DEPENDABILITY Research Group

Our research focuses on issues related to software and architectures in high-value systems - computing systems of extreme importance to society whose failure would have a severe negative impact whether measure in terms of time, money or loss of life. Our work encompasses safety-critical systems, such as medical devices, avionics, unmanned aircraft systems, weapons systems; critical infrastructures such as financial networks, transportations systems, and power systems; and emergent networked information systems that increasingly play a strategically vital role in such diverse industries as finance, healthcare, pharmaceuticals, and aerospace.

John Knight

Professor
knight@cs.virginia.edu
www.cs.virginia.edu/~jck/

Department of Computer Science
University of Virginia
Charlottesville, VA
434.982.2216

"Don't put too much trust in computers."

UNIVERSITY of VIRGINIA
ENGINEERING

Architectures

The Helix self-regenerative architecture is designed to be a next generation network security architecture. Helix employs a combination of defense mechanisms that is both highly effective and metamorphic, thereby presenting attackers with a continuously changing attack surface, i.e., a metamorphic shield that is altered routinely and as attacks progress. An attack that manages to overcome these defenses is then faces with the Helix innate response mechanism which creates a more aggressive system metamorphosis. This metamorphosis seeks to contain the effects of the attack and to reconfigure to provide rapid recovery and continued service. Finally, the Helix adaptive response mechanism examines the basic application system design at the level of its implementation and effect repairs that will ensure that future attacks of the same or similar forms will be deflected, either by removing the path to vulnerabilities or the vulnerabilities themselves.

Formal Verification

Mathematical certainty that a software system implies the associated specification, i.e., formal verification, is highly desirable. Although a proof does not obviate the need for a full complement of software engineering techniques, a proof does increase confidence in software considerable. Echo is a formal verification technique that has been shown to be effective, efficient and scalable. With Echo formal verification becomes practical for large programs.

Rigorous Safety Case Technology

Rigorous safety cases based on the systematic use of explicit arguments have emerged as a powerful assessment mechanism for safety-critical systems. Explicit arguments document the developers' rationale for believing that a delivered system is adequately safe. Safety cases address many of the difficulties faced by traditional prescriptive standards.
Assurance Based Development is a software engineering technology that adapts rigorous argument to assurance of software dependability. Assurance Based Development introduces the concept of synergistic constructions of a critical computing system and an assurance case that sets out the dependability claims for the system. The assurance case argues that the available evidence justifies those claims. Co-developing the system and its assurance case helps software developers make technology choices that address the specific dependability goal of each component.

RECENT RESEARCH DEVELOPMENTS

- Confidence arguments enable clearer arguments in existing safety case technology
- Echo software verification technology. Echo facilitates the complete formal verification of software functionality for large software systems
- Secretless security. The N-variant system architecture provides comprehensive security without requiring secrets such as cryptographic keys

RECENT GRANTS

- DOD/Air Force/iARPA-Helix-A Self-Regenerative Architecture
- NSF-Practical Formal Verification by Specification Extraction
- NSF-A Rationale Approach to Safety-Critical Software Development

SEAS Research Information
Pamela M. Norris, Associate Dean
University of Virginia
Box 400242
Charlottesville, VA 22903
pamela@virginia.edu
434.243.7683

UNIVERSITY of VIRGINIA ENGINEERING