Warum CRA und DevSecOps zusammengehören

von Oliver Roth

Wer in der EU nach dem 11. Dezember 2027 „Produkte mit digitalen Elementen“ mit CE-Kennzeichnung verkaufen möchte, muss sein Portfolio rechtzeitig auf die CRA-Anforderungen vorbereiten. Dabei sollte der Aufwand für erforderliche Neuentwicklungen nicht unterschätzt werden. Das betrifft insbesondere maßgeschneiderte Geräte für spezielle Anwendungen.

In Teil 1 der Serie wurde auf die Dringlichkeit von Bedrohungsanalysen nach dem TARA-Standard hingewiesen. Maschinenbauer und Industrieausrüster erfahren so, wie sie ihre Produkte CRA-konform gestalten und das vom CRA geforderte Maß an „Security by Design“ erreichen können. Dazu werden in vielen Fällen grundlegende Neuprogrammierungen, Hardware-Upgrades oder sogar komplette Neuentwicklungen erforderlich sein. Je höher der Aufwand zur Erlangung der CRA-Konformität ist, desto dringlicher wird die Frage, ob digitale Elemente wie Controller weiterhin individuell entwickelt werden sollen oder ob der Einsatz von Standardlösungen sinnvoller ist.

So oder so müssen Embedded-Systeme nach Inkrafttreten des CRA durchgängig „Security by Design“ bieten. Betrachtet man ein System als Haus, muss sich die Sicherheit durch alle Ebenen ziehen. Das beginnt beim Hardware-Fundament, wo selbst bei Mikrocontrollern der Einsatz eines TPM-Moduls (Trusted Platform Module) sinnvoll sein kann, und endet nicht auf der Betriebssystemebene, die Secure Boot gemäß UEFI-Spezifikation unterstützen und ein sicheres Betriebssystem einschließlich sicherer Anwendungen enthalten sollte.

Besondere Beachtung verdient in dieser Architektur das Updateportal. Es ist erforderlich, um funktionale oder sicherheitsrelevante Updates bereitzustellen. Gleichzeitig muss es selbst abgesichert werden. Dazu sind Sicherheitsmechanismen sowohl im Updateportal als auch auf dem Zielsystem erforderlich. Die Secure-Boot-Funktion wird dabei zum zentralen Element, das die gesamte Kette von ROM über Bootloader und Betriebssystem bis zu den Anwendungen umfasst und jeden Systemstart sowie jedes signierte Update überprüft.

Die skizzierten Anforderungen stellen hohe Ansprüche an die Entwicklung. Um ihnen gerecht zu werden, hat die Grossenbacher Systeme AG ihren Entwicklungsprozess weiterentwickelt – weg von punktuellen Sicherheitsmaßnahmen hin zu einem ganzheitlichen und kontinuierlichen Sicherheitsprozess. Dazu wurde der etablierte DevOps-Ansatz um Sicherheitsaspekte zum DevSecOps-Modell erweitert.

DevSecOps statt DevOps

DevOps führt Entwicklung und Betrieb in einer Organisation zusammen, wobei die Verantwortung für Produktqualität und betriebliche Effektivität gemeinsam getragen wird. Diese Zusammenarbeit ermöglicht es Unternehmen, Entwicklungsprozesse zu beschleunigen und Produkte schneller bereitzustellen. DevSecOps erweitert diesen Ansatz um Sicherheitsaspekte. Der Grundgedanke lautet: Sicherheitsprüfungen dürfen nicht erst am Ende eines Projekts stattfinden, sondern müssen kontinuierlich in Entwicklungs- und Bereitstellungsprozesse eingebunden werden. Für CRA-konforme Embedded-Produkte schafft DevSecOps damit organisatorische und technische Voraussetzungen.

DevSecOps lässt sich als geschlossener Kreislauf mit acht Phasen verstehen. Jede Phase liefert Ergebnisse, die in die nächste Entwicklungsiteration einfließen. Trotz dieses zyklischen Prozesses wird DevSecOps häufig als liegende Acht dargestellt. Diese Visualisierung verdeutlicht die kontinuierliche Wechselwirkung zwischen Entwicklungs- und Betriebs-

Bild: AI Stocks/stock.adobe.com

prozessen sowie den Rückfluss von Sicherheits- und Betriebsdaten in neue Entwicklungszyklen.

Die „liegende Acht" als Visualisierung von DevSecOps

• Die linke Schleife der liegenden Acht beginnt mit der **Plan**-Phase. Hier werden Sicherheitsanforderungen bereits zu Beginn systematisch definiert. Grundlage sind die Ergebnisse der Bedrohungsanalyse, die im Idealfall nach dem TARA-Standard erfolgt ist. Für vernetzte Embedded-Geräte müssen dabei grundlegende Entscheidungen getroffen werden, etwa zu Authentifizierungsmechanismen, verschlüsselter Kommunikation oder sicheren Update-Konzepten. Ziel ist es, „Security by Design" von Anfang an im Produkt zu verankern.

• Darauf folgt **Develop**. In dieser Entwicklungsphase entstehen Software, Firmware und Systemfunktionen unter Anwendung sicherer Entwicklungsprinzipien. Sichere Programmierstandards, Code-Reviews, automatisierte Sicherheitsanalysen und ein kontrollierter Umgang mit Bibliotheken sollen verhindern, dass Schwachstellen in das Produkt gelangen. Sicherheit wird damit Teil des laufenden Entwicklungsprozesses.

• In der anschließenden **Build**-Phase werden aus dem Quellcode reproduzierbare Artefakte erzeugt. Für Embedded-Systeme betrifft dies insbesondere Firmware-Images und Softwarepakete. Sicherheitsrelevant sind abgesicherte Build-Pipelines, Signaturen, Integritätsprüfungen sowie die Erstellung einer Software Bill of Materials (SBOM), die Transparenz über die verwendeten Komponenten schafft.

• Ein **Test** schließt die linke Schleife der Acht ab. Neben funktionalen Prüfungen erfolgen gezielte Sicherheitsbewertungen, beispielsweise durch automatisierte Schwachstellenanalysen oder Penetrationstests. Zudem werden Kommunikationsschnittstellen, Firmware und Update-Mechanismen überprüft. Ziel ist es, Risiken möglichst früh vor der Auslieferung zu erkennen.

• In der Mitte der liegenden Acht erfolgt der Übergang von Entwicklung zu Betrieb – ein zentraler Gedanke von DevSecOps. Ergebnisse aus beiden Bereichen fließen kontinuierlich ineinander.

• Die rechte Schleife beginnt mit der **Release**-Phase. Dabei wird entschieden, ob ein Produkt die definierten Qualitäts- und Sicherheitsanforderungen erfüllt und freigegeben werden kann. Dokumentation, Compliance-Prüfungen und Nachweise über Sicherheitsmaßnahmen spielen insbesondere im CRA-Umfeld eine wichtige Rolle.

• Anschließend folgt **Deploy**, also die sichere Bereitstellung des Produkts beim Anwender. Für vernetzte Embedded-Systeme umfasst dies die geschützte Installation, die sichere Konfiguration sowie die zuverlässige Verteilung von Firmware- oder Software-Updates, häufig über OTA-Verfahren (Over the Air).

• Mit **Operate** beginnt der eigentliche Lebenszyklus im Feld. Anders als in klassischen Entwicklungsmodellen endet die Verantwortung des Herstellers nicht mit der Auslieferung. Laufende Überwachung, Patch-Management, Incident Response und das Management neu entdeckter Schwachstellen gehören dauerhaft zum Betrieb.

• Die letzte Phase, **Monitor**, schließt den Kreislauf und führt zurück an den Anfang der Acht. Betriebsdaten, Telemetrie, Sicherheitsereignisse und Erkenntnisse aus dem Feld werden analysiert und fließen unmittelbar in neue Planungs- und Entwicklungszyklen ein.

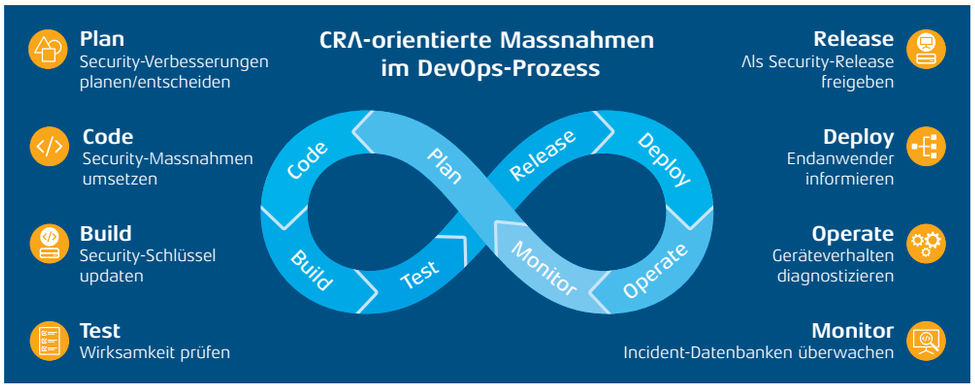

Bild: Grossenbacher Systeme

Die liegende Acht symbolisiert den geschlossenen Kreislauf der DevSecOps-Entwicklung.

Security by Design als dauerhafte Aufgabe

Die Grossenbacher Systeme AG empfiehlt an dieser Stelle den erneuten Einsatz einer strukturierten Bedrohungsanalyse nach dem TARA-Standard. Sie wird damit zur dauerhaften Aufgabe und zu einem zentralen Bestandteil moderner Embedded-Entwicklung nach dem DevSecOps-Prinzip. Dieser Rückkopplungsmechanismus setzt den Gedanken von „Security by Design“ kontinuierlich um und unterstützt die dauerhafte Konformität mit dem Cyber Resilience Act. Sicherheit wird nicht punktuell geprüft, sondern auf Basis einer kontinuierlichen Bedrohungsanalyse fortlaufend weiterentwickelt. Hierfür bietet Grossenbacher Systeme ein Cybersecurity-Monitoring an, das gemeldete Schwachstellen erfasst und in den Entwicklungsprozess zurückführt. Dies stellt Unternehmen vor organisatorische und technische Herausforderungen.

Ein neues Verhältnis zu externen Partnern

DevSecOps verändert die Verantwortlichkeiten und die Zusammenarbeit zwischen den Stakeholdern innerhalb eines Unternehmens sowie mit externen Entwicklern und Security-Experten. Diese begleiten Produkte zunehmend über deren gesamten Lebenszyklus, ähnlich wie im IT-Umfeld.
Maschinenbauer und Industrieausrüster, die sich nicht dauerhaft auf diese Weise binden möchten und auf maßgeschneiderte, applikationsspezifische Geräte verzichten können, sollten den Einsatz von Standardprodukten prüfen. Vor dem Hintergrund des CRA sprechen dafür insbesondere zwei Aspekte: Anpassungen an die eigenen Anforderungen binden in der Regel weniger interne Ressourcen, und die Verantwortung für die CRA-Konformität liegt überwiegend beim Lieferanten. Unternehmen, die bereits mit Partnern zusammenarbeiten, die sowohl Standardprodukte als auch Individualentwicklungen beherrschen und entsprechende Hard- und Software entwickelt haben, verfügen damit über günstige Voraussetzungen. Diese Erfahrung kann die Umsetzung der CRA-Anforderungen und die Auswahl geeigneter Lösungen unterstützen. ag

Web-Tipp

Lesen Sie Teil 1 der Artikelserie ‚CRA-Compliance in der Praxis‘ online:
https://bit.ly/3Tx3sGF

Oliver Roth
ist CEO von Grossenbacher Systeme und der Amalthea Gruppe.