control, and are conducted to a great degree to ensure the security protocols are carried out. However, only 65% agreed with the statement, "Members help each other solve security issues." Normative support for security tasks was not clearly perceived among OSS participants. In line with this, it perhaps is not surprising that only 51% thought that they received encouragement and expectation from their peers to work securely in OSS communities, while more than 20% did not agree that they had been influenced by other members regarding secure software development. The OSS participants did not perceive strong norms in their communities, something that could promote and reward behavior that serves the security quality of their software products.

Table 7. Descriptive analysis of the Subjective Norm dimension

Item	Frequency (Percentage)					Mean
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	
Trust	8 (3%)	19 (7%)	28 (11%)	95 (37%)	104 (41%)	4.06
Supportiveness	15 (6%)	23 (9%)	51 (20%)	84 (33%)	81 (32%)	3.76
Expectation	24 (9%)	31 (12%)	69 (27%)	76 (30%)	54 (21%)	3.41

5.7 Governance

Regarding the complement of security expertise in OSS communities, less than half of the survey population (46%, Table 8) clearly reported that there were security teams (or at least one person) dealing with software security in their communities, implying that a considerable portion of participant communities (54%) did not possess sufficient expertise to fully address complex security risks. OSS projects do not usually have the monetary resources in software security that companies producing proprietary software have. The people hosting the project have to do it in their spare time, making the level and motivation of security conduct questionable. This situation could also result in fewer security policies and a low implementation rate for secure practices in OSS development. In this study, security governance in OSS communities was either weak or problematic, as only half of the respondents (51%) agreed with the statements about the situations in the two measurement items, policies and implementation.

Table 8. Descriptive analysis of the Governance dimension

Item	Frequency (Percentage)					Mean
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	
Expertise	21 (8%)	35 (14%)	83 (33%)	68 (27%)	47 (19%)	3.33
Policies	25 (10%)	44 (17%)	56 (22%)	71 (28%)	58 (23%)	3.37
Implementation	21 (8%)	47 (19%)	57 (22%)	65 (26%)	64 (25%)	3.41

5.8 Communication

ASONAM '19: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining August 2019, Pages 863–870
https://doi.org/10.1145/3341161.3343520

Only 41% of respondents reported that dedicated communication channels related to security subjects existed in the community (Table 9). We found that only 35% of participants agreed with the statement, "It is easy for me to find specific security information in the community," and nearly 40% disagreed. OSS projects normally publish their own coding guidelines, a set of conventions (sometimes arbitrary) about how to write code for that project. However, OSS projects rarely address the security requirements in documentation to help drive the team to understand the prioritized security needs of the entire project. Thus, newcomers might feel that comprehending security requirements from exploring the website is hopeless; thus, they prefer to start with programming. In contrast to striving for codified security information, respondents felt at ease in asking for guidance or recommendations using available communication channels in their communities. Nearly 70% of respondents said they knew where to go for advice about security for their personal needs.

Table 9. Descriptive analysis of the Communication dimension

Item	Frequency (Percentage)					Mean
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	
Infrastructure	34 (13%)	65 (26%)	51 (20%)	57 (22%)	47 (19%)	3.07
Codification	27 (11%)	69 (27%)	68 (27%)	54 (21%)	36 (14%)	3.01
Personalization	17 (7%)	28 (11%)	34 (13%)	96 (38%)	79 (31%)	3.76

6. Discussion

We identified that a key inhibitor of OSS culture is the "it's not my responsibility" attitude. The survey data in table 4 showed that there was a strong reliance in participants' mindset on other methods (members, processes, and technology) to take care of software security. The lack of responsibility could occur when security is not considered part of a developer's everyday duties, or when developers expect security is handled elsewhere, such as by the core team or other community members. Given the openness and freedom of OSS, it is not surprising that OSS developers ease their workload by passing the responsibility for software security to others when possible. As long as developers are not held responsible for security tasks related to their code, they would rather spend their time on aspects for which they will be held responsible.

Developers want to write more secure code, but this might not be a priority for their work. They focus on contributing software code with the perception to become good application developers, but not necessarily security experts. Getting code out quickly, albeit with vulnerabilities that they discover and fix later, may be a better fit with their personal goals. As the analysis results shown for the measurement of Positivity (Table 4), it is not that OSS developers do not want to develop secure products, but they are more interested in delivering new functionality to increase the features of their software products. Furthermore, in the aspect of risk-taking, depicted in table 5, OSS participants might think that hackers are not interested in their applications, or that they are not famous enough to be a target for attacks. Thus, they see no perceived risk, and security efforts lack value. This perhaps indicates that OSS communities still have some way to go in ensuring that software security is high on the list of project priorities, and gets participants' attention in promoting positive security best practice.

In addition to the lack of incentives to focus on strengthening security, our study revealed a missed set of means in terms of security practice reinforcement, and demonstrates a clear knowledge gap that must be addressed by OSS communities. The data analysis in Table 6 indicated that two-thirds of respondents evaluated themselves as equipped with security competency, but the other one third did not. Still, OSS developers today are, most likely, unaware of the many ways they can introduce security problems into their code, and do not have the wherewithal to fix them when they

are found. In view of the gap in the skills and knowledge necessary for secure OSS development, the lack of appropriate security competencies is clear. OSS developers or other participants do not traditionally receive formal education or training about software security [75]. Programmers make security errors because they are unaware that their code will be attacked, and have no knowledge of methods by which their code can be secured [76]. Knowledge is not the motive for human information security behavior; however, the lack of knowledge is a barrier to developing the desired behavior [77]. We believe a closer look at security training also seems to be needed.

To effectively deal with security problems, OSS participants need greater awareness of specific errors in the context of their own development. Thus, security knowledge transfer within the OSS community is required to help them know about the threat to their own products, so they are motivated to respond. They also need to know what it means to write secure code, and how to find and correct the errors that cause the security flaws. Improving participants' competence in security can improve their confidence when a user is placed in the adverse condition of using the software [78]. It also makes the participants feel that their behavior will not have any bad consequences. With respondents' broadly positive attitude to security, OSS communities clearly need to place more focus on providing members with information related to security subjects, offering opportunities for learning and supporting self-development of security knowledge.

However, based on the analysis of subjective norms in Table 7, weak subjective norms support security culture in OSS communities. Only half of the respondents thought that they were encouraged by community members in terms of secure software development. Thus, OSS communities should enforce adherence to the mutual norm of security aspects, making cooperation between developers a goal, as well as part of the success of the project. Research indicated that in teams where security was part of the organizational culture and support for security tasks was available, individuals were more motivated to focus on security [79, 80]. This could be because they are confident performing their security tasks, especially when they feel support from peers. This behavior could result in a snowball effect, and lead to motivating more community members to recognize the importance of considering security as their peers do.

Developers do not like to feel exploited. If they believe that the other members of the project will not contribute equally, the norm of reciprocity is violated [81]. In the context of OSS development, peers' positive encouragement or expectation of secure coding behavior could increase developers' bonds with their teams, for example, with the feeling that they see the value of the community, and thus, perform the expected behavior for the team. As a result, rather than performing security tasks purely to follow an order or commission, the participants internalized such work, accepted it, and experienced willingness to act. This internalization of security has a statistically significant positive effect on persistence and performance [82]. We believe that OSS communities will greatly benefit from a security culture where an individual takes more responsibility for the security of the collective he or she is a part of, and is assured help if he or she encounters security crimes.

This study also exposes a problem that there was very weak security governance to support security culture. OSS communities differ from common enterprises in their coordination and organizational structure. The work is done on a voluntary basis, and there are less guidelines regarding time and intensity of work. Software security should not only be the domain of the core developers. On the one hand, those responsible for core development tasks must understand the importance of the scope of software function protection. On the other hand, participants must be informed the general process and methods to provide protection during the entire software development cycle. In this regard, OSS communities can utilize a security team or experts to define security requirements and best practices, help perform code reviews, and provides the necessary security knowledge for the software development staff. The team acts as the known point of escalation for security issues encountered by developers, if local champions cannot resolve them. It

is also responsible for sympathetically setting standards or practices, as developer members will have working knowledge of how security practices are best implemented.

To design functional and effective security governance, OSS projects must not only be responsible for security expertise coordination, but have the abilities to execute corresponding security policies. Security policies or guidelines have to be readily accessible or available to participants to ensure that they will not be ignored. Therefore, OSS communities must have the ability to convey the criticality of maintaining security to the whole project team. However, as this study reveals, Communication gains the lowest score among the six dimensions of security culture in OSS communities. To overcome the communication problems, OSS communities need to provide a communication strategy to ensure that participants have reached security information, the codification knowledge, when they need it, and importantly, are aware of where they can locate it. For example, specific security web pages can be included in the project website or repository, serving as an information clearinghouse. With just a glance, participants understand they need to pay attention, and take any recommended action immediately. Through this structural mechanism, the security knowledge gains valuable insights from the community, and further, facilitating discussion and decision making and sharpening personalization knowledge.

7. Limitations

Several limitations of this study should be noted. First, the survey relied heavily on self-reported data from participants about their perceptions and activities in secure OSS development. Respondents may have wanted to portray an ideal image of their security attitude, behavior, or knowledge within the workplace, rather than the reality. Although participants were not required to name their project and were given assurances of anonymity, respondents may still have reticent in reporting their actual behaviors. Second, the samples were chosen opportunistically from GitHub repositories, and the number of responses obtained from the survey was small compared with the enormous number of OSS projects and field workers today. Thus, there is a need for further research efforts focused on accumulating more evidence that is empirical, and data to break through the limitations. These efforts should improve the generalizability of this study to the entire OSS development phenomenon, by considering a larger number of responses covering a range of diverse OSS projects.

8. Conclusion

In this paper, we present a security cultural analysis in the context of OSS development. Measurements of security culture and the corresponding issues that must be addressed in OSS communities were defined and discussed. OSS is a core part of virtually all software applications today. The number of OSS projects has increased significantly over the last 5 years [9]. It is easier than it has ever been to create a new OSS project, as well as use other projects from other members of the community. The barrier to entry has decreased, so that a large number of enthusiastic amateur developers build a variety of apps and share their code in their spare time. This diversity of OSS projects is fantastic, but there is a shortage of developers entering the profession with software security expertise. With the increasing speed of development and sharing, convincing developers of the importance of security is challenging. Previously, OSS projects were focused on functionality and speed to market as their main goals. However, under pressure from a rising number of malicious threats and with tighter privacy protection laws coming into force, OSS communities have had to rethink their priorities. As the diversity of OSS products and projects increases, there will no longer be a single approach (e.g., practice, tool, heroic effort, or checklist) for achieving an optimal security culture suited to all communities. We believe that every technology developer has a responsibility to implement and participate in such a process. This is fundamental to achieving a security culture in a

software organization. Furthermore, OSS communities should establish rules and norms, roles, and methods, that is, to cultivate and maintain a culture that values positive security attitude and behaviors.

Funding: This research received no external funding.

Conflicts of Interest: The author declares that they have no conflict of interest.

Ethical Approval: All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki declaration and its later amendments or comparable ethical standards Informed consent was obtained from all individual participants included in the study.

References

[1] Humes, L.L. (2007), "Communities of Practice for Open Source Software", in Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives, IGI Global. pages 610-623.

[2] Godfrey, M.W. and Q. Tu (2000), "Evolution in open source software: A case study". in Software Maintenance, 2000. Proceedings. International Conference on. IEEE.

[3] Scacchi, W., et al. (2006), "Understanding free/open source software development processes". Software Process: Improvement and Practice, volume 11, issue 2, pages 95-105.

[4] Feller, J. and B. Fitzgerald (2002), "Understanding open source software development". volume: Addison-Wesley London.

[5] Feller, J., et al. (2006), "Developing open source software: a community-based analysis of research", in Social Inclusion: Societal and Organizational Implications for Information Systems, Springer. pages 261-278.

[6] NorthBridge, B., "2016 Future of Open Source Survey", Electronic document. http://www.northbridge.com/2016-future-open-source-survey-results. (Accessed on)

[7] BlackDuck Software, "2017 Open Source Security and Risk Analysis", Web: https://www.blackducksoftware.com/open-source-security-risk-analysis-2017. (Accessed on)

[8] Wen, S.-F. (2017), "Software Security in Open Source Development: A Systematic Literature Review". in Proceedings of the 21st Conference of Open Innovations Association FRUCT. Helsinki, Finland.

[9] snyk, "The state of open source security - 2019"; Available from: https://snyk.io/opensourcesecurity-2019/. (Accessed on 3-29-2019)

[10] Kowalski, S. (1994), "IT insecurity: a multi-discipline inquiry", PhD Thesis, Department of Computer and System Sciences, University of Stockholm and Royal Institute of Technology, Sweden. ISBN: 91-7153-207-2. pages.

[11] Scacchi, W. (2002), "Understanding the requirements for developing open source software systems". in IEE Proceedings--Software. IET.

[12] Fox, W.M. (1995), "Sociotechnical system principles and guidelines: past and present". The Journal of Applied Behavioral Science, volume 31, issue 1, pages 91-105.

[13] Zeitlyn, D.J.R.p. (2003), "Gift economies in the development of open source software: anthropological reflections". volume 32, issue 7, pages 1287-1291.

[14] Schlienger, T. and S. Teufel (2003), "Analyzing information security culture: increased trust by an appropriate information security culture". in 14th International Workshop on Database and Expert Systems Applications, 2003. Proceedings.: IEEE.

[15] Schlienger, T. and S. Teufel (2002), "Information security culture", in Security in the Information Society, Springer. pages 191-201.

[16] Chen, C.C., et al. (2008), "A cross-cultural investigation of situational information security awareness programs". volume 16, issue 4, pages 360-376.

[17] Ouchi, W.G.J.A.s.q. (1980), "Markets, bureaucracies, and clans". volume, issue, pages 129-141.

[18] Von Krogh, G., et al. (2012), "Carrots and rainbows: Motivation and social practice in open source software development". MIS quarterly, volume 36, issue 2, pages 649-676.

[19] Ouchi, W.G. (1979), "A conceptual framework for the design of organizational control mechanisms", in Readings in accounting for management control, Springer. pages 63-82.

[20] Kirsch, L.J.J.O.S. (1996), "The management of complex tasks in organizations: Controlling the systems development process". volume 7, issue 1, pages 1-21.

[21] Ducheneaut, N. (2005), "Socialization in an open source software community: A socio-technical analysis". Computer Supported Cooperative Work (CSCW), volume 14, issue 4, pages 323-368.

[22] Madey, G., V. Freeh, and R. Tynan (2002), "The open source software development phenomenon: An analysis based on social network theory". AMCIS 2002 Proceedings, volume, issue, pages 247.

[23] Gasser, L., et al. (2003), "Understanding continuous design in F/OSS projects". in In 16th. Intern. Conf. Software & Systems Engineering and their Applications. Citeseer.

[24] Cusumano, M.A. and R.W. Selby (1998), "Microsoft secrets: how the world's most powerful software company creates technology, shapes markets, and manages people". volume: Simon and Schuster.

[25] Olchi, W.G.J.A.o.M.J. (1978), "The transmission of control through organizational hierarchy". volume 21, issue 2, pages 173-192.

[26] Sharma, S., V. Sugumaran, and B.J.I.S.J. Rajagopalan (2002), "A framework for creating hybrid-open source software communities". volume 12, issue 1, pages 7-25.

[27] Schmidt, D.C. and A. Porter (2001), "Leveraging open-source communities to improve the quality & performance of open-source software". in Proceedings of the 1st Workshop on Open Source Software Engineering. Citeseer.

[28] Nelson, M., R. Sen, and C. Subramaniam (2006), "Understanding open source software: A research classification framework". Communications of the Association for Information Systems, volume 17, issue 1, pages 12.

[29] Vadalasetty, S.R.J.S.I. (2003), "Security concerns in using open source software for enterprise requirements". volume, issue, pages.

[30] Ye, Y. and K. Kishida (2003), "Toward an understanding of the motivation Open Source Software developers". in Proceedings of the 25th international conference on software engineering. IEEE Computer Society.

[31] Von KROGh, G. and S. Spaeth (2007), "The open source software phenomenon: Characteristics that promote research". The Journal of Strategic Information Systems, volume 16, issue 3, pages 236-253.

[32] Grodzinsky, F.S., et al. (2003), "Ethical issues in open source software". volume 1, issue 4, pages 193-205.

[33] Von Krogh, G. and E. Von Hippel (2003), "Special issue on open source software development", Elsevier. pages.

[34] Yamauchi, Y., et al. (2000), "Collaboration with Lean Media: how open-source software succeeds". in Proceedings of the 2000 ACM conference on Computer supported cooperative work. ACM.

[35] Dhillon, G. (1997), "Managing information system security". volume: Macmillan International Higher Education.

[36] Martins, A. and J. Elofe (2002), "Information security culture", in Security in the information society, Springer. pages 203-214.

[37] Ngo, L., W. Zhou, and M. Warren (2005), "Understanding Transition towards Information Security Culture Change". in AISM.

[38] Al Sabbagh, B. and S. Kowalski (2012), "Developing social metrics for security modeling the security culture of it workers individuals (case study)". in Communications, Computers and Applications (MIC-CCA), 2012 Mosharaka International Conference on. IEEE.

[39] Da Veiga, A. and J.H. Eloff (2010), "A framework and assessment instrument for information security culture". Computers & Security, volume 29, issue 2, pages 196-207.

[40] Ramachandran, S., S.V. Rao, and T. Goles (2008), "Information security cultures of four professions: a comparative study". in Hawaii International Conference on System Sciences, Proceedings of the 41st Annual. IEEE.

[41] Van Niekerk, J. and R. Von Solms (2005), "A holistic framework for the fostering of an information security sub-culture in organizations". in Issa.

[42] Martins, N., A. Da Veiga, and J.H.J.S.A.B.R. Eloff (2007), "Information security culture-validation of an assessment instrument". volume 11, issue 1, pages 147-166.

[43] Furnell, S.J.C. and Security (2007), "From the Editor-in-Chief: IFIP workshop-Information security culture". volume 26, issue 1, pages 35.

[44] Bulgurcu, B., H. Cavusoglu, and I.J.M.q. Benbasat (2010), "Information security policy compliance: an empirical study of rationality-based beliefs and information security awareness". volume 34, issue 3, pages 523-548.

[45] Woon, I.M. and A.J.I.J.o.H.-C.S. Kankanhalli (2007), "Investigation of IS professionals' intention to practise secure development of applications". volume 65, issue 1, pages 29-41.

[46] Ajzen, I., M.J.J.o.p. Fishbein, and S. Psychology (1973), "Attitudinal and normative variables as predictors of specific behavior". volume 27, issue 1, pages 41.

[47] Chia, P., S. Maynard, and A.J.P.o.P.J. Ruighaver (2002), "Understanding organizational security culture". volume 158, issue, pages.

[48] Kruger, H.A., W.D.J.c. Kearney, and security (2008), "Consensus ranking–An ICT security awareness case study". volume 27, issue 7-8, pages 254-259.

[49] Cox, A., S. Connolly, and J.J.V. Currall (2001), "Raising information security awareness in the academic setting". volume 31, issue 2, pages 11-16.

[50] Harnesk, D., J.J.I.M. Lindström, and C. Security (2011), "Shaping security behaviour through discipline and agility: Implications for information security management". volume 19, issue 4, pages 262-276.

[51] Dhillon, G. (2001), "Challenges in managing information security in the new millennium", in Information security management: Global challenges in the new millennium, IGI Global. pages 1-8.

[52] Bakanauskienė, I. and J. Martinkienė (2011), "Determining managerial competencies of management professionals". Management of Organizations: Systematic Research volume, issue 60, pages 29-43.

[53] Campion, M.A., et al. (2011), "Doing competencies well: Best practices in competency modeling". volume 64, issue 1, pages 225-262.

[54] Heinsman, H., et al. (2007), "Competencies through the eyes of psychologists: A closer look at assessing competencies". volume 15, issue 4, pages 412-427.

[55] Koeppen, K., et al. (2008), "Current issues in competence modeling and assessment". volume 216, issue 2, pages 61-73.

[56] Farrior, M.J.B.P.F. (2005), "Breakthrough strategies for engaging the public: Emerging trends in communications and social science". volume, issue, pages.

[57] Singh, M.P.J.A.T.o.I.S. and Technology (2013), "Norms as a basis for governing sociotechnical systems". volume 5, issue 1, pages 21.

[58] Avery, D., et al. (2016), "Externalization of software behavior by the mining of norms". in Proceedings of the 13th International Conference on Mining Software Repositories. ACM.

[59] Venkatesh, V. and S.A.J.M.q. Brown (2001), "A longitudinal investigation of personal computers in homes: adoption determinants and emerging challenges". volume, issue, pages 71-102.

[60] Sheeran, P. and S.J.J.o.A.S.P. Orbell (1999), "Augmenting the theory of planned behavior: roles for anticipated regret and descriptive norms 1". volume 29, issue 10, pages 2107-2142.

[61] Koh, K., et al. (2005), "Security Governance: Its Impact on Security Culture". in AISM.

[62] Dallas, S. and M.J.G.I. Bell (2004), "The need for IT governance: Now more than ever". volume, issue, pages.

[63] Weill, P. and R. Woodham (2002), "Don't just lead, govern: Implementing effective IT governance". volume, issue, pages.

[64] Sholler, D., "Community Call – Governance strategies for open source research software projects"; Available from: https://www.r-bloggers.com/community-call-governance-strategies-for-open-source-research-software-projects/. (Accessed on March 3, 2019)

[65] Koskosas, I. (2011), "Web Banking: A Security Management and Communications Approach". International Journal of Computer Science & Engineering Technology volume 2, issue 7, pages 146-154.

[66] DeVito, J.A. (2002), "Human communication", Boston: Allyn & Bacon. pages.

[67] Anne, A. and M.A.J.C.A. Sasse (1999), "Users are not the enemy". volume 42, issue 12, pages 40-46.

[68] Boh, W.F. (2007), "Mechanisms for sharing knowledge in project-based organizations". Information and organization, volume 17, issue 1, pages 27-58.

[69] Schlienger, T. and S. Teufel (2005), "Tool supported management of information security culture". in IFIP International Information Security Conference. Springer.

[70] Siponen, M., et al. (2014), "Employees' adherence to information security policies: An exploratory field study". volume 51, issue 2, pages 217-224.

[71] Berry, L.M. and J.P. Houston (1993), "Psychology at work: An introduction to industrial and organizational psychology". volume: Brown & Benchmark/Wm. C. Brown Publ.

[72] GitHub, "Github user search"; Available from: https://github.com/search?q=type:user&type=Users. (Accessed on March 3, 2019)

[73] GitHub, "Celebrating nine years of GitHub with an anniversary sale"; Available from: https://github.com/blog/2345-celebrating-nine-years-of-github-with-an-anniversary-sale. (Accessed on March 3, 2019)

[74] Gousios, G., et al. (2014), "Lean GHTorrent: GitHub data on demand". in Proceedings of the 11th working conference on mining software repositories. ACM.

[75] Wen, S.-F. (2018), "Learning secure programming in open source software communities: a socio-technical view". in Proceedings of the 6th International Conference on Information and Education Technology. ACM.

[76] Lawton, G.J.C. (2002), "Open source security: opportunity or oxymoron?". volume, issue 3, pages 18-21.

[77] Khan, B., et al. (2011), "Effectiveness of information security awareness methods based on psychological theories". volume 5, issue 26, pages 10862-10868.

[78] Shaw, R.S., et al. (2009), "The impact of information richness on information security awareness training effectiveness". volume 52, issue 1, pages 92-100.

[79] Merhi, M.I. and V. Midha (2012), "The impact of training and social norms on information security compliance: A pilot study". volume, issue, pages.

[80] Siponen, M.T.J.I.M. and C. Security (2000), "A conceptual foundation for organizational information security awareness". volume 8, issue 1, pages 31-41.

[81] Benbya, H. and N. Belbaly (2010), "Understanding developers' motives in open source projects: a multi-theoretical framework". volume, issue, pages.

[82] Ryan, R.M. and E.L.J.A.p. Deci (2000), "Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being". volume 55, issue 1, pages 68.