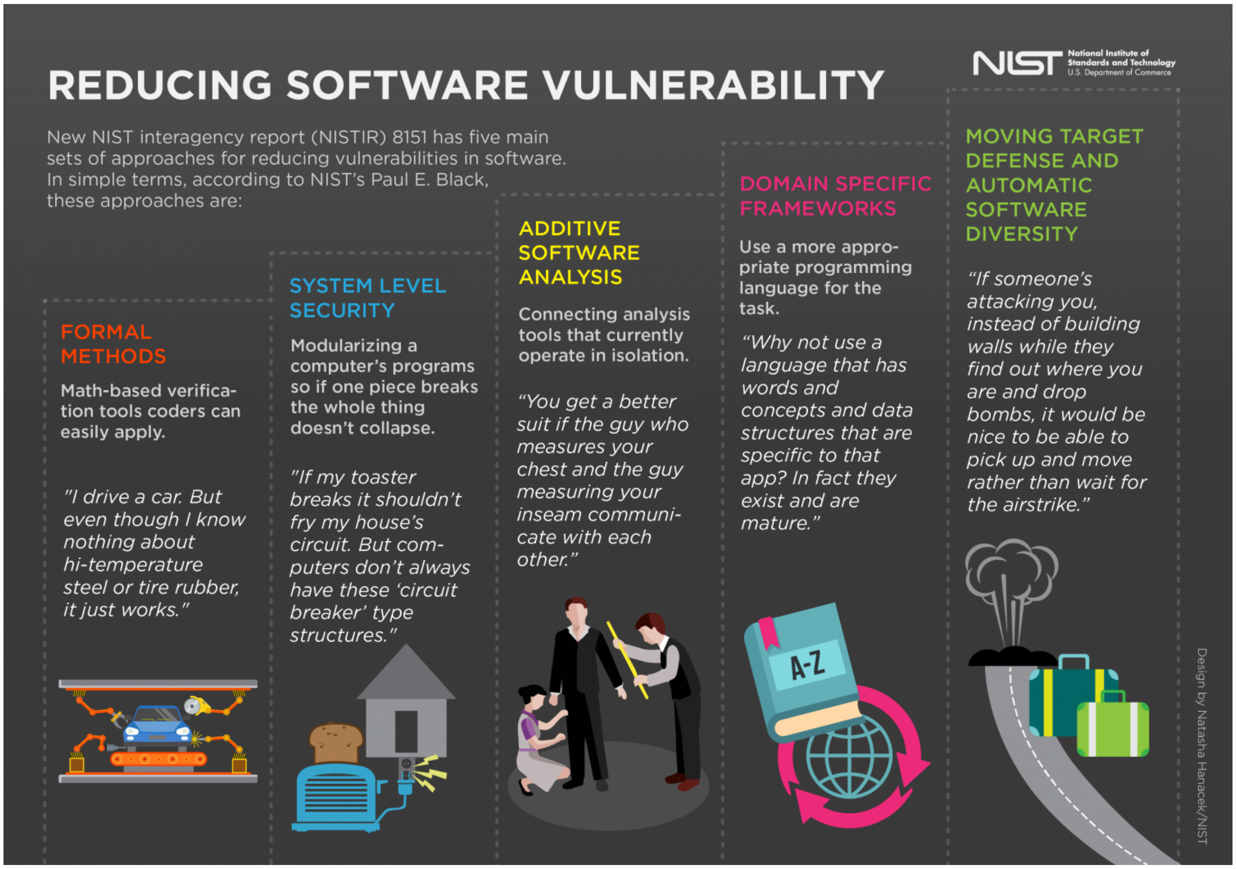
PHYS.ORG

Safer, less vulnerable software is the goal of new NIST computer publication

December 5 2016



REDUCING SOFTWARE VULNERABILITY

NIST National Institute of Standards and Technology U.S. Department of Commerce

New NIST interagency report (NISTIR) 8151 has five main sets of approaches for reducing vulnerabilities in software. In simple terms, according to NIST's Paul E. Black, these approaches are:

FORMAL METHODS

Math-based verification tools coders can easily apply.

"I drive a car. But even though I know nothing about hi-temperature steel or tire rubber, it just works."

SYSTEM LEVEL SECURITY

Modularizing a computer's programs so if one piece breaks the whole thing doesn't collapse.

"If my toaster breaks it shouldn't fry my house's circuit. But computers don't always have these 'circuit breaker' type structures."

ADDITIVE SOFTWARE ANALYSIS

Connecting analysis tools that currently operate in isolation.

"You get a better suit if the guy who measures your chest and the guy measuring your inseam communicate with each other."

DOMAIN SPECIFIC FRAMEWORKS

Use a more appropriate programming language for the task.

"Why not use a language that has words and concepts and data structures that are specific to that app? In fact they exist and are mature."

MOVING TARGET DEFENSE AND AUTOMATIC SOFTWARE DIVERSITY

"If someone's attacking you, instead of building walls while they find out where you are and drop bombs, it would be nice to be able to pick up and move rather than wait for the airstrike."

Design by Natasha Hanacek/NIST

A new NIST report recommends five main approaches for reducing software vulnerabilities, described in lay terms by NIST computer scientist Paul E. Black. Credit: Hanacek / NIST

We can create software with 100 times fewer vulnerabilities than we do today, according to computer scientists at the National Institute of Standards and Technology (NIST). To get there, they recommend that coders adopt the approaches they have compiled in a new publication.

The 60-page document, NIST Interagency Report (NISTIR) 8151: Dramatically Reducing Software Vulnerabilities, is a collection of the newest strategies gathered from across industry and other sources for reducing bugs in software. While the report is officially a response to a request for methods from the White House's Office of Science and Technology Policy, NIST computer scientist Paul E. Black says its contents will help any organization that seeks to author high-quality, low-defect computer code.

"We want coders to know about it," said Black, one of the publication's coauthors. "We concentrated on including novel ideas that they may not have heard about already."

Black and his NIST colleagues compiled these ideas while working with software assurance experts from many private companies in the computer industry as well as several government agencies that generate a good deal of code, including the Department of Defense and NASA. The resulting document reflects their cumulative input and experience.

Vulnerabilities are common in software. Even small applications have hundreds of bugs by some estimates. Lowering these numbers would bring many advantages, such as reducing the number of computer crashes and reboots users need to deal with, not to mention decreasing the number of patch updates they need to download.

The heart of the document, Black said, is five sets of approaches, tools and concepts that can help, all of which can be found in the document's second section. The approaches are organized under five subheadings that, despite their jargon-heavy titles, each possess a common-sense idea as an overarching principle (see downloadable infographic).

These approaches include: using math-based tools to verify the code will work properly; breaking up a computer's programs into modular parts so that if one part fails, the whole program doesn't crash; connecting analysis tools for code that currently operate in isolation; using appropriate programming languages for the task that the code attempts to carry out; and developing evolving and changing tactics for protecting code that is the target of cyberattacks.

In addition to the techniques themselves, the publication offers recommendations for how the programming community can educate itself about where and how to use them. It also suggests that customers should request the techniques be used in development. "You as a consumer should be able to write it into a contract that you want a vendor to develop software in accordance with these principles, so that it's as secure as it can be," Black said.

Security is, of course, a major concern for almost everyone who uses technology these days, and Black said that the White House's original request for these approaches was part of its 2016 Federal Cybersecurity R&D Strategic Action Plan, intended to be implemented over the next three to seven years. But though ideas of security permeate the document, Black said the strategies have an even broader intent.

"Security tends to bubble to the surface because we've got adversaries who want to exploit weaknesses," he said, "but we'd still want to avoid bugs even without this threat. The effort to stymie them brings up general principles. You'll notice the title doesn't have the word 'security' in it anywhere."

More information: Dramatically Reducing Software Vulnerabilities, doi.org/10.6028/NIST.IR.8151

Provided by National Institute of Standards and Technology

Citation: Safer, less vulnerable software is the goal of new NIST computer publication (2016, December 5) retrieved 25 April 2024 from https://phys.org/news/2016-12-safer-vulnerable-software-goal-nist.html