Vier Spezialisten veröffentlichen: „Sichere ABAP-Programmierung“

Unbreakable ABAP?

Auch ABAP-Programme sind das Ziel von Hacker-Angriffen. Wo liegen die Risiken? Dieser Beitrag stellt das neue Buch „Sichere ABAP-Programmierung“ vor, das auf jahrelangen Erfahrungen in den Bereichen SAP-Penetrationstests und ABAP-Quellcodeaudits basiert.

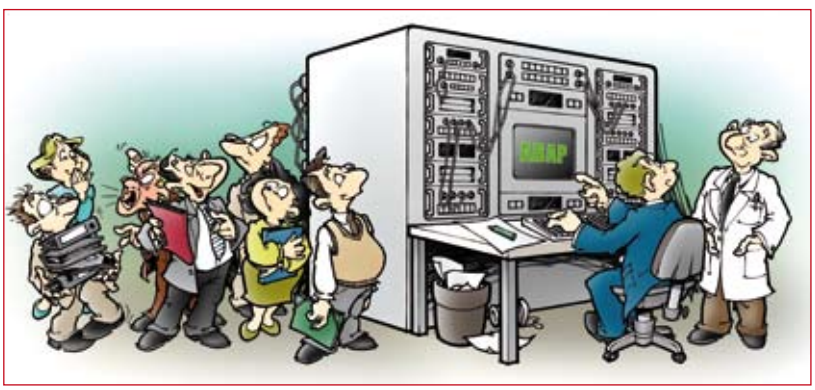
Unsere SAP-Sicherheitsanalysen zeigen es immer wieder: die Sicherheitsdefekte, mit denen kein Unternehmen rechnet, sind die im eigenen Code. Das gilt für ABAP noch mehr, als für Java. Irgendwie hat sich in der Vergangenheit bei den Verantwortlichen der Mythos gebildet, ABAP Programme seien automatisch „unbreakable“. Das ist natürlich falsch. In ABAP kann man genauso unsicheren Code schreiben wie in jeder anderen Programmiersprache. Da SAP-Systeme zunehmend mit externen Systemen vernetzt und auch via Internet erreichbar sind, spielt sichere Programmierung eine immer größere Rolle. „Die Aussage, dass man sich mit der Sicherheit von ABAP-Code nicht näher beschäftigen muss, gilt längst nicht mehr so pauschal wie noch vor zehn Jahren.“ sagt auch Dr. Gunter Bitz von der SAP im Vorwort zu “Sichere ABAP-Programmierung”. Es zeigt sich, dass insbesondere selbst entwickelter ABAP-Code Sicherheitsdefekte enthält. Aber auch Beratungshäuser und ISV’s liefern nicht unbedingt sicheren Code ab. Bei Audits von SAP-Systemen haben wir schon öfter verheerende Fehler im Code von ISV’s entdeckt. Eine E-Recruitment Komponente eines Drittanbieters erlaubte beispielsweise jedem Benutzer aus dem Internet die Dokumente aller Bewerber unberechtigt einzusehen. Solche Fehler sind nicht nur schlecht für den guten Ruf. Sie können auch ernsthafte rechtliche Konsequenzen haben, wenn durch sie Gesetzte verletzt werden, wie in diesem Fall das Bundesdatenschutzgesetz.

Geringster Widerstand

Es reicht also nicht aus, SAP Systeme richtig zu konfigurieren, Verschlüsselungstechniken einzusetzen und Rollen und Berechtigungen korrekt zu pflegen. Der Angreifer wird immer den Weg des geringsten Widerstandes gehen. Und das ist bei SAP-Installationen derzeit der selbst entwickelte Code. Aber warum gibt es so viele Sicherheitsfehler im ABAP-Code? Ganz einfach: Weil es bis dato keine umfassende Publikation über Sicherheit in der Programmiersprache ABAP gab. Dadurch wusste praktisch niemand von dem Problem und entsprechend gab es keine Anforderungen oder (Test-) Anleitungen für sicheren ABAP-Code. ABAP-Entwickler hatten bislang also gar keine Möglichkeit sich qualifiziert zu informieren, wie sie sicheren ABAP-Code schreiben können. Entsprechend groß ist das Problem in der Industrie heute: Milliarden Zeilen von ABAP werden täglich weltweit ausgeführt um die Geschäftsprozesse der größten Unternehmen der Welt zu unterstützen. Und die allerwenigsten davon wurden auch nur ansatzweise sicher entwickelt. Viele große Unternehmen trauen sich schon gar nicht ihren Code auf Sicherheitsprobleme untersuchen zu lassen, aus Angst vor den Folgekosten für die Fehlerbehebung. Das ist Risikomanagement der besonderen Art. Da von mindestens einer kritischen Sicherheitslücke in 2000 Zeilen ABAP-Code ausgegangen werden kann, grenzt dies an Fahrlässigkeit. Bislang gab es schon einige gute Publikationen zu anderen Bereichen der Sicherheit. Dabei möchten wir zwei Bücher hervorheben: „Sicherheit und Berechtigungen in SAP-Systemen“ von Mario Linkies und Frank Off und „Programmierhandbuch SAP NetWeaver Sicherheit“ von Martin Raepple - dort wird beschrieben, wie einige Sicherheitsfunktionen des SAP-Standards verwendet werden. Beides reicht jedoch nicht aus, wenn der Code Sicherheitslücken aufweist.

Umfassendes Werk

Wir haben jetzt das Buch „Sichere ABAP-Programmierung“ geschrieben, damit es endlich ein umfassendes Werk zum Thema Sichere ABAP-Programmierung gibt. Insbesondere soll das Buch Architekten, Entwicklern und Entscheidern als Leitfaden und Nachschlagewerk für die Entwicklung von sicheren ABAP-Programmen dienen. Aber wir sprechen damit auch Auditoren an, Ihre Prüflisten zu erweitern. Für Auditoren dürften insbesondere Hintertüren interessant sein, also versteckte Anweisungen, die bestimmten Benutzern mehr Rechte geben, als das Berechtigungskonzept für sie vorsieht. Wir haben Anfang dieses Jahres in einer Sicherheitsuntersuchung eine besonders gefährliche Hintertür entdeckt. In einer Webseite (BSP) war für vier festegelegte Benutzer eine besondere Funktionalität eingebaut. Während sich die Seite allen anderen Benutzern als leer darstellte, erlaubte sie den vieren, den gesamten Inhalt jeder beliebigen Tabelle im SAP-System einzusehen. Wie war dies möglich, obwohl die Benutzer gar keine Be-



Die Autoren

Andreas Wiegenstein ist Geschäftsführer der Virtual Forge GmbH. Bereits mit 15 Jahren entdeckte er erste Sicherheitslücken in einem Programm. Seit 2002 beschäftigt er sich mit der Sicherheit in SAP-Szenarien. Er hält Vorträge auf internationalen Konferenzen, wie auch schon mehrfach auf der SAP TechEd.

Frederik Weidemann konzentrierte sich schon früh auf die IT-Sicherheit und im speziellen auf die angewandte Applikationssicherheit. Er publiziert nicht nur Fachartikel, sondern hält auch Trainings zu sicherer Programmierung ab. Sein Talent als Sicherheitstester entdeckte er während seines Studiums an der TU Darmstadt.

Sebastian Schinzel ist seit mehreren Jahren Entwickler und Sicherheitsberater in einem stetig wachsenden Spektrum von Technologien und Domänen. Bereits in seiner Studienzeit befasste er sich mit der Sicherheit von verteilten Systemen und statischer Codeanalyse. Auch er schreibt Artikel über sichere SAP-Anwendungen.

Dr. Markus Schumacher war bei der SAP als Product Manager (NetWeaver Security) tätig. Bei Virtual Forge beschäftigt er sich aus betriebswirtschaftlicher Sicht mit dem Thema Softwaresicherheit. Er hat im Fachgebiet Informatik promoviert sowie zahlreiche Artikel und Fachbücher veröffentlicht.

Sichere ABAP-Programmierung,
372 Seiten, Galileo Press 2009
ISBN 978-3-8362-1357-8

rechtigung hatten, auf diese Tabellen zuzugreifen? Ganz einfach: Sie brauchten keine. ABAP-Code kann technologisch auf alle Daten des Systems zugreifen. Soll ein Benutzer bestimmte Aktionen nicht ausführen, so muss dazu im ABAP-Code explizit eine Berechtigungsprüfung stattfinden. Je nach Ergebnis dieser Berechtigungsprüfung, führt dann der ABAP-Code die Aktion aus oder nicht. Aber der ABAP-Code könnte die Aktion ebenso auch ohne die Berechtigungsprüfung ausführen. Fehler im Code - seien sie nun absichtlich oder unabsichtlich - können somit zu erheblichen Sicherheitsrisiken führen.

„You hacker out there“

In unserem Buch gehen wir das Thema sichere Programmierung aus mehreren Richtungen an, die sich an die verschiedenen Zielgruppen richten. Nach der Einleitung erklären wir zunächst, was bei der ABAP-Programmierung anders ist, als bei anderen Sprachen. Anschließend stellen wir Methoden und Werkzeuge für die Entwicklung sicherer ABAP-Software vor. Nachdem die ersten Kapitel eher allgemein gehalten sind, folgen danach einige technische Kapitel. Diese beginnen mit allgemeinen Best Practices für die Programmierung, gehen über zu spezifischen Problemen bei der klassischen ABAP-Programmierung und zeigen schließlich, worauf man bei der Webprogrammierung mit ABAP achten muss. Pro Abschnitt erläutern wir die Anatomie der Schwachstellen, Risiken, Maßnahmen und Anleitungen für Selbsttests. In diesen Kapiteln werden die wichtigsten technischen Grundbausteine gelegt und viele gefährliche ABAP-Programmiertechniken vorgestellt. Wir beschreiben Schwachstellen wie SQL Injection, ABAP-Command Execution und Cross-Site Scripting. Im folgenden Kapitel betrachten wir das ganze mehr aus Sicht der zu verwendenden Technologien: Was ist bei Dateizugriffen zu beachten? Was sind die Risiken von Business Server Pages (BSP's)? Worauf muss man bei Web Dynpro achten? Natürlich enthalten alle Kapitel Checklisten, die für den täglichen Gebrauch gedacht sind. Abschließend betrachten wir ABAP-Sicherheit im Kontext ausgewählter Business-Szenarien wie z.B. E-Recruitment.

Wir haben versucht, das eher trockene Thema Sicherheit durch zahlreiche Anekdoten zu beleben, die sich im Laufe der Jahre bei uns angesammelt haben. Wir hoffen, mit diesem Buch einen Grundstein für zukünftig sichere ABAP-Programmierung gelegt zu haben, der es den Hackern da draußen schwerer macht und dadurch den Sicherheitsverantwortlichen einen ruhigeren Schlaf verschafft.

www.sap-press.de