Safety ist anders, Security auch

Wie begegnet man emergenten Systemeigenschaften?

Dr. Thomas Liedtke, ICS AG

Der steigende Anteil an Software realisierter Anforderungen in Systemen führt zu einer immer größer werdenden Anzahl von Verwundbarkeiten und möglichen Angriffsvektoren, um Schwachstellen auszunutzen.
Safety-Risiken werden häufig durch deduktive (oftmals mathematische) Ansätze unter Anwendung bewährter Methoden und Wahrscheinlichkeiten bestimmt. Sehr unwahrscheinliche und damit in ihrer Auswirkung kaum einschätzbare Risiken können aufgrund ihrer geringen Auftretenswahrscheinlichkeit vernachlässigt werden. „Nur" mit aller Vernunft vorstellbare Ursachen müssen in Risikoanalysen betrachtet werden. Im schlimmsten Fall wird versucht einen als Fail-Safe definierten Zustand einzunehmen.
Bei der Betrachtung von Security-Risiken gehören auch sehr unwahrscheinliche Bedrohungen und komplexe Verwundbarkeiten zu potentiellen Gefährdungen. Durch den Interessenskonflikt bei Angriffen müssen zur Risikoanalyse insbesondere auch induktive und zufällige Szenarien durchdacht werden. Dazu kommen mögliche Sicherheitslücken durch emergente Systemeigenschaften. Antifragilität wird zur notwendigen Eigenschaft, um vorbereitet zu sein.

0. Einleitung

„Es kommt nicht darauf an, die Zukunft vorauszusagen, sondern auf die Zukunft vorbereitet zu sein."
Perikles (490 – 429 v. Chr.)

Der Begriff der Sicherheit ist in der deutschen Sprache mehrdeutig belegt:

- Safety: auch Funktionssicherheit: stellt sicher, dass sich ein System konform zur erwarteten Funktionalität verhält. Es geht um die Sicherheit des Menschen.
- Security: bezeichnet die Aufrechterhaltung von Funktionalität im Falle eines Angriffes, den Schutz eines technischen Systems und die Einhaltung von Grundwerten vor seiner Umwelt (z.B. Angreifer).

Sowohl die Komplexität der Anforderungen an technische Systeme (z.B. Dezentralisierung von Steuerungen, unterschiedlichste zu unterstützende Endgeräte und Bedienungskonzepte, immer intelligenter werdende Aktoren und Sensoren, erhöhte Anzahl von Schnittstellen, immer raffiniertere Services) als auch die Komplexität der Entwicklungsprozesse (wechselnde Infrastrukturen, unterschiedlichste zu unterstützende Standards) steigen rasant.

War die Systemarchitektur bislang eher einfach, im Sinne von einer zentralen Steuerung, die von einer unabhängigen Kontrolleinheit überwacht wurde [IEC61508], sind Komponenten eines modernen Systems selbst für Sicherheit zuständig (z.B. Fahrerassistenzsysteme im Automobil) [ISO26262]. Hinzu kommt, dass sich Produkte offen und dynamisch zur Betriebszeit selbst konfigurieren können

müssen und nichtmehr nur geschlossen statisch sind (Stichworte Industrie 4.0, Losgröße 1).

1. Safety vs. Security

Safety ist definiert als der Schutz vor (unbeabsichtigter) Fehlfunktion (bei bestimmungsgemäßer Verwendung) und der Abwesenheit von unzumutbaren Risiken. Die realisierte Ist-Funktionalität entspricht der spezifizierten Soll-Funktionalität.

Bei der Entwicklung sicherheitsgerichteter Systeme werden in den anzuwendenden Normen bewährte Vorgehensweisen gefordert:

- Als Vorgehensmodell wird das V-Modell vorgegeben
- Je nach Kritikalität der zu implementierenden Funktion ist ein mehr oder weniger strenges Rollenkonzept und/ oder Personaltrennung zu erfüllen
- Die Vorgehensweisen bei Test, Validierung und Dokumentation sind streng hierarchisch den horizontalen Ebenen des V-Modells angepasst.
- Als Testendekritieren sind strukturelle oder funktionale Abdeckungskriterien definiert

Vorgehensweisen und Methoden im Safety-Prozess sind häufig Top-Down-orientiert, mathematisch und mit viel Disziplin und Fleiß durchzuführen. Aus fragilen Systemen sollen robuste werden, die bei entsprechenden Geschehnissen das System notfalls in einen sicheren Zustand führen.

Security ist definiert als der Schutz vor (absichtlichen) Angriffen (und Zufällen/ Unglücke). Security bedeutet die funktionale Korrektheit bei aktiven Attacken.

Spätestens seit Stuxnet [Stuxnet] gibt es in der Security-Welt eine Zeitenwende. Die große Motivation, der Breite des Angriffs, die hohe Komplexität und der enorme Aufwand waren vorher so nie da gewesen. Auch dass man derart tief in vermeintlich proprietäre Systeme eingreifen kann war vorher nicht erwartet worden.

Security-Risiken können reduziert, vermieden (System wird so umstrukturiert, dass die Gefährdung wegfällt), übernommen (akzeptiert) oder delegiert werden.
Im Unterschied zu Safety gibt es bei Security zunächst keinen definierten sicheren Zustand. Ein sicheres System soll stets am Laufen gehalten werden, ein Denial of Service unter allen Umständen vermieden werden.

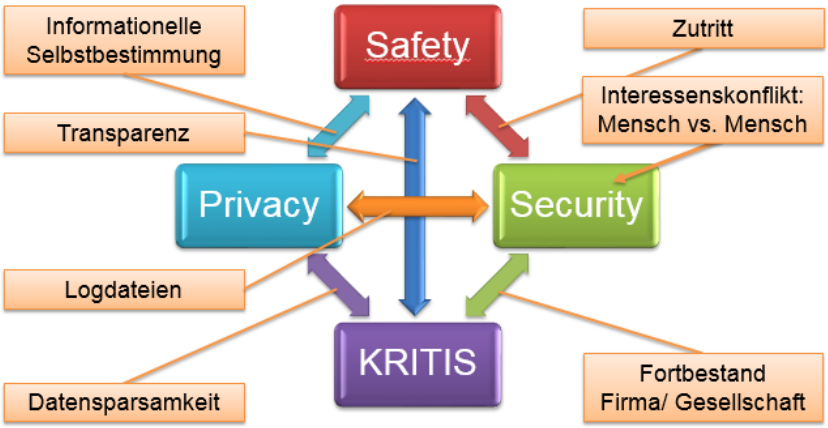
Nachdem für Safety das Schutzziel immer Leib und Leben ist, müssen für Security System-abhängig Schutzziele definiert werden. Je nach Schutzziel sind unterschiedliche Design Prinzipien (wie z.B. „Sichtbarkeit und Transparenz“) und Design Strategien (wie z.B. „Datensparsamkeit“) anzuwenden.

2. Zielkonflikte

Durch den (bei Safety fehlenden) Interessenskonflikt zwischen Angreifer und Betreibern sind bei der Security-Risiko- und Bedrohungsanalyse eine ganze Reihe zusätzlicher Themen abzusichern. So sind z.B. Implementierungsangriffe,

Seitenkanalangriffe (Re-Engineering) viel stärker zu berücksichtigen. Security-Angriffe haben außerdem häufig die Eigenschaft mit großem Einsatz an Ressourcen aus der Ferne geführt zu werden (z.B. Industriespionage). Kommunikationskanäle und Schnittstellen des Systems nach außen stellen besonders gefährdete Sicherheitslücken dar.

Bei der Entwicklung sicherheitsgerichteter Systeme mit Safety- und Security-Anforderungen muss zwischen orthogonalen Zielwerten priorisiert werden [LiHo16]. Privacy (Datenschutz) und KRITIS (Sicherung kritischer Infrastrukturen) können als weitere Dimensionen gesehen werden (s. Figur 1).



Figur 1: Beispiele für Zielkonflikte

Von der Methodik her gibt es ähnliche Vorgehensweisen, so dass auch Security als paralleler Managementprozess zur Entwicklung verstanden wird. Das „verweben“ mit Safety ist Thema vieler aktueller Berichte (s. z.B. [GGH+15]).

Bei der traditionellen Gefahren- und Risikoanalyse geht man von einer identifizierten Gefahr aus, um zur Auswirkung und Eintrittswahrscheinlichkeit zu gelangen (z.B. [Sto09]). Entsprechende Risikoakzeptanzkriterien bestimmen die weiteren Maßnahmen in der Entwicklung. Im nächsten Kapitel wird aufgezeigt, dass für Security ein anderer Weg eingeschlagen werden muss.

3. Antifragilität und Emergenz

Die Norm IEC62443 [IEC62443-3] gibt zur Risikobetrachtung eines Systems zunächst eine Top-Down-Zerlegung vor. Das System wird dazu in Komponenten (Zonen) und Kanäle (Conduits) zerlegt. Die Komponenten und Kanäle werden dann einzeln für sich analysiert.

Emergente Eigenschaften komplexer Systeme erfordern das Umgehen mit unbekannten Systemeigenschaften. Systeme müssen mehr als robust ausgelegt werden: sie müssen antifragil [Tal14] sein.

Antifragilität wird als das Gegenteil von Fragilität definiert. Dazwischen wird im Sinne einer Triade der Begriff Robustheit eingeordnet. Mögliche Charakteristika sind in Tabelle 1 angegeben.

	fragil	robust/ resilient/ stabil	antifragil (Gegenteil von fragil)
Unbeständigkeit/ Unordnung	angreifbar	nicht angreifbar/ nicht nützlich	profitiert
Zufälle/ seltene Ereignisse/ Ungewissheit/ Irrtümer	ist gefährdet/ leidet/ wird schwächer/ geht mit der Zeit zu Bruch	bleibt gleich	wächst und gedeiht/ wird besser
Meßbarkeit	Fragilität meßbar/ vergleichbar		Eintritt von Risiken nicht kalkulierbar
Top-down-Dynamik	erhöht Fragilität		blockiert Antifragilität
Bottom-up-Strukturen	beruhen eher auf Schulbildung		profitieren von Stress und Unordnung; offensive Bereitschaft zum Risiko
Fehler	selten, bei Eintritt gravierend, unumkehrbar		klein, harmlos, umkehrbar, schnell verschmerzt

Tabelle 1: Mögliche Beschreibungen Fragilität vs. Antifragilität

Eine Reihe von Beispielen zeigt wie vorteilhaft sich der Gedanke an Antifragilität auswirken kann und wie wünschenswert diese Eigenschaft für sichere (Security-) Systeme ist. Antifragilität ist mehr als das Härten von Systemen, es ist die Überkompensation und das „fit“ machen um für zukünftige Anforderungen (Angriffe) gewappnet zu sein. Nicht alle durch Emergenz entstehende Sicherheitslücken sind zur Entwicklungszeit bekannt. Sie werden (u.U. durch Trial und Error) erst im Laufe der Zeit „gefunden“. Das bedeutet, dass Systeme mit Security-Vorkehrungen ständig aktualisiert und gehärtet werden müssen.

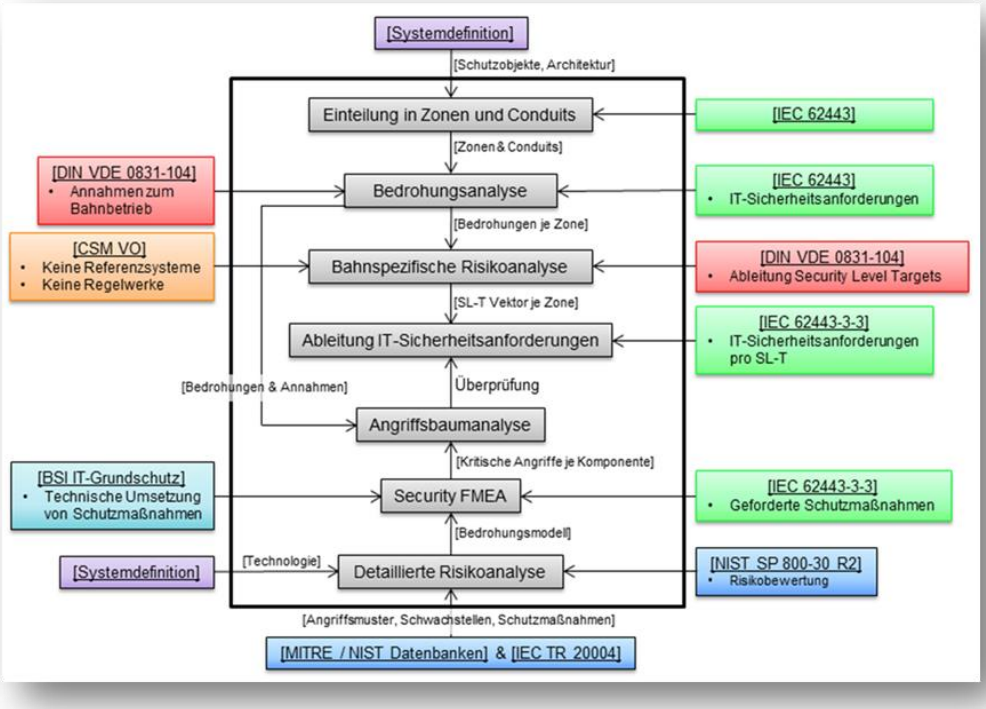
Bei der Betrachtung von Security-Analysen sind u.a. folgende Trugschlüsse zu vermeiden:

- Was man nicht sieht/ versteht gibt es nicht (s. Stuxnet, 911, …)
- Weil seltene Ereignisse im Nachhinein erklärbar sind, geben sie den Eindruck man hätte sie vorausgeahnt (Illusion der Vorhersagebarkeit)

Im Gegensatz zur Fragilität ist Antifragilität nicht messbar. Die Anfälligkeit eines Porzellan-Tellers gegenüber einem Plastikteller, die Gesundheit eines alten Menschen gegenüber eines jungen kann bewertet werden. Wie gut ein System gegen Zufälle und seltene (bislang unbekannte) Ereignisse geschützt ist, ist wesentlich schwieriger zu bewerten.

Nachdem man mit schädlichen Folgen besser umgehen kann als mit der Vorhersage seltener Ereignisse, müssen Vorhersagen und Risikomanagement auf den Kopf gestellt werden.

Ein mögliches Beispiel für eine Risikoanalyse aus dem Bereich Transportation wird in Figur 2 aufgezeigt.



Figur 2: Beispiel einer Risikoanalyse kritischer Systeme

Im letzten Schritt wird eine detaillierte Risikoanalyse gegen potentielle bereits bekannte Bedrohungen durchgeführt. Dabei spielen verwendete Quellen für Schwachstellen, Verwundbarkeiten und bekannte Angriffe eine wichtige Rolle.

Dem Zufall und Trial undError von Angriffen können auf Dauer nur antifragile Systeme standhalten. Eine Reihe von möglichen Betrachtungen/ Überlegungen zur Emergenz aus Security-Sicht umfassen damit z.B.:

- Definition möglicher Gegenmaßnahmen für den Fall, dass ein Angriff schon teilweise erfolgreich ist (z.B. eine sichergeglaubte „äußere“ Mauer überwunden hat)
- Honeypots bringen Erfahrung zu Angriffsszenarien. Angreifer agieren häufig nach Trial und Error
- Erfahrungsdatenbanken über bekannte Bedrohungen, Schwachstellen nutzen
- Testverfahren mit Zufallscharakter anwenden: Fuzzy-, Penetrationstests, …
- Angriffe antizipieren (Intrusion Detection; Prevention)
- Interessenskonflikte (z.B. bei Implementierungsangriffen, Seitenkanalangriffen, Social Engineering, …) in Security-Analysen ausreichend berücksichtigen
- Die „dunkle Seite der Macht“ ist organisiert. Das was möglich ist, passiert irgendwann (existierende Sicherheitslücken werden ausgenutzt, die Motivation von Angreifern ist hoch)
- Risiko und Bedrohungsanalysen wiederholen, Systeme regelmäßig härten

4. Zusammenfassung

Bewährte Safety-Betrachtungen und –Methoden sind nicht ausreichend um auch Security zu garantieren. Die Komponenten Absicht, Interessenskonflikt, Zufall, Trial und Error, hohe Motivation von Angreifern müssen sehr viel stärker ins Kalkül gezogen werden als bei reinen Gefahrenanalysen für Safety.

Um auf die Zukunft vorbereitet zu sein, sind starre Grenzen (z.B. Virenscanner auf PC's) nicht ausreichend. Dynamische Abwehrstrategien, die sich ständig erneuern und aktualisieren, müssen implementiert werden.

Literatur

[GGH+15] Benjamin Glas, Carsten Gebauer, Jochen Hänger, Andreas Heyl, Jürgen Klarmann, Stefan Kriso, Priyamvadha Vembar, Philipp Wörz, „Integration Challenges", Konferenz Automotive Safety and Security 2015

[IEC-61508] Standard „Functional safety of electrical/ electronic/ programmable electronic safety-related systems"

[IEC62443-3] "Security for industrial process measurement and control – Network and system security"

[ISO26262] „Road vehicles – Functional safety"

[LiHo16] Thomas Liedtke und Bernhard Hohlfeld „Wie passen Safety und Security zusammen? Beispiel eCall". „Forum Safety & Security"; Tagungsband 06.-07. Juli 2016 München. ISBN 978-3-645-50168-2

[Sto09] Ketil Stolen, SINTEF & UiO, „Security Analysis: The CORAS Approach"

[Stuxnet] Wikipedialink: https://de.wikipedia.org/wiki/Stuxnet (eingesehen, 20.10.2016)

[Tal14] Nassim Nicholas Taleb „Antifragilität: Anleitung für eine Welt, die wir nicht verstehen". Btb, 2014

Autor

Dr. rer. nat. Thomas Liedtke (geb. 1966) ist Leiter der Business Unit Research und Development der Informatik Consulting Systems AG (ICS AG). Seine Hauptarbeitsgebiete sind Projektmanagement und Methoden in sicherheitsgerichteten Entwicklungen (Safety, Security, Privacy) von Systemen: theoretisch, praktisch und in der Lehre, sowie die Leitung der Competence Center der ICS AG. Zusätzlich ist er IT-SiBe und bDSB.

Kontakt

Internet: www.ics-ag.de
E-Mail: thomas.liedtke@ics-ag.de