elbformat – BLOG-ARTIKEL

SOFTWARE ROSTET NICHT.

Wir verstehen Wartung als die Aufrechterhaltung der gewünschten Funktion. Für viele Alltagsgegenstände ist das leicht verständlich. Ölwechsel und Bremsbeläge am Auto oder die Reinigung und Entkalkung des Kaffeevollautomaten sind normal für uns. Aber Software hat keinen physischen Verschleiß. Software kennt keine Korrosion. Was also hat Softwarewartung eigentlich für eine Bedeutung für uns?

Nach der Abnahme kommt die Wartung

Ein offensichtlicher Grund für Wartung sind Fehler und Störungen, die bei Test und Abnahme nicht aufgefallen sind. Wenn also VW Fahrzeuge wegen Problemen mit dem Bremspedal zurückgerufen werden, dann können am Ende KBA Fahrzeuge stillgelegt werden, die nicht repariert wurden. Diesen Prozess sollten wir für die Wartung unserer eigenen Softwareprojekte im Hinterkopf behalten.

Es gibt öffentlich zugängliche Datenbanken über bekannte Sicherheitslücken. Wir können diese Informationen wie einen Rückruf betrachten und geeignete Maßnahmen ergreifen. Meistens stellen Anbieter*innen relativ kurzfristig Updates bereit, die wir dann übernehmen. In kritischen Fällen gibt es für eine Übergangszeit bis zur endgültigen Lösung oft Empfehlungen für eine Absicherung.

Software hat einen Kontext

Moderne Software hat in der Regel mehrstufige Abhängigkeiten zu Modulen von Fremdanbieter*innen. So wie VW seine Bremsen nicht selbst baut, entwickeln wir für Standardfunktionen die Software nicht selbst. Teile bringen die Betriebssysteme mit, vieles lösen wir mit Open Source Komponenten und oft kaufen wir eben auch Module ein.

Die Wartung von Betriebssystemen ist mittlerweile schon sehr gut automatisiert. Wir aktualisieren die Systeme über das Paketmanagement und zentrale Quellen der Hersteller*innen. Nachträglich entdeckte Fehler und Schwachstellen werden so behoben. Auch für Standardsoftware ist dieser Weg oft gut geeignet.

Für eigene Softwareprojekte müssen wir auch einen eigenen Wartungsprozess etablieren. Wir bauen dabei auf dem Entwicklungsprozess auf und beziehen ein Software Supply Chain Management wie Dependency Track mit ein. So steuern wir neue Versionen auch über externe Ereignisse.

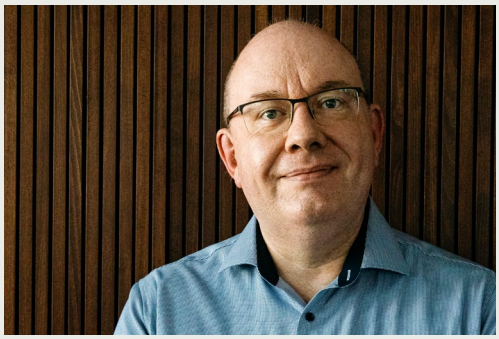
„Wer sucht, der findet auf Anhieb diverse Beispiele für signifikante Schäden.“

Wartung und Nutzen

Der Nutzen von Wartung ist nicht offensichtlich. Wartungsstau und daraus resultierende Mängel bzw. Schwachstellen erkennen nur Expert*innen. Wir brauchen eine viel breitere Akzeptanz für die Kosten der Softwarewartung. Sie ist ein wirksames Mittel um Schaden von uns und anderen fernzuhalten. Wir haben gelernt zu akzeptieren, dass ein Auto mit defekten Bremsen nicht in den Straßenverkehr gehört. Wir schicken unsere Fahrzeuge regelmäßig zur Überprüfung durch Profis (Hauptuntersuchung). Und wir legen sie still, wenn sicherheitsrelevante Mängel nicht beseitigt werden.

Wem die Problematik bezüglich möglicher Schäden zu abstrakt ist, sollte sich fragen, wie gut gewartet die Software für die Notbremsfunktion im Auto ist. Oder die für die Beatmungsmaschine auf der Intensivstation, die Steuerung des Stromnetzes etc. Wer sucht, der findet auf Anhieb diverse Beispiele für signifikante Schäden.

KONTAKT



Jens Dibbern
Projektleiter

jens.dibbern@elbformat.de