

# Inhalt

Vorworte .....	17
----------------	----

<b>1 Einleitung .....</b>	<b>27</b>
1.1 Evolution von Sicherheitsrisiken .....	28
1.1.1 Angriffe auf die Applikation .....	28
1.1.2 Symptome versus Ursachen .....	30
1.1.3 Risiken durch ABAP-Code .....	32
1.2 (Un-)Sichere ABAP-Programme .....	34
1.2.1 Top Ten der falschen Annahmen .....	34
1.2.2 Reality Check .....	35
1.3 Ziel des Buches .....	40
1.3.1 Aufbau des Buches .....	42
1.3.2 Verwendete Konventionen .....	45
1.3.3 ABAP-Terminologie .....	45
1.3.4 Zielgruppen .....	46
1.4 Danksagung .....	50

## Teil I Grundlagen

<b>2 ABAP-Entwicklung aus Sicherheitssicht .....</b>	<b>53</b>
2.1 Charakteristika von ABAP .....	54
2.1.1 Mächtigkeit von ABAP und ABAP-Entwicklern .....	55
2.1.2 Angriffsoberfläche von ABAP-Programmen ....	56
2.1.3 Datenaustausch bei SAP-Systemen .....	58
2.1.4 Sicherheitsdefekte in ABAP .....	61
2.2 ABAP-Entwicklungsprozesse .....	62
2.3 Besonderheiten bei externer Entwicklung .....	64

<b>3 Methoden und Werkzeuge zur Entwicklung sicherer Software .....</b>	<b>65</b>
3.1 Reifegrad von Sicherheitsprozessen .....	65
3.2 Spezifikation .....	68
3.2.1 De-facto-Standards .....	70
3.2.2 Grundschutzbaustein SAP .....	71
3.2.3 Individuelle Best Practices .....	72

3.2.4	Schulungen .....	73
3.2.5	Spezifisches Threat Modeling .....	73
3.3	Architektur und Design .....	74
3.3.1	Bewährte Designs .....	75
3.3.2	Dokumentierte Angriffsoberfläche .....	76
3.4	Implementierung und Programmierung .....	77
3.4.1	Training für Entwickler und QA-Teams .....	77
3.4.2	Code-Audit .....	78
3.4.3	Code Inspector .....	83
3.5	Test .....	84
3.6	Betrieb und Wartung .....	86

## Teil II Anwendung und Praxis

<b>4</b>	<b>Sichere Programmierung .....</b>	<b>89</b>
4.1	Ursachen von Sicherheitsproblemen .....	93
4.1.1	Softwareentwicklung – Theorie vs. Praxis .....	93
4.1.2	Unwissen, Zeitmangel und technologischer Fortschritt .....	95
4.2	Organisatorische Maßnahmen .....	97
4.2.1	Erfassung des Ist-Zustands der Softwaresicherheit .....	98
4.2.2	Prozess für sichere Software etablieren .....	99
4.3	Sicherheitsprinzipien in der Softwareentwicklung .....	101
4.3.1	Prinzip #1 – Sicherheit als Priorität .....	102
4.3.2	Prinzip #2 – Risikobewusstsein .....	103
4.3.3	Prinzip #3 – Denken wie ein Angreifer .....	105
4.3.4	Prinzip #4 – Angreifer mit internem Wissen .....	106
4.3.5	Prinzip #5 – Prüfung aller Eingabewerte .....	106
4.3.6	Prinzip #6 – Reaktion auf alle Fehler .....	108
4.3.7	Prinzip #7 – Mehrschichtiges Schutzkonzept .....	110
4.3.8	Prinzip #8 – Möglichst kleine Angriffsoberfläche .....	111
4.3.9	Prinzip #9 – Überprüfung der Annahmen .....	112
4.3.10	Prinzip #10 – Handeln nach Standards .....	112
4.3.11	Prinzip #11 – Ständige Erweiterung des Wissens .....	114

4.4	Filterung und Validierung von Benutzereingaben .....	114
4.4.1	Repräsentation von Daten in Rechnersystemen .....	115
4.4.2	Validierung von Benutzereingaben .....	116
4.4.3	Behandlung unerwarteter Daten .....	119
4.4.4	Typische Fehler bei der Validierung mit Filtern .....	121
4.5	Encodierung von Ausgaben .....	123
4.5.1	Encodierungsprobleme .....	124
4.5.2	Encodierung von Daten .....	126
4.5.3	Typische Fehler bei der Encodierung .....	128
4.6	Indirektion .....	130
4.7	Checkliste für sichere Programmierung .....	132
<b>5</b>	<b>Sichere Programmierung mit ABAP .....</b>	<b>135</b>
5.1	Fehlende Berechtigungsprüfungen bei Transaktionen .....	138
5.1.1	Anatomie der Schwachstelle .....	138
5.1.2	Risiko .....	138
5.1.3	Maßnahmen .....	139
5.1.4	Selbsttest .....	143
5.2	Hintertüren – hart codierte Berechtigungen .....	143
5.2.1	Anatomie der Schwachstelle .....	144
5.2.2	Risiko .....	146
5.2.3	Maßnahmen .....	146
5.2.4	Selbsttest .....	146
5.3	Fehlende Berechtigungsprüfungen in RFC-fähigen Funktionen .....	148
5.3.1	Anatomie der Schwachstelle .....	149
5.3.2	Risiko .....	150
5.3.3	Maßnahmen .....	150
5.3.4	Selbsttest .....	151
5.4	Debug-Code in Assert Statements .....	151
5.4.1	Anatomie der Schwachstelle .....	153
5.4.2	Risiko .....	154
5.4.3	Maßnahmen .....	155
5.4.4	Selbsttest .....	155
5.5	Generischer und dynamischer ABAP-Code .....	156
5.5.1	Anatomie der Schwachstelle .....	157
5.5.2	Risiko .....	164
5.5.3	Maßnahmen .....	165

5.5.4	Selbsttest .....	165
5.6	Generische Funktionsaufrufe .....	166
5.6.1	Anatomie der Schwachstelle .....	167
5.6.2	Risiko .....	168
5.6.3	Maßnahmen .....	168
5.6.4	Selbsttest .....	168
5.7	Generische Reports (ABAP Command Injection) .....	169
5.7.1	Anatomie der Schwachstelle .....	170
5.7.2	Risiko .....	170
5.7.3	Maßnahmen .....	172
5.7.4	Selbsttest .....	172
5.8	SQL-Injection .....	173
5.8.1	Anatomie der Schwachstelle .....	176
5.8.2	Risiko .....	181
5.8.3	Maßnahmen .....	182
5.8.4	Selbsttest .....	184
5.9	Directory Traversal .....	185
5.9.1	Anatomie der Schwachstelle .....	187
5.9.2	Risiko .....	189
5.9.3	Maßnahmen .....	189
5.9.4	Selbsttest .....	190
5.10	Aufrufe in den Kernel .....	191
5.10.1	Anatomie der Schwachstelle .....	193
5.10.2	Risiko .....	196
5.10.3	Maßnahmen .....	197
5.10.4	Selbsttest .....	197
5.11	System Command Injection und System Command Execution .....	198
5.11.1	Anatomie der Schwachstelle .....	198
5.11.2	Risiko .....	206
5.11.3	Maßnahmen .....	207
5.11.4	Selbsttest .....	208
5.12	Checkliste für sichere ABAP-Programme .....	209

## **6 Sichere Webprogrammierung mit ABAP ..... 213**

6.1	Probleme von browserbasierten User Interfaces .....	215
6.1.1	Informationssicherheit .....	219
6.1.2	Berechtigungen .....	220
6.1.3	Integrität .....	221
6.1.4	Funktionsumfang .....	221

6.2	Sicherheitslücken in Web-Frontends .....	223
6.2.1	Günstiger Webshop .....	223
6.2.2	Verstecktes Passwort .....	225
6.2.3	Vermeintlicher Schutz vor Manipulation .....	226
6.3	Cross-Site Scripting .....	226
6.3.1	Anatomie der Schwachstelle .....	228
6.3.2	Risiko .....	234
6.3.3	Maßnahmen .....	236
6.3.4	Selbsttest .....	244
6.4	Cross-Site Request Forgery .....	246
6.4.1	Anatomie der Schwachstelle .....	248
6.4.2	Risiko .....	253
6.4.3	Maßnahmen .....	254
6.4.4	Selbsttest .....	259
6.5	Forceful Browsing .....	259
6.5.1	Anatomie der Schwachstelle .....	261
6.5.2	Risiko .....	265
6.5.3	Maßnahmen .....	267
6.5.4	Selbsttest .....	269
6.6	Phishing .....	270
6.6.1	Anatomie der Schwachstelle .....	271
6.6.2	Risiko .....	274
6.6.3	Maßnahmen .....	276
6.6.4	Selbsttest .....	277
6.7	HTTP Response Tampering .....	279
6.7.1	Anatomie der Schwachstelle .....	281
6.7.2	Risiko .....	282
6.7.3	Maßnahmen .....	282
6.7.4	Selbsttest .....	283
6.8	Checkliste für UI-Programmierung .....	283

## 7 Sichere Programmierung in den ABAP-Technologien ..... 285

7.1	Verarbeitung von Dateien .....	286
7.1.1	Zugriff auf Dateien .....	287
7.1.2	Verarbeitung von Dateiinhalten .....	288
7.1.3	Dateiaustausch zwischen Client und Server .....	289
7.1.4	Zusammenfassung .....	290

7.2	Datenbankzugriffe .....	291
7.2.1	Datenbankabfragen mit Open SQL .....	292
7.2.2	Datenbankabfragen mit Native SQL .....	293
7.2.3	Validierung der Daten .....	293
7.2.4	Zusammenfassung .....	295
7.3	SAP GUI-Anwendungen .....	296
7.3.1	Ablauf der Interaktion zwischen SAP GUI und Backend .....	296
7.3.2	Varianten des SAP GUI .....	298
7.3.3	Maßnahmen im Backend .....	299
7.3.4	Zusammenfassung .....	300
7.4	SAP NetWeaver Application Server ABAP .....	301
7.4.1	Funktionsweise des AS ABAP .....	301
7.4.2	Hilfsmittel zur sicheren Entwicklung .....	302
7.4.3	Zusammenfassung .....	303
7.5	Business Server Pages .....	303
7.5.1	Entwicklung von BSP-Anwendungen .....	304
7.5.2	Absicherung der Interaktion .....	306
7.5.3	Zusammenfassung .....	308
7.6	Internet Transaction Server .....	309
7.6.1	Entwicklung von Webanwendungen mit dem ITS .....	309
7.6.2	Sichere Entwicklung mit dem ITS .....	310
7.6.3	Zusammenfassung .....	311
7.7	Web Dynpro ABAP .....	312
7.7.1	Entwicklung mit Web Dynpro .....	312
7.7.2	Sichere Entwicklung mit Web Dynpro .....	313
7.7.3	Zusammenfassung .....	314
7.8	Anbindung indirekter User Interfaces und externer Systeme .....	314
7.8.1	Sichere Anbindung externer User Interfaces .....	315
7.8.2	Sichere Anbindung externer Systeme .....	315
7.8.3	Zusammenfassung .....	316
7.9	Checkliste für SAP-Technologien .....	317

## **8 Risiken in Business-Szenarien ..... 319**

8.1	E-Recruitment .....	320
8.1.1	Angriffsmotive .....	321
8.1.2	Angriffsszenarien .....	322
8.1.3	Maßnahmen .....	324

8.2	Employee Self-Services .....	325
8.2.1	Angriffsmotive .....	326
8.2.2	Angriffsszenarien .....	326
8.2.3	Maßnahmen .....	327
8.3	Customer Relationship Management .....	328
8.3.1	Angriffsmotive .....	328
8.3.2	Angriffsszenarien .....	329
8.3.3	Maßnahmen .....	330

## **9 Schlussfolgerungen und Ausblick ..... 331**

9.1	Schlussfolgerungen .....	331
9.2	Ausblick .....	333
9.3	Was Sie mitnehmen sollten ... ..	335

## **Teil III Anhang**

A	Checklisten und Übersichten .....	339
B	Literatur- und Quellenverzeichnis .....	351
C	Die Autoren .....	355
	Index .....	359