Leviathan redux

John L. Manferdelli
Intel Science and Technology Center for Secure Computing
UC, Berkeley

Joint work with Tom Roeder (Google), Fred Schneider (Cornell)
And Kevin Walsh of Mt Holyoke College

Hobbes

- “Every man is Enemy to every man; wherein men live without other security ... In such condition, there is no place for Industry; because the fruit thereof is uncertain [and people live in] continuall feare (his spelling, not mine), and danger of violent death; And the life of man, solitary, poore, nasty, brutish, and short.”

Are you better off than you were in 2001? (1647?)

- Yes:
  - PC infections: 1% down from >2%
  - Mobile data traffic grew 81% in 2013 with 4G devices accounting for 30% of total mobile traffic [Cisco 2013]. Reported infection rate of 0.0009% [NDSS, 2013].
  - New study
    - iPhone: infection rate .22%
    - Android phone: infection rate .39%
- But:
  - This bug shows that in 2014, your machine can still be owned by a picture

New stuff

- Situational awareness (big data)
  - Adversarial machine learning
  - Text mining
  - Suspicious URLs
- CFI/XFI
- Formal methods
- Artificial diversity
- NX, ASLR
- *Sandboxing*
- *Hardware roots*

Also new

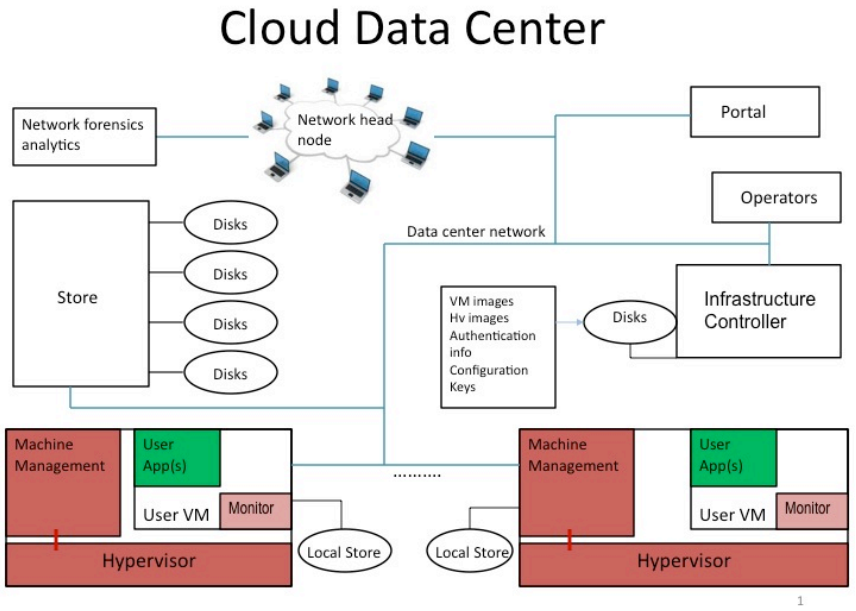
- Nation state attacks on commercial computers
  - Not *so* new
- Carders, Silk Road
- Social computing
- Wearables
  - Pictures everywhere
  - Control model?
- Embedded systems
  - Also not so new (space shuttle)
  - But ubiquitous … and connected
- Hardware attacks
  - Especially “core” parts

Were the old solutions inadequate?

- *Isolation*
- *Least privilege*
- Verification/certification

CloudProxy: protection model

- Adversaries: Co-tenants, insiders (but not “inside” developer), eavesdroppers, technicians
- Protect: Key disclosure, integrity violation, data (maybe code)
- Ensure: Correct operation, configuration (affecting confidentiality and integrity)
- Avoid: Large software services written by others, CAs



Building a secure distributed application

Currently

- Write the programs implementing the application correctly
- Deploy the program safely (no changes)
- Configure the operating environment correctly
- Ensure other programs can't (or don't) interfere with safe program execution
- Generate and deploy keys safely
- Protect keys during use and storage
- Ensure data is not visible to adversaries and cannot be changed in transmission or storage
- Add new program elements during operation
- Ensure trust infrastructure is reliable
- Manually audit to provide confidence this all happened during operations

With CloudProxy

- Write the programs implementing the application correctly

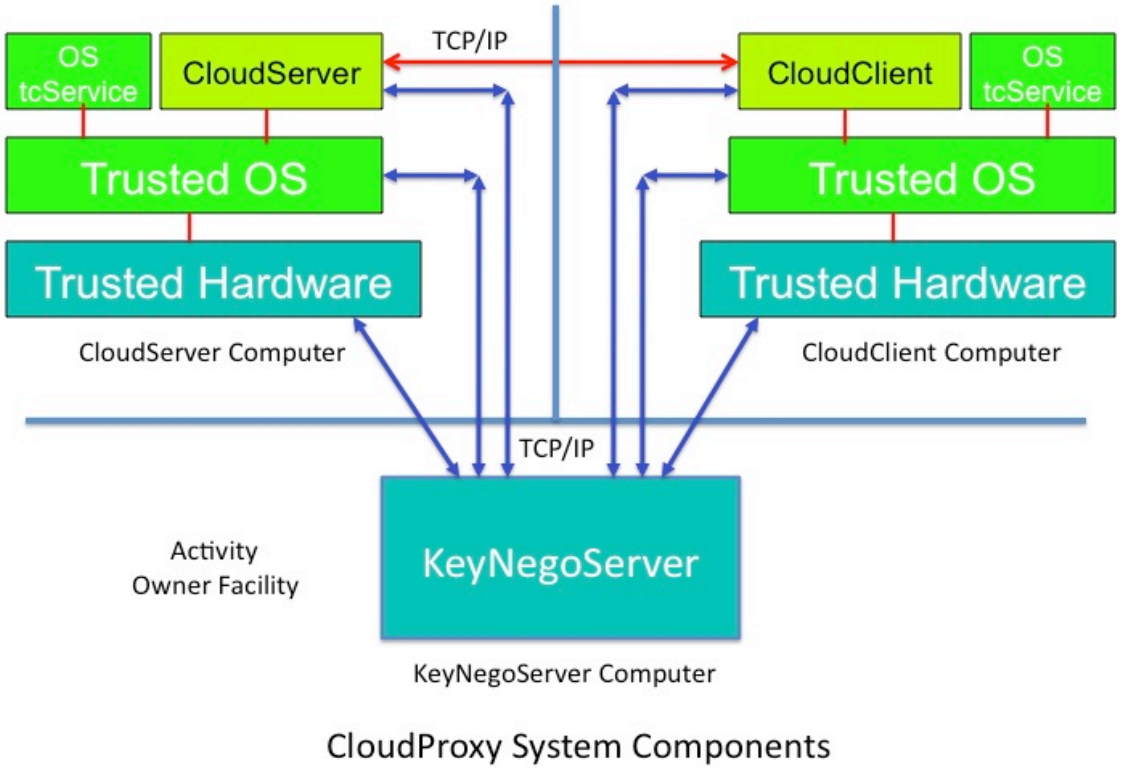
- Properties
  - Fail safe *remotely* verifiable operation
  - Simple programming model
    - Support multiple layers of familiar software stack (Application, OS, Hypervisor, Hardware)

The “secret sauce”

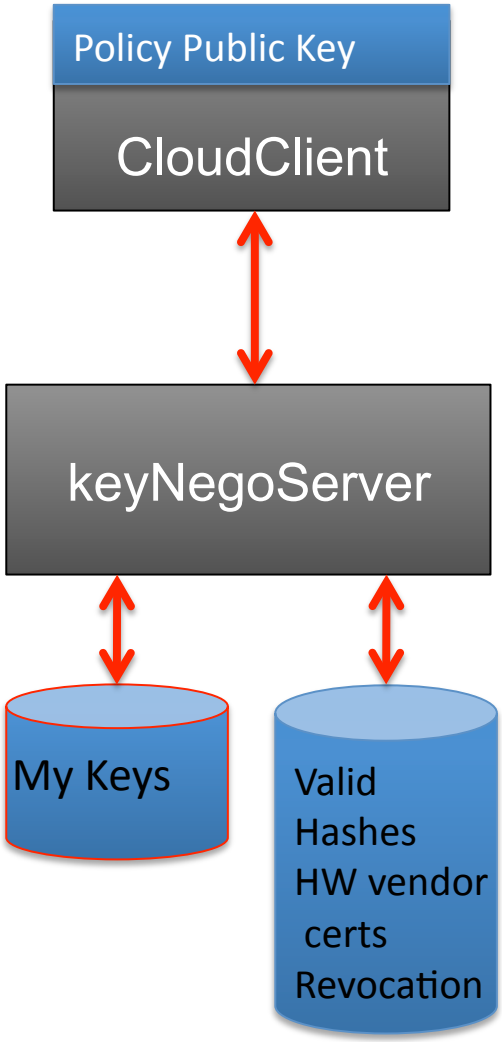
Host system provides:

- Isolation for measurement-based principals
  - Hosted programs
- Services for measurement-based principals (the Tao)
  - Restricted use of cryptographic keys to encrypt and decrypt secrets for a measurement-based principal
  - Key management for the principals
  - Policy enforcement anchor (authentication and authorization)
  - Tao services same at all levels of the stack (hypervisor, OS, app)
  - *Base layer rooted in hardware (CPU, chipset, TPM)*

Initialization

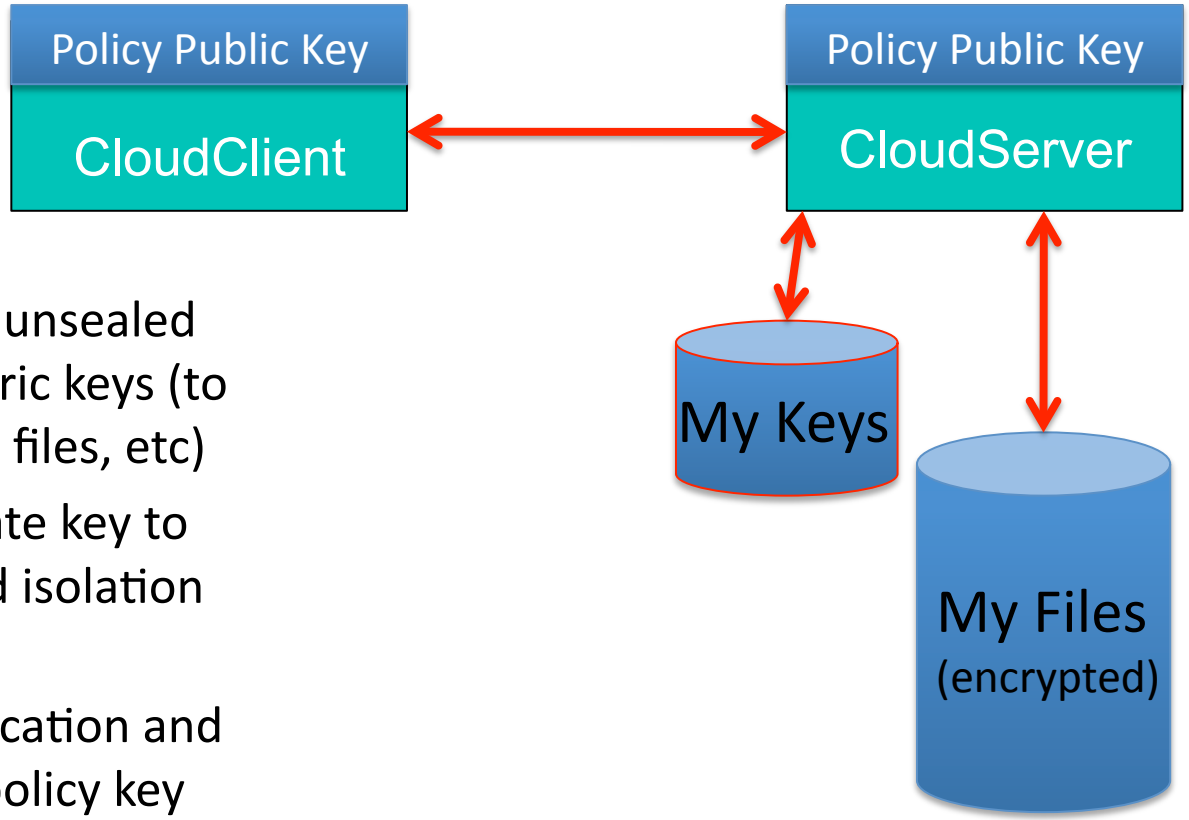


CloudProxy System Components



- CloudClient has private key that only it can access, while isolated and Policy principal signed certificate for the public key.
- Policy principal signed certificate can be used to
  1. Establish SSL channel
  2. Authenticate program and isolation regime

Operation

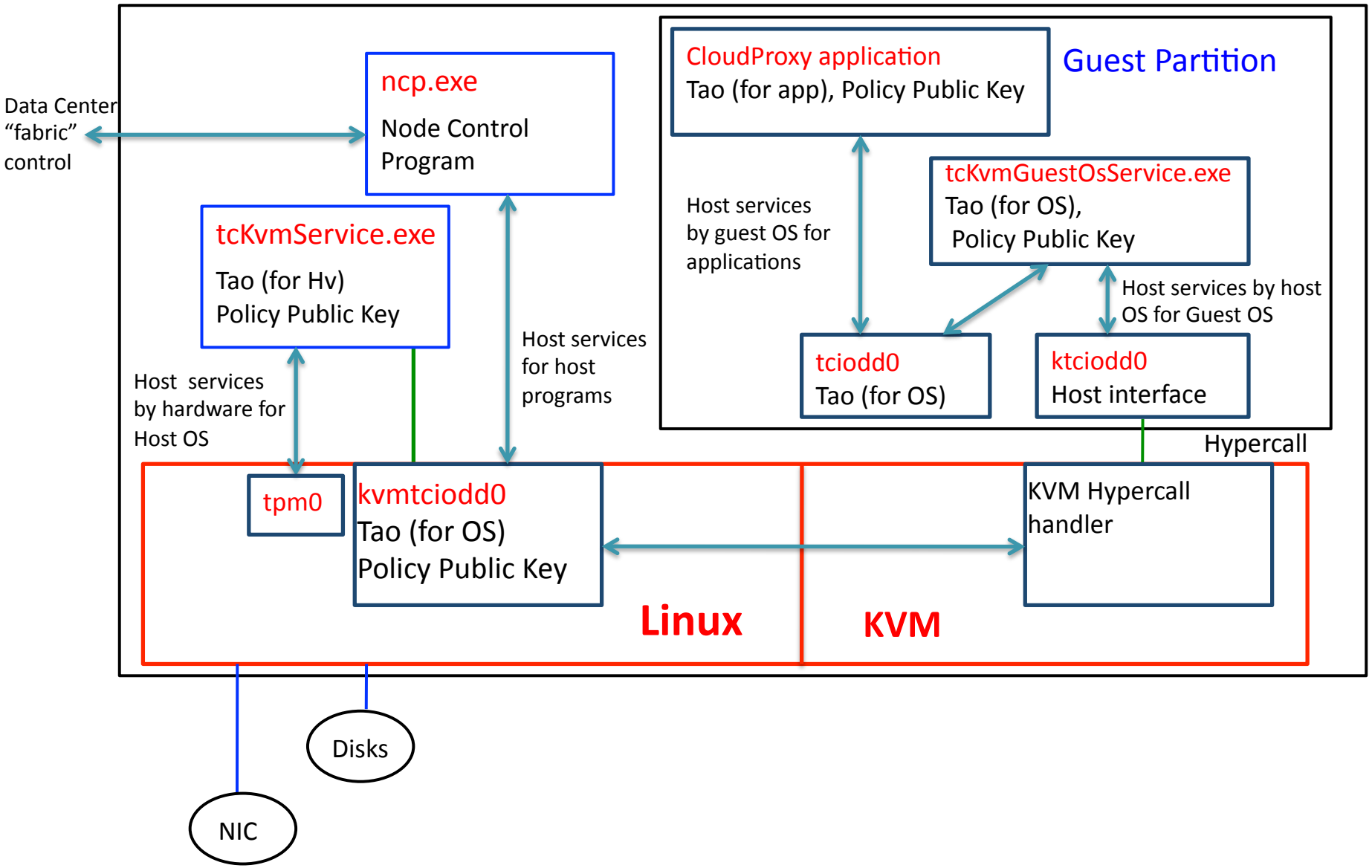


- Programs have access to unsealed private keys and symmetric keys (to encrypt/integrity protect files, etc)
- Program can use its private key to authenticate identity and isolation regime
- Program can do authentication and authorization rooted in policy key
- Trusted programs have authenticated, encrypted integrity protected channel to communicate.

Properties

- All trust rooted in "Policy Key"
- All private keys sealed to code identity
- Policy key is part of identity

KVM Hypervisor (old)



You get what you deserve

- You get what you deserve instead of what you're dealt.
- Maybe it's not what you want but it's what you need.