HOL-TESTGEN/FW
An Environment for Specification-Based Firewall Conformance Testing

Achim D. Brucker\textsuperscript{1}, Lukas Brügger\textsuperscript{2}, and Burkhart Wolff\textsuperscript{3}

\textsuperscript{1} SAP AG, Vincenz-Priessnitz-Str. 1, 76131 Karlsruhe, Germany
achim.brucker@sap.com
http://www.brucker.ch/

\textsuperscript{2} Information Security, ETH Zurich, 8092 Zurich, Switzerland
lukas.a.bruegger@gmail.com

\textsuperscript{3} LRI, Université Paris-Sud, 91405 Orsay Cedex, France
wolff@lri.fr
http://www.lri.fr/~wolff

Abstract. The HOL-TESTGEN environment is conceived as a system for modeling and semi-automated test generation with an emphasis on expressive power and generality. However, its underlying technical framework Isabelle/HOL supports the customization as well as the development of highly automated add-ons working in specific application domains.

In this paper, we present HOL-TESTGEN/fw, an add-on for the test framework HOL-TESTGEN, that allows for testing the conformance of firewall implementations to high-level security policies. Based on generic theories specifying a security-policy language, we developed specific theories for network data and firewall policies. On top of these firewall specific theories, we provide mechanisms for policy transformations based on derived rules and adapted code-generators producing test drivers. Our empirical evaluations show that HOL-TESTGEN/fw is a competitive environment for testing firewalls or high-level policies of local networks.

Keywords: symbolic test case generations, black box testing, theorem proving, network security, firewall testing, conformance testing.

1 Introduction

HOL-TESTGEN \cite{6,7} (http://www.brucker.ch/projects/hol-testgen/) is a generic model-based testing environment. Built as an extension of the Isabelle framework \cite{15}, HOL-TESTGEN inherits, among other things, the front-end PIDE, the Isar language for HOL specifications and proofs, and code- and documentation generators from the Isabelle framework. HOL-TESTGEN extends the framework by an infrastructure to develop formal test plans, i.e., descriptions of test goals, their decomposition into abstract test partitions, and their transformation to concrete tests with the help of constraint solvers like Z3 \cite{12}. Finally, customized code-generators produce code of concrete test drivers which can be run against real implementations following a black-box testing approach.
HOL-TestGen as such is conceived as an interactive, flexible environment that draws from the abundant expressive power and generality of HOL; test plans are therefore typically mixtures of very powerful automated partitioning and selection tactics, their configurations, and intermediate small-step tactics that help to turn the results into a suitable form for the next step. HOL-TestGen was used successfully in large case studies from various domains, see [7] for details.

In this paper, we present the novel HOL-TestGen/fw environment, which is an add-on of HOL-TestGen for a specific problem domain: the specification-based conformance test of network components. Such components can be stateless packet filters, stateful firewalls, routers, devices performing network address translation (NAT), etc. In the sequel we just refer to them as firewalls. We describe the underlying generic theories for modeling network data and firewall policies using a generic security-policy language called the **Unified Policy Framework** (UPF) [3, 8], mechanisms for policy transformations (for which formal proofs of correctness have been established [2]) and adapted code generators producing test drivers. We present application scenarios as well as experimental evaluations which show HOL-TestGen/fw as a competitive environment for testing firewalls or high-level policies of networks.

## 2 The HOL-TestGen/fw Workflow

HOL-TestGen/fw is an environment for the specification-based conformance testing of firewalls.

---

**Fig. 1.** The HOL-TestGen/fw Architecture

---

1 HOL-TestGen/fw (including case studies) is, since version 1.7.1, part of the HOL-TestGen distribution. HOL-TestGen is available from: [http://www.brucker.ch/projects/hol-testgen/]