Remote Attestation for Cloud-Based Systems

Dr. Perry Alexander¹  Dr. Andrew Gill¹  Dr. Prasad Kulkarni¹
Adam Petz¹  Paul Kline¹  Justin Dawson¹
Jason Gevargizian¹  Mark Grebe¹  Edward Komp¹
Edward Bishop²

¹Information and Telecommunication Technology Center
Electrical Engineering and Computer Science
The University of Kansas

²Southern Cross Engineering

May 6, 2015
The promises of the cloud are substantial

- reduced hardware and software costs
- reduced resource consumption
- improved availability and reliability

The structure of the cloud complicates assurance

- not under the desk
- ambiguous and changing runtime environment
- unknown and unknowable actors in the same environment

Is trust possible in the cloud environment?

- unambiguous identification
- confirmation of uninhibited execution
- direct or trusted indirect observation of good behavior
Virtual Blinking Lights

Provide new capabilities that establish and maintain trustworthy cloud-based application deployment

- Establish trust in cloud applications
  - trust in cloud infrastructure
  - trust in user-space applications
  - trust in application cohorts
- Promote informed decision making
  - confirm data confidentiality
  - confirm execution and data integrity
- Autonomous run-time response and reconfiguration
  - respond to attack, failure, reconfiguration, and repair
  - appraisal informs response
Semantic Remote Attestation

- Appraiser requests a quote
  - specifies needed information
  - provides a nonce
- Target gathers evidence
  - measures application
  - gathers evidence of trust
- Target generates quote
  - measurements and evidence
  - original nonce
  - cryptographic signature
- Appraiser assesses quote
  - good application behavior
  - infrastructure trustworthiness
Trusted Platform Module

» Provides and Protects Roots of Trust
  » Storage Root Key (SRK) - root of trust for storage
  » Endorsement Key (EK) - root of trust for reporting

» Quote generation
  » high integrity quotes - (\{RS\}_{AIK−}, SML, \{n, PCRC_{Comp}\}_{AIK−})
  » high integrity evidence - (⟨E, n⟩, \{∥⟨E, n⟩∥, PCR\}_{AIK−})

» Sealing data to state
  » \{D, PCR\}_{K+} will not decrypt unless PCR = current PCR
  » data is safe even in the presence of malicious machine

» Binding data to TPMs and machines
  » \({K−}_{SRK+,K}\) - \{D\}_{K+} cannot be decrypted unless \(SRK−\) is installed
  » \({J−}_{K+,J}\) - \{D\}_{J+} cannot be decrypted unless \(K−\) and \(SRK−\) are installed
The Cloud Challenge
Chasing the bottom turtle

Platform
TPM

Platform
TPM
The Cloud Challenge
Chasing the bottom turtle

- Hypervisor
- Dom0 vTPM Manager
- Platform TPM
The Cloud Challenge
Chasing the bottom turtle

Platform
TPM

Dom0
vTPM
Manager

Hypervisor

Virtual Platform vTPM
The Cloud Challenge
Chasing the bottom turtle

Establishing Roots-of-Trust
The Cloud Challenge
Chasing the bottom turtle

Platform
TPM

Virtual Platform
vTPM

Dom0
vTPM
Manager

Application

Platform
TPM

Transfering
Roots-of-Trust?

Application

Hypervisor

Platform
TPM
Enabling Technologies

- Trustworthy protocol execution
  - executable and analyzable protocol representation
  - generates evidence of trustworthiness
  - negotiates attestation details
  - designed for highly focused appraisal

- Application specific measurement
  - managed and traditional execution environments
  - compile-time assistance for measurer synthesis
  - specialized measurement bundled with applications

- Lightweight trust infrastructure
  - abstract communications capability
  - migration support
  - strong identity
Armored Application Architecture

M&A targeted to an application

- Appraiser makes attestation requests
- Attester responds to attestation requests
- Measurer gathers evidence from application
- Influenced by the *Trusted Research Platform* and *Principles of Remote Attestation*
Privacy CA Attestation

App $\rightarrow$ d, $N_{App}$, $PCR_m$

Att

$\downarrow$

$make\_and\_load\_identity$

$\rightarrow$ TPM

$\leftarrow AIK_h$

$\downarrow$

$\rightarrow AIK^+$

$\leftarrow \{K, |AIK^+|\}_{EK^+}, \{||AIK^+||_{CA^-}\}_K$

$\downarrow$

$activate\_identity(AIK_h, \{K, |AIK^+|\}_{EK^+})$

$\downarrow$

$\leftarrow K$

$\downarrow$

$d \rightarrow Meas$

$\downarrow$

$e$

$\downarrow$

$quote( AIK_h, PCR_m, |(e, N_{App}, \{||AIK^+||_{CA^-}\}) | )$

$\rightarrow PCR_c, \{||PCR_c|, |(e, N_{App}, \{||AIK^+||_{CA^-}\})|\}_AIK^-$

$\downarrow$

$\leftarrow e, N_{App}, PCR_c, \{||AIK^+||_{CA^-}\}_AIK^-$
First-class structure for protocols
- encapsulates a protocol-centered computation
- semantics provide a basis for static analysis
- based loosely on the Reader monad

Abstract communication primitives
- extended RPC-style capability
- requests remote execution
- defines send and receive operations
- abstracts away communication details

\[
do \{ \\
    f(x); \\
y \leftarrow f(x); \\
send a x; \\
y \leftarrow receive a \\
\}
\]
Negotiating a Protocol
Respecting privacy

- Typical negotiation
  - request sent to Attester
  - Attester generates proposal
  - Appraiser selects protocol
  - Attester executes protocol
- Three kinds of requests
  - execute protocol 22
  - provide \{OS\_config, http\_stat, firewall\_stat\}
  - execute protocol do \{ ... \}
- Three negotiation criteria
  - ability to satisfy the request
  - satisfaction of appraiser and attester privacy policies
  - previously obtained evidence
Negotiation Protocol
Request and Select

- Requests an attestation
- Receives proposals
- Selects from proposals

```
do { send t r;
    q <- receive t;
    e <- case {p:q | (policy? p)} of
          ∅ : None
          p : send t (choose p)
    end;
    case e of
          Some v : (appraise v)
          None : None
    end }```

Negotiation is a protocol that can itself be selected or negotiated.
Negotiation Results

- Evidence and Protocol pairs
- Satisfies privacy policy of attester
- Provide some or all of requested information

```plaintext
(((ID,SIGHASH,SIGSRC),
  do { id <- getVCID;
      sig <- getSigFileEvidence;
      src <- getSigFileSrc;
      e <- createEvidence(id,sig,src);
      returnEvidence(e) })
```
Generated negotiation protocol code (currently by hand):

P = CreateChannel (AChannel "attesterChan") Target
  $ Send ANRequest (AChannel "attesterChan")
  $ Receive (Var "counterOffer") (AChannel "attesterChan")
  $ CalculateFinalRequest (Var "finalReq")
      ANRequest
      (Var "counterOffer")
  $ Send (Var "finalReq") (AChannel "attesterChan")
  $ Receive (Var "finalConfirmation")
      (AChannel "attesterChan")
  $ Case (Var "finalConfirmation") [(Var "finalReq")]
      (HandleFinalChoice (Var "result") (Var "finalReq")
       (Result (Var "result")))
      (Stuck "finalConf and finalReq match error")
Performing Measurement and Attestation

Selection  Instantiation  Execution  Measurement

Request \(<d,n>\)

Attestation Protocol
Attestation Protocol Instance
Measurement Instances

Appraiser
vTPM
Application
Environment
Targets

\(<q,n>_{k-1}\) Quote
Single Realm Attestation

Protocol for gathering virus checker evidence

```
do {
    id <- getVCID;
    sig <- getSigFileEvidence;
    src <- getSigFileSrc;
    e <- createEvidence(id, sig, src);
    returnEvidence(e)
}
```

and generates evidence of the form:

\[\langle (id, sig, src), \{\| (id, sig, src)\|, PCRComp_0 \})_{AIK_0^-}\rangle\]

Appraisal replays the protocol up to crypto operations with known good measurements
Multi-Realm Attestation

Nested attestation requests evidence from the signature server directly:

\[
\text{do } \{ \text{id} \leftarrow \text{getVCID}; \\
\quad \text{sig} \leftarrow \text{getSigFileEvidence}; \\
\quad \text{src} \leftarrow \text{getSigFileSrc}; \\
\quad \text{srcEvidence} \leftarrow \text{send src r}; \\
\quad \text{e} \leftarrow \text{createEvidence(id, sig, src, srcEvidence)} \\
\quad \text{returnEvidence(e)} \}
\]

and generates bundled evidence:

\[
\text{let } b = \langle (e), \{ ||e|, \text{PCRComp}_{1} \}^{A_{IK_1}} \rangle \text{ in } \\
\langle (id, sig, src, b), \{ ||(id, sig, src, b)|, \text{PCRComp}_{0} \}^{A_{IK_0}} \rangle
\]
Trusting Evidence
Why bundling is hard

- Trusting evidence
  - hashes and TPM quotes
  - measure and appraise the attestation infrastructure
  - gather evidence of good protocol execution

- Trusting bundled evidence
  - appraisers do not know the source of evidence \textit{a priori}
  - no global name space for evidence sources
  - bundled appraisals vs bundled evidence

- Trusting the appraiser
  - negotiated protocols must satisfy privacy policies
  - trust may not be transitive for applications and infrastructure
  - global policy is not an answer
Current Status
Demos available

- Attestation and Appraisal development
  - CA-Based attestation protocol execution example
  - simple dynamic appraisal of attestation results
  - integrated negotiation protocol and attestation protocols

- Measurement development
  - HotSpot-based Java VM run time measurements
  - detect and report several runtime anomalies
  - standard mechanism for extending measurement capabilities

- Infrastructure development
  - vchan, TCP/IP and socket communication infrastructure
  - initial certificate authority implementation
  - language-based interface with TPM 1.2
  - integrated Berlios TPM emulator
  - JSON-based data exchange formats
Ongoing Work
Goals for 2015

- Establish roots-of-trust and trust argument
  - measured launch and remeasurement of ArmoredSoftware
  - establish trust in the Xen/OpenStack infrastructure
- Executable protocol representation and protocol semantics
  - evidence of proper execution
  - static trust analysis
  - protocol-centered appraisal
- More capable measurement
  - compiler directed measurement
  - continuous measurement—tripping and trending
- Publicly available libraries and infrastructure


