GELDINSTITUTE
Bank-IT und Organisation für Entscheider.

Qualitätssicherung bei Banken gemäß BAIT

Ein Artikel von **Florian Kleimann, Senior Consultant Innobis & Alexandra Finke, Öffentlichkeitsarbeit Innobis** | 14.09.2020 - 06:17

Banken-IT bedeutet über Jahre gewachsene und komplexe Systemlandschaften mit in der Regel zahlreichen Eigenentwicklungen. Deren Qualität ist zu sichern und sie sind vor Manipulation zu schützen. Die entsprechenden Vorgaben in der Anwendungsentwicklung sind in den Bankaufsichtlichen Anforderungen an die IT (BAIT) formuliert. Wie lassen sich Quellcodes im Rahmen der Anwendungsentwicklung überprüfen?

Ende 2017 veröffentlichte die Bundesanstalt für Finanzdienstleistungsaufsicht (BaFin) erstmalig *Bankaufsichtliche Anforderungen an die IT (BAIT).* Das Regelwerk trat mit Veröffentlichung sofort in Kraft. Ziel war es, „den Geschäftsleitungen der Institute die Erwartungen der Bankenaufsicht hinsichtlich der sicheren Ausgestaltung der IT-Systeme sowie der zugehörigen Prozesse […] transparent zu machen." Im entsprechenden Rundschreiben der BaFin, das am 14. September 2018 aktualisiert wurde, finden sich 61 Anforderungen an die Banken-IT zur Gewährleistung von IT-Governance und Informationssicherheit. Diese werden für eine Bank als ebenso wichtige Überlebensfaktoren gesehen wie Kapital und Liquidität.

Code Review für sichere Eigenentwicklungen in Banken

Unsere Erfahrung zeigt, dass im Zuge der Erfüllung der BAIT immer noch zu wenig Institute auf eine *sichere Softwarequalität bei den Eigenentwicklungen achten*. Dabei ist diese Anforderung mit einem sogenannten Code Review sehr gut zu erfüllen. Hierbei wird der Quellcode geprüft, um frühzeitig in der Entwicklung Fehler zu identifizieren, zu beseitigen und damit die Softwarequalität zu verbessern beziehungsweise zu sichern.

Zudem: *Fehler, die im Review auffallen, können häufig bedeutend kostengünstiger behoben werden*, als wenn diese erst während der Testdurchführung gefunden werden. Keine Aufsichtsbehörde verlangt explizit den Code Review, aber das Ergebnis muss stimmen: eine sichere Software!

BAIT Vorgaben an die Anwendungsentwicklung

Alexandra Finke ist seit 2011 für die Öffentlichkeitsarbeit von Innobis verantwortlich. Sie betreut bereits seit 1997 zahlreiche IT-Unternehmen im Rahmen ihrer Kommunikation.

Vorab ein Blick in das Regelwerk der BAIT. Unter Punkt 6 „IT-Projekte, Anwendungsentwicklung (inkl. durch Endbenutzer in den Fachbereichen)“ werden die Anforderungen an die Anwendungsentwicklung formuliert und mit der Untersuchung des Quellcodes ein möglicher Lösungsweg genannt.

Satz 36: „Für die Anwendungsentwicklung sind angemessene Prozesse festzulegen, die Vorgaben zur Anforderungsermittlung, zum Entwicklungsziel, zur (technischen) Umsetzung (einschließlich Programmierrichtlinien), zur Qualitätssicherung, sowie zu Test, Abnahme und Freigabe enthalten. Anwendungsentwicklung umfasst beispielsweise die Entwicklung von Software zur Unterstützung bankfachlicher Prozesse oder die von Endbenutzern in den Fachbereichen selbst entwickelten Anwendungen (z. B. Individuelle Datenverarbeitung – IDV). Die Ausgestaltung der Prozesse erfolgt risikoorientiert.“

Satz 39: „Im Rahmen der Anwendungsentwicklung müssen Vorkehrungen getroffen werden, die erkennen lassen, ob eine Anwendung versehentlich geändert oder absichtlich manipuliert wurde. Eine geeignete Vorkehrung unter Berücksichtigung des Schutzbedarfs kann die Überprüfung des Quellcodes im Rahmen der Anwendungsentwicklung sein. Die Überprüfung des Quellcodes [Code Review] ist eine methodische Untersuchung zur Identifizierung von Risiken.“

Kommunikation und Akzeptanz für den sicheren Weg

Trotz der Anforderungen an die Softwarequalität gemäß BAIT, potenzieller Sanktionen und Risiken bei Softwarefehlern oder Manipulation findet bei vielen Banken die Auseinandersetzung mit dem Code der Eigenentwicklungen immer noch unzureichend statt. Der Aufwand erscheint zu groß, schwer

fassbar, und über Jahre gewachsene organisatorische Strukturen erweisen sich als Hemmschuh. Die Herausforderung besteht darin, den Verantwortlichen und betroffenen Mitarbeitern in der IT und den Fachabteilungen die Notwendigkeit zu kommunizieren, Akzeptanz für einen neuen Weg in der Anwendungsentwicklung zu schaffen und damit langjährig bestehende Strukturen aufzubrechen.

Code Review bedeutet, dass die IT den Code prüft. Er muss in Ordnung sein beziehungsweise die Software funktionieren. Dabei gibt es bewährte Software-Tools, die beim Code Review zum Einsatz kommen und einen Großteil der Arbeit übernehmen. Vornehmlich die Entwickler vor Ort gilt es von den „neuen“ integrierten Verfahren mit dem Ziel sicherer Software zu überzeugen.

Am Beispiel: Tool-basierter Code Review in SAP-Systemumgebungen

Die derzeit am Markt verfügbaren Tools für den Code Review rund um bankspezifische Eigenentwicklungen in SAP-Systemlandschaften sind in ihrer Funktionsweise ähnlich. Damit ist die Tool-Auswahl auch kein entscheidender Erfolgsfaktor. Aufgrund der Komplexität dieser Eigenentwicklungen ist der Code Review und der Einsatz eines Tools für die systematische und vollständige Überprüfung aber alternativlos. Ein möglicher Ad-Hoc-Review, in dem Kollegen in regelmäßigen Abständen den Code im Mehraugenprinzip prüfen, ist hier allein unzureichend, da die Qualität vom Know-how, Blickwinkel und der Zeit abhängig ist.

Florian Kleimann ist Senior Consultant bei Innobis. Der studierte Diplom Wirtschaftsinformatiker ist im Bereich Development & Integration Services tätig. Er betreut fortwährend Projekte bei Banken und Kreditinstituten rund um SAP Anwendungsentwicklung, Softwareentwicklungs- sowie ITSM-Prozesse. © Daniel Reinhardt

Aus unserer Sicht haben sich zwei Werkzeuge besonders für die Analyse kundeneigener Entwicklungen im SAP-Umfeld bewährt. Beide lassen sich gut in die bestehende Entwicklungsumgebung sowie den Entwicklungsprozess integrieren: der *SAP Code Vulnerability Analyzer (CVA)* der SAP sowie der *Code Profiler for ABAP der Firma Virtual Forge*. Bei beiden handelt es sich um Code-Scanner für ABAP-Eigenentwicklungen. Die Kostenspanne liegt dabei – je nach Anforderung – zwischen etwa 15.000 bis zu 200.000 Euro. Lizenz und Wartung des SAP-Tools CVA sind mit einem Hauptvertrag bei SAP schon abgedeckt.

Die Tools bieten zahlreiche, automatisierte Prüfungen an, die je nach Bedarf ausgewählt und eingestellt werden können. Es handelt sich dabei um prioritäre Prüfungssets beispielsweise im Hinblick auf Sicherheit und Compliance sowie sekundäre Prüfungen mit Blick auf Bereiche wie Performance, Wartbarkeit und Robustheit. Wichtig ist, dass ein Verantwortlicher die Hoheit über den Software-basierten Code Review innehat. Er wählt die Prüfungssets aus, setzt die Prioritäten, macht entsprechende Vorgaben für die Entwickler und verantwortet die Code Quality. Dabei kann diese Rolle als Schnittstelle zwischen den Entwicklern, dem Management und der Informationssicherheit verstanden werden.

Umgang mit Altbefunden und Berechtigungen

Soll die Software bei der Anwendungsentwicklung dauerhaft reibungslos funktionieren, müssen zunächst alle über das Tool identifizierten Altbefunde in den *bestehenden Eigenentwicklungen* bearbeitet werden. Ansonsten steht das System jedes Mal still, wenn es an die entsprechenden Code-Stellen kommt. Um jedoch keine wertvolle Zeit zu verlieren ist eine bessere Alternative, die Altbefunde vorläufig zu akzeptieren und dann parallel in Extraprojekten schrittweise zu beheben.

In den Code-Review-Tools sind Prüfungen auf fehlende oder inkonsistente Berechtigungsprüfungen ein wesentlicher Bestandteil, um die Daten vor dem Zugriff Unberechtigter zu schützen. Daneben gibt es natürlich auch das zentrale Berechtigungsmanagement mit technischen und fachlichen Rollen sowie in der Vergabe in der Regel mit einem Vieraugenprinzip.

Zusammen mit dem Berechtigungsmanagement muss eine ganzheitliche Sicht auf die Vergabe von Berechtigungen zur Sicherung des Gesamtsystems entstehen. Eine einzelne Auswertung zu schützen ist nicht ausreichend, wenn aus der Historie heraus Unberechtigte etwa Berechtigungen für die Tabellenanzeige oder den SAP QuickViewer haben (beides SAP-Standard-Transaktionen). Der Abgleich dieser Beziehung und Wechselwirkung zueinander ist sehr aufwändig. Es ist nahezu unmöglich alle Sicherheitslücken zu schließen. Was aber möglich ist: Werden über den Code Review Sicherheitslücken in den Berechtigungen gefunden, lassen sich gleichzeitig Empfehlungen für das zentrale Berechtigungsmanagement aussprechen.

Der Nutzen rechtfertigt Kosten und initialen Aufwand

Software-basierter Code Review bietet großen Nutzen, da er wesentlich bei der Erfüllung der BAIT im Hinblick auf die Anwendungsentwicklung unterstützt. Oberstes Ziel ist dabei, möglichst sichere, manipulationsfreie und gut wartbare Software zu produzieren. Zu Beginn ist ein größerer Arbeitsaufwand notwendig, um das Tool einzuführen, sich damit vertraut zu machen sowie für die Bereinigungsphase der Altbefunde. Nach dieser anfänglichen Hürde ist der Mehrwert aber entsprechend groß. Weitere Vorteile für Banken im Überblick:

In den IT-Abteilungen wird zusätzliches Know-how durch die Problemlösung bei wiederkehrenden Befunden und das Ausarbeiten von Best-Practice-Ansätzen zur Schaffung von einheitlichen, wiederverwendbaren Lösungen aufgebaut. Diese sollten in die Architekturvorgaben aufgenommen werden, damit Korrekturen von Befunden nach stets dem gleichen Schema erfolgen.

Der Wissensaustausch und das Teilen von Information intensivieren sich durch die mit dem Code Review neu geschaffenen Verantwortlichkeiten und Prozesse. Es ist damit nicht mehr möglich, dass das Wissen über den Code eines Programms bei nur einer Person liegt. Dies ist insbesondere dann kritisch, wenn die Verantwortlichkeit von historisch gewachsener Software für sensible Bereiche wie z.B. Risikocontrolling oder Meldewesen bei nur einem Mitarbeiter liegt.

Und nicht zuletzt erhöht sich die Reaktionsfähigkeit in der laufenden Entwicklung – Fehler sind über Altbefunde sowie über die Best-Practice-Lösungen bekannt und können schnell korrigiert bzw. von Anfang an vermieden werden.

Fazit

Einige Banken kennen die Tools, haben diese im Einsatz, aber aus Erfahrung nicht so scharf geschaltet, dass sie im Zweifel etwas verhindern könnten. Wichtig ist es, eine „Linie zu markieren“ bis wohin alles erlaubt bzw. genehmigt ist. Dabei ist der Start mit einer O-Linie und der sukzessive Ausbau des Prüfungsset nach Prioritäten durchaus legitim. Aufgabe eines IT-Dienstleisters, der bei der Einführung der Code-Review-Software unterstützt, ist es, das Institut langfristig in die Lage zu versetzen, das System selbst einzustellen und korrekt anzuwenden. Der Nutzen ist dann in jeder Hinsicht maximal.