VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui

School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@ecs.vuw.ac.nz

Why Do Programmers Do What They Do? A Theory of Influences on Security Practices

Lavanya Sajwan

Supervisors: James Noble, Craig Anslow

Submitted in partial fulfilment of the requirements for Bachelor of Engineering with Honours in Software Engineering.

Abstract

Technologies are continually adapting to match ever-changing trends, and as this occurs, new vulnerabilities are exploited by malignant attackers and can cause significant economic damage to companies. Programmers are therefore repeatedly having to expand their knowledge and skills to protect software. Programmers do make mistakes, and this is why we must understand the thinking behind their decisions and influences to interpret how they implement and adopt security practices. Understanding these decisions can help inform design and educational decisions around improving programming language security. This report will cover the full progress of the project "Why Do Programmers Do What They Do? A Theory of Influences on Security Practices"

Acknowledgements

I would like to thank my supervisors James Noble and Craig Anslow for their ongoing support and direction as I have completed this project.

To dad and mum, thanks for harassing friends to potentially participate in this study. With the addition of my brother; thank you for having no idea about what I have been doing all year long, but also for leaving me fairly alone for these last two weeks as I recluse into a hole where I write, write, write and write.

And to Janaye; a massive thanks for reading over this report and being my grammar and punctuation whizz since high school.

Contents

Figures

Chapter 1

Introduction

As software is now ubiquitous across many industries, it is impossible to not have a presence in the tech sphere. Consequently, software security has become so significant, programmers have to ensure that the security processes that they implement are resilient to any attacks. Lack of attack prevention can cause leakage of sensitive information, major economic damage and danger to massive numbers of users and employees. Consequently, this opens businesses, clients and end-users to exploitation by external bodies. Unfortunately, every day, we hear of compromised organisations [1]. Last year New Zealand experienced a significant security breach within the Tū Ora Compass Health (Tū Ora) Primary Health Organisation (PHO) [2]. Tū Ora is one of the largest PHO's in the country, and it governs the greater Wellington region [2]. Personal information of up to one million New Zealander's was exposed, and the effects of this are ongoing [2]. Costs have been high in attempts to mitigate any effects to the public. Dedicated call centres have been set-up as well as dedicated mental health lines [3, 2]. Regular updates are released in order to maintain communication transparency and assurance quality measures are ongoing [3, 2].

Qualitative research is often neglected and overlooked in favour of more quantitative reasoning and technical traits such as the security method used or the programmers task-completion rate [4]. Programmers provide a human aspect to a technical solution and there should therefore be a shift towards understanding the more background 'soft' processes that occur when making decisions; why are the choices made based on past influences, and how do they affect the programmers work in the present?

Exploring this topic is essential as it allows for a more comprehensive understanding of how and why programmers think the way they do, and of the human and social aspects of Software Engineering [5]. We want to understand what solutions programmers are using to implement in their security practices, if at all. The findings from this study can support programmers in terms of education and the better design of security methods in programming [6] that have an emphasis on usability. The findings of this project can also be used to identify what security methods programmers find as beneficial in their programming. This will allow programmers to complete their work to a high standard, by adhering to proper security protocols, thus overall making their work of a higher value both in a secure and professional sense.

Beyond the research aspect, when security and privacy issues do occur in real-world scenarios, programmers are blamed first as it is the faults in their implementation which allow for the exploitation of vulnerabilities [7]. Programmers do make mistakes, which is why they need the support to make better security decisions, and this support is currently

lacking in the industry [7]. Education is limited past initial acceptance within organisations, and often programmers have a blasé attitude regarding security, instead expecting other teams to fix the issue [8]. Furthermore, security mechanisms often have increased complexity, which makes them challenging to understand and to then use [9].

This project will investigate how programmers implement and adopt security practices in the work they do in order to develop an understanding of what influences and impact decisions surrounding their technical work. This project uses Grounded Theory, a method which aims to establish a theory when there is none [10], and it is commonly used for data analysis. Interviews will take place to collect the data, to then analyse and present a theory on standard security practices in the professional workplace. This project builds upon works by Hala Assal, Charles Weir and Nathan Newton [6, 1, 11].

Participants will be recruited by posting on tech-related groups (i.e. From OWASP Meetup Group and LinkedIn) and mailing lists and also using my own and my supervisor contacts. At this point, semi-structured interviewing can take place with 10-20 interested individuals on the topic of their security practices while programming. Practices include; libraries, frameworks, protocols and specific languages. These people will all be programmers in New Zealand that are in varying stages of their careers and career paths to allow for a broader range of responses and a case study relevant to New Zealand. Examples of appropriate job titles include;

- DevOps Engineer
- Software Developer
- Software Engineer
- Front-end security developer
- Database administrator
- Tester
- Security Architect
- Security Consultant.

This project will lead to a new, more in-depth understanding of the psychology of the decisions made by programmers which can support security education programmes in tertiary education providers and within the workplace. The research undertaken by this project could lead to future qualitative research on another under-developed topic on why programmers do what they do. Paired with this research, future studies could help build a profile of a programmer and their thought processes. Data collected could also be the foundation that allows a programmer to build tools that help other programmers implement proper security practices; a Grammarly for security. It can also help the further development of existing static analysis tools such as Infer developed by Facebook, Tricorder used by Google, Coverity and Raygun [12, 13, 14, 15]. These tools all work by providing security detection modules and are used for quality assurance and security.

1.1 COVID-19 Effect

In trimester one, during the thick of COVID-19 in New Zealand, there were no issues with this project. However, as the country went back to normal, I found it increasingly more

difficult to schedule-in time with participants as the move back for them was disruptive. They were doing half-and-half working from home and in-the-office and also dealing with their transitions back to ”normal”. Therefore, there were many non-respondents, no-shows and general lack-of-communication from participants due to external reasons related to the pandemic. This pushed back the timeline I followed by weeks.

I also found that this semester the workload in other courses was a lot more than last semester even with the supposed reductions due to the pandemic. This made it hard to dedicate the much needed time into this project. I also burned out during the last three weeks.

Mentally the second-half of semester has been difficult. With the borders closed no one has been able to leave New Zealand and there have been some extended family members who have died in India. Despite the separation, Indian family units are quite close so that has definitely been a shock.

Chapter 2

Background

Conceptually, this project aims to gather data in order to ultimately form a theory on the influences of security practices in New Zealand. Therefore, It is essential to know enough information about the topic of security so that interviews with professional programmers are conducted in a knowledgeable manner. This chapter will include some background on what security actually is, the importance of information security practices, widely used security practices and summaries on existing research on similar subject matter.

2.1 What is Security?

Three objectives define security; Confidentiality, Integrity and Availability, most commonly referred to as the CIA triad. The triad measures, controls and protects information assets which include hardware, software, firmware, information data and telecommunications **INSERT NIST CITATION**. A now common synonym for security is the term coined; cybersecurity **INSERT NIST CITATION**.

Confidentiality is the act of maintaining protection of information from threat agents **INSERT BOOK CITATION**. This term also encapsulates privacy where "individuals control or influence what information related to them may be collected and stored and by whom and to whom that information may be disclosed **INSERT BOOK CITATION**. Integrity is the maintenance of data, so that information and programs are only changed when needed by authorised bodies **INSERT BOOK CITATION**. It also covers system integrity where "a system performs its intended function in an unimpaired manner, free from deliberate or inadvertent unauthorised manipulation of the system" **INSERT BOOK CITATION**. Availability is the act of ensuring reliable access to information to authorised users **INSERT BOOK CITATION**. Any breach of these principles can result in significant adverse impacts on reputation, health and safety, service delivery as well as finances **INSERT IMPACTS CITATION**

2.2 Information Security in New Zealand

A study run by Aura Security showed a 10% increase in cyber-attacks on New Zealand businesses between the years of 2018-2019 [16]. In NZ, the rise is attributed to the digitisation of day-to-day way-of-life; as new technologies continually develop, there is an urgency to deploy products. There is a lot of pressure on programmers to finalise these products, and the deviation of attention to the finished output means that there are many new threats to security [17]. Malicious attackers are also becoming more sophisticated. Individual threat

attackers now seem to have the same knowledge and resources as nation-backed threat actors [17].

Areas of improvement identified by the New Zealand government are [17]:

1. **Cyber security-aware and active citizens:** Increased regular awareness campaigns and education opportunities for the public in regards to best personal security practices.
2. **Strong and capable cybersecurity workforce and ecosystem:** Increased promotion and support of the development of the cyber industry in New Zealand.
3. **Internationally active:** Detect and prevent any breaches as well as proactively maintaining international relationships regarding information security and participating in any rule reforms.
4. **Resilient and responsive New Zealand:** Supporting infrastructure, businesses, charity organisations, community organisations, individuals in improving security capabilities and resilience.
5. **Proactively tackle cybercrime:** Increasing support to impacted parties, preventing and encouraging reporting of any cybercrimes.

These five principles are planned to continue to improve upon till 2023 [17]. It is expected that aspects of these will influence the security practices that programmers use in NZ industry.

2.3 Secure Programming Practices

Secure refers to protecting and deterring any security incidents due to vulnerabilities in program code **(CITE THIS)**. Without a focus on security while programming, there is a lack of confidence in the outputted security, resulting in the lack of use and waste of time, effort and money. Using unsecured programs poses significant adverse impacts to businesses and individuals.

The Open Web Application Security Project (OWASP) has outlined a checklist to ensure that code is secure. They have provided this document which spans various languages and technologies as the non-profit foundation aims to improve the security across all software. It states that it is "much less expensive to build software than to correct security issues" [18].

Significant items from the checklist include [18]:

- **Input Validation:** This is the act of validating that all inputs are trusted. Examples of inputs are data from databases, file streams, as well as client-provided data. Anything classed as 'untrusted', should fail and result in a rejection. The application should have one input validation routine, a specified standard character set and all validation should occur on a preset trusted system.
- **Output Encoding:** Data sent back the client needs to be output encoded. To make this secure, a started test routine has to be utilised, and similar to input validation; all encoding must occur on a preset trusted system.

- **Authentication and Password Management:** This is verifying whether a user is allowed to act. Best practice is restricting all resources except the ones intended to be public. All authentication controls should be able to fail in order to maintain security, and passwords need hashing. In order to pass, authentication has to occur first, and every detail has to match the protected records.
- **Session Management:** A session should only associate the same client ID. It can only occur after authentication.
- **Access Control:** Set users access based on system permissions. It should use only preset, trusted systems.
- **Cryptographic Practices:** All assets should be protected by cryptography. A policy should be established in how to manage the public and private keys.
- **Error Handing and Logging:** All application errors must be 'caught' and handled. They should not disclose any information in the error responses, and instead, only show in the logs. A few can only access the logs themselves, and mechanisms should be in place to analyse the logs.
- **Data Protection:** Protect most assets, communications and caches. Encrypt any stored information that holds significant value.
- **Communication Security:** Encrypt any communication channels. Certificates should be valid, and the protocol Transport Layer Security should be in use.
- **System Configuration:** Ensure that everything is up to date. This includes servers, frameworks, components, languages, IDE's and libraries. Remove unnecessary information before deployment; test code, TODO's, HTTP methods and response headers.
- **Database Security:** Maintain short connections. Authentication needs to be checked prior use, and access control needs to be restricted.
- **File Management:** Require authentication and access control before any interaction with files. Validate file headers. Implement safe uploading by scanning for any malicious intent; viruses and malware.
- **Memory Management:** Check buffers. If larger than expected, potentially has malware or viruses. Avoid the use of vulnerable functions - *printf, strcpy*
- **General Coding Practices:**
  - Use tested and trusted components for tasks
  - Restrict users from altering code in any way
  - Review all third party components; code and libraries
  - Initialise all variables and fields

2.4 Related Work

This project expands upon ideas on the following works; A Survey with Software Developers - Think secure from the beginning; and Kirk and Tempero's, Software Development in New Zealand.

2.4.1 "Think secure from the beginning" – Assal and Chiasson

In "A Survey with Software Developers" [6], the authors pursue to understand the human behaviours and motivations surrounding factors of software security. The authors specified a series of questions targeted toward software developers through an online survey. The research examined responses to support the professional development of programmers further, both in theory and practice. This was chosen as background reading as it aligns well with the research goals of "Why Do Programmers Do What They Do?", as it aims to form a theory on the influences and effects of decisions surrounding technical work.

The results outlined the following common groups:

1. **Work Motivation:** Developers did not lack motivation in their job. They performed based on self-determination.
2. **Understanding of software security:** Developers had a sound understanding of software security. They grasped the importance of securing technical work and the discussed various methods of doing so and specifying at what stages in the project life-cycle they should implement these based on "best practices".
3. **Security Issues:** Majority of the participants believed their software could be compromised, despite being comfortable with the approaches to protecting the software. The majority has also experienced a security issue, whether that be a breach or vulnerable code.

From these common groups, the overarching theme displayed was that the developers were not purposefully ignorant about maintaining security practices; the majority were proactive and willing to learn. However, it was the importance of functionality and lack of ongoing support from organisations which made working towards a more secure software challenging.

This paper was valuable to read as there are strong similarities between the research topic developed in this and the subject of this ENGR489 project. The methodologies are different; however, this paper's findings display a programmers personal perspective rather than a theorised view. It is a direct comparison to our project, and we can further outline questions directed to the future work stated in the article's conclusion; "to explore potential relationships between motivations, deterrents and strategies for software security" and "investigate security procedures and attitudes in companies that have experienced security breaches and compare it to others who have not".

Similar questions from the survey were used for the interview process in this study. Important questions identified are:

- How does security fit in the development life-cycle in real life?
- What are the current motivators and deterrents to developers paying attention to security?

The differences and similarities can be easier understood in the table below:

	This Research	**Assal and Chiasson's Research**
Data Collection	Interview	Survey
Participant Group	Any type of programmer	Software Developers
Location	New Zealand	North America
Data Type	Confidential	Anonymous
Results	Analysis of participant answers and observations	Based on participant answers
Topic	Security	Security

Table 2.1: Summary of comparison with "Think secure from the beginning"

2.4.2 "Software Development Practices in New Zealand" – Kirk and Tempero

In "Software Development Practices in New Zealand" [19], the report authors look to "developing and applying a range of software productivity techniques and tools to enhance the performance of the New Zealand software industry". Like Assal and Chiasson, the authors of this study outlined a series of questions in a survey targeted towards known Information Technology organisations. The survey aimed to understand the practices used by industry and in the findings can be used to make recommendations on best-use development practices for organisations. Kirk and Tempero's report is similar in output to the ENGR489 project as the findings can be used to make suggestions for teams adopting and developing security practices. This was chosen as a background reading as it provides insights on the current state of industry in New Zealand, which is what I aimed to develop in the research outlined in this report.

The key findings of this study were:

1. Organisations and individuals **do not follow** standard agile process models.
2. New Zealand is generally more **implementation-focused** in software development. There is an emphasis on this over other aspects of the software development life-cycle such as security and testing.
3. Decision-making is a **collaborative effort** with individuals involved in different stages and traits of the development life-cycle.
4. While most New Zealander's state they are "agile" this is not supported as frequent contact with clients and stakeholders is not upheld. However, there is a **highly iterative aspect** to the work individuals do on projects which do maintain agile principles.
5. There is a **weakness in requirements gathering** which results in a widely noticed lack of clarity on scope details.
6. This point also relates to point 2, there is a noticed severe **lack of code quality** whether this is in design, reviewing and testing stages, or with general coding best practices.
7. Most **do not develop around** tools such as libraries - rather they use them as a support. This can be derived as not being "best-practice" and can be more time-consuming.

Not much was asked specific to security, but finding number 6 links to poor practices around secure programming.

The report had a limitation in which it did not make any recommendations at this stage, but it did mention that these findings can be used by organisations to obtain a view of the software practices in New Zealand. From here, organisations can make their own decisions on what to focus on to better their specific operations.

Comparing to the described ENGR489 project, the methodology is different, and while the topics to are differing, they are similar enough to make interesting comparisons between the two. Kirk and Tempero focus on software development practices, while this project will research security practices. A comparison that can be made could be between the findings, as much like the prior related work; the findings are that of a personal perspective of the participants rather than an objective view which Grounded Theory supplies.

The differences and similarities are outlined in the table below:

	This Research	Kirk and Tempero's Research
Data Collection	Interview	Survey
Participant Group	Any type of programmer	If developing software
Location	New Zealand	New Zealand
Data Type	Confidential	Anonymous
Results	Analysis of participant answers and observations	Based on participant answers
Topic	Security	Software Best Practices

Table 2.2: Summary of comparison with "Software Development Practices in New Zealand"

Chapter 3

Methodology

This chapter outlines the methodology that was followed to obtain findings in this study. It then deconstructs the methodology further to describe the data collection and data analysis process. The data collection subsection provides a summary of the participants.

3.1 Grounded Theory

Grounded Theory (GT) is the chosen methodology for this study. It was an appropriate choice as the study focuses on human aspects, and GT is a way of analysing qualitative data with the end-goal of defining a new theory from the sampled data [5]. There are several steps in the GT methodology in order to make the final theory. The theory is expected to be explanatory, focusing on describing elements of the findings rather than stating. It should also be based on the collected responses and observations, rather than pre-conceived ideas. As such, extensive literature reviews should be avoided.

Initially, researchers must choose a topic and research whether the Grounded Theory method is the right one for the research project [5]. After contacting potential participants, the iterative process of data collection occurs. Questions adapt between the rounds of the collection as the data is analysed. This refinement of questions happens to delve deeper into the traits of the emerging theory. When saturation is achieved, no further new findings are displayed which contribute to the final theory. This results in the "grounded" nature of the overall theory.

In the study outlined in this report, at approximately ten interviews, the concepts obtained from the data collection started to become quite similar. This is called saturation point and is when some new lower-level ideas are being obtained, but no overly new information. This was the grounding point of this specific Grounded Theory.

Initially, I aimed to get more technical empirical evidence on programmers and their security practices. However, as early as the third interview, it was identified that technical security programming practices are not as influential as initially thought, and the interview questions changed to match this gradual shift in focus. This is another benefit of Grounded Theory as questions do not have to stay the same; instead, they undergo iterations of development as an area of interest. shifts.

Similar projects at the university have been conducted using this methodology. Siva Dorairaj's, "The Theory of One Team: Agile Software Development with Distributed Teams";

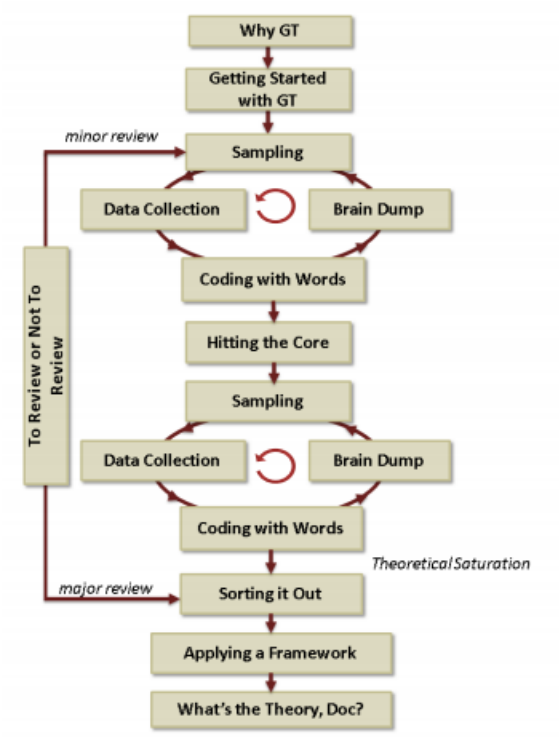

Figure 3.1: The Grounded Theory Pattern (Hoda, Noble, & Marshall, 2011)

Rashina Hoda's, "Self-Organising Agile Teams: A Grounded Theory"; Brendan Julian's, "Agile Practices: A Theory of Agile Adoption and Evolution"; Nathan Newton's, "Information Security in Agile Software Development: A Critical Success Factor Perspective"; and Aaron Pangs, "What Programming Languages Do Developers Use? A Theory of Static vs Dynamic Language Choice" [20].

3.2 Data Collection

3.2.1 Recruitment

A recruitment post of this study supported by a webpage [21] was sent to mailing lists and websites like OWASP Meetup and LinkedIn. These are shown in Appendix E and Appendix F, respectively. There was a brief halt in the recruiting through the Meetup website as I got banned for posting in groups as I was marked as spam. However, I did not get much interest from the promotion through both Meetup and Linkedin and instead relied on personal contacts (friends and family-friends) and supervisor contacts for initial recruitment. Participants also helped with recruitment by reaching out to others in the industry and telling them about this study.

3.2.2 Interviews

Fifteen interviews were agreed upon and conducted of participants across New Zealand. These participants were from fourteen unique organisations. Participants had varying job titles, years of experience, and were in a range of different sectors and organisation fields. The diversity within the participants allowed for connections to be made across a broader

range of people.

Human Ethics approval had to be gained before the interview process could commence. This document is displayed in Appendix B. Initial questions were shortlisted based on a small pilot study run on two recent graduates of the School of Engineering at Victoria University of Wellington. Overall, this small pilot study was essential in providing some more clarity on the interview process. It also supplied practice in conducting interviews and has helped in the continuous process of defining questions.

After Human Ethics had been approved the interview process started. Questions were adapted dependent on the analysis between the interviews and changed as a result of participants answers within the interview. This was so any intriguing information could be queried further during the semi-structured interview. Therefore, the list of interview questions in Appendix D changed with only the "Participant Background" section remaining as a constant.

Due to the disruptive year, interviews were predominantly conducted over Zoom. Consent from the participants was obtained through an email response as per Human Ethics committee request. Interviews ran for periods ranging between 30-90 minutes. A summary of the participant sample is provided in the table below.

Alias	**Role**	**Experience (Years)**
P1	Enterprise Architect and Domain Lead	15<
P2	Principal Product Architect	15<
P3	Senior Security Architect	10-15
P4	Senior Software Engineer	15<
P5	Software Developer	<2
P6	Level 2 Security Analyst	2-5
P7	Site Reliability Engineer	<2
P8	DevOps Engineer	<2
P9	Portfolio Architect	10-15
P10	Full-stack Software Developer	2-5
P11	Cloud Engineer	2-5
P12	Consultant (Cloud Engineering)	2-5
P13	Development Manager and Technical Lead	10-15
P14	Senior Software Engineer	10-15
P15	Business Rule Consultant	15<

Table 3.1: Participant Aliases and Summary

3.3 Data Analysis

After transcribing interviews and pairing them with written observations), the selecting coding process occurs [5]. Key points from each transcript are paired with a simplified summary of the points [5]. The constant comparison method is used where codes are compared to others from within the same interview and also with other interviews [5]. These comparisons

continue to occur, and higher abstractions are found, also referred to as Axial coding **INSERT CITATION** . They are further narrowed to become categories for the theory [5], also referred to as Selective coding [5]. Therefore, this coding process has three key steps. Theoretical notes are written throughout this coding process to research relationships between concepts and categories [5]. An emergent theory is then formed which aims to explain a practice or a phenomenon.

Chapter 4

A Theory of Influences on Security Practices

This chapter will outline the emergent theory that has been found through the analysis of the findings. The three-step coding process described in the methodology section was used to achieve this theory. The three found categories will be further explained and the relationship between each of them.

4.0.1 Emergent Theory

The emergent theory consists of three main categories on what influence programmers:

1. Culture in groups of individuals
2. Organisational structuring and practices
3. Industry trends

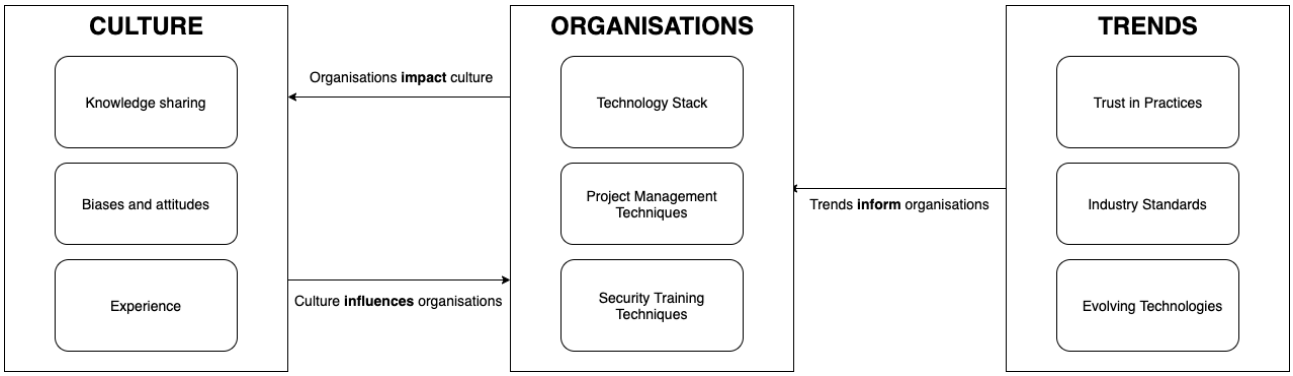

Figure 4.1: The diagram of the emergent Influences on Security Practices theory

A prior iteration of the diagram had shown one more category; ”Teams”. However, as the constant comparison process occurred, the factors of teams seemed to be more-so attributed to organisational structuring and practices. This meant that this category title was discarded and the factors were merged into ”Organisations”. Another discarded category was the influence of stakeholders. At the beginning of the interview process, clients and vendors seemed to have significant influence to the security practices implemented by programmers, but as the analysis process developed, it was not as strongly quoted in the data

collection. Even when prompted, the idea seemed to be foreign to many. Participants were disclosing that decisions were not dependent on any stakeholders and that set practices were already stipulated.

4.1 Culture

This category is a combination of the culture we see in groups of individuals. Important concepts relating to this were mentioned by the participants and were discovered in the data analysis. The term culture refers to the customs that programmers have in order to maintain security; knowledge sharing, biases and attitudes and experience.

4.1.1 Knowledge Sharing

Knowledge sharing was highly regarded among the participants. Participants identified communication between team-mates as being a key way to learn and better current and new security practices. These opportunities are done within teams and are often done when peer-programming or simply asking questions. However, often the vital knowledge sharing moments happened in passing. When programmers were stuck with using a library, framework or even learning a newer language nuance, they appreciated the ability to turn to the person next to them and get help.

"...we've got a really flat structure, if I needed to I could pretty much directly ping someone..." - P10

The lack of knowledge sharing was often highlighted as deterrence in learning and using practices. P11 described this as *"tech bro-culture"* where there is a view that asking questions makes you seem lesser than, and therefore, there was a cultivated culture on this lack-of-communication. They stated that a lot of the time in this industry people would say that they know something when they do not.

A participant *[P8]* defined that the open communication streams between people across business and experiences made it easier to obtain answers. Walking between desks to other teams and also sending anyone emails no-matter the hierarchical nature of jobs titles helped facilitate these exchanges. These were often more attainable in smaller organisations where individuals were more familiar *P5, P8, P9*.

"It's a lot easier when there are only seven people in my company. - P8"

4.1.2 Biases and attitudes

It was observed participants had very strong preferences to how security practices aligned with their work. Those were strictly developers *[P4, P5]* typically found it a deterrence in completing their work, while the others had more of a positive attitude towards security. This has been something also picked up by participants. This also showed that often, people had a more positive demeanour when dealing with technologies based on familiarity.

"... when a change happens that they don't like, there have been people who have straight up left and quit their jobs" - P12

A participant *[P11]* acknowledged the personal relationships within the workplace as contributor to biases in the *"tech bro-culture"* . They gave the example of a vendor-client relationship that they had witnessed in the workplace where the vendor accepted any requests into production because of the close friendship with the client who had not given any comments and had caused the build pipeline to fail many times and still was not working. Therefore, this *"tech bro-culture"* also favours explicit favouritism between small groups of people in an organisation. None of the other participants from my sample group bought this up and it was an interesting view-point.

4.1.3 Experience

Experience pertains to the level of expertise an individual has in the industry. This was not only measured by the amount of years that they have worked, but also whether they have worked on a range of projects. One participant *[P11]* is an example of this; they have worked 2-5 years in industry, but have worked in multiple government and private organisations in their short-time in the workforce. Therefore, they heavily advice the seniors in their team on the work that they do as their experiences are related to their background in the industry.

Almost all the participants asked identified this trait as being a major difference in how less experienced and more experienced individuals deal with security practices. Often less experienced team members are wanting and willing to learn, but still are lacking in the ability to identify key threats and risks which a more experienced team member is more versed in doing.

"New team members, I find, that they're kinda charged-up, ready-to-go, and that they want to prove themselves. I do find the more experienced people are a little more humble and they're a little bit more set-back... it's a bit more of a different energy-vibe; you see the new people come in, so ready to learn and they want to do security and then you get the people who have been there for ages like oh yeah that's easy and just do it." - P6

4.2 Organisations

This category is the influence of organisational practices and structuring on security practices. The attributes within this category are the parts of an organisation which motivated programmer choices; technology stack, project management techniques and security training techniques.

4.2.1 Technology Stack

The technology stack references the security tools, frameworks, libraries and languages in use in an organisation. Throughout the study it was prevalent that organisations have preselected security programming languages that they use. These are chosen based on legacy technologies already in use [P14] and legacy products that organisations provide clients [P3, P7]. Not a lot flexibility is given to programmers to choose and if they try lobby a change there is a very long term process in approving *[P10]* which dissuade them in trying to make any changes to the current process. When asked about whether languages changed dependent on security practices and/or requirements the answer was often always unanimously,

"No." - P4.

Though, one participant did mention that their libraries and tools did not change based on the security needs as they use ones that can be used across languages [P14].

There were also two outliers that were able to change languages related to security. Both the cloud engineers *[P11, P12]* stated that languages were changed based on their security needs; not on features or familiarity. Especially when working on the security of pipelines, but pipelines themselves were mandated by the organisation.

"...we picked this tool... we also couldn't pick our pipelining tool, that was Jenkins and that's a corporate mandate... I think what makes us different is that we deal with CI/CD so need to keep up-to-date and research more..." - P11

4.2.2 Project Management Techniques

Often project management techniques are set by the organisational structuring and choices of more senior people within companies. Participant indicated these techniques as being a deterrence in implementing security in their work. All, but one participant *[P10]* stated that waterfall was not used by them, and many of the other participants *[P3, P4, P5, P7, P8, P11, P12, P13, P15]* identified that project management techniques like agile or DevOps are never explicitly followed, thus forming a hybrid of all the three approaches. This can make incorporating security into project life cycles difficult. When managed correctly, the techniques did benefit the participants a lot. It was easier getting the security teams involved throughout the process in a more DevOps approach, and with an agile approach each deliverable are checked at the end of each sprint, but as stated, the strict management styles were not enforced. Traditional waterfall-type thinking still leaks through and there is an emphasis of security at the beginning and end of projects rather than throughout as a means to reduce *"tech-debt" [P7]* which does the opposite as programmers then struggle to fix multiple issues at the end.

Participants with more senior roles did not like the wave of DevOps management; it was identified as being too time consuming or just a fad. However, the feelings to do with security were different as one was a developer and one was a security architect. Also the later critique stemmed from a hatred of buzz-words a they mentioned DevSecOps; a more security specific iteration of DevOps project management.

"It's just annoying and gets in the way of my work" - P4

"... I don't like it. No one actually follows anything anyway, and an ongoing security approach should be there already..." - P3

The project management techniques also influenced how teams interacted with each other. With a DevOps approach, the support from the security team was continuous and talking between teams occurred more often. Often a security person was fully involved in the entire project and consequently this increased collaboration which meant that security issues were being identified earlier, not just near deadlines or after completion. However, a deterrence to this is money and different business units charge for time *[P12]*.

"The cross-functional teams, is how I've kind of been told to refer to them; they're really good... Anyway the point was, yeah, a lot of it is around your organisational structure and the way you form your teams. It's a major problem with like waterfall-type projects where you do like dev, and then

you do test and then you do security at the end, but that's, it's slowly going out, but I mean like very slowly, and they've left a lot of things behind, or rather forgotten to update a bunch of processes to match this sort of stuff." - P11

4.2.3 Security Training Techniques

Security training techniques and methods are legislated internally by the organisations. Participants state that they have not been exposed to much security during their prior education so these training's are key in educating employers in how to protect assets and mitigate risks and are ongoing *[P1, P2, P3, P6, P7, P9, P10, P11, P12, P13, P14, P15]*. However, training methods are still quite traditional. They are more policy and protocol related and often employees just have to do readings, watch videos or listen to talks in the office by either internal teams or external businesses. The training techniques are more informative to-do's of what to watch out for rather than learning any practical mitigation techniques. Organisations do not expect employees to know the policies and protocols, but are expected to know the technical programming-side or should pick them up themselves.

"It's kind of expected, we don't do any formal coding security training, we've got other general security training about data and procedures, but nothing like technical related... I've done, I've had some, listened to some talks throughout about general security risks. They talk about you know, the different areas that our company gets kind of attacked. More, of just of a FYI." - P7

Personal training is encouraged in organisations and a budget is put aside for this. Participants stated that they could ask to do their own training at work using Linkedin videos or Pluralsight *[P10, P11, P12, P13]*. While the subject of the solo-learning videos are more technically focused, they are still theory-based. This is why most participants identified that they learn best from talking to other people in their teams as you learn more technical skills when you apply your knowledge.

"I just ask and someone will help me out and they'll teach me" - P10

"I think the best training you can get is from your peers..." - P14

There were two unique points brought up by participants from my sample group which shows a gradual change in how organisations aim to educate programmers on security. A participant *[P10]* recalled the use of technical workshops with an external company being a great way to learn how to program more securely to protect applications. This method was a more active approach than what had been described by others. They stated it was similar to following live-coding during university lectures.

"I think at work actually, they also made us do a workshop on SQL injection attacks and stuff and they made us do a bunch of things and we actually had to follow along on our own computers and that's really the only way you learn is by doing." - P10

Another newer way of learning which was introduced in the interviews *[P13, P14]* was corporate Hackathons. These are events which the organisation runs for its employees. The two participants talked highly about it as being fun and casual while also prompting people to learn concepts. The Hackathon topics would be centred around the organisations field (eg. Fintech) in order to translate learning's from these sessions to their current work.

"We have our own in-house, we call it Hackathon, you know, programme and base-line programme where anybody [can join], it's a kind of mandatory training programme where everybody [will go through] - [they] will be given a kind of a training or walk through by one of our security team, team member[s] actually, to how it works, how does it work in reality... People will be given some level of platform information, that what [does] this Hackathon [mean] and it's a game, you know, game! Where can you show me where is [the] problem. Can you fix [the] problem? [Do] you think there is a problem here? Do you think [with this] given website, will you be able to hack some data from my, from the machine, you know? So it's all kind of doing, rather than just doing, a PPT programme, a presentation, [it's] more getting your hands dirty and getting things done. " - P13

"[I'm] fortunate enough to work in a company where there is a Hackathon every month somewhere in the world right. And, and it's amazing" - P14

4.3 Trends

This category is the influence of industry trends on a programmer. The attributes within this category are the functions which make this category and impact the security related decisions which programmers make; trust in practices, industry standard, evolving technologies.

4.3.1 Trust in Practices

While those who worked in financial tech and in consultancy firms *[P3, P6, P7, P9, P11, P12, P13]* favoured enterprise tools and libraries. It provided them with more security in protecting their assets. Those participants did acknowledge that while enterprise was favoured over open-source due to more trust, there were times where open-source was needed when coming across a new problems without any enterprise solution *[P12]*. The participants also used open-source when they wanted to adapt anything to best suit their practices without breaching any legal agreements with the enterprise solutions. Often the trust was based on what other companies were doing. However, one *[P13]* did state that the enterprise solution that they have is quite robust and can be used for many needs and languages.

"We do prefer enterprise, but to avoid breaking any SLA's [service level agreements] sometimes we have to look towards other open-source alternatives. We don't prefer it, but have to do it." - P12

Every participant stated that while the security team has to vet new software, they do not check libraries. This in turn gives programmers free reign over choice. But the majority did not check the open-source libraries that they use *[P1, P2, P3, P4, P5, P7, P8, P9, P10, P11, P12, P13, P5]* and one participant *[P10]* stated that all that gets checked is the functionality of the library after it has already been used. Therefore, there is a trust in this.

"I honestly, like, I don't know, like, I'm really allowed to like code, and use the library and play around with it. - P10"

There was only one participant who checked open-source libraries before using them *[P14]*. They stated that open-source libraries are great to use because they have so many different people working on them at once, but it was naive not to check them for any discrepancies or issues, especially since there are so many readily available which can make the choosing process overwhelming. It defines programmers and their ethics as it can provide

entry-ways to threats.

".... there are a certain giveaways which you can look in the code-base and footprints to actually see if the tools are leaning towards the good-side or bad-side... there are a few giveaways like the test coverage of a tool, linting in the tool. How many commits do you actually do? Do you write any footprint doc for it? How do you add a feature? Is it commented? Types, annotations." - P14

The participants in government organisations or within smaller start-ups had to outsource a lot of the security work *[P1, P5, P8, P9, P15]*. They do not have the resources to dedicate the time into this aspect of programming and consequently, they also do not robustly test any outputs of the vendors against security, only the functionality. They have a trusting relationship with their vendors. It was acknowledged that this was not the best practice *[P5]* , and another noted that vendors and client both lie so it was better to ask as many questions and do as much testing as possible *[P11]*. Another participant stated that a breach due to a vendor designed product was the catalyst for change within their organisation for hiring an internal software team *[P10]*.

4.3.2 Industry Standard

Much of the processes of how programmers worked with security was defined by industry standard. There is a rush to match others in the same field, and also to work inside the legal and best-practice compliance standards.

Participants cited SOC **[insert soc citation]** and ISO **[insert iso citation]** compliance *[P2, P3, P4, P6, P7]* as being the key frameworks that were followed when coding. These are mandates so that programs can be used externally by clients and other parties. This enforced clear coding practices upon programmers such as encryption of their work and separation between applications, *"zero-trust architecture" [P1]*. Programmers also follow the best-practices outlined by OWASP, which were described in section 2.3 of this report. This does not only include the actual code, but also documentation.

"With the various compliance regimes that we're under, so there's ISO27001, PCI and some stuff for the government we have to demonstrate that we are following the processes." - P2

There is a pressure to keep on par with other similar people and companies in industry. This is to still stay relevant in the area of expertise and especially for vendor firms to look appealing for their clients.

"We haven't [stuck] ourselves in any particular way. Whatever industry is responding [to] and whatever the new features and challenges are coming, we adopt it, we adopt as early as possible." P13

4.3.3 Evolving Technologies

Participants emphasised the emergent and evolving technologies in industry as being a motivator in adapting security practices. Cloud products was one that was consistently bought up in the interviews, and the first participant [P1] stated that in following interviews, I should focus on cloud in order to match the newer ways of working rather than just the old. This emergence of the cloud involves server migration to services like AWS and Azure or data migration. AWS and Azure are prevalent in industry as they are the two cloud services approved my government. The migration has been slower in New Zealand compared to the

rest of the world due to data legally having to be stored on-shore.

"Any organisation private or public, they are putting extra [effort] and going out of the box to [move] onto cloud, and the cloud is totally different compared to the on-prem, legacy infrastructure. - P1"

More resources are also being directed into automation of services *[P15]*. While it costs in terms of time and money in the present, much like server and data migration, there are many long-term savings.

These evolving technologies mean that newer processes have to be learnt on how to deal with security. However, this is difficult as not a lot of people have the robust knowledge to truly understand these yet, especially within a client and vendor relationship.

"... a lot of people at the start didn't necessarily know [understand], it wasn't intentional. Sometimes it's also clients, they just go on and add these rules..." -P12

4.4 Connections between Categories

This section describes the relationships between each of the categories; trends inform organisations, organisations impact culture and then vice versa with culture influencing organisations.

4.4.1 Trends Inform Organisations

Emergent trends in industry heavily informed organisational practices. This relationship is there when newer technologies and standards are released, as there is a rush for organisations to pick up the traits in order to seem relevant, up-to-date and/or within the bounds of the law. However, this rush is all based on the organisations decisions, and the informing nature means trends just exhibit and provide opportunities for organisations to make any changes.

When new technologies become popular in general industry, organisations adopt it in their technology stack, as seen in the rise of React into the in-use languages *[P3, P5, P7, P10, P13]*. They can also become mandated by the organisation as with Jenkins for the pipeline *[P11]*. The choice of that is because of an industry standard, not explicitly stated by compliance standards or by legislation, but the desire to match and be on-par with other companies.

Project management techniques are informed by industry trends as well. The phasing out of traditional waterfall is due to the emergence of agile, and now the adoption of DevOps and consequently DevSecOps is another trend. However, changing to match the trends seem to be premature and confusing for programmers and how they deal with security as these management styles are not exactly used as how they were designed to be.

There is more of a shift and understanding that there needs to be more technical support provided in workplaces. However, not quite a strong trend yet, but as established companies *[P10, P2, P14]* continue to do live-coding workshops and internal Hackathon's, this will further inform other organisations of the benefits of such security training techniques.

There has been some small changes already with two participants *[P3, P7]* having completed a Hackathon when joining the organisation, but this change needs to be long-lasting and there should be constant practical ways to learn about secure programming much like the once a month opportunities that *P14* has.

4.4.2 Organisations Impact Culture

Organisations impact the culture of employees within industry. The term impact means to force change. The ways of working in an organisation impact the three aspects of culture; knowledge sharing, biases and attitudes and experience.

Organisational structure impacts the ability to knowledge share. This is most clearly done by the project management techniques in use. With a more DevOps approach, there is talking going on throughout the project life-cycle which involves different teams. Participants identified this way of working as being extremely helpful. This is similarly identified in the agile management styles as best practice is pulling in security team members to assess at the end of each sprint. These approaches as opposed to waterfall also promote communication within teams as the constant assessing makes internal teams discuss whether their solutions are best-practice and they share ways of changing.

Biases and attitudes are impacted and shaped by an organisation as well. An organisation which fosters security involvement throughout a project and also promotes ongoing security education for programmers will be an organisation that finds people more willing to change and more positive about having their code going through multiple iterations of security. *[P10]* stated that they were happy doing the waterfall approach because they were unwilling to change despite the difficulties they bought up in being blocked by the security team for long periods of time. [P4] also was observed to have a distaste in dealing with security when programming as they were not provided with ongoing training opportunities.

Experience is impacted by two aspects of the organisation; security training techniques and the technology stack. Theoretical security training whether it be about policy and protocols, or lecture-style talks and Linkedin videos are a way of exposing employees to past and current threats to an organisation. As the case and how it was or can be managed is explained to employees (eg. SQL injections) they are able learn ways of mitigation and increase their knowledge. However, increase in knowledge does not increase their experiences. These are best done with the internal Hackathon's. *[P14]*, described giving cases related to the nature of the organisations work and this provided a mock scenario which is able to build up the experiences of employees in a safe environment without any consequences. As the organisation increases the technology stack employees experience more languages, tools, frameworks, libraries. The range and growth of the experiences which are provided by the organisation in-turn increase the ability for employees to deal with security issues.

4.4.3 Culture influences Organisations

While organisations impact culture; culture also influences organisations. The important distinction between the two is that impact is forcing change, while influence is a way to coerce. Culture does this to the three defined aspects of organisations; technology stack, project management techniques and security training techniques.

Individuals prefer using technologies which are familiar. This is reinforced by *[P11]*, who stated that the team chooses security frameworks and libraries which are familiar to them. This is an example of experience influencing the technology stack. A more experienced participant *[P14]*, described that the recent adoption of React to their programming languages was informed by other organisations and what was in use, but also that the newer, and fresh graduates with more familiarity with the language reinforced the decision as they liked using it.

Biases and attitudes influenced the project management techniques. As stated throughout this section, no one follows a truly real interpretation of waterfall, agile or DevOps. This is because of their own ingrained biases. These biases consequently make it hard to involve security as an ongoing component of a project life-cycle. The attitudes also cause difficulties in making changes to organisational management regarding security as some people simply do not want any disruption in the current norm. This also makes it hard to implement any changes in involving security teams and practices.

Biases and attitudes, paired with experience do effect the way organisations implement security training techniques. Based off the participants within the sample group, the more experienced members did not participate in as frequent training sessions compared to the lesser experienced individuals. This provides the bias to organisations that experienced members will not benefit from the training and some places did not provide it ongoing. Attitudes in regards with the security training techniques were also a big influence to organisations with a participant noting that if fun events like Hackathon's are enforced, people do not want to participate *[P14]*.

Chapter 5

Evaluation

The contents of this chapter will, compare and contrast the chosen methodology to others, justify the processes used within the Grounded Theory framework, evaluate the overall theory using Glaser's criteria and evaluate the findings against existing literature.

5.1 Chosen Methodology

Alternative qualitative methodologies were not explored in the beginning of this project as the initial project brief described that this project would be conducted as a grounded theory study. However, two alternatives are compared with the project methodology to see what, if any, might have been a better approach to the study.

5.1.1 Ethnography

CITE: https://www-emerald-com.helicon.vuw.ac.nz/insight/content/doi/10.1108/03090560510581782/fu
theory-ethnography-and-phenomenology-a-comparative-analysis-of-three-qualitative-strategies-
for-marketing-research

Ethnography is the study of people, or "folk" within a group. It is an intensive study-type as participants need to be regularly contacted and different forms of data collection methods should be used from interviews, to surveys to observations. In regards to findings, this methodology focuses explicitly on culture. Usually these occur in-person, this is so the researcher is able to understand the behaviours and view-points of the participants clearer. While predominantly qualitative, quantitative methods can be employed to support the other type of data.

Ethnography would have fit to the initial brief of the project. However, while Grounded Theory aims to present a new theory, Ethnography aims to find patterns between the ethno (folk) within the sample. This would have provided no potentially new insights and it would have focused on culture of programmers rather than other traits. Ethnography would not have been suitable to the COVID-19 situation as the data gathering process calls for interviewers to directly interact with the participants. This is something that would have been impossible to do in the earlier half of the year. Another additional negative to this approach is that long-term interaction and relationships with participants may introduce biases in the findings; something which Grounded Theory aims to minimise. Biases can skew results which make them more inaccurate and consequently results can be considered weak.

A benefit of ethnography compared to Grounded Theory is that there is a lot more data being provided to the study from a lot of different inputs. This can strengthen findings. However, this also means that the pattern formation is far more robust that the already long process of Grounded Theory analysis which makes ethnography best suited for a longer-term project; not just two trimesters.

5.1.2 Phenomenology

CITE: https://www-emerald-com.helicon.vuw.ac.nz/insight/content/doi/10.1108/03090560510581782/fu
theory-ethnography-and-phenomenology-a-comparative-analysis-of-three-qualitative-strategies-
for-marketing-research

Phenomenology is the study of human perception. It relies on the understanding of participant experiences. In this methodology, interviews are the only source of data so an interview is key. The interview is then transcribed and studied in full by the researcher. This is so the researcher is able to understand the full narratives of participants. Researchers then can pull out key points from the transcripts. As these key points get added upon as more interviews are undertaken a structure is defined. This structure is then reduced to the key aspects, and used to describe the phenomena. Researchers are then able to go back to interview participants for more data gathering and confirmed approval of the outputted description.

Phenomenology would have fit to the initial brief of the project as it a study of a experiences, in this case the experiences of programmers in regards to security practices. However, while it only aims to define practices towards the development of a theory, Grounded Theory aimed to provide a new theory and consequently newer insights and ideas. Phenomenology would have been suitable for the lockdown situation we had in the beginning of the year as interviews could have been conducted over Zoom, much like the ones done for this project and under the Grounded Theory methodology.

Data is only coming from one source in this type of methodology. This reduces the opportunity to produce biases compared to Grounded Theory. However, there is not a lot of analysis which has to be undertaken then and results can potentially be less concrete.

5.1.3 Grounded Theory

This project followed a Grounded Theory methodology using interviews and observations for gathering data, as was outlined in chapter three - methodology. Key factors for choice were that it has a strength for exploring human and social aspects of concepts [5]. Specific to influences to security practices, it was important to have a qualitative method as this allows for intake of observations and answers as supporting evidence [5]. It also aims to focus more on this collection of data, rather than existing knowledge on the topic to minimise any biases. Above all, Grounded Theory allows for new theory formation to gain new insights and perspectives in the area of secure programming [5].

Compared to the two alternatives which were explored in this section, Grounded Theory methodology was best suited to the initial brief. It aimed to provide a theory while ethnography and phenomenology seem to be methodologies which can support theory building. Ethnography focuses on one aspect, culture, as the overarching topic of the research questions, and phenomenology aims simply to concisely define existing experiences. However,

ethnography could have been an interesting methodology to pursue given a longer period of time as it pulls in various sources of data and allows for the researcher to be immersed in the lives of participants to gain better understanding of their answers.

5.2 Internal Methodology Processes

This section evaluates the participant recruitment and data gathering strategy used in this project. It outlines what was done well and why, and what could have been improved upon.

5.2.1 Participant Recruitment

While the recruitment aimed to find people from Linkedin and Meetup groups such as OWASP NZ, this became difficult. The constant recruitment across Meetup groups without actually going to Meetups (despite the COVID-19 lockdown) made it so my account was marked as spam, and I was unable to post anymore. Linkedin proved to be ineffective as well as groups do not often show notifications of new posts to users by default. The topic of security also made people hesitant to participate and so did the disruptions in people's lives due to the pandemic. Therefore, a lot of recruitment was done by asking family friends and friends. This also made it comfortable to ease into the data gathering process.

Using family friends and friends for this investigation did not skew or introduce biases to the results which was the initial worry. This was because everyone came from a range of study backgrounds, had varying years of experiences, and worked across the industry in various different roles. With the other individuals as a part of this sample, and expanding it to all of New Zealand, the theory had depth to it.

5.2.2 Data Collection

Interviews were chosen as the method of data collection as this is what is the norm in Grounded Theory studies [5]. Interviews are semi-structured and are more open-ended compared to the likes of a diary study or a survey. This is beneficial as it allows for participants to feel more comfortable and have more control over their answers. It also allows an interviewer to correspondingly make observations of the participant and make connections with their responses [5]. A more concrete emergent theory is then formed as it is built upon two sources. However, this also can introduce biases by the researcher as observations can hold stereotypes.

Light observations were undertaken. As this year was unprecedented, very early on the data gathering process shifted to be one predominantly online. Only two participants were interviewed in person; *[P6]* and *[P14]*. This meant that robust observations could not be taken of their ways of working in the office (most were working from home), but some could still be undertaken through Zoom. Facial expressions and tone of voice were big indicators as the face was the focus of the video. From this, the observation about the differences in attitudes of developers and that of more security related roles such as architects and analysts, was able to be made.

Fifteen semi-structured interviews were undertaken. For the time that this project had for completion it was enough and it had also hit saturation point where a minimal amount of new data was being added to the theoretical findings. Appendix D provides the interview template that was used. It shows the the four primary areas of questioning. The participant

background was the only section which remained consistent across all interviews and the rest changed. The other three sections provided questions that could have been asked to a person dependent on the participant background, but not all were at each time. Questions were also adapted dependent on the participants ability to answer the questions. Due to the nature of the topic, there were some privacy concerns and often the field of the organisation meant that participants had signed non-disclosure agreements as part of the contract.

After completing the data gathering process, I think it would have been good to perhaps split current work into two sections, one on the more ”hard” influences and another on the ”soft” influences. This is so more answers could be prompted pertaining to technical impacts on participants. While Grounded Theory aims to leave questions open-ended, I think the specific section would have been a benefit regardless as it would have provided the theory with more fresh and new inputs.

5.3 Glaser's Criteria

There are standards in which an emergent Grounded Theory is assessed for quality; fit, work, relevance, modifiability **INSERT BOOK CITATION**. This criteria will be used to assess this projects theory.

5.3.1 Fit

*”Fit refers to the emergence of conceptual codes and categories from the data rather than the use of preconceived codes or categories from extant theory”**INSERT BOOK CITATION**.*

This criteria means that findings should not be based on existing theories. This was done for the theory presented because robust background reading was not undertaken and the related works described in the background chapter were to provide an understanding on types of qualitative studies that could be undertaken rather than drawing from the conclusions. The result of my theory was different to what was expected, less technical reasons for the influences on security, which meant that there was a degree of unknowing starting this research. Numerous quotes were provided to further support that the theory was supported by new data.

5.3.2 Work

*”Work refers to the ability of the grounded theory to explain and interpret behaviour in a substantive area and to predict future behaviour.”**INSERT BOOK CITATION**.*

This criteria means that the theory's categories and connections should be explained. The Theory chapter describes these in detail and does so by providing further support by the use of numerous quotes from participants. Future behaviour is mentioned briefly in that chapter when discussing the shift to DevOps, more technical-oriented training and evolving technologies. It is also further supported when comparing the theory findings to literature in the section after this (5.4 Literature Comparison).

5.3.3 Relevance

”Relevance refers to the theory's focus on a core concern or process that emerges in a substantive area. Its conceptual grounding in the data indicates the significance and relevance of this core concern or

*process thereby ensuring its relevance."**INSERT BOOK CITATION**.*

This criteria means that the theory must reference the core question and and the object area of study. This is done so by the categories of the theory answering the initial question, Why do programmers do what they do? They are further described and related to the area of interest which was to find the influences of security practices. The categories are also interlinked which shows the process of the external reasons as to why security decisions are made.

5.3.4 Modifiability

*"Modifiability refers to the theory's ability to be continually modified as new data emerge to produce new categories, properties or dimensions of the theory. This living quality of grounded theory ensures its continuing relevance and value to the social world from which it has emerged. "**INSERT BOOK CITATION**.*

This criteria means that the theory should be open to adaption as more data is gathered. This was proven in the data collection and analysis as when more people were interviewed, the gathered data was able to be iteratively analysed and became codes. The modifiable aspect to this was the continuous editing to the categories if they were not considered viable to the theory as more interviews were occurring and also the changes to the questions.

5.4 Literature Comparison

This section outlines interesting findings from within the theory and compares it to existing literature.

5.4.1 "Corporate hackathons, how and why? A multiple case study of motivation, projects proposal and selection, goal setting, coordination, and outcomes" - Pe-Than, Nolte, Filippova, Bird, Scallen, Herbsleb

CITE: https://hackathon-planning-kit.org/files/Pethan-HCI-2020.pdf

The investigation outlined in this report followed five teams in a large coporate hackathon. Researchers Pe-Than, Nolte, Filippova, Bird, Scallen and Herbsleb interviewed all team members immediately before, immediately after and four months afterwards the hackathon, and during the event they simply made observations. They focused on three research questions:

R1 - *"what were the team processes, and how did they differ between PETs and FTs? "*

R2 - *"what were the conditions that contributed to sustaining the projects after the event?"*

R3 - *"what impacts did participants believe the event had on them?"*

There question most related to the theory is research question three which is the one that will be compared to the general views of internal hackathons from the category aspect explanations in chapter four. The "perceived impacts to individuals skills" section in "Corporate hackathons, how and why? A multiple case study of motivation, projects proposal and selection, goal setting, coordination, and outcomes" outlines that participants became

more confident in their skills as they learnt more, the majority also enjoyed themselves. The participants attitudes had also changed to become more positive and confident about learning new skills by themselves. This relates to the theory as the organisations category impacts culture and they correspond to the aspects security education practices and biases and attitudes. Aspects in the theory also described how people managed to learn new skills as they were applying them practically and learning as they went; this was described in the research done by Pe-Than, Nolte, Filippova, Bird, Scallen and Herbsleb.

The idea of internal, corporate hackathons is fairly new as research is limited. However, it is prevalent in industry in the United States of America. As New Zealand matches international trends, it will start to become more popular domestically as well.

5.4.2 "Challenging Software Developers: Dialectic as a Foundation for Security Assurance Techniques" - Weir, Rashid, Noble

In "Challenging Software Developers: Dialectic as a Foundation for Security Assurance Techniques"[22], the authors identify that security is more reliant on developers, but that the developers are not providing the security that is needed. Similar to the study outlined in the ENGR489 study, this was conducted using a Grounded Theory Methodology. In contrast, there were two parts of the survey, and each one was a separate grounded theory mini-study.

The authors' interview participants in order to obtain data so they can find ways to "help programmers themselves to improve security given existing constraints".

The two major findings of each survey showed:

1. Developer security is based on challenges in order to motivate better practice. These challenges are often fun adversary questioning usually to do with review and advisory. This emerged as the core theory as it was interwoven through most of the participant responses.

2. Six assurance techniques were identified in being the most helpful; threat assessment, stakeholder negotiation, configuration review, vulnerability scan, source code review and penetration testing. They all help provide software security.

These two ideas are linked, as not only do those six assurance techniques mitigate any challenges, they also provide challenges to the developer when dealing with them.

This paper was valuable to read as it provided another method of conducting grounded theory; by the use of two separate surveys that can be combined to provide their own intermingled findings. There are also similarities to the topic of this ENGR489 project. In a direct comparison, the authors outlined that they wanted to help programmers improve their security, the project outlined in this final report was initially more focused on the improvement of the technical design of security tools, frameworks and libraries. Ultimately, as the categories found in the theory are defined as culture, trends and organisations, these are all aspects which influence a programmer in terms of security, so combined with the findings of "Challenging Software Developers: Dialectic as a Foundation for Security Assurance Techniques", perhaps a unique and strong security education in the workplace can outlined.

5.4.3 ”A Survey on Developer-Centred Security” - Tahaei, Vaniea

Tahaei and Vaniea undertake an expansive literature review of 49 publications on security studies with participants whom were software developers. They then present an overview of the methodologies and current research in the area. This subsection will focus on the latter.

There were eight major themes in the results shown by the authors. They were ”Organisations and Context”, ”Structuring Software Development”, ”Privacy and Data”, ”Third Party Updates”, ”Security Tool Adoption”, ”Application Programming Interfaces (API's), ”Programming Languages” and ”Testing Assumptions”. Organisations and Context focuses on developers working within the context of organisations, teams and cultures. They are inline with the ways of working of such constraints which in turn impacts the development. This lines with the theory as one of the categories is organisations and another is culture. ”A Survey on Developer-Centred Security” specifically mentioned dedicated security teams and security oriented organisations. As described by the authors the dedicated security teams are a deterrence as developers still are not impacted by them as much as teams are small and not involved with an entire project. This was mentioned in a category in the presented theory. Security oriented organisations pertain to security being involved in all stages of development and it is cited as being a benefit. This was another aspect of the theory which was mentioned.

The section ”Programming Languages” also stated that often developers do not have free reign on programming languages able to be used. This was exhibited in the findings of this projects theory in ”Technology Stack”. Participants stated that often they just use what is already established and are not given flexibility in choice, which ultimately can effect the security libraries, tools and frameworks in place.

Chapter 6

Conclusions

6.1 Limitations and Future Work

The theory is developed and relationships are outlined between each category, but there were some limitations in regards to the data collection. While not significant there was a gender dis-balance in the participants. One third were female while the rest male. However, as women are underrepresented in STEM fields [insert citation - bookmark 4], it can be argued that this is indicative of the actual population. However, as women in this industry typically have different experiences to the majority, it would have been interesting to have more varying data to draw codes from [insert citation -bookmark 4]. If a 50:50 ratio did occur, I do think that the theory would have stayed relatively the same.

Another difficulty that limited the theory when data collecting was the refusal to answer questions. It was understood during the Human Ethics Application process that this was allowed, however some participants could not respond to simple questions such as "what language do you use to program with at work?" due non-disclosure agreements. This made it difficult to draw appropriate conclusions during the middle of this study. This could have also introduced slight biases in collection as observational inferences were made based on the non-responses.

For future work, while it was interesting to get a diverse range of experiences within the sample group of participants, the investigation would benefit from reducing this scope. By choosing one type of role, one sector or similar years of experience, the theory would have had some more pointers to technical influences. This would have resulted in key traits for a programmer to directly improve upon in their security education. Right now, the theory is more about the external influences which have to adapt to support a programmer. It could also provide more insights as to why developers seem to find security a hindrance and how to change this. If future studies could focus on organisational differences, it would be interesting as smaller companies had laxer security controls and support when programming compared to the bigger companies. However, they seemed to be more willing and open to change than those in a established organisation.

A separate study on internal hackathons should be undertaken. As it is a relatively new concept in New Zealand and generally is a paucity of research worldwide so would be interesting to combine the two; Corporate Hackathons in New Zealand. Specifically focusing on the differences between less experienced people and more experienced people in terms of work experience and the impacts before, during and after a hackathon using the Phenomenology methodology could form a strong understanding of the effects on a varying

group's experiences due to such an event. It can be used to market the idea of having regular internal hackathons within organisations in New Zealand.

6.2 Conclusion

This research was investigated in order to find a reason as to why programmers do what they do, specific to the influences to their security practices. The project was completed and three overarching categories were found - culture, organisations and trends.

Culture referenced knowledge sharing, biases and attitudes. The category is a combination of the culture we see in groups of individuals and the term culture refers to the customs that programmers have in order to maintain security. Knowledge sharing was internal team communications between members. Biases and attitudes were existing thoughts and feelings about security. Experience was the number of years and consequently number of threats programmers had been exposed to.

Organisations included the attributes, technology stack, project management techniques and security training techniques. Technology stack was the programming languages, libraries, frameworks and/or tools that were in use by the participant. Project management techniques were management structures enforced within the organisation; whether this be Waterfall, Agile or DevOps. The security training techniques were the avenues of educational support provided by different organisations.

Trends specified that trust in practices, industry standards and evolving technologies where due to the influences of industry trends on a programmer. Trust in practices was the lack of checking open-source libraries, industry standard was changing policies, compliance and best-practice standards. Evolving technologies referenced the ever changing adoption of new technology to business streams.

The emergent theory has been evaluated Glaser's criteria of fit, work, relevance and modifiablity. It provides a theory which is thoroughly explained, has drawn codes from newly gathered information, is relevant to the topic and is able to adapt long-term. The theory has also been evaluated against three works of literature and the findings are synonymous to those of those reports. Overall, the findings of this research can provide a more robust educational programme in the workplace and allow for companies to reevaluate how they respond to trends, how they structure their organisations and what they do to improve their culture.

Bibliography

[1] C. Weir, I. Becker, J. Noble, L. Blair, M. A. Sasse, and A. Rashid, "Interventions for long-term software security: Creating a lightweight program of assurance techniques for developers," *Software: Practice and Experience*, vol. 50, no. 3, pp. 275–298, 2020.

[2] M. of Health, "Cyber security incident." [Online]. Available: `https://www.health.govt.nz/our-work/emergency-management/cyber-security-incident`, [Accessed May 31 2020].

[3] M. of Health, "Update on tū ora cyber security incident at 8 october 2019." [Online]. Available: `https://www.health.govt.nz/system/files/documents/pages/health_report_8_october_20191935_redacted.pdf`, [Accessed May 31 2020].

[4] M. John, F. Maurer, and B. Tessem, "Human and social factors of software engineering: Workshop summary," *ACM SIGSOFT Software Engineering Notes*, vol. 30, pp. 1–6, 07 2005.

[5] R. Hoda, J. Noble, and S. Marshall, "grounded theory for geeks," in *ACM International Conference Proceeding Series*, 2011.

[6] H. Assal and S. Chiasson, "think secure from the beginning," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI 19*, ACM Press, 2019.

[7] M. Green and M. Smith, "Developers are not the enemy!: The need for usable security apis," *IEEE Security Privacy*, vol. 14, no. 5, pp. 40–46, 2016.

[8] C. Weir, I. Becker, J. Noble, L. Blair, M. A. Sasse, and A. Rashid, "Interventions for long-term software security: Creating a lightweight program of assurance techniques for developers," *Software: Practice and Experience*, vol. 50, no. 3, pp. 275–298, 2020.

[9] S. G. Lorrie Faith Cranor, *Security and Usability*. O'Reilly Media, 2005.

[10] B. Julian, J. Noble, and C. Anslow, "Agile practices in practice: Towards a theory of agile adoption and process evolution," pp. 3–18, 04 2019.

[11] N. Newton, C. Anslow, and A. Drechsler, *INFORMATION SECURITY IN AGILE SOFTWARE DEVELOPMENT PROJECTS: A CRITICAL SUCCESS FACTOR PERSPECTIVE*, pp. 8–14. 06 2019.

[12] I. Facebook, "A tool to detect bugs in java and c/c++/objective-c code before it ships." [Online]. Available: `https://fbinfer.com/`, [Accessed May 31 2020].

[13] C. Sadowski, J. van Gogh, C. Jaspan, E. Soederberg, and C. Winter, "Tricorder: Building a program analysis ecosystem," in *International Conference on Software Engineering (ICSE)*, 2015.

[14] I. Synopsys, "Coverity scan static analysis." [Online]. Available: https://scan.coverity.com/, [Accessed May 31 2020].

[15] Raygun, "Raygun." [Online]. Available: https://raygun.com/, [Accessed May 31 2020].

[16] A. I. Security, "Cyber security market research report." [Online]. Available: https://www.kordia.co.nz/aura-cyber-security-market-research-2019, [Accessed May 25 2020].

[17] D. of the Prime Minister and Cabinet, "New zealand's cyber security strategy 2019." [Online]. Available: https://dpmc.govt.nz/sites/default/files/2019-07/Cyber\%20Security\%20Strategy.pdf, [Accessed May 25 2020].

[18] OWASP, "Owasp secure coding practices." [Online]. Available: https://owasp.org/www-pdf-archive/OWASP_SCP_Quick_Reference_Guide_v2.pdf, [Accessed May 25 2020].

[19] D. Kirk and E. Tempero, "Software development practices in new zealand," in *2012 19th Asia-Pacific Software Engineering Conference*, vol. 1, pp. 386–395, 2012.

[20] SPG, "Alumni." [Online]. Available: https://ecs.wgtn.ac.nz/Groups/SPG/Alumni, [Accessed May 31 2020].

[21] L. Sajwan, "Why do programmers do what they do? recruitment webpage." [Online]. Available: https://homepages.ecs.vuw.ac.nz/~sajwanlava/, 2020.

[22] C. Weir, J. Noble, and A. Rashid, "Challenging software developers: Dialectic as a foundation for security assurance techniques," Apr. 2020.

Appendix A

Application

ResearchMaster

Human Ethics Application

Application ID :	0000028506
Application Title :	Why Do Programmers Do What They Do?
Date of Submission :	28/05/2020
Primary Investigator :	Lavanya Sajwan; Principal Investigator
Other Personnel :	Prof James Noble; Supervisor
	Dr Craig Anslow; Associate Investigator

Research Form

Application Type

Is this application for:*

◉ Research ○ Teaching only

Please select **'Research'** below and then select **'Save'** to access the rest of the form.

*

Research

Research Overview

Application Details

1. Application ID

0000028506

2. Title of project
*(Click the ? icon for more info)**

Why Do Programmers Do What They Do?

3. School or research centre*

Engineering and Computer Science

4. The following questions will help the committee assess whether your application is categorised as a Category A (more than low risk) or Category B (low risk).

Please check all of the boxes that apply. You will be asked for more information about some of these questions later in the application.

Check the box if your study:

4a Is health research*

○ Yes ◉ No

4b Is an intervention study*

○ Yes ◉ No

4c Involves the use, collection or storage of human tissue*

○ Yes ◉ No

4d Involves processes that use EEG, ECG, MRI, TMS, FMRI, EMG, radiation, invasive or surface recordings*

○ Yes ◉ No

4e Involves collection of information about illegal behaviour, or information that has been obtained illegally*

○ Yes ◉ No

4f Involves people who are not giving consent to be part of the study (other than observational research in a public place)*

○ Yes ◉ No

4g Involves participants under the age of 16*

○ Yes ◉ No

4g (i) Will a parent/guardian be asked to give consent for the child/young person to participate in research?*

○ Yes ○ No ◉ N/A

4g (ii) Will more than one meeting be held with the child/young people without others present? *

○ Yes ○ No ◉ N/A

4h Involves participants whose ability to consent freely is compromised due to context (e.g. people in prison), or a limited capability to make independent rational decisions (e.g. those with a serious intellectual disability).*

○ Yes ◉ No

4i Involves the use of concealment or covert observations, including those conducted online or conducted in social media. *
○ Yes ◉ No

4j Involves the use of previously collected personal information, other data, or biological samples for the collection of which there was no explicit consent for use in research.*
○ Yes ◉ No

4k Involves deception of the participants, including concealment of the true purpose of the research*
○ Yes ◉ No

4l Involves the use of highly sensitive information (see policy for definition)*
○ Yes ◉ No

4m Involves a focus on, has particular importance for, or impacts on Māori*
○ Yes ◉ No

4n Involves any other group (for example cultural or religious), other than Māori, and has the potential to cause discomfort or disruption to members of that group*
○ Yes ◉ No

4o Involves any direct financial interest in the outcome of the research by any member of the research team or external sponsor*
○ Yes ◉ No

4p Involve a conflict of interest or the appearance of a conflict of interest for the researcher (for example, where the researcher is also the lecturer/teacher/treatment provider/colleague/manager or employer of the participants)*
○ Yes ◉ No

4q Involve any situation which may put the researcher at risk of harm (e.g. overseas in politically unstable countries)*
○ Yes ◉ No

4r Involve a reasonable expectation that participants may experience (at a greater level than in everyday life) physical discomfort, emotional discomfort, or psychological or spiritual harm (e.g. asking participants to recall upsetting events)*
○ Yes ◉ No

Relationship to other Projects

5. Does this application relate to any previous applications submitted to an ethics committee (at VUW or other Institute)?*
◉ Yes
○ No

5a. If this was a Victoria University of Wellington human ethics applications, please search and add the related ethics applications from the below search window.

Search by title (partial or complete) or application ID (partial or complete):

This question is not answered.

5a (i). If you can't find the application above, please enter the application number here.

0000024839

5a (ii). If this was an ethics committee from another institution, please upload supporting documentation (such as a letter of approval) in the document section.

5a (iii). If your research has been assessed by a Health and Disability Ethics Committee (HDEC) and found to be Out of Scope, please upload a copy of the Scope of Review form submitted to HDEC and the Out of Scope letter on the Documents page.

Personnel

Personnel
To add other researchers, enter their user name, if known, or search for their first or last name (whichever is the most unusual). Click the search icon to run the search, and select the person from the list view. Click on '**OK**' at the bottom right corner to save the person record.

1	First Name	Lavanya
	Last Name	Sajwan
	Full Name	Lavanya Sajwan
	AOU system code	Engineering and Computer Science
	Position	Principal Investigator
	Primary Investigator?	Yes
2	First Name	Craig
	Last Name	Anslow
	Full Name	Dr Craig Anslow
	AOU system code	Engineering and Computer Science
	Position	Associate Investigator
	Primary Investigator?	No

6. Are any of the researchers from outside Victoria?*

- ○ Yes
- ◉ No

7. Is the principal investigator a student?*

- ◉ Yes
- ○ No

Student Researcher

7a. What is your course code (e.g. ANTH 690)?*

ENGR489

7b. Supervisor

To add your supervisor enter their user name, if known, or search for their first or last name (whichever is the most unusual). Click the search icon to run the search, and select the person from the list view. Click on '**OK**' at the bottom right corner to save the person record. *

1	Given Name	James
	Surname	Noble
	Full Name	Prof James Noble
	AOU	Engineering and Computer Science
	Position	Supervisor

7c. What is your email address? (this is needed in case the committee needs to contact you about this application)*

sajwanlava@myvuw.ac.nz

Scope of Research

Project Details

8. Describe the aims and objectives of this project
Provide a brief summary in plain language of the purpose, research questions/hypothesis, and objectives of your project. *

This project will investigate how software programmers implement and adopt security practices in the work they do, in order to develop an understanding of the influences and impacts on decisions surrounding their technical work. Examples of influences can include: type of education, whether the organisation further educates the employees, and what programming languages have they had experiences with.

This project will be done by using grounded theory and interviews will to take place to collect the data, to which then analysis of answers will then need to occur.

9. Describe the benefits and scholarly value of the project
Briefly place the project in perspective, explaining its significance and worthwhile outcomes. Include how this project will build on relevant literature, including references if appropriate.
*

Qualitative research is often neglected and overlooked in favour of more quantitative reasoning and technical traits; the processing speed, the programmers task-completion rate. Programmers provide a human aspect to a technical solution and therefore, there should be a shift to understanding the more background 'soft' processes that occur when making decisions; why are the choices made based on past influences, and how they affect the programmers work in the present?

Exploring this topic is essential as it allows for more understanding on how and why programmers think the way they do, and it builds a more robust understanding of the human and social aspects of Software Engineering The findings of this can be used to identify what security methods developers find as beneficial in their programming. This will allow programmers to complete their work to a higher standard, thus overall making their work of a higher value.

10. Explain any ethical issues your research raises for participants, yourself as the researcher, or wider communities and institutions, and how you will address these. This is an opportunity to present what you think the key risks are in your project and show how you have taken them into account.*

The project aims to investigate how security practices are implemented and adopted in the work programmers in industry work on. Steps will need to be taken to ensure that the interview questions will be written in a way that does not reveal the individuals personal passwords or suggest any hints to it. As they will be professionals, the questions will also have to be worded in a way that does not expose specific business secrets. By doing both of these, the privacy of the person and the company can be maintained and protected. Participants will not require approval from organisations to partake in this study, as the participants will not be named in any produced texts, and therefore, will not be linked to any specific organisation. It will be explained to participants that they should not reveal any business secrets or passwords at the start of the interview, and this piece of information will also be stated in the supporting participant information sheet document.

There is also a risk of discovering an organisation's security practices, which means the interview data needs to be kept secure and confidential. For instance, any reports or publications should avoid including compromising information and allow interviewees to see their transcripts to comment and edit any information.

Key Dates

If approved, this application will cover this research project from the date of approval for up to 3 years.

11. Proposed start date for data collection*

06/07/2020

12. Proposed end date for data collection*

28/02/2021

13. Proposed end date for research project*

28/02/2021

Proposed source of funding and other ethical considerations

14. Indicate any sources of funding

- Internally: by a University grant, such as the University Research Fund
- Externally: funding from an external organisation for this project, or a scholarship awarded by an external organisation
- Self-funded: paying for research costs such as travel, postage etc. from your own funds

Tick all that apply:

- [x] Internally funded
- [ ] Externally funded
- [x] Self-funded

15. Is any professional code of ethics to be followed?*

- ( ) Yes
- (•) No

16. Do you require ethical approval from any other organisation, such as another tertiary institution in New Zealand or overseas, or a District Health Board?*

- ( ) Yes
- (•) No

Data Collection and Recruitment

Data Collection

17. Please select all forms of data collection you will use in your project*

- [x] Interviews
- [ ] Focus groups
- [ ] Questionnaires
- [ ] Observation
- [ ] Other

18. Provide an explanation of the sampling rationale for your study.
E.g. representative sampling of a particular population, purposive sampling, convenience sampling. Include here your eligibility criteria for potential participants -- will there be particular criteria for participants to be included in your study, or criteria that will exclude them? *

The study will follow purposeful sampling as the participants will be those working in the industry and ideally have a range of job titles associated with programming and years in the field in order to find interesting comparisons during the interview process in the way security practices are adopted in the developed software. They can be a part of any type of organisation; government, private, non-profit etc.

Examples of appropriate jobs associated with the individuals can be; devops engineers, front-end security developers, database administrators, security architects etc.

Participants will consequently be filtered by appropriate job titles and anyone younger than the age of 18.

19. How many participants will be involved in your research?
Please specify how many groups and how many participants in each group. *

Up to 30 participants from the industry who are familiar with security practices through the professional work they do.

20. What are the characteristics of the people you will be recruiting?*

The key characteristic of the individuals will be their programming experience, security education, and age. They will have a technical role within the organisation. The study hopes to capture a range of ages and levels experience in their role in order for in-depth comparisons for analysis. Therefore, there is no specific minimum or limit on the years of programming experience, or what kind of experience, and security education sought. However, participants do have to be at least 18 years old. Participants jobs will all be New Zealand based. Participants can also work across a range programming fields eg. gaming, education, financial, analysis.

21. Outline in detail the method(s) of recruitment you will use for participants in your study. Include here how potential participants will be identified, who will contact them and how. Please include copies of all advertisements, online posts or recruitment emails in the 'Documents' section. *

I will make contact with sample groups with a request of participation by posting on groups, mailing lists and by using supervisor and my own connections. Such groups could be security groups on the websites, Meetup and Linkedin. Mailing lists can consequently be obtained from those groups as well.

A pilot study will be conducted with 2-3 participants. These participants will be recruited directly by me and are all personal contacts; friends, and family friends.

22. Explain the details of the method of data collection. For example, describe the location of your research procedures, if appropriate (e.g. where your interviews will take place). If necessary, upload a research protocol in the 'Documents' section.*

The interviews will be confidential rather than anonymous as they are done in-person and to also allow for follow up questions if necessary. Participants will be welcome to leave at any time during the interview, do not have to answer any questions they do not want to and can choose to leave the study within a week after the interview if they would like.

Interviews should be done in safe and private environments, with all participants clear on earthquake and fire evacuation procedures relevant to their location. Due to the recent COVID-19 outbreak, I will offer zoom interviews for those who are unable or unwilling to meet directly. Consent can be obtained over email by sending the information sheet and consent form content to the participant in an email body and they reply to that email with "I consent".

For in-person interviews, I will provide a box of tissues, a bottle of sanitiser, and maintain social distancing. The small meeting room will be wiped down and disinfected between interviews.

The interviews will be voice recorded, and will then be transcribed.

23. Will your research project take place overseas?*

- ○ Yes
- ◉ No

24. Does the research involve any other situation which may put the researcher at risk of harm (e.g. gathering data in private homes)?*

- ○ Yes
- ◉ No

Participants and Informed Consent

25. Does your research target members of a vulnerable population?
This includes, but is not limited to, children under the age of 16, people with significant mental illness, people with serious intellectual disability, prisoners, employees and students of a researcher, and people whose health, employment, citizenship or housing status is compromised. Vulnerability is a broad category and encompasses people who may lack the ability to consent freely or may be particularly susceptible to harm.*

- ○ Yes
- ◉ No

26. Have you undertaken any consultation with the groups from which you will be recruiting, regarding your method of recruitment, data collection, or your project more widely?*

- ○ Yes
- ◉ No

27. Will your participants receive any gifts/koha in return for participating?*

- ◉ Yes
- ○ No

27a. Describe the gifts/koha and the rationale.*

All individuals who participated in the interviews will be given a $10 supermarket voucher, and all will go into the draw to win a $60 supermarket voucher to thank them for their time and provide them with an incentive.

28. Will your participants receive any compensation for participation (for instance, meals, transport, or reimbursement of expenses)?*

- ○ Yes
- ◉ No

29. How will informed consent be obtained? (tick all that apply to the research you are describing in this application)*

- ☐ Informed consent will be implied through voluntary participation (anonymous research only)
- ☐ Informed consent will be obtained through a signed consent form
- ☑ Informed consent will be obtained by some other method

29a. Describe the other method*

Participants can consent over email by simply sending the consent form content to the participant in an email body and they reply to that email with "I consent".

Treaty of Waitangi

Treaty of Waitangi

30. How does your research conform to the University's Treaty of Waitangi Statute? (you can access the statute from Victoria's Treaty of Waitangi page)*

The study does not involve knowledge directly related to Te Āo Māori but care should be taken to encourage the participation of Māori under the principle of Whai wāhi (participation). To encourage this, the study will be advertised to organisations where there is more emphasis on using Te Tiriti o Waitangi as part of their core values.

Project Risks

Minimisation of Harm

31. Is it possible that participants may experience any physical discomfort as a result of the research?*

- ○ Yes
- ◉ No

32. Is it possible that participants may experience any emotional or psychological discomfort as a result of the research? (E.g. asking participants to recall upsetting events, viewing disturbing imagery.)*

- ○ Yes
- ◉ No

33. Will your participants experience any deception as a result of the research?*

- ○ Yes
- ◉ No

34. Is any third party likely to experience any special hazard/risk including breach of privacy or release of commercially sensitive information? This may occur in the instance participants are asked to discuss identifiable third parties in the research.*

- ◉ Yes
- ○ No

34a. Give details and indicate how you will manage this*

As this study involves questioning individuals on the security practices they implement in the software they develop there is a risk of exposing the company they work for. However, as outlined in question 10 of this application, there will be steps to prevent this happening by being careful of how questions are phrased. These include not publishing the interviewee and corresponding organisation name in any texts that are the result of this research.

35. Do you have any professional, personal, or financial relationship with prospective research participants? *

- ○ Yes
- ◉ No

36. What opportunity will participants have to review the information they provide? (tick all that apply)*

- ☐ Will be given a full transcript of their interview and given an opportunity to provide comments
- ☐ Will be given a full transcript of their interview and NOT given an opportunity to provide comments
- ☐ Will be given a summary of their interview
- ☑ Other opportunity
- ☐ Will not have an opportunity to review the information they provide

36a. Please give details*

Participants can be given a recording of their interview and have the opportunity to add/edit/remove details up to 1 week after the interview has taken place.

Confidentiality and Anonymity

37. Will participation in the research be anonymous?
'Anonymous' means that the identity of the research participant is not known to anyone involved in the research, including researchers themselves. It is not possible for the researchers to identify whether the person took part in the research, or to subsequently identify people who took part (e.g., by recognising them in different settings by their appearance, or being able to identify them retrospectively by their appearance, or because of the distinctiveness of the information they were asked to provide).*

- ○ Yes
- ◉ No

38. Will participation in the research be confidential?
'Confidential' means that those involved in the research are able to identify the participants but will not reveal their identity to anyone outside the research team. Researchers will also take reasonable precautions to ensure that participants' identities cannot be linked to their responses in the future.*

- ◉ Yes
- ○ No

38a. How will confidentiality be maintained in terms of access to the identifiable research data? (tick all that apply)*

- ☐ Access to the research will be restricted to the investigator
- ☑ Access to the research will be restricted to the investigator and their supervisor
- ☐ Focus groups will have confidentiality ground rules
- ☐ Transcribers will sign confidentiality forms
- ☐ Other

38b. How will confidentiality be maintained in terms of reporting of the data? (tick all that apply)*

- ☑ Pseudonyms will be used
- ☑ Data will be aggregated
- ☑ Participants will be referred to by role rather than by name
- ☐ Other

38b (i). Please provide details*

Participants in the study will not be named in any reports or summaries produced. The study uses grounded theory methodology so the people and the answers will be grouped to form a final theory on what influences programmers to make the decisions they do. The purpose of the study is not to focus on the individuals, but rather draw conclusions and similarities from their responses. Instead, the study will likely compare different groups of people based on key variables such as computing experience. Participants will not be directly quoted, and will be referred to by role or pseudonyms.

39. Will participation in the research be neither confidential nor anonymous, and participants will be identifiable in any outputs or publications relating to the research? *

- ○ Yes
- ◉ No

Data Management

Access, Storage, Use, and Disposal of Data

40. Which of the following best describes the form in which data generated in your study will be stored during the study?
See help text for guidance on these terms. Further info available on human ethics website.*

- ○ Identifiable
- ○ Potentially identifiable
- ◉ Partially de-identified
- ○ De-identified
- ○ Anonymous
- ○ Other

41. Which of the following best describes the form in which data generated in your study will be stored after the study is completed?
See help text for guidance on these terms. Further info available on human ethics website.*

- ○ Identifiable
- ○ Potentially identifiable
- ○ Partially de-identified
- ◉ De-identified
- ○ Anonymous
- ○ Other

42. Proposed date for destruction of identifiable research data (i.e. the date when data will be de-identified and personal information on participants destroyed)

*

31/12/2021

43. Proposed date for destruction of de-identified research data, including anonymous data

*

31/12/2021

44. Will any research data be kept for longer than 5 years after the conclusion of the research?*

- ○ Yes
- ◉ No

45. Who will have access to identifiable, de-identified or anonymous data, both during and at the conclusion of the research?*

- ○ Access restricted to the researcher only (whoever is named as PI)
- ◉ Access restricted to researcher and their supervisor
- ○ Access restricted to researcher and immediate research team, e.g. co-investigators, assistants
- ○ Other

46. Are there any plans to re-use either identifiable, de-identified or anonymous data?*

- ○ Yes
- ◉ No

47. What procedures will be in place for the storage of, access to and disposal of data, both during and at the conclusion of the research? (Check all that apply)
Information regarding appropriate data storage is available on the human ethics website. Note that storing research data on USB drives is strongly discouraged for security reasons. *

- ☑ All hard copy material will be stored securely e.g. in a locked filing cabinet
- ☑ All electronic material will be held securely, e.g. only on University servers, password protected
- ☑ All hard copy material will be appropriately destroyed (e.g. shredded) on the dates given above
- ☑ All electronic data will be deleted on the dates given (ITS should be consulted on proper method)

Dissemination

Dissemination

48. How will you provide feedback to participants?*

I will offer participants the opportunity to request for a copy of their interview recording.

49. How will results be reported and published? Indicate which of the following are appropriate. The proposed form of publications should be indicated to participants on the information sheet and/or consent form*

- ☑ Publication in academic or professional journals
- ☑ Dissemination at academic or professional conferences
- ☑ Availability of the research paper or thesis in the University Library and Institutional Repository
- ☐ Other

50. Is it likely that this research will generate commercialisable intellectual property?
*(Click the ? icon for more info)**

○ Yes
◉ No

Supporting Documents

Documents

51. Please upload any documents relating to this application. Sample documents are available on the Human Ethics web page.

- Ensure that your files are small enough to upload easily, and in formats which reviewers can easily download and review.
- To upload a document click on the green arrow to the right of the named document. Follow the on screen instructions which will be displayed to upload a document.
- To replace a document, click the tick in the column to the right of the document you want to replace, and follow the screen instructions to continue.
- To add a new document click on 'New Document', at top right of the documents table. You **must** enter the document name in the box that appears. Click on 'OK'. Click on the green arrow which appears to the right of the file name to continue.
- Collate all your documents into one PDF or Word file, and upload as a new document. This should be labelled as 'Combined Documents'.

*

Description	Reference	Soft copy	Hard copy
Participant information sheet(s)	Participant_Information_Sheet.pdf	✓	
Participant consent form(s)	Consent_to_Interview.pdf	✓	
Interview questions or guide	Interview_Guide.pdf	✓	
Application Changes	Changes_Summary.docx	✓	
Recruitment Posts	Recruitment_Post.pdf	✓	

Application Sign Off

Application Feedback

Feedback

This page will be used to provide feedback between the Research Office and the researcher/s during the application review process.

Committee response to your application

Date

15/05/2020

Comments

Hi,
Please refer doc attached.
Thanks

Documents

28506_Sajwan_Revise and Resubmit.docx

Researcher response to requirements: please enter your comments below and then save.

This question is not answered.

If you have been asked to provide a response to the committee's comments in a document, please upload that document here and then **save**.

Changes_Summary.docx

Further required changes

Date

This question is not answered.

Comments

This question is not answered.

Documents

This question is not answered.

Researcher response to requirements: please enter your comments below and then save.

This question is not answered.

If you have been requested to respond to further feedback by uploading documentation here, please upload and then **save**.

This question is not answered.

Final Review Outcome

Final Review Outcome

This question is not answered.

Formal Notice of Approval

This question is not answered.

Risk Category

Based on answers to the screening questions (Q4), your application will be assessed as either Category A (higher than low risk), or Category B (low risk). The risk category will be reviewed by the Research Office before it is processed by the committee.

The risk assessment for this application is currently:

Category B (LOW RISK)

Incident Report (admin only)

Incident Reporting

Incident Report

Adverse incidents are instances of potential or actual physical harm to participants or researchers; emotional harm or distress to participants or researchers; and any other unforeseen events that raise ethical issues.

Research teams must immediately advise the Human Ethics Committee if an adverse incident occurs in the course of their research project.

An incident report form can be obtained by sending an email to Ethics Administrator

The full incident report should be returned by email, and the committee administrator will upload it to this application

For Admin Use

Is there an incident to report?

☐ Yes

This question is not answered.

Do you have a second incident to report?

☐ Yes

This question is not answered.

Appendix B

Participant Information Sheet

Why do Programmers Do What They Do?

INFORMATION SHEET FOR PARTICIPANTS

You are invited to take part in this research. Please read this information before deciding whether or not to take part. If you decide to participate, thank you. If you decide not to participate, thank you for considering this request.

Who am I?

Kia ora koutou! My name is Lavanya Sajwan and I am an Honours student in the school of engineering and computer science at Victoria University of Wellington. This research project is work towards partial fulfilment of the requirements for Bachelor of Engineering with Honours in Software Engineering.

What is the aim of the project?

This project is an informative investigation on how software programmers implement and adopt security practices in the work they do in order to develop an understanding of what influences and impact decisions surrounding their technical work. Your participation will support this research by providing insight to the solutions that programmers are using to implement in their security practices. Essentially this research aims to gather your experiences and compare them with those of your peers to understand the use of security practices in the New Zealand industry. This research has been approved by the Victoria University of Wellington Human Ethics Committee (Research Master Application ID: 0000028506).

How can you help?

You have been invited to participate because you are a professional programmer in the industry, familiar with working with security practices, and are over the age of 18 and this project aims to develop a theory as to why programmers implement and adopt security practices in the work, they do by interviewing professional programmer. It is important to gather data from individuals with varying career timelines and progressions as the data can be thoroughly analysed to form connections. I will interview you at Victoria University of Wellington's Kelburn campus, via Zoom or at another venue of your choice. I will ask you questions about your security practices and habits in your day-to-day work and how you make decisions regarding your methods of choice. It is advised that you refrain from naming any 3rd parties. If you agree to take part the interview will take 30-60 minutes of your time. I will audio record the interview with your permission and write it up later. You can choose to not answer

any question or stop the interview at any time, without giving a reason. You can withdraw from the study up to 1 week after the interview has taken place. If you withdraw, the information you provided will be destroyed. You may also choose to receive a copy of the interview recording which will be emailed to you, and you will have the opportunity to edit, appended and remove details up to 1 week after receiving the recording. You will be given a $10 supermarket voucher and go into the draw for a $60 supermarket voucher as koha to thank you for your time.

What will happen to the information you give?

This research is confidential, but may be limited due to small sample and nature of the research. You and your organisation will not be named in the final report, but title may be named and persons within or familiar with your organisation and title may be able to identify you and your organisation based on the distinctiveness of the information you provide. However, you may choose to not have any potential identifying information published.

Please do not name any 3rd parties and reveal any business secrets and passwords.

Your name and email address will be used to contact you in the event of winning the overall prize, and when a recording of your interview is sent if you choose to receive it.

Only my supervisors and I will read the notes of the interview. The interview summaries and any recordings will be kept securely and destroyed on the 31st of December 2021.

What will the project produce?

The information from my research will be used in my Honours report and presentation. You will not be identified in either of these materials or in any supplementary reports such as publications in academic or professional journals.

If you accept this invitation, what are your rights as a research participant?

You do not have to accept this invitation if you don't want to. If you do decide to participate, you have the right to:

- choose not to answer any question;
- ask for the recorder to be turned off at any time during the interview;
- withdraw from the study up to 1 week after the interview has taken place;
- ask any questions about the study at any time;
- receive a copy of your interview recording;
- edit/append/remove any details up to 1 week after the interview.
- be able to read any reports of this research by emailing the researcher to request a copy.

If you have any questions or problems, who can you contact?
If you have any questions, either now or in the future, please feel free to contact any of either one of the people listed:

Student Researcher:

Name: Lavanya Sajwan
University email address: sajwanlava@myvuw.ac.nz

Primary supervisor:

Name: James Noble
Role: Professor of Computer Science, Associate Dean (Postgraduate Research)
School: Engineering and Computer Science, Victoria University of Wellington
Phone: 04 463 6736
kjx@ecs.vuw.ac.nz

Secondary supervisor:

Name: Craig Anslow
Role: Senior Lecturer
School: Engineering and Computer Science, Victoria University of Wellington
Phone: 04 463 6449
craig.anslow@ecs.vuw.ac.nz

Human Ethics Committee information

If you have any concerns about the ethical conduct of the research you may contact the Victoria University HEC Convenor: Dr Judith Loveridge. Email hec@vuw.ac.nz or telephone +64-4-463 6028.

Appendix C

Participant Consent Form

Why do Programmers Do What They Do?

CONSENT TO INTERVIEW

This consent form will be held for 5 years.

Researcher: Lavanya Sajwan, School of Engineering and Computer Science, Victoria University of Wellington.

- I have read the Information Sheet and the project has been explained to me. My questions have been answered to my satisfaction. I understand that I can ask further questions at any time. The end research date is: 28th of February 2021.
- I agree to take part in an audio recorded interview.

I understand that:

- I may withdraw from this study up to 1 week after the interview has taken place and any information that I have provided will be destroyed.
- Any information gathered will be securely stored on an ECS lab machine and any information I have provided will be destroyed on the 31st of December 2021.
- I understand that the findings may be used for an Honours report and a summary of the results may be used in academic reports and/or presented in conferences.
- I understand that the any information I provide will be kept confidential to the researcher and the supervisors.
- My name and my organisation name will not be used in reports, but persons within or familiar with the organisation practices may be able to identify me based on distinctiveness of the information I provide.
- I have the right to edit/append/remove details up to 1 week after the interview.
- I would like a copy of the recording of my interview: Yes ☐ No ☐
- I would like to receive a copy of the final report and have added my email address below. Yes ☐ No ☐

Signature of participant: _____

Name of participant: _____

Date: _____

Contact details: _____

Appendix D

Interview Template

Why do Programmers Do What they Do? – Potential Interview Questions

Participant Background

- How many years' experience do you have?
- What is your current role and what does this entail?
- Please describe any security education you had before working at your current job?
- Would you like to tell me anything more about your background or experience in the industry?

Current Work

- Can you give a brief overview of the security features (protocols, frameworks, libraries, tools) that you regularly use at work?
- Do you know those practices have been tested within your team/organisation?
- Do you change languages based on the security practice you use? Why?
- Please share if you and your team have adapted any of the approaches to suit your work better
- Does your work provide any training, or is it expected that you how to program "securely"?
- What are some difficulties in joining a team with an established process?
- How does security fit in the development life-cycle in real life?

Impacts

- How have your chosen practices affected your ability to deliver within constraints?
- Have you experienced any data/security breaches as a consequence of your work and how was it managed?
- Have there been any lessons learned in your own personal projects and style?

Experience-Based (Dependent on role and years of experience)

- What recommendations would you generally give to teams looking to adopt your security practice of choice?

- Do you give other teams/individuals feedback on their security practices?
- What differences do you see between new and experienced people?
- What common “mistakes” do you see?
- What factors contribute to a successful education on how to use security practices?
- What difficulties did the teams you have worked with share?
- What changes have you observed in the security programming landscape in New Zealand?
- What are the current motivators and deterrents to programmers in terms of paying attention to security?

- IF FROM OVERSEAS OR WORKED OVERSEAS, WHAT DIFFERENCES HAVE THEY NOTICED IN SECURITY IN WORKPLACES IN NZ COMPARED TO OVERSEAS.

Any other issues or comments?

Appendix E

Recruitment Post

Groups/Mailing List Post:

Kia ora koutou!

My name is Lavanya Sajwan and I am a Software Engineering student at Te Herenga Waka - Victoria University of Wellington. I am working on an honours project this year which is about security practices in industry and investigating the influences impacting the decisions of the choice that developers make. This research has been approved by the Victoria University of Wellington Human Ethics Committee (Research Master Application ID: 0000028506).

I am looking to interview professional programmers who work with security practices, and will need to be over 18 years old and New Zealand based. Participants can also work across a range programming fields eg. gaming, education, financial, analysis.

If this is you and are interested in participating in this study please send me an email stating your interest including name and job title.

If you participate in the interview, you will be given koha of a $10 voucher, and go into the draw to win a $60 supermarket voucher to thank you for your time!

If you are interested in participating or hearing more about the study please contact me over email – sajwanlava@myvuw.ac.nz

Ngā mihi

Appendix F

Recruitment Webpage