Bart van Delft

Paragon : practical programming with information flow control
- Overview and Roadmap

Paralocks is an information flow policy specification language developed at Chalmers and has found its way as an extension to the Java language. This extension, called Paragon, provides a practical way to write secure programs.

Classical protection mechanisms in computer security only regulate the access to resources but are not concerned with how the information obtained from these resources afterwards flows through a process. These mechanisms are therefore incapable of preventing information leakage. In the field of Language Based Security the analysis of process-internal information flows haw received much attention in the past decade and multiple approaches to addressing this problem have been put forth. Many of these results are theoretical however and most of the practical implementations (of information flow analyzers) are either highly complicated to work with or written for academic purposes only.

Paralocks is a serious attempt to create simple, semantically well-defined but at the same time highly expressive policy specification language.
Policies can
be expressed in a simple non-restraining logic. That is, we do not force the policies to address the confidentiality, integrity, taint-level or similar property of information flow, but leave this in the hands of the policy writer.
Paragon is the embodiment of Paralocks in the Java programming language, where we continue to aim for a simple and practical language.

In this presentation we take a look at the current features of Paralocks and Paragon, as well as future extensions that we are planning.