An Infrastructure for Faithful Execution of Remote Attestation Protocols

Adam Petz
ampetz@ittc.ku.edu
University of Kansas
Lawrence, Kansas

ABSTRACT

Experience shows that even with a well-intentioned user at the keyboard, a motivated attacker can compromise a computer system at a layer below or adjacent to the shallow forms of authentication that are now accepted as commonplace[3]. Therefore, rather than asking “Can we trust the person behind the keyboard”, a still better question might be: “Can we trust the computer system underneath?”. An emerging technology for gaining trust in a remote computing system is *remote attestation.* Remote attestation is the activity of making a claim about properties of a target by supplying evidence to an appraiser over a network[2]. Although many existing approaches to remote attestation wisely adopt a layered architecture–where the bottom layers measure layers above–the dependencies between components remain static and measurement orderings fixed. For modern computing environments with diverse topologies, we can no longer fix a target architecture any more than we can fix a protocol to measure that architecture.

Copland [1] is a domain-specific language and formal framework that provides a vocabulary for specifying the goals of layered attestation protocols. It also provides a reference semantics that characterizes system measurement events and evidence handling; a foundation for comparing protocol alternatives. The aim of this work is to refine the Copland semantics to a more fine-grained semantics that is closer to a concrete implementation of an *attestation manager*–a high-privilege thread of control responsible for invoking attestation services and bundling evidence results. This refinement consists of two cooperating components called the Copland Compiler and the Attestation Virtual Machine (AVM). The Copland Compiler translates a Copland protocol description into a sequence of primitive attestation instructions to be executed in the AVM. When considered in combination with advances in virtualization, trusted hardware, and high-assurance system software components–like compilers, file-systems, and OS kernels–a formally verified remote attestation infrastructure creates exciting opportunities for building system-level security arguments.

REFERENCES

[1] John D Ramsdell B, Paul D. Rowe, Perry Alexander, Sarah C Helble, Peter Loscocco, J Aaron Pendergrass, and Adam Petz. 2019. *Orchestrating Layered Attestations.* Vol. 1. Springer International Publishing. 197–221 pages. https://doi.org/10.1007/978-3-030-17138-4

[2] George Coker, Joshua Guttman, Peter Loscocco, Amy Herzog, Jonathan Millen, Brian O’Hanlon, John Ramsdell, Ariel Segall, Justin Sheehy, and Brian Sniffen. 2011. Principles of Remote Attestation. *International Journal of Information Security* 10, 2 (June 2011), 63–81.

[3] P D Rowe. 2016. Confining adversary actions via measurement. *Third International Workshop on Graphical Models for Security* (2016), 150–166.