Transparent, performant, non-privileged provenance tracing through library interposition

Abstract

System-level provenance tracing is the idea of automatically capturing how computational artifacts came to be (e.g., what version of the program wrote this file). While provenance is often discussed in the context of security, it also fills an important niche in computational science, providing data for reproducibility, debugging, and assigning credit. Unlike a security administrator, computational scientists do not necessarily have root-level access to the machine on which they want to trace provenance. Prior work proposes recompiling with instrumentation, ptrace, and kernel-based auditing, which at best achieves two out of three desirable properties: transparency, performance, and non-privilege. We present PROBE, a system-level provenance tracer that uses library interpositioning to achieve all three. We evaluate the performance of PROBE on system microbenchmarks and scientific applications. We also discuss the completeness of the provenance that PROBE collects compared to other provenance tracers.