An Investigation of Security Conversations in Stack Overflow: Perceptions of Security and Community Involvement

An Investigation of Security Conversations in Stack Overflow

Perceptions of Security and Community Involvement

Tamara Lopez[†], Thein T. Tun[†], Arosha Bandara[†], Mark Levine[⋆], Bashar Nuseibeh[†‡], Helen Sharp[†]

[†] School of Computing & Communications, The Open University, Milton Keynes, UK, firstname.lastname@open.ac.uk
[⋆] Department of Psychology, University of Exeter, Exeter, UK, M.Levine@exeter.ac.uk
[‡] Lero - The Irish Software Research Centre, University of Limerick, Limerick, Ireland, bashar.nuseibeh@lero.ie

ABSTRACT

Developers turn to Stack Overflow and other on-line sources to find solutions to security problems, but little is known about how they engage with and guide one another in these environments or the perceptions of software security this may encourage. This study joins recent calls to understand more about how developers use Internet sources to solve security problems. Using qualitative methods, a set of questions within the security channel of Stack Overflow were selected and examined for themes. Preliminary findings reveal more about this community of practitioners: who are the askers and commenters, how security questions are asked and how developers frame technical information using social and experience-based perceptions of security.

CCS CONCEPTS

• **Security and privacy → Software security engineering**; • **Software and its engineering → Collaboration in software development**;

KEYWORDS

secure software development, collaborative environments, empirical studies

1 INTRODUCTION

Many real-world security vulnerabilities in software relate to a few known classes of attack such as code injection. A number of practices and technologies for detecting and preventing vulnerabilities in software are likewise established, such as input sanitisation and non-escaping strings. However, it is not clear why many professional software developers do not adopt these practices and technologies as a matter of course.

Security is, in part, a social phenomenon. Peer interaction, experience of security failures, and an awareness about the impact of security failures on people's well-being influence the decisions individuals make about whether or not to be secure in their personal lives [10] and on the job [17]. Social interaction is also key to developers' motivation to work; characteristics of the feedback received influence a developer's actions [20], and the nature of peer-to-peer interaction can satisfy or frustrate a developer's motivational needs [7].

The study reported here joins recent calls to understand more about how developers use guidance found on-line [3] and to identify methods for studying developers' security behavior [1]. We hypothesize that social factors are effective in motivating developers to write secure code and postulate that these factors will be evident within interactions undertaken between developers communicating in an on-line community setting.

This preliminary qualitative analysis examines on-line discussion within posts made in the security channel of Stack Overflow, asking:

> *How do developers talk to one another in Stack Overflow posts about security?*

2 BACKGROUND

Developers bring to the desk a degree of awareness about security formed on the job and in wider engagement in the world[12], but also must engage with it in practice [8]. The ideal within organisations is to achieve a "security culture", in which behaving securely is an implicit part of behavior [12]. A range of voices and skills contribute to this process: people with different levels of individual commitment to being secure[13] alongside those who could be described as "champions" [5].

Security has been described as a secondary concern to developers, one that must be prioritized and managed alongside other tasks developers need to complete [2]. Nonetheless, it is likely that developers are similar to other workers who exhibit a sense of responsibility toward security and their organizations [17]. Research in information security suggests that workers are motivated to be secure by social factors, that their attitudes and perceptions of security are influenced in part by information they receive from peers [17]. However, little is understood about what motivates developers to adopt secure practices [1].

Motivation significantly influences productivity and code quality in software development projects [11]. Successful developers are motivated to learn new technologies but are rarely motivated by reading documentation or studying manuals [6]. Instead, they engage in peer-to-peer interactions and observations, both of which have been found to bring about lasting cultural change within the wider software developer community [4, 27]. This phenomenon is evident, for example, in the widespread adoption of object-oriented technologies and agile development practices [18, 19]. Motivation is also an internal state, driven by individual characteristics and

SEAD 2018, May 2018, Gothenberg, Sweden

ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

intrinsic needs [16]. Any attempt to study it must therefore adopt the perspective of the individual.

Among on-line sources, Stack Overflow is reported to be the most popular source for self-learners [1]. Gamification features within the site encourage developers to participate, promising status and recognition within the on-line community, two known motivators of developers in workplace environments [6]. A number of other workplace motivators that might bring developers to Stack Overflow or drive them to engage have been identified, including a need for social connection, peer interaction, and identification with the task [22].

Recent studies within software engineering have looked at individual Stack Overflow channels, examining how knowledge is shared and formed within the R channel[26], and finding evidence for differences in use between this environment and other channels such as mailing lists [28]. Research looking at open-source discussion environments has established that social qualities of on-line interactions in technical fora are apparent [15] and can affect contributions made to open source projects [25].

Given the opportunity, it has been shown that developers turn to Stack Overflow to find solutions to security problems, however the code samples taken from security posts may not be as robust or correct as other information sources like books and vendor supplied documentation [2]. Another recent study has investigated other types of on-line sources of guidance about secure software development that are available to software developers, finding that developers must rely on diverse sources of information because there are gaps in coverage [3].

In this paper, focus is placed on the interactions between developers involved in the security channel of Stack Overflow, to understand both how developers talk about security and the nature of community involvement.

3 METHOD

This study is part of a larger program of research that is investigating ways to initiate and sustain secure software culture. Building upon frameworks of personal motivation and team culture [6, 24], the project has two aims:

- A1 Develop an empirically-grounded model of why and how non-specialist developers can be motivated to adopt secure coding practices and technologies into their software development practice.
- A2 Develop guidelines for creating and propagating a security culture across software teams.

To address these research aims, this project is conducting a series of ethnographic studies [23], within on-line developer communities and within professional settings.

The ethnographic method is used to study peoples' actions and accounts of actions. The method allows researchers to develop understanding about what practitioners working in socio-technical environments do and why they do it. The ethnographic stance allows researchers to consider experience from the perspective of the insider. Within this program of research, understanding developers' individual perspectives is key, as motivation is an internal state and the impact of peer-to-peer interaction is subtle rather than overt.

In taking the individual developer's perspective, the intention is not to assess the quality of the information developers provide to one another, but to understand how they engage with and guide one another in practice when dealing with issues related to security. As a starting point, this study examines how developers talk to one another in the comment streams for questions and answers given the "security" tag in Stack Overflow.

4 DATA SELECTION

The aim is to investigate the security community of practitioners within Stack Overflow. To identify a set of data within this community, gamification features of the site were leveraged. The site encourages participation through features that reward developers with points and badges when their posts are "voted up"[2] . Having a higher reputation grants access to different opportunities for contribution.

A search of questions within the meta help site and queries in the data explorer made it apparent that reputation and status are important to developers on the site. Given this, a decision was taken to select threads that are valuable to the community of developers, as indicated through scoring features.

Data associated with the twenty highest scored questions given the tag "security" were extracted from the Stack Exchange data explorer data dump of 14 January 2018. Queries were run within the hosted version of the data explorer[3]. Data were selected using the following criteria:

Evident need. The top-scored questions were chosen to form the set rather than top-scored answers to provide access to developers that showed evidence of a need to or interest in writing secure code, but with gaps in knowledge or understanding.

Non-specialists. The guiding aim within the overarching project is to understand more about the security practices of developers who are not specialists in security. For this reason, data was drawn from the main Stack Overflow site rather than the Information Security Stack Exchange site.

Stable data. In choosing the highest scored postings, the list of issues also broadly reflects the list of askers given for "All Time"[4]. This time period is conducive to analysis because these posts are less active than recent top rated posts. Because the list includes issues that are several years old, it is also possible to explore features of community development across a longer span of time.

5 ANALYSIS

Analysis began with the examination of a single question in the set (Q2 in Table 2). The question was examined, as were all answers to the question, commentary and a subset of user profiles. Cursory evidence was found within these materials to suggest that developers share more than technical information with each other about the security problem. This finding was used to initiate a literature review within motivation, software engineering and information

[1] See HackerRank's 2018 Developer Skills Report at: https://research.hackerrank.com/developer-skills/2018/

[2] https://stackoverflow.com/help/privileges/vote-up

[3] http://data.stackexchange.com/help

[4] 'security' top users: https://stackoverflow.com/tags/security/topusers

Table 1: Dimensions of security engagement on-line.

Dimension	Description	Related Papers
Security Advice & Assessment	Descriptions of the security problem. Includes technical guidance, advice and assessments of the response and attack.	[14, 17, 21]
Values & Attitudes	Statements that reveal individual attitudes and beliefs about the security problem or software development.	[6, 8, 10]
Community Involvement	Indications of sustained, patterned engagement between respondents; mindset; rules of practice.	[15, 19, 28]

security research. Analysis then turned to examination of the larger set of data.

At time of reporting, a set of 20 questions and 137 comments made about those questions have been cataloged and given preliminary codes. The profile pages within Stack Overflow and Stack Exchange for each developer have also been consulted. Following principles associated with thematic analysis[9], the coding process is inductive and iterative, and is supported by investigation of related literatures, shown in Table 1.

5.1 Characteristics of the Data

The data set includes details about the question asker, the provider of the accepted answer, and commenters. It also includes the set of questions, the accepted answers, and comments associated with each question and accepted answer. In this section, descriptive information is given about questions, the Askers and Commenters. Summary data about the questions and associated comments are in Table 2.

Questions. The top 20 security-related questions in Stack Overflow are dominated by issues about password handling, user authentication and SQL (see Figure 1). The questions were asked between August, 2008 (Q20) and December, 2012 (Q1). Three of the questions were last active in January of 2018; fourteen were active within 2017. The "Last Active" date reflects edits that are made for curatorial purposes, however, analysis has shown that commenting activity remains active within question and answer streams for years after the question is asked. All questions except three (Questions 3, 7 and 11) have at least one comment given about the question that was asked.

Askers. As Table 3 indicates in green, 6 Askers participated in the comment stream for the question they posted to the community. Within the set, no developer asked more than one question. Eight of the developers are also members of the Information Security Stack Exchange site. However, information within profiles shows that in most cases, Askers joined that channel after posting their question. Only five have been active within that site in the last year (indicated with Y/Y). While Askers remain active in Stack Overflow, they do not often participate in posts given the security tag.

Commenters. Comments were given by 109 users. Six commenters are also Askers, who provided additional detail or clarifying information about their question, sometimes more than once.

Figure 1: Tags assigned to each question; the tag "security" was removed.

Among the others, only 18 provided more than one comment within a stream (see Table 4). The remaining 85 left a single comment. This finding is not surprising, given the small size of the set in relation to the size of the channel, which had more than 40,000 posts at the time the data were collected.

Three of the developers who commented described themselves as security aware in their profile description; they are depicted within Table 4 in rows that are green. The activity of commenters, as might be expected, suggest a greater variation in security related activity. Several have answered or asked more than 40 questions given the tag "security"; 7 have written more than 20 comments on posts with the tag.

6 FINDINGS

The previous sections described qualitative methods employed in this study. In this section, findings are given that characterize engagement by developers within the set along two dimensions: security advice and assessment, and attitudes and values.

6.1 Security Advice and Assessment

In keeping with the requirements of the Stack Overflow channel, specific questions are asked, often about a particular technology. Within the comment stream the question is clarified and developed alongside broader conversation that includes perceptions about the security problem.

Table 2: Top 20 "security" questions by score - the dates of creation, last activity and number of comments (C).

Qu. (C)	Asked	Last Active	Qu. (C)	Asked	Last Active
Q1 (27)	Dec 2012	Dec 2017	Q11 (0)	Feb 2009	Feb 2017
Q2 (12)	Jan 2012	Jan 2018	Q12 (7)	Dec 2008	June 2017
Q3 (0)	July 2011	Oct 2015	Q13 (9)	Dec 2008	March 2017
Q4 (1)	July 2011	Oct 2017	Q14 (1)	Oct 2008	Nov 2016
Q5 (17)	April 2011	Jan 2018	Q15 (1)	Sept 2008	July 2017
Q6 (6)	Feb 2011	Nov 2017	Q16 (9)	Sept 2008	Dec 2017
Q7 (0)	Aug 2010	Nov 2017	Q17 (3)	Sept 2008	May 2017
Q8 (9)	June 2010	Nov 2015	Q18 (5)	17 Sep 2008	Sept 2017
Q9 (5)	April 2010	Nov 2017	Q19 (6)	28 Aug 2008	Jan 2018
Q10 (18)	Feb 2010	May 2017	Q20 (1)	11 Aug 2008	Dec 2017

To illustrate, a developer might ask a question about using data types securely. The question conveys a need for technical information, but is also framed within an assessment about the cost that undertaking the secure coding recommendation will have:

> *"Why does String pose a threat to security when it comes to passwords? It feels inconvenient to use char[]* ***(Q2)***

Such *appraisals* or assessments are made about recommended techniques and about the threats themselves.

> *"Can you cite the source of the suggestion? I can't really think of a reason for 'char[]' being more secure except maybe the most amateurish of threats."* ***(Q2.C1)***

In the following example the commenter suggests that the effort required to follow the recommendation isn't worth the developer's time, but is perhaps necessary to meet demands of the job:

> *"But still, it's a pointless tick in your employer's pointless tick list."* ***(Q2.C4)***

Appraisals are traded among developers alongside techniques and other kinds of information that comment on the efficacy of a response, circumstances and scenarios. Examples of these kinds of comments are given below.

Efficacy. In addition to assessments of the costs associated with security techniques, developers comment on the efficacy, or ability of a technique to avert a threat[17].

> *"The scheme can be made arbitrarily obfuscated to make patching difficult, but it's a certainty that the code can be patched to avoid any check."* ***(Q8.C8)***

> *"Md5 is now completely unsafe."* ***(Q12.C2)***

Table 3: Askers involvement in threads tagged with "security". Highlighted rows indicate Askers who participated in the comment stream.

Asker	Info Sec member?	Active last year?	"security" posts	"security" comments
As1	N	N	1	4
As2	Y	N	2	2
As3	Y	Y	7	1
As4	Y	N	6	2
As5	N	N	4	2
As6	Y	N	3	2
As7	N	N	3	0
As8	Y	Y	843	559
As9	N	N	2	1
As10	N	N	5	5
As11	N	N	2	0
As12	Y	Y	5	0
As13	Y	Y	15	3
As14	Y	Y	10	1
As15	N	N	4	0
As16	N	N	5	2
As17	N	N	1	1
As18	N	N	2	1
As19	N	N	1	0
As20	N	N	2	0

Circumstances. Many comments describe the context of application for a security response within an implementation, and in broader scenarios of use.

> *"If you are expecting an integer, use ctype_digit...IN most case you shoudl [sic] surround it with " " or ' '. and escape in variable matching quotes..."* ***(Q5.C11)***

> *" ... it's a common practice in forensics to ensure that no computers requiring investigation are turned off before their memory is dumped, in case they are utilizing software encryption ..."* ***(Q2.C8)***

6.2 Attitudes and Values

In this section, evidence is presented indicating that developers also qualify the advice they provide using statements that convey personal values and attitudes. This is notable, as these factors have been shown to influence security behaviour in the general public[10].

Comments frequently indicate evidence of a security principle that is being recommended or followed. Principles do not always reflect formal sources of information. They can also be framed as "should" statements or received wisdom.

> *"It sounds like you may be using 'security by obscurity' if your payment processing scheme relies on the operation of the client remaining secret..."* ***(Q1.C1)***
>
> *"...you should never design or even implement any cryptography parts yourself and instead use openly available standards."* ***(Q1.C7)***

Other statements make appeals to trust; in the case that follows, that is trust in experts, or in the second, faith in the "big" guys.

> *"How many times must security experts repeat this before it finally sinks in: **The most common and most serious security threats are always INTERNAL.**"* ***(Q10.C11)***
>
> *"Google itself tried to tackle on piracy by saving encrypted apks in '/mnt/asec', starting in JB."* ***(Q1.C17)***

There are a few examples in this set of appeals to fear. In the first example below, an expression of responsibility toward the Asker is also given:

> *"On the other hand the OP may now have the false security that their native code is not (easily) readable."* ***(C1.14)***
>
> *"At best, rejecting such a name for technical reasons is a band-aid; at worst, it's false security."* ***(Q5.C15)***

As might be expected, developers note responsibilities toward users, requirements, and in this example, policy:

> *"Note that...there may be regulatory and legal policy that affects your app and could potentially expose you to severe penalties: see PCI compliance, starting with http://www.pcicomplianceguide.org/pcifaqs.php"* ***(Q1.C10)***

Finally, comments often also invoke the notion of a developer's responsibility to the code; they appeal to developer pride and capability.

> *"There is nothing stopping you from implement [sic] a secure system. Use password resets, if they don't know their password then they should make a new one"* ***(Q10.C5)***

It is important to note that in this case, the developer asserts that the responsibility to write good and secure code is more important than another non-functional requirement, user experience. This suggests that though security may be a secondary concern [2], it is actively managed among other competing demands.

7 DISCUSSION

Participants in security posts frame questions and advice about security with assessments and attitudes that speak both to the response and the attack [17], illuminating the broad landscape a developer must survey in making security decisions. In this section, reflections are made about the questions, interactions between developers and the development of community in the Stack Overflow security channel.

7.1 Asking Security Questions

> *"Any help is appreciated."* ***(Q19)***

There are many discussions around the effective method for storing passwords, which shows some awareness about the security risks of storing plain-text passwords. Is hashing or encrypting more appropriate for storing passwords? If salt is used, is it OK to store it together with the password hash? There are suggestions for using tools such as bcrypt. Fundamentally, there appears to be a lack of standard tool support for storing passwords in popular database engines.

SQL injection attacks are also the subject of several popular questions. Here the questions tend to be more specific to a particular programming language, or the effectiveness of a particular input sanitization method. OAuth is another topic of popular interest: what it means, how it works and how to use it are widely-read threads.

There are very few questions about cryptographic primitives, protocols or issues related to network security. Given the nature of the Stack Overflow community, this is not surprising. Although security issues are quite specific, there is a lot of diversity in questions primarily because the programming/scripting languages (Java, PHP, JavaScript etc.) and the operating systems involved (iOS, Android, Windows etc.) are numerous.

Table 4: Top commenters by number. Highlighted rows indicate developers who describe themselves as security aware.

Question	Comments	Info Sec member?	Active last year?	"security" posts	"security" comments
Q1	4	Y	Y	1	4
Q1	3	Y	Y	0	3
Q1	2	N	N	0	2
Q1	2	Y	Y	17	21
Q2	2	Y	Y	0	2
Q2	2	N	N	0	4
Q5	2	N	N	0	2
Q5	2	N	N	1	3
Q5	2	Y	Y	10	24
Q5	2	Y	Y	13	28
Q5	2	Y	Y	41	34
Q8	2	Y	Y	47	22
Q10	3	Y	Y	69	63
Q10/As8	2	Y	Y	843	559
Q10	2	N	N	5	2
Q10	2	Y	Y	10	10
Q10	2	Y	Y	5	5
Q16	2	Y	Y	14	14

7.2 From Questions to Conversations

> *"There might be an even better way..."* ***(Q5.C13)***

The sense is given that questions within the security channel belong to the community, not the asker. This is perhaps most clearly demonstrated by an Asker who enters into the comment stream some four years after the post was created, noting that his question was changed after he wrote it(Q6).

The developers who participate in the questions and comment streams are aware that their problems have a security aspect. Sometimes, the information traded is precise and includes references to established security principles. However, the way awareness is signaled within questions indicates that awareness often grows - out of "suggestions" given by other developers, or things that have been "seen around".

The dynamic nature of security problems and the impact of time on the development of knowledge is well established: knowledge must be updated, people must constantly react and respond to changes in the threat environment [6].

This is exemplified within the set studied here by issues in which a commenter points out that something in the accepted answer is now "completely unsafe" or that a different answer offers a "better approach" than the answer that was accepted. Such comments are sometimes added years after questions are asked.

Most of the posts remain active, and while it is clear that some activity is curatorial or may be done to boost individual profiles, other activity is relevant to the security problem. Developers drop in on these posts, to add new information, to update old information and, it can be surmised, to learn.The set includes firm evidence of different kinds of engagement within the comment streams. As might be expected given the nature of the forum, people ask for help in understanding a technical detail, and other commenters take the time to provide it. A large number of the comments provide factual information about techniques, suggesting packages or tools, or providing links to external and internal sites.

However, in looking beyond questions or individual comments toward exchanges between developers, evidence is emerging of other kinds of involvement. Developers frame technical advice and guidance within values and attitudes like responsibility, trust, and fear. The link between rhetorical devices and rationale has been noted in studies of other on-line fora[15] and these factors have been shown to determine or influence security behavior in users[10]. While their function in this forum has yet to be established, their presence indicates that community members provide more than technical information about secure coding practice to each other.

7.3 Community Involvement

"You and I know it, but not everyone who reads this will understand..." ***(Q16.C4)***

The commenters with the most activity in the security channel are also members of the Information Security site, suggesting that developers with an interest in security are active on both sites, and may have a higher degree of security awareness and knowledge than Askers. Taken together, activity within the group suggests that the developers are primarily non-specialists but exhibit a range of levels of activity within the security channel and the Information Security Stack Exchange site.

The majority of these developers are very likely also "ordinary insiders". Experts use analytic approaches, while non-specialists base their perceptions on experiential analyses, that is, on feelings about the risk associated with their actions. The way in which technical information is framed within assessments and attitudes suggests that these developers may be influenced by and depend upon their interactions with peers in the process of making decisions that will impact the security of their code [17].

Some askers and commenters stand out. Among the askers, only one describes himself as a security aware developer (Table 4, AS8). At some point in time this developer also became a top 20 Answerer, and this is also true of two of the commenters. Other commenters are active within the security channel but don't describe themselves as security aware or appear within the ranking features of Stack Overflow.

The notion of identity and roles that developers play for one another within communities of practitioners is of particular interest for this study and within the parent research project. Later stages of this work will investigate how security identities develop.

8 LIMITATIONS AND NEXT STEPS

Selection drew upon the gamification features of stack overflow; the set was identified out of those posts that had the highest score. Cursory examination suggests differences in other groups of posts. For example, the top 20 questions for the past thirty days appear to be more narrowly focused on technical approaches. Likewise, only questions have been examined in any detail. Understanding will certainly enlarge once we examine answer streams.

Currently, understanding about secure coding practice among practitioners extends only to developers who write on Stack Overflow. Many developers report using the site in their daily work, but don't ask questions or participate in discussions[5].

To address these limitations in the coming months, future research will:

- Broaden investigation to include a fuller reporting on this set of data to include answers and answer comment streams.
- Contextualise this data against different sets within Stack Overflow (e.g. top-rated posts of the last 30 days, wiki posts, negative score posts).
- Identify and describe the community of security practice within Stack Overflow.
- Explore connections between the Stack Overflow security community of practitioners and the work practices of professional developers in organisational settings.

9 CONCLUSION

This report describes a preliminary examination of how a particular community of practitioners conceives of and manages issues related to security. Findings suggest that security conversations within Stack Overflow are rich, providing evidence about perceptions of security, developer values and community involvement. In short, there are indications that Stack Overflow is influential in shaping perceptions and values of non-specialist developers where security is concerned. It is reasonable to assume that this may also translate into changes in behavior.

ACKNOWLEDGMENTS

The work presented in this paper was supported by the UK National Cyber Security Centre, as part of the Developer-Centred Security

[5]For an overview of participation types, see: https://insights.stackoverflow.com/survey/2017#community

research programme, the ERC Advanced Grant 291652 (ASAP) and SFI grant 13/RC/2094.

REFERENCES

[1] Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle L. Mazurek, and Christian Stransky. 2016. You Get Where You're Looking For: The Impact Of Information Sources On Code Security. In *IEEE Symposium on Security and Privacy*. IEEE, 289–305.

[2] Yasemin Acar, Sascha Fahl, and Michelle L. Mazurek. 2016. You are not your developer, either: A research agenda for usable security and privacy research beyond end users. In *Cybersecurity Development (SecDev), IEEE*. IEEE, 3–8.

[3] Yasemin Acar, Christian Stransky, Dominik Wermke, Charles Weir, Michelle L. Mazurek, and Sascha Fahl. 2017. Developers Need Support, Too: A Survey of Security Advice for Software Developers. In *Cybersecurity Development (SecDev), 2017 IEEE*. IEEE, 22–26.

[4] Giovanni Asproni. 2004. Motivation, teamwork, and agile development. *Agile Times* 4, 1 (2004), 8–15.

[5] Ingolf Becker, Simon Parkin, and M. Angela Sasse. 2017. Finding Security Champions in Blends of Organisational Culture. *Proc. USEC* 11 (2017).

[6] Sarah Beecham, Nathan Baddoo, Tracy Hall, Hugh Robinson, and Helen Sharp. 2008. Motivation in Software Engineering: A systematic literature review. *Information and software technology* 50, 9-10 (2008), 860–878.

[7] Sarah Beecham, Helen Sharp, Nathan Baddoo, Tracy Hall, and Hugh Robinson. 2007. Does the XP environment meet the motivational needs of the software developer? An empirical study. In *Agile Conference (AGILE)*. IEEE, 37–49.

[8] Marcus Beyer, Sarah Ahmed, Katja Doerlemann, Simon Arnell, Simon Parkin, M. A. Sasse, and Neil Passingham. 2015. Awareness is only the first step. *A framework for progressive engagement of staff in cyber security, Hewlett Packard, Business white paper* (2015).

[9] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative research in psychology* 3, 2 (2006), 77–101.

[10] Lynne Coventry, Pamela Briggs, John Blythe, and Minh Tran. 2014. Using behavioural insights to improve the public's use of cyber security best practices. *Government Office for Science* (2014).

[11] S. A. Frangos. 1998. Motivated humans for reliable software products. *Microprocessors and Microsystems* 21, 10 (1998), 605–610.

[12] Steven Furnell and Anish Rajendran. 2012. Understanding the influences on information security behaviour. *Computer Fraud & Security* 2012, 3 (2012), 12–15.

[13] Steven Furnell and Kerry-Lynn Thomson. 2009-02-01. From culture to disobedience: Recognising the varying user acceptance of IT security. 2009, 2 (2009-02-01), 5–10.

[14] Allen C. Johnston and Merrill Warkentin. 2010. Fear appeals and information security behaviors: an empirical study. *MIS quarterly* (2010), 549–566.

[15] Andrew J. Ko and Parmit K. Chilana. 2011. Design, discussion, and dissent in open bug reports. In *Proceedings of the 2011 iConference*. ACM, 106–113.

[16] Daniel H. Pink. 2011. *Drive: The surprising truth about what motivates us*. Penguin.

[17] Clay Posey, Tom L. Roberts, Paul Benjamin Lowry, and Ross T. Hightower. 2014. Bridging the divide: A qualitative comparison of information security thought patterns between information security professionals and ordinary organizational insiders. *Information & Management* 51, 5 (2014), 551–567.

[18] Hugh Robinson and Helen Sharp. 2005. The social side of technical practices. In *International Conference on Extreme Programming and Agile Processes in Software Engineering*. Springer, 100–108.

[19] Hugh Robinson and Helen Sharp. 2009. The emergence of object-oriented technology: the role of community. *Behaviour & Information Technology* 28, 3 (2009), 211–222.

[20] Rien Sach. 2013. *The Impact of Feedback on the Motivation of Software Engineers*. PhD Thesis. The Open University.

[21] Bruce Schneier. 1999. Attack trees. *Dr. Dobb's Journal* 24, 12 (1999), 21–29.

[22] Helen Sharp, Nathan Baddoo, Sarah Beecham, Tracy Hall, and Hugh Robinson. 2009. Models of motivation in software engineering. 51, 1 (2009), 219–233. http://www.sciencedirect.com/science/article/pii/S0950584908000827

[23] H. Sharp, Y. Dittrich, and C. R. B. de Souza. 2016-08. The Role of Ethnographic Studies in Empirical Software Engineering. 42, 8 (2016-08), 786–804.

[24] Helen Sharp, Hugh Robinson, and Mark Woodman. 2000. Software engineering: community and culture. *IEEE Software* 17, 1 (2000), 40–47.

[25] Igor Steinmacher, Marco Aurelio Graciotto Silva, Marco Aurelio Gerosa, and David F. Redmiles. 2015. A systematic literature review on the barriers faced by newcomers to open source software projects. *Information and Software Technology* 59 (2015), 67–85.

[26] Bogdan Vasilescu, Alexander Serebrenik, Prem Devanbu, and Vladimir Filkov. 2014. How social Q&A sites are changing knowledge sharing in open source software communities. In *17th ACM conference on Computer supported cooperative work & social computing*. ACM, 342–354.

[27] Jim Witschey, Olga Zielinska, Allaire Welk, Emerson Murphy-Hill, Chris Mayhorn, and Thomas Zimmermann. 2015. Quantifying developers' adoption of security tools. In *10th Joint Meeting on Foundations of Software Engineering*. ACM, 260–271.

[28] Alexey Zagalsky, Carlos Gómez Teshima, Daniel M. German, Margaret-Anne Storey, and Germán Poo-Caamaño. 2016. How the R community creates and curates knowledge: a comparative study of stack overflow and mailing lists. In *13th International Conference on Mining Software Repositories*. ACM, 441–451.