ArmoredSoftware:
Establishing and Maintaining Trust in the Cloud

Perry Alexander, Andrew Gill, Prasad Kulkarni
Information and Telecommunication Technology Center, The University of Kansas

Edward Bishop, Ciro Pinto-Coelho
Southern Cross Engineering

Introduction

A system is *trustworthy* if it can be strongly identified; acts in an uninhibited fashion; and is directly or indirectly observed by a trusted third party to behave as expected.

ArmoredSoftware is a new architecture for establishing and maintaining trust in the cloud. Cloud software processes operate in an armor capsule that assesses the operational environment, protects the application, and assesses the application for the environment.

ArmoredSoftware ends the need for *a priori* trust in cloud environments. Armored components automatically assess and react to their operational environments based on user parameters. They use the information they gather to take actions ranging from simple reconfiguration and reporting to migrating among available execution platforms.

ArmoredSoftware provides evidence of its own trustworthiness. Armored components evaluate their applications to provide static and run-time evidence used for appraisal. Applications can establish mutual trust relationships.

Platforms

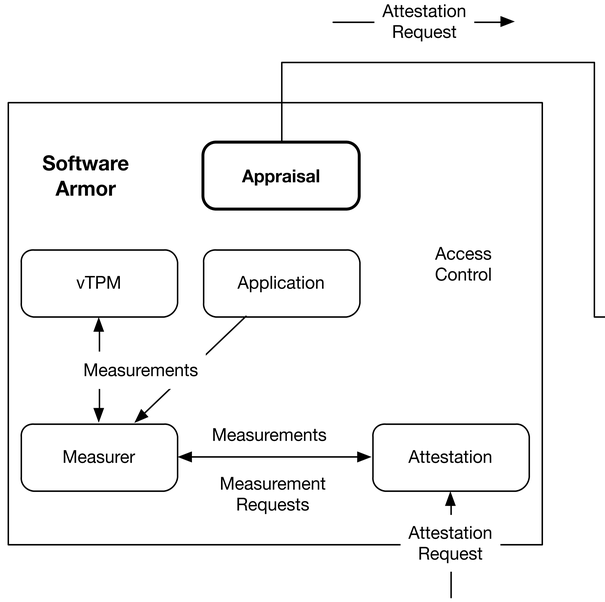
ArmoredSoftware currently targets the Xen virtualization environment running in an OpenStack cloud infrastructure. MiniOS VMs provide simple services while CirrOS and CentOS Linux VMs provide operational environments for VMs and applications. XSM will provide access control enforcement for armored components.

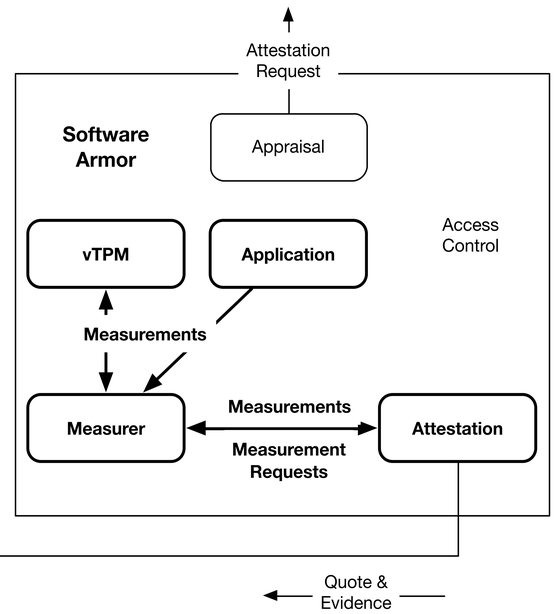
Armored implementation is in Haskell with Cloud Haskell providing an abstract mechanism for communicating among VMs.

ArmoredSoftware Component Architecture

Applications protected and managed by Armor

An *appraiser* requests an attestation from a target and assesses trustworthiness based on received quote and evidence.





A *target* responds to an attestation request by gathering evidence, generating a cryptographic quote, and returning quote and evidence.

Approach

The ArmoredSoftware approach is based on trusted computing techniques as described in *Principles of Remote Attestation* by Coker et. al. Using remote attestation, an appraiser requests information from a target that it can then use to assess its behavior and state. The appraiser makes a trustworthiness assessment based on that information

An *appraiser* requests a quote from a target system. The request specifies what information is desired, provides a nonce to ensure freshness, and identifies the source of the request as appropriate.

The *target* gathers evidence describing its state and behavior. The evidence includes both static measurements stored by the target and on-demand measurements taken when requested. A vTPM is used as a root-of-trust for storage.

The *target* generates a quote attesting to the integrity of gathered evidence. The quote includes cryptographic signatures for assessing authenticity, freshness, and provenance of evidence. A vTPM serves as a root-of trust for attestation.

The *appraiser* assesses the quote to ensure integrity of evidence. Given a good quote, it then assesses evidence to determine trustworthiness of the target. Assessment looks for evidence of correct boot, good parts, and good run-time behavior.

Access control enforced by the virtualization environment and by individual components enforces policy over ArmoredSoftware components and their associated applications.

Literature

George Coker, Joshua Guttman, Peter Loscocco, Amy Herzog, Jonathan Millen, John Ramsdell, Ariel Segall, Justin Sheehy, Brian Sniffen, “Principles of Remote Attestation” in *International Journal of Information Security*, 2012

Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pragg, Andrew Warfield, “Xen and the Art of Virtualization”, *Proceedings of the ACM Symposium on Operating Systems Principles (SOSP'03)*, Boldon Landing, NY, 2013

Andrew Martin, *The Ten-Page Introduction to Trusted Computing*, 2008, <http://www.cs.ox.ac.uk/files/1873/RR-08-11.PDF>

Acknowledgements

We gratefully acknowledge the support of the United States Department of Defense

Further Information

ArrmoredSoftware website:

http://armoredsoftware.github.io

All software is freely available. For further information, please contact Dr. Perry Alexander at palexand@ku.edu.