Synergieeffekte von Safety und Security

Thomas Bötner, Prof. Dr. Hartmut Pohl, softScheck GmbH

Roboter enthalten zwar grundsätzlich Safety-relevante (IEC 61508 Funktionale Sicherheit) Schutzmechanismen, um Schäden an Leib und Leben entgegen zu wirken. Allerdings ist gleichermaßen Security (Angriffssicherheit: ISO 27034 und IEC 62443) von (vernetzten) Robotern unverzichtbar, da Security die Safety beeinflussen kann: Ein Angreifer kann durch Ausnutzen von Sicherheitslücken in Software oder Firmware implementierte Schutzmechanismen außer Kraft setzen und angeschlossene Roboter fehlsteuern.

Vernetzte Produktionsprozesse sind ein zunehmend beliebtes Ziel für Cyberattacken. Die fortschreitende Digitalisierung von Fertigungsanlagen und die Vernetzung von Produktions- mit Office-Netzen vergrößern die Angriffsfläche und wecken gleichzeitig immer mehr Begehrlichkeiten von Angreifern. In der Praxis lässt sich ein Wildwuchs von ad-hoc Sicherheitsaktivitäten finden - eine Strategie fehlt. Einen Lösungsansatz zum Management aller Sicherheitsaktivitäten bietet die ISO 27034, die sich mit Hilfe eines Prozessleitfadens leicht in die Praxis umsetzten lässt.

Die Angriffsziele reichen vom Daten-Diebstahl (Spionage) und Erpressung bis hin zur Sabotage (!) von Produktionsprozessen. Wird der Angriff bekannt, entsteht dem Unternehmen (zusätzlich) ein Imageschaden; dieser ist häufig unabhängig davon, ob der Prozesseigentümer selbst oder Dritte - wie Zulieferer und Serviceanbieter - für die ausgenutzte Sicherheitslücke verantwortlich sind.

Voraussetzung für alle (erfolgreichen) Angriffe ist nämlich mindestens eine dem Angreifer bekannte und vom Angriff ausnutzbare Sicherheitslücke. Daher gilt es, möglichst alle Sicherheitslücken in Soft- und Hardware zu identifizieren und zu patchen – dies gilt insbesondere für bislang nicht-erkannte (unveröffentlichte) Sicherheitslücken (Zero-Day-Vulnerabilities).

Der Patch-Aufwand einer Sicherheitslücke steigt allerdings exponentiell mit dem Reifegrad der Software und ist nach dem Release am größten. Daher sollten Sicherheitsaspekte bereits ab den ersten Phasen der Requirements- und Risikoanalyse sowie dem Design im Entwicklungsprozess berücksichtig werden.

ISO 27034-1

Einen allgemeinen Ansatz zum Management der Entwicklung sicherer Software liefert ISO 27034-1 „Application Security“. Diese Norm bietet eine hersteller- und technologieunabhängige Grundlage; sie definiert Konzepte, Frameworks und Prozesse, die Unternehmen helfen, Application Security in ihren Software-Entwicklungsprozess zu integrieren. Entscheidender Kern der Norm ist eine unternehmensweite Bibliothek mit allen für die Softwareentwicklung nützlichen Sicherheitsaktivitäten. Gemäß den Security-Requirements für das jeweilige Software-Projekt werden ausgewählte Sicherheitsaktivitäten eingesetzt und in der Verifikationsphase auf ihre erfolgreiche Implementierung überprüft.

Die ISO 27034 lässt sich ebenso gut auf den Einkauf von Produkten oder Outsourcing der Entwicklung anwenden. So können für eingekaufte Produkte oder Leistungen unternehmensweite einheitliche Sicherheitsniveaus gefordert und überprüft werden.

Security Testing
Eine wesentliche Sicherheitsaktivität stellt das Security Testing dar: Mit den von der Norm vorgeschlagenen folgenden 5 Methoden werden bekannte Sicherheitslücken und insbesondere bisher nicht-erkannte, unveröffentlichte Zero-Day-Vulnerabilities identifiziert:

Security Requirements Analysis: Identifizierung und Überprüfung exakter Sicherheitsanforderungen.

Threat Modeling (Security by Design) überprüft die Sicherheitsarchitektur der Software und Firmware kritischer IT-Infrastrukturen und Netzwerke. Da etwa die Hälfte aller Sicherheitslücken auf Designfehler zurückzuführen sind, müssen Sicherheitsmaßnahmen bereits vor bzw. während der Designphase implementiert und überprüft werden.

Static Source Code Analysis (Code Review): Ab der Implementierungsphase wird die Konformität des Quellcodes der Zielsoftware mit formalen Methoden auf Einhaltung syntaktischer Programmierkonventionen der Programmiersprache und auf Einhaltung der Programmierrichtlinien überprüft. Dieses Verfahren ist vergleichbar einem Parser, der eine lexikalische, syntaktische und semantische Analyse des Programmcodes durchführt.

Aufgrund lexikalischer Regeln der verwendeten Programmiersprache und den semantischen Zugehörigkeiten benötigen die einzelnen Fehler im Allgemeinen einen manuellen Audit, um false positives auszuschließen und entsprechende Behebungsstrategien zu entwerfen. Die Qualität und Quantität des Analyse-Resultats hängt somit maßgeblich von der Auswahl geeigneter Tools ab.

Penetration Testing: Dynamische Sicherheitsprüfung mit bekannten Angriffen zur Identifizierung bekannter Sicherheitslücken.

Dynamic Analysis – Fuzzing: Fuzzing stellt eine halb-automatisierte Methode zur Identifizierung von (mit Angriffen ausnutzbaren) Sicherheitslücken in Software und Hardware/Firmware dar: Tool-gestützte Eingabe von Testdaten in ein Target System (Programm, Firmware) werden verwendet, um unvorhergesehene - im Programmcode unberücksichtigte - Eingabedaten zu erkennen. Die falsche oder unzureichende Verarbeitung dieser (im Programm nicht berücksichtigten) Daten führt zu einem unerwarteten Verhalten (Crash, hoher Verbrauch an Ressourcen wie Rechenzeit, Speicher) des Zielprogramms. Dieses anomale Verhalten des Programms wird mit Hilfe eines Monitoring-Tools protokolliert, voranalysiert und dargestellt. Durch die Analyse der Monitorergebnisse können falsche Hinweise (False Positives) ausgesondert werden. Sicherheitslücken werden durch Reproduzierung der Anomalie und Entwicklung eines Exploits nachgewiesen.

Neben der Auswahl der richtigen Methoden ist die die Festsetzung des Zeitpunkts, wann im Produktlebenszyklus eine Methode angewandt wird, für den Erfolg und die Effizienz der Methode wichtig.

Prozessleitfaden
Zur praktikablen Umsetzung dieser Norm ist ein Prozessleitfaden zur Entwicklung sicherer Software hilfreich; ein aufwändiges Einarbeiten aller Projektbeteiligten in die Norm ist bei der Verwendung des Prozessleitfadens nicht notwendig. Bestehende Sicherheitskonzepte und Maßnahmen lassen sich in den Prozessleitfaden integrieren. Er ist unabhängig vom verwendeten Software Development Lifecycle (traditionelle Modelle wie Wasserfallmodell, über V-Modell bis hin zu agilen Ansätzen wie Scrum).

Der Prozessleitfaden unterstützt also Unternehmen bei der ISO 27034 konformen Entwicklung sicherer Software.

ONF & ANF

Den Kern der ISO 27034 bilden die zwei Hauptprozesse Organisation Normative Framework (ONF) Process und das Application Security Management Process (ASMP), durch den das Application Normative Framework (ANF) erstellt und angewandt wird. Erster beschreibt den Aufbau und die Pflege des ONF. Im ONF werden alle im Unternehmen verwendeten Richtlinien, Regularien, Best Practices etc. in einer Unternehmensweiten Bibliothek zusammengefasst. Anschließend wird für jedes Projekt durch den ASMP ein ANF gebildet, indem die notwendigen Richtlinien, Regularien, Best Practices etc. zusammen mit den zugehörigen Application Security Controls (ASC) „Sicherheitsaktivitäten" aus dem ONF in den Produktlebenszyklus, insbesondere in den Entwicklungsprozess integriert werden.

Eine Risikoanalyse im ersten Schritt des ASMP dient als Grundlage zur Bestimmung des angestrebten Sicherheitsniveaus (Target Level of Trust) und einer anschließenden Definition von Sicherheitsanforderungen für die zu entwickelnde Software. Darauf wird das ANF entsprechend der Projekteigenschaften, wie verwendete Technologien, lokale gesetzliche Rahmenbedingungen, Unternehmensrichtlinien und dem angestrebten Sicherheitsniveau aus dem ONF abgeleitet.

Das ANF beschreibt, welche Sicherheitsaktivitäten wann in dem Produktlebenszyklus ausgeführt werden sowie wann und wie die erfolgreiche Ausführung nachgewiesen bzw. überprüft wird. Hierbei wird der Produktlebenszyklus in die zwei Phasen Provisioning und Operating Application unterteilt, wobei das Rollout den Übergangspunkt der beiden Phasen bildet. Zudem werden Meilensteine gesetzt, an denen das aktuelle mit dem angestrebten Sicherheitsniveau verglichen wird. Wurden alle, bis zu diesem Zeitpunkt geplanten, Sicherheitsaktivitäten erfolgreich umgesetzt? Falls nicht, muss der Grund dafür erörtert werden und ggf. das ANF oder das angestrebte Sicherheitsniveau angepasst werden.

Des Weiteren wird die Notwendigkeit zusätzlicher oder neuer Sicherheitsaktivitäten durch Veränderungen im Projekt, z.B. durch neue Anforderungen, überprüft werden.

Werden neue Sicherheitsaktivitäten benötigt oder müssen selbige angepasst oder aktualisiert werden, wird dies von dem Projektteam an das ONF Team kommuniziert.

Durch diesen Feedback-Prozess verändern sich das ONF und die Sicherheitsaktivitäten entsprechend dem aktuellen Stand der Technik kontinuierlich.

Schulterschluss zur funktionalen Sicherheit

Funktionale Sicherheit und Angriffssicherheit sind sowohl bei bestehenden, sicheren Systemen wie auch in der aktuellen Softwareentwicklung zentrale Anforderungen. Allerdings werden beide Bereiche meist unabhängig voneinander betrachtet – was zu Sicherheitslücken, erhöhter Entwicklungszeit und somit steigenden Kosten führt. Dabei liegt die Lösung auf der Hand: Ein Prozessleitfaden, der beide Bereiche umfasst und Synergien nutzt.

Safety (funktionale Sicherheit) und die zugehörige Norm IEC 61508 sind in der Softwareentwicklung bereits weitgehend bekannt. Die Zusammenhänge und insbesondere die Abhängigkeiten zwischen Security und Safety werden jedoch oft nicht beachtet. Safety-Funktionen können durch bösartige Angreifer manipuliert werden. Daher müssen notwendigerweise funktional sichere Systeme ebenfalls secure sein.

Zudem lassen sich durch eine gemeinsame Betrachtung von Safety und Security Synergien nutzen. So verfolgen beide Bereiche die Verfügbarkeit als Ziel. Die Risiken für Ausfälle und Fehlfunktionen wurden bisher nur getrennt untersucht.

Die Herausforderung für die Zukunft besteht darin, die bislang zweigleisige Vorgehensweise zu verbinden. Denn nicht nur in konkreten Projekten kommt es zunehmend zu Überschneidungen zwischen Safety und Security, auch die noch eigenständigen Prozessleitfäden in den jeweiligen Bereichen bieten Verknüpfungspunkte. Ein Zusammenführen zu einem einzigen Leitfaden ermöglicht somit Synergieeffekte, wodurch der Kunde erhebliche Entwicklungskosten einspart. Gleichzeitig gewährleistet ein kombinierter Leitfaden aber vor allem zwei wichtige Aspekte: Zum einen stellt er sicher, dass die Software gemäß beider Normen entwickelt wurde und somit zertifizierbar ist. Zum anderen bietet er erstmals die dringend benötige Chance, Security- und Safety-Anforderungen bzw. Ziele in Abhängigkeit voneinander zu formulieren. So lassen sich die Anforderungen von Beginn an aufeinander abstimmen und optimieren. Zeitgleich identifiziert und bewertet auch erst eine gemeinsame Risikoanalyse Bedrohungen, die bis dato nicht so einfach erkannt werden konnten. So lassen sich Maßnahmen exakter formulieren.

Zertifizierung
Dabei ist zum einen der Entwicklungsprozess (mit ONF und ASM Prozess) zertifizierbar. Zum anderen lassen sich auch die mit diesem Prozess hergestellten Produkte einfacher und günstiger zertifizieren.
Die Zertifikate werden durch eine anerkannte Zertifizierungsstelle vergeben.

Somit erhält der Software-Hersteller nicht nur ein zertifiziertes (sicheres) Produkt, sondern auch einen Wettbewerbsvorteil und kann diesen auch bewerben.

Zusammenfassung
Automatisierungssysteme sind ein zunehmend beliebtes Ziel für Cyberattacken. Die fortschreitende Digitalisierung von Fertigungsanlagen und die Vernetzung von Produktions- mit Office-Netzen vergrößern die Angriffsfläche und wecken gleichzeitig immer mehr Begehrlichkeiten von Angreifern. Funktionale Sicherheit und Angriffssicherheit sind sowohl bei bestehenden, sicheren Systemen wie auch in der aktuellen Softwareentwicklung zentrale Anforderungen. Allerdings werden beide Bereiche meist unabhängig voneinander betrachtet – was zu Sicherheitslücken, erhöhter Entwicklungszeit und somit steigenden Kosten führt. Dabei liegt die Lösung auf der Hand: Ein Prozessleitfaden, der beide Bereiche umfasst und Synergien nutzt.

Autoren
Autor Herr Bötner ist als IT-Sicherheitsberater bei der softScheck GmbH insbesondere in den Bereichen Mobile App Security, Thread Modeling und Entwicklung sicherer Software tätig.

Co-Autor Prof. Dr. Hartmut Pohl ist Geschäftsführender Gesellschafter der IT-Sicherheitsberatung softScheck GmbH.

Co-Referent Gregor Schmitt ist als Key Account Manager Safety bei der infoteam Software AG beschäftigt.