Jonas Wolf, Vector Informatik GmbH

Titel: System-Sicherheitsnachweis durch Introspektion in der KI-gestützten Entwicklung

KI-Agenten erzeugen Code schneller, als Menschen ihn prüfen können, auch für eingebettete, sicherheitsrelevante Steuergeräte. Damit bestimmt nicht mehr die Implementierung das Entwicklungstempo, sondern der Nachweis, dass das Ergebnis sicher ist. Die Funktionalität ändert sich im kontinuierlichen Entwicklungsprozess, daher lässt sich dieser Nachweis nicht als einmaliges Dokument am Projektende führen. Er muss mit jeder Änderung neu entstehen.

Wir stellen einen Ansatz vor, bei dem die Evidenz für den Sicherheitsnachweis als Nebenprodukt der Entwicklung entsteht. Grundlage ist ein Aktor-Framework für verteilte Echtzeitsysteme in Rust, aufgebaut auf einem deterministischen asynchronen Laufzeitsystem. In der sicheren Untermenge von Rust sind Speicherfehler und Data Races zur Übersetzungszeit ausgeschlossen, sodass bei KI-generiertem Code ein Teil der Prüfung vom Review in den Compiler wandert. Das Aktor-Framework stellt menschlichen Entwicklern ebenso wie KI-Agenten einen stabilen Rahmen für die eigene Entwicklung bereit. Das Laufzeitsystem bietet Introspektion ohne zusätzliche Hardware: welche Aufgabe wann ausgeführt wird, wie Daten durch das System fließen und wie viel Rechenzeit und Speicher jede Komponente verbraucht. Verteilte Ausführungen lassen sich aufzeichnen und deterministisch wiederholen.

Diese Introspektion nutzen wir zweifach.
Erstens erhält der KI-Agent einen geschlossenen Regelkreis: Er beobachtet das Verhalten seines Codes auf der Zielhardware, statt nur Tests zu bestehen. Zweitens verdichtet die CI/CD-Pipeline dieselben Messdaten zu Evidenz, etwa zu gemessenen Ausführungszeiten und daraus abgeleiteten Abschätzungen der Worst-Case-Antwortzeiten von Regelfunktionen. Mit dieser Evidenz verknüpft, macht ein Assurance Case in GSN sichtbar, welche Argumente eine Änderung betrifft und welche Evidenz neu erzeugt werden muss.

Wir zeigen den Ist-Stand an einem Demonstrator.
Im Ausblick diskutieren wir zwei offene Fragen: Wie viel Vertrauen verdient Evidenz, wenn Code und Prüfung aus derselben KI-gestützten Werkzeugkette stammen? Und wie lässt sich die Introspektion aus der Entwicklung in die Feldüberwachung übertragen, um Timing- und Ressourcenannahmen des Sicherheitsnachweises auch im Betrieb laufend zu überprüfen?