Fraunhofer FOKUS

FRAUNHOFER-INSTITUT FÜR OFFENE KOMMUNIKATIONSSYSTEME FOKUS

STANCE

Kontakt
Dr.-Ing. Jens Gerlach
Forschungsgruppenleiter Verifikation
System Quality Center – SQC
Tel. +49 30 3463-7458
jens.gerlach@fokus.fraunhofer.de

Fraunhofer FOKUS
Kaiserin-Augusta-Allee 31
10589 Berlin

www.fokus.fraunhofer.de/go/STANCE

An die Steuerungssoftware von sicherheitskritischen Systemen, z. B. in Zügen oder Autos, werden hohe Sicherheitsanforderungen gestellt. Solche Systeme müssen in jedem Fall zuverlässig und sicher funktionieren, damit keine Gefährdungen entstehen. Der Begriff »Sicherheit« hat hier zwei Bedeutungen, die im Englischen durch die Worte »safety« und „security" beschrieben werden. Die Betriebssicherheit (engl. »safety«) befasst sich mit der Frage, ob durch den Betrieb des Systems unakzeptable Risiken für Menschen oder die Umgebung auftreten können. Bei der Angriffssicherheit (engl. »security«) soll das System vor Angriffen von außen, z. B. durch Hacker, geschützt werden. Das Projekt STANCE (Source Code Analysis Toolbox for Software Security Assurance) befasst sich vor allem mit dem Problem der Angriffssicherheit von Software.

Vertrauen in die Sicherheit vor Angriffen

Die Sicherheit von Informations- und Kommunikationstechnologien in unserer Umgebung vor Angriffen muss garantiert sein, damit die Menschen Vertrauen in sie haben können. Der Anspruch ist: Technologie soll nicht nur Dinge ermöglichen, sondern auch ihre Nutzerinnen und Nutzer schützen! Aus diesem Grund hat es sich das Projekt STANCE zur Aufgabe gemacht, Technologien zu erforschen und zu entwickeln, die die Sicherheit von Software vor Angriffen garantieren. Zehn europäische Partner aus Wissenschaft und Industrie wollen eine Toolbox für Quellcode-Analysen bereitstellen, mit der verifiziert werden kann, dass Softwaresysteme immun gegen bestimmte Kategorien von Angriffen sind. Mit diesen Technologien sollen Softwareentwickler beweisen können, dass Programme und deren Funktionalitäten stets ihrem zuvor spezifizierten Verhalten entsprechen. Wenn dies möglich ist, entsteht bei Nutzern ein starkes und gut begründetes Vertrauen in die entwickelten Systeme.

IM PROJEKT STANCE WERDEN ANALYSETOOLS FÜR DIE VERIFIKATION DER ANGRIFFSSICHERHEIT VON KOMPLEXEN SOFTWARESYSTEMEN ENTWICKELT

Schwachstellenanalyse in Programmen

Im Projekt sollen eine Reihe von Programmanalysetools definiert, implementiert und validiert werden, die in der Lage sind, die Angriffssicherheit von komplexen Softwaresystemen zu verifizieren. Programmanalyse bezeichnet verschiedene, teilweise formale, Methoden, mit denen ungeplantes Verhalten von Softwaresystemen semiautomatisch entdeckt werden kann. Bisher werden formale Methoden bei der Programmanalyse nur selten im Bereich der Angriffssicherheit von Software eingesetzt. Das liegt unter anderem daran, dass die populären Programmiersprachen – darunter Java und C++ – sehr komplex sind, was den Einsatz formaler Methoden erschwert. Im Projekt werden daher Werkzeuge und Methoden entwickelt, mit denen die Sicherheitseigenschaften der kritischen Komponenten von Programmen umfassend analysiert werden können. Das Programmanalysetool Frama-C von CEA-LIST sowie das Verifikationswerkzeug VeriFast der KU Leuven, die bisher hauptsächlich für C einsetzbar sind, sollen in diesem Rahmen so erweitert werden, dass auch Java- und C++-Software untersucht werden kann. Vorhandene Lösungen (formale Methoden, moderne statische und dynamische Programmanalysetools, bestehende Expertise zur Security-Evaluation und relevantes industriespezifisches Wissen) werden im STANCE-Projekt sowohl genutzt als auch signifikant erweitert.

Statische und dynamische Analyse

Das Projekt will eine Brücke zwischen statischen und dynamischen Programmanalysetools bauen. Dynamische Analyse bezeichnet die Analyse von Programmen, die stattfindet, während das Programm ausgeführt wird – mit anderen Worten: Es wird getestet. Dabei läuft der kompilierte Softwarecode mit festgelegten Testwerten ab. Auf diese Weise soll unerwartetes Verhalten des Programms entdeckt werden. Dabei wird der Quellcode oder auch der kompilierte Code untersucht, um Schwachstellen, die ein Einfallstor für Angriffe sein können, frühzeitig zu identifizieren. Der Fokus im Projekt liegt auf der statischen Analyse, allerdings werden auch dynamische Methoden berücksichtigt. Zunächst werden typische Schwachstellen von Software formalisiert (d. h. auf einer höheren Abstraktionsebene mit mathematischer Strenge beschrieben), damit diese im Verifikationsprozess von den Analysetools werkzeuggestützt gefunden werden können. Damit die fertige Software so gut wie möglich gegen potenzielle Angreifer gewappnet ist und möglichst keine Angriffsflächen und Schwachstellen mehr bietet, werden im STANCE-Projekt dynamische und statische Analyse kombiniert. Diese Kombination ist sinnvoll, da der Einsatz statischer Analysemethoden für mehr Effizienz und damit geringere Kosten bei der Software-Qualitätssicherung sorgt.

Wir machen Städte schlau

bad | 1402 (Fotos: Matthias Heyde; iStockphoto.com)