

Inhaltsverzeichnis

1	Einleitung	1
1.1	Entwicklung sicherer Informationssysteme	1
1.2	Problembeschreibung	5
1.3	Forschungsfragen und Methodik	6
1.4	Einordnung der Arbeit in das DFG-Projekt SecVolution	7
1.5	Aufbau der Arbeit	10
2	Grundlagen	13
2.1	Entwicklung sicherer Software	13
2.1.1	Grundlegende Begriffe	13
2.1.2	Softwareentwicklungsprozesse	15
2.1.3	Ausgewählte Techniken zur Sicherheitsanalyse	17
2.2	Wissensmanagement und Sicherheitswissen	19
2.2.1	Wissensmanagement im Software Engineering	19
2.2.2	Ressourcen für Sicherheitswissen	22
2.3	Experten- und Kritiksysteme	23
2.4	Computerlinguistik	26
2.4.1	Syntax, Semantik und Pragmatik	26
2.4.2	Verarbeitung von natürlichsprachlichen Texten	27
2.4.3	Ressourcen für maschinelle Sprachverarbeitung	31
2.4.4	Semantische Ähnlichkeit von Wörtern	33
3	Heuristischer Lösungsansatz	35
3.1	Artefakte in der Softwareentwicklung	36
3.2	Bewertungsheuristiken für Artefakte	37
3.2.1	Allgemeine Begriffserklärung	37
3.2.2	Modelltheoretischer Ansatz	39
3.2.3	Formalisierung	40
3.2.4	Gütemaße	42
3.3	Prozess zur heuristischen Sicherheitsprüfung von Entwicklungsartefakten	43

3.4	Kritiksystem zur automatischen Sicherheitsprüfung	45
3.4.1	Identifikation von Schwachstellen	46
3.4.2	Wissensbasis und Akquisition von Wissen	47
3.4.3	Heuristiken für die Sicherheitsprüfung	47
3.4.4	Anforderungen an das Kritiksystem	48
3.5	Wissenschaftliche Herausforderungen	49
4	Modellierung von sicherheitsbezogenem Wissen	51
4.1	Sicherheitsbezogenes Wissen	52
4.2	Ansätze zur Modellierung und Aufnahme von sicherheitsbezogenem Wissen	53
4.2.1	Systematische Literaturrecherche	53
4.2.2	Analyse und Ergebnisse	58
4.3	Entwurf einer Ontologie für sicherheitsbezogenes Wissen	65
4.3.1	Modellierung primärer Begriffe und Beziehungen	66
4.3.2	Erweiterung der Ontologie für spezifische Anforderungen	69
4.4	Verwandte Arbeiten	69
5	Akquisition von sicherheitsbezogenem Wissen aus natürlichsprachlichen Informationen	71
5.1	Aufbau und Erweiterung von sicherheitsbezogenem Wissen	72
5.1.1	Eignung der Wissensquellen von Informationssystemen	72
5.1.2	Extraktion natürlichsprachlicher Informationen	73
5.2	Graphbasierte Repräsentation von natürlichsprachlichen Informationen	74
5.3	Instanzbasierte Klassifikation von Wörtern	75
5.4	Semiautomatisches Verfahren zur Akquisition von Wissen	77
5.4.1	Aktives Lernen mit natürlichsprachlichen Informationen	78
5.4.2	Einbeziehung der heuristischen Befunde	79
5.4.3	Erweiterung der Ontologie	80
5.5	Verwandte Arbeiten	81
6	Identifikation von sicherheitsrelevanten Schwachstellen in Entwicklungsartefakten	85
6.1	Struktur und Inhalt des Analysemodells	86
6.1.1	Modellierung der ablauforientierten Beschreibung	86
6.1.2	Modellierung der Abhängigkeiten zwischen Artefakten	89
6.2	Überführung der Artefakte in das Analysemodell	91
6.3	Überprüfung des Analysemodells	93
6.3.1	Analyse der Ablaufbeschreibung eines Artefakts	93
6.3.2	Berücksichtigung der inhaltlichen Abhängigkeiten zwischen den Artefakten	97
6.4	Verwandte Arbeiten	98

7	Anwendung der Sicherheitsprüfung in der Anforderungsphase	101
7.1	Schwachstellenanalyse von natürlichsprachlichen Anforderungen	102
7.1.1	Struktur und Inhalt von Anwendungsfällen	103
7.1.2	Überführung der Anwendungsfälle ins Analysemodell	104
7.2	Prototypische Umsetzung	106
7.3	Evaluierung mit der Fallstudie iTrust	109
7.3.1	Aufbau der Fallstudie	110
7.3.2	Ergebnisse	114
7.3.3	Diskussion der Ergebnisse	124
7.3.4	Validität der Ergebnisse	127
7.4	Verwendung heuristischer Befunde zur Entscheidungsdokumentation	129
7.4.1	Dokumentation und Modellierung von Entscheidungen	129
7.4.2	Systematische Überführung von heuristischen Befunden	130
7.5	Verwandte Arbeiten	132
8	Eignung der Sicherheitsprüfung für andere Entwicklungsphasen	137
8.1	Entwurfsphase	137
8.1.1	Sicherheitsbezogenes Wissen in der Entwurfsphase	138
8.1.2	Schwachstellenanalyse von Entwurfsmodellen in UML	139
8.1.3	Anwendungsbeispiel für einen Geldautomaten	141
8.1.4	Verwandte Arbeiten	145
8.2	Implementierungsphase	147
8.2.1	Sicherheitsbezogenes Wissen in der Implementierungsphase	148
8.2.2	Schwachstellenanalyse von Quelltext	149
8.2.3	Explorative Fallstudie mit Apache Tomcat	152
8.2.4	Akquisition von sicherheitsbezogenem Wissen durch die Demonstration von Angriffen	158
8.2.5	Visualisierung der heuristischen Befunde in der Software	161
8.2.6	Verwandte Arbeiten	163
9	Zusammenfassung und Ausblick	167
9.1	Ergebnisse und Beitrag der Arbeit	168
9.2	Grenzen der heuristischen Sicherheitsprüfung	169
9.3	Ausblick	171
A	Material zur systematischen Literaturrecherche	173
B	Penn-Treebank-Tagset	177
C	Material zur Fallstudie mit iTrust	179
D	Material zum Anwendungsbeispiel für einen Geldautomaten	187

Definitionsverzeichnis	191
Abbildungsverzeichnis	193
Tabellenverzeichnis	195
Literaturverzeichnis	197