Konzeption einer Unterstützung für Softwareentwickler im Bereich IT-Sicherheit mit Hilfe mentaler Modelle

Heike MÄRKI, Bettina ABENDROTH

Institut für Arbeitswissenschaft, Technische Universität Darmstadt
Otto-Berndt-Straße 2, D-64287 Darmstadt

Kurzfassung: Dem Fortschritt durch Digitalisierung stehen steigende Fallzahlen im Bereich von Cybercrime gegenüber. Schwachstellen der IT-Sicherheit, z. B. durch falsche Anwendung und Implementierung eigentlich sicherer Kryptographie, bieten Angreifern die Möglichkeit großen Schaden anzurichten. Grund ist häufig ein fehlendes gemeinsames Verständnis von Kryptographie-Experten als Designer von Sicherheitslösungen und Softwareentwicklern als deren Nutzer. Ziel der hier vorgestellten Arbeit war die Erarbeitung einer Unterstützung für Softwareentwickler bei der Auswahl und Integration geeigneter Sicherheitslösungen. Es fanden hierfür Erhebungen der mentalen Modelle von Softwareentwicklern und Kryptographie-Experten, sowie Interviews statt. Die gewonnenen Erkenntnisse wurden zur Erarbeitung eines Prototyps in Form einer Webseite verwendet.

Schlüsselwörter: Mentale Modelle, Softwareentwicklung, Kryptographie, Prototyp, Usability

1. Motivation

Cyberkriminalität ist ein globales wirtschaftliches Problem, das der Weltwirtschaft Schäden in Höhe von mehr als 600 Billionen US-Dollar (CSIS 2018) und der deutschen Wirtschaft in Höhe von 55 Milliarden Euro pro Jahr verursacht (Bitkom 2017). Eine der Hauptursachen für Verletzungen der Sicherheit stellen Software-Schwachstellen dar (De Win et al. 2002; Pohlmann 2017). Nach Angaben des Bundesamtes für Sicherheit in der Informationstechnik wurden im Jahr 2015 beispielsweise 847 kritische Schwachstellen in den elf häufigsten Softwareprodukten gefunden (BSI 2016). In Zeiten voranschreitender Digitalisierung und Vernetzung sowohl im Arbeits- als auch im privaten Kontext führt dies zu einem erhöhten Bedrohungspotential (BKA 2017). Ein Grund hierfür wird darin gesehen, dass nicht alle Softwareentwickler über ausreichend Kenntnisse und Erfahrung im Kontext von Sicherheit verfügen (De Win et al. 2002; BSI 2014). Softwareentwicklung ist eine sich ständig verändernde Branche. Die Technologie schreitet schnell voran. Geschäftsanforderungen ändern sich ständig und entwickeln sich weiter. Es herrscht ein hoher Produktivitätsdruck und ein ständiger Bedarf an Innovationen sowie rasche, spezifische Reaktionen (Nandico 2011). Zusätzlich steigt die Streuung, Raffinesse und Professionalität von Angriffen (Pohlmann 2017). Dies führt dazu, dass das Berufsbild der Softwareentwickler sich ändert (Nandico 2011) und stetige Weiterbildung angeraten ist. Das übergeordnete Ziel dieser Forschung besteht darin, ein Unterstützungstool zu entwickeln, welches Softwareentwickler im Kontext von Sicherheitslösungen und Kryptographie unterstützt. Im Fokus steht dabei ein gemeinsames Verständnis, bzw. mentales Modell (Yu & Petter 2014) von Kryptographie-Experten als Designer von

Sicherheitslösungen und Softwareentwicklern, die in diesem Kontext als Benutzer angesehen werden können (Besnard & Arief 2004). Sie sind ebenso wie Administratoren oder Endbenutzer Teil des Systems und wirken sich auf dessen Sicherheit aus.

2. Methode

Im Produktentwicklungsprozess ist eine frühzeitige Einbindung der Benutzer unerlässlich (Butz & Krüger 2014; Kujala 2003), um die Benutzerfreundlichkeit zu verbessern (Bekker & Long 2000) und um zusätzliche Kosten durch spätere Anpassungen zu vermeiden (Noyes et al. 1996). Im Rahmen dieser Arbeit wurde sich am menschzentrierten Gestaltungsprozess aus DIN EN ISO 9241-210 orientiert. Aus diesem Grund wurden in einem ersten Schritt die mentalen Modelle von Softwareentwicklern erhoben um den Nutzungskontext verstehen und beschreiben zu können. Im Zuge eines gemeinsamen Verständnisses wurden diese mit denen im Rahmen einer weiteren Studie erhobenen mentalen Modellen von Kryptographie-Experten verglichen und aus diesem Vergleich Nutzungsanforderungen und Gestaltungshinweise für eine Unterstützung von Softwareentwicklern bei der Auswahl und Integration kryptographischer Lösungen abgeleitet. Im Anschluss daran wurden Interviews mit weiteren Softwareentwicklern durchgeführt, deren Fokus ausschließlich auf der Ableitung von Gestaltungshinweisen lag. Basierend auf den gewonnenen Erkenntnissen wurde dann ein erster Prototyp in Form einer Webseite erarbeitet, der im Sinne eines iterativen Gestaltungsprozesses von Softwareentwicklern getestet und bewertet wurde. Die einzelnen Schritte sind im Folgenden zusammengefasst dargestellt.

2.1 Erfassung mentaler Modelle von Softwareentwicklern

Falsche Denkmodelle sind oft der Grund für ineffiziente Problemlösung und unerwünschte Ergebnisse (van Boven & Thompson 2003). Ziel einer ersten Studie war es deshalb, die mentalen Modelle von Softwareentwicklern zu den Themen *Sicherheit*, *Angriffe* und *Kryptographie* zu erfassen. In Anlehnung an Märki et al. (2016) werden als mentale Modelle im Rahmen dieser Forschung dynamische und funktionale Repräsentationen der Realität verstanden, deren Zuverlässigkeit mit zunehmenden Wissen im jeweiligen Kontext zunimmt.

Zur Erfassung der mentalen Modelle wurde, im Rahmen der in Märki et al. (2016) ausführlich dargestellten Studie, die sogenannte Heidelberger Struktur-Lege-Technik (Scheele & Groeben 1984) verwendet. Ergebnis der Studie sind demnach bildliche Darstellungen der mentalen Modelle der fünf befragten Softwareentwickler.

2.2 Vergleich mentaler Modelle von Softwareentwicklern mit denen von Experten

In einer Folgestudie wurden, als nächster Schritt bezüglich der Erarbeitung einer Unterstützung für Softwareentwickler, die mentalen Modelle von Kryptographie-Experten mit denen der Softwareentwickler aus der ersten Studie verglichen. Zur Erfassung der mentalen Modelle der Experten wurde die gleiche Methode verwendet. Der verwendete Interviewleitfaden wurde dafür überarbeitet und um Fragen gekürzt, die entweder in der ersten Studie gar nicht verwendet wurden oder in keiner Weise zur Modellerstellung beigetragen haben.

Im Rahmen der Interviews, die zwischen 50 und 65 min dauerten, wurden die Teilnehmer zunächst über den Umgang mit ihren Daten und den Hintergrund der

Studie aufgeklärt. Die Interviews, die entsprechend der ersten Studie in die Themengebiete *Sicherheit*, *Angriffe* und *Kryptographie* strukturiert waren, wurden mittels Tonaufnahme aufgezeichnet. Diese Aufnahmen dienten im weiteren Verlauf zur Transkription und Auswertung. Die Auswertung gliederte sich in drei Schritte. Zunächst wurde der wörtliche Text im Detail analysiert um Konzepte und verwendete Relationen ableiten zu können. Diese wurden in einem nächsten Schritt an das Regelwerk der Methode der Heidelberger Struktur-Lege-Technik (Scheele & Groeben 1984) angepasst und schließlich als Struktur gelegt. Im Rahmen eines zweiten Termins nach zwei Wochen wurden, basierend darauf, zusammen mit jedem der Experten jeweils drei resultierende Strukturen der entsprechenden mentalen Modelle gelegt.

Es konnten sechs Kryptographie-Experten (vier männliche und zwei weiblich) zur Teilnahme gewonnen werden, deren Erfahrung auf diesem Gebiet mit mindestens drei Jahren, bis zu mehr als neun Jahren als ausreichend angesehen wird.
Für den anschließenden Vergleich der erhaltenen mentalen Modelle wurden jeweils die der Kryptographie-Experten und die der Softwareentwickler nebeneinandergelegt und thematisch zusammengehörige Bereiche farbig hervorgehoben und anschließend bezüglich Gemeinsamkeiten und Unterschieden analysiert. Es ergaben sich daraus unterschiedliche Gestaltungshinweise, die zu acht Design Guidelines zusammengefasst wurden. Diese beziehen sich auf eine Unterstützung in Form einer Webseite.

2.3 Halb-standardisierte Interviews mit Fokus auf der Ableitung von Gestaltungshinweisen

Um Gestaltungshinweise speziell in Bezug auf ein Unterstützungstool in Form einer Webseite zu erhalten, wurden im dritten Schritt im Rahmen dieser Forschung Interviews mit Softwareentwicklern durchgeführt. Hierfür wurde zunächst ein Interviewleitfaden erarbeitet, der nach wünschenswerten Inhalten und Eigenschaften eines solchen Tools und nach allgemeinen Hinweisen bezüglich der Verbesserung des Softwareentwicklungsprozesses fragt. Insgesamt konnten fünf Softwareentwickler hierfür akquiriert werden, deren Alter zwischen 24 und 39 und deren Berufserfahrung zwischen 2 und 9 Jahren lag.

2.4 Integration der gefundenen Gestaltungshinweise zu einem Unterstützungsprototypen in Form einer Webseite

Die in den Vorarbeiten gefundenen Gestaltungshinweise bezüglich eines Unterstützungstools wurden als nächstes für die Entwicklung eines Prototyps in Form einer Webseite verwendet. Nach einer inspektionsbasierten Evaluierung durch drei Probanden ohne Erfahrung in der Softwareentwicklung bewerteten drei Softwareentwickler die Webseite im Anschluss im Rahmen eines Usability Tests und eines Fragbogens. Die Probanden wurden im ersten Teil gebeten einige Aufgaben auf der Webseite zu erfüllen, wie z. B. „Um Schwachstellen in der eigenen Software zu finden, können Penetrationstests genutzt werden. Finden Sie einen Leitfaden, den Sie für die Durchführung von Penetrationstests nutzen können.“ Während der Bearbeitung sollte die Methode des *lauten Denkens* angewendet werden. Neben der Beobachtung wurde das Gesagte mittels Tonaufnahmegerät festgehalten und die Bearbeitungszeit gestoppt. Im Anschluss daran erfolgte eine Bewertung der Webseite mittels eines Fragebogens.

3. Ergebnisse

Die Ergebnisse der ersten Studie zur Erfassung der mentalen Modelle der Softwareentwickler wurden bereits in Märki et al. (2016) dargestellt. Aus diesem Grund soll an dieser Stelle mit dem Vergleich der mentalen Modelle der Softwareentwickler mit denen der Kryptographie-Experten begonnen werden. Die interessantesten Erkenntnisse hierzu waren, dass beide Gruppen sehr abstrakte Definitionen für Sicherheit verwendeten. Im Kontext von *Sicherheit* zeigte sich deutlich eine höhere Komplexität der mentalen Modelle der Experten. Darüber hinaus ist erwähnenswert, dass mehrere Experten es als notwendig ansehen, dass Softwareentwickler unterstützt werden und die eigene Gruppe der Experten dabei in der Verantwortung sehen auf die Entwickler zuzugehen. Die Thematik einer solchen Unterstützung lässt sich dagegen in keinem der mentalen Modelle der Softwareentwickler finden.

Beide Gruppen betonen die konträren Ziele von Funktionalität und Effizienz auf der einen und Sicherheit auf der anderen Seite. Die Experten geben dies als notwendigen Trade-Off an, während die Softwareentwickler die Sicherheit nur als untergeordnetes Ziel ansehen. Allerdings enthalten die Modelle der Entwickler im Gegensatz zu denen der Experten praktische Einschränkungen der Sicherheit, wie Zeit- und Kostendruck Kundenwünsche und -ansprüche. Auch in Bezug auf *Angriffe* und *Kryptographie* stellen sich die Modelle der Experten erwartungsgemäß konkreter, vielfältiger und strukturierter dar, als die der Softwareentwickler. Hier zeigte sich, dass die Experten betonen, dass zu Beginn des Entwicklungsprozesses die Sicherheit in Bezug auf das Gesamtkonzept betrachtet und die Kryptographie in dieses integriert werden muss. In den mentalen Modellen der Softwareentwickler erscheint Kryptographie dagegen als modulare Softwarekomponente, die auch per „copy + paste“ eingebaut werden kann. Davon raten die Experten wiederum ab und erwähnen die Wichtigkeit einer behutsamen Implementierung. In der Frage danach, welche Kriterien bei der Auswahl einer geeigneten kryptographischen Lösung herangezogen werden, stimmen beide Gruppen überein. Hier werden beiderseits Kriterien wie z. B. die Häufigkeit der Verwendung der Lösung in der Praxis, die jeweilige Programmiersprache und der Urheber der jeweiligen Lösung genannt.

Aus dem Vergleich und vor allem aus den gesamten mentalen Modellen konnten einige Hinweise für die notwendigen Inhalte einer Unterstützung abgeleitet werden, wie beispielsweise, dass ein Überblick über Theorie (z. B. Definitionen oder Argumentationsgrundlage in Bezug auf konträre Ziele), eine Austauschmöglichkeit für Softwareentwickler und Kryptographie-Experten und eine Sammlung von bekannten Gefahren, Fehlern und Maßnahmen gegeben werden sollte.

Die durchgeführten Interviews mit Softwareentwicklern mit dem Fokus auf eine zu erarbeitende Unterstützung machten es möglich weitere mögliche Inhalte eines solchen Tools zu identifizieren. Häufig gewünscht wurde eine To-Do-Anleitung, ein Leitfaden oder sogar eine automatische Auswahl passender kryptographischer Algorithmen auf Basis abgefragter Kriterien z. B. der Höhe des akzeptierten Performanz Verlustes oder Speicherverbrauchs. Die Softwareentwickler würden bezüglich Sicherheitslücken gerne auf dem neusten Stand gehalten werden. Insgesamt sollen sämtliche Informationen dabei möglichst verständlich und einfach dargestellt werden. Auch die Möglichkeit der Testung einer verwendeten Lösung sollte durch das zu erarbeitende Tool unterstützt werden.

Auf Basis der gesammelten Hinweise wurde im nächsten Schritt ein erster Prototyp der Unterstützung in Form einer Webseite erarbeitet. Dieser behielt die Gliederung in die Themen *Sicherheit*, *Angriffe* und *Kryptographie* bei. So lassen sich unter

dem Menüpunkt *Sicherheit* z. B. Sicherheitsmodelle, Methoden zum Aufzeigen von Sicherheitslücken oder Informationen zu Zugriffsrechten, bzw. -kontrollen finden. Das zum Thema *Angriffe* zugeordnete Menü führt zu Informationen über unterschiedliche Arten von Angriffen und Schadenseingrenzungen, während die Inhalte des Menüpunktes *Kryptographie* z. B. Grundwissen, Leitfäden zur Implementierung kryptographischer Algorithmen oder Auswahlkriterien für standardisierte Bibliotheken enthält. Darüber hinaus bietet der Menüpunkt *Tools/Links* die Möglichkeit diese genannten Inhalte übersichtlich aufbereitet und ergänzt mit weiteren Unterstützungsmöglichkeiten, wie andere Webseiten oder entsprechende Literatur zu finden. Eine Suchfunktion, ein Forum und für den Austausch zu Verfügung gestellter Speicherplatz runden das Angebot ab. Dieses Grobkonzept wurde mittels ausführlicher Recherche mit entsprechenden exemplarischen Inhalten befüllt und zunächst im Rahmen einer inspektionsbasierten Evaluierung überprüft und entsprechend angepasst.

Im Anschluss daran wurde ein Usability-Test mit drei Softwareentwicklern durchgeführt. Dieser führte zu Gesamtbewertungen von „gut“ (n=2) und „befriedigend“ (n=1). Alle Teilnehmer gaben an, diese Webseite beruflich nutzen zu wollen und weiterzuempfehlen. Alle Teilnehmer vermissten darauf aber jeweils Kurzübersichten über die Inhalte der unterschiedlichen Menüpunkte, da Softwareentwickler mit geringer Erfahrung sich unter den Bezeichnungen allein eventuell nichts vorstellen könnten und ein kurzer Überblick auch erfahrenen Entwicklern ermöglichen würde, sich schnell und effizient zu orientieren. In diesem Sinne wurden darüber hinaus eine Hilfefunktion, eine FAQ-Sektion und eine besser ausgebaute Suchfunktion, bzw. auch eine Auflistung der Inhalte von A-Z zur manuellen Suche, vorgeschlagen.

4. Diskussion und Ausblick

Das Vorgehen der Produktentwicklung auf Basis der Erhebung der mentalen Modelle bietet sowohl Vor- als auch Nachteile. So sind die sich ergebenden Inhalte eventuell vielschichtiger, strukturierter und umfassender, als die Ergebnisse, z. B. auf Basis von Interviews. Andererseits handelt es sich jeweils um eine individuelle Struktur, die erst mit anderen mentalen Modellen in den richtigen Zusammenhang gebracht werden muss. Young (2008) schlägt hierzu eine Erfassung der mentalen Modelle direkt im Bezug auf ein zu entwerfendes Produkt vor. Die Webseite als konkretes „Produkt“ ergab sich im Rahmen dieser Arbeit allerdings erst im Laufe der Erhebungen.

In Bezug auf die Erstellung einer solchen Webseite in diesem Kontext wäre es natürlich effizienter die Gestaltung und Ausarbeitung der inhaltlichen Darstellung in professionelle Hände zu geben. Es stellt sich dabei allerdings generell die Frage nach Übernahme der Verantwortung auch in Zukunft, was Wartung und Aktualität der Seite betrifft. Interessant könnten die im Rahmen dieser Forschung erhaltenen Erkenntnisse für Anbieter ähnlicher Webseiten für Anpassungen ihres Angebotes sein. Als Beispiele seien hier das BSI oder The Open Web Application Security Project (OWASP) genannt. Darüber hinaus sei erwähnt, dass nahezu alle Softwareentwickler die Wichtigkeit der Integration von Sicherheitsaspekten in Aus- und Weiterbildung erwähnten, um mit dem digitalen Wandel, der auch diese Berufsgruppe sehr betrifft, besser umgehen zu können.

5. Literaturverzeichnis

Bekker, Mathilde; Long, John (2000): User involvement in the design of Human—Computer interactions: Some similarities and differences between design approaches. In: People and Computers XIV—Usability or Else!: Springer, S. 135–147.

Besnard, Denis; Arief, Budi (2004): Computer security impaired by legitimate users. In: Computers & Security 23 (3), S. 253–264.

Bitkom (2017): Spionage, Sabotage, Datendiebstahl: Deutscher Wirtschaft entsteht jährlich ein Schaden von 55 Milliarden Euro. Online verfügbar unter https://www.bitkom.org/Presse/
Presseinformation/Spionage-Sabotage-Datendiebstahl-Deutscher-Wirtschaft-entsteht-jaehrlich-ein-Schaden-von-55-Milliarden-Euro.html, zuletzt geprüft am 11.12.2018.

BKA (2017): Cybercrime. Bundeslagebild 2017. Bundeskriminalamt. Online verfügbar unter https://www.bka.de/SharedDocs/Downloads/DE/Publikationen/JahresberichteUndLagebilder/Cybercrime/cybercrimeBundeslagebild2017.html?nn=28110, zuletzt geprüft am 11.12.2018.

BSI (2014): Software-Schwachstellen oder -Fehler. Online verfügbar unter https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/g/g04/g04022.html, zuletzt geprüft am 11.12.2018.

BSI (2016): Die Lage der IT-Sicherheit in Deutschland 2015. Bundesamt für Sicherheit in der Informationstechnik. Online verfügbar unter https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/
Publikationen/Lageberichte/Lagebericht2015.pdf?__blob=publicationFile&v=5, zuletzt geprüft am 11.12.2018.

Butz, Andreas; Krüger, Antonio (2014): Mensch-Maschine-Interaktion. München: Oldenbourg Wissenschaftsverlag.

CSIS (2018): Economic Impact of Cybercrime. Center for Strategic & International Studies. Online verfügbar unter https://www.csis.org/analysis/economic-impact-cybercrime, zuletzt geprüft am 11.12.2018.

De Win, Bart; Piessens, Frank; Joosen, Wouter; Verhanneman, Tine (2002): On the importance of the separation-of-concerns principle in secure software engineering. In: Workshop on the Application of Engineering Principles to System Security Design, S. 1–10.

Kujala, Sari (2003): User involvement: a review of the benefits and challenges. In: Behaviour & Information Technology 22 (1), S. 1–16, zuletzt geprüft am 03.03.2016.

Märki, Heike; Maas, Miriam; Kauer-Franz, Michaela, Oberle, Marius (2016): Increasing Software Security by Using Mental Models. In: Denise Nicholson (Hg.): Advances in Human Factors in Cybersecurity. Proceedings of the AHFE 2016 International Conference on Human Factors in Cybersecurity, Bd. 501. Walt Disney World®, Florida, USA, 27.-31.7., S. 347–359.

Nandico, Oliver F. (2011): Effizientere Software-Entwicklung durch Industrialisierung der Prozesse. In: Jochen Ludewig und Axel Böttcher (Hg.): Software Engineering im Unterricht der Hochschulen. SEUH. München, 24.-25.2., S. 3–8.

Noyes, J. M.; Starr, A. F.; Frankish, C. R. (1996): User involvement in the early stages of the development of an aircraft warning system. In: Behaviour & Information Technology 15 (2), S. 67–75.

Pohlmann, Norbert (2017): Cyber Security – 10 aktuelle Problemfelder (01), S. 24–27. Online verfügbar unter https://norbert-pohlmann.com/app/uploads/2017/07/356-Cyber-Security-10-aktuelle-Problemfelder-Problembewusstsein-muss-zun%C3%A4chst-entwickelt-werden-Prof.-Norbert-Pohlmann.pdf, zuletzt geprüft am 11.12.2018.

Scheele, Brigitte; Groeben, Norbert (1984): Die Heidelberger Struktur-Lege-Technik (SLT): ein Dialog-Konsens-Methode zur Erhebung subjektiver Theorien mittlerer Reichweite: Beltz.

van Boven, Leaf; Thompson, Leigh (2003): A look into the mind of the negotiator: Mental models in negotiation. In: Group Processes & Intergroup Relations 6 (4), S. 387–404.

Young, Indi (2008): Mental models: Aligning design strategy with human behavior. Brooklyn, New York: Rosenfeld Media.

Yu, Xiaodan; Petter, Stacie (2014): Understanding agile software development practices using shared mental models theory. In: Information and Software Technology 56, S. 911–921.

Danksagung: Ein ganz besonderer Dank gilt Frau Miriam Maas, Frau Ernestine Dickhaut, Herrn Christian Friedrich und Herrn Markus Weidmann für die Unterstützung im Rahmen der in dieser Studie beschriebenen Erhebungen.

Gesellschaft für
Arbeitswissenschaft e.V.

Arbeit interdisziplinär
analysieren – bewerten – gestalten

65. Kongress der
Gesellschaft für Arbeitswissenschaft

Professur Arbeitswissenschaft
Institut für Technische Logistik und Arbeitssysteme
Technische Universität Dresden

Institut für Arbeit und Gesundheit
Deutsche Gesetzliche Unfallversicherung

27. Februar – 1. März 2019

GfA-Press

Bericht zum 65. Arbeitswissenschaftlichen Kongress vom 27. Februar – 1. März 2019

Professur Arbeitswissenschaft, Institut für Technische Logistik und Arbeitssysteme, Technische Universität Dresden;
Institut für Arbeit und Gesundheit, Deutsche Gesetzliche Unfallversicherung, Dresden

Herausgegeben von der Gesellschaft für Arbeitswissenschaft e.V.
Dortmund: GfA-Press, 2019
ISBN 978-3-936804-25-6

NE: Gesellschaft für Arbeitswissenschaft: Jahresdokumentation

Als Manuskript zusammengestellt. Diese Jahresdokumentation ist nur in der Geschäftsstelle erhältlich.

Schriftleitung: Matthias Jäger
im Auftrag der Gesellschaft für Arbeitswissenschaft e.V.

Screen design und Umsetzung
© 2019 fröse multimedia, Frank Fröse
office@internetkundenservice.de · www.internetkundenservice.de