Secure Virtual Platform Research

OpenXT Summit

June 7, 2016

Peter A. Loscocco

Trusted Systems Research

National Security Agency

Trusted Systems Research

- Conduct/sponsor research into:
  - Providing information assurance for national security systems
  - Enabling safe operation in risky or compromised environments
  - Advancing cryptographic algorithms and protocols, system analysis and design methods, trust mechanisms, and understanding system behavior
- Creators of SE Linux, Xen Security Modules, Linux Kernel Integrity Monitor, and SE for Android

Our Motivation

- Desire to improve upon solid security foundation provided by SELinux
  - Minimize impact of kernel vulnerabilities
  - Minimize exposure of secrets to unauthorized components
  - Prevent unauthorized OS/configuration
  - Improve assurance of security solutions
- Desire to build assurable security solutions using commodity technology
- Support wider variety of customer environments

Secure Virtual Platform (SVP)

- Body of research conducted/sponsored by NSA to investigate how to secure systems more effectively
  - Research dating back to 2002
  - Demonstration of concepts through series of prototypes
- Explored the use of emerging hardware support for virtualization and trusted computing to
  - Address some of the limitations of SELinux systems
  - Create security architectures targeting assurance using commercial-grade components
- Targeted influence/advances in key areas rather than building complete SVP system

Specific Goals for SVP

- Use virtualization for security as well as functionality
- Advance Trusted Computing concepts to better meet security goals
- Create general-purpose architecture able to be applied to a variety of use cases
  - Focus on sound security architecture and proper distribution of security functionality
  - Achieve flexibility through configuration
- Enable evolution towards higher assurance
  - Investigate using security architecture to achieve greater assurance result using lesser assured components
  - Facilitate assurability by limiting assurance burden about a given component to arguments about functionality of that component
  - Overcome barriers of traditional high assurance solutions

SVP Strategy - Architecture

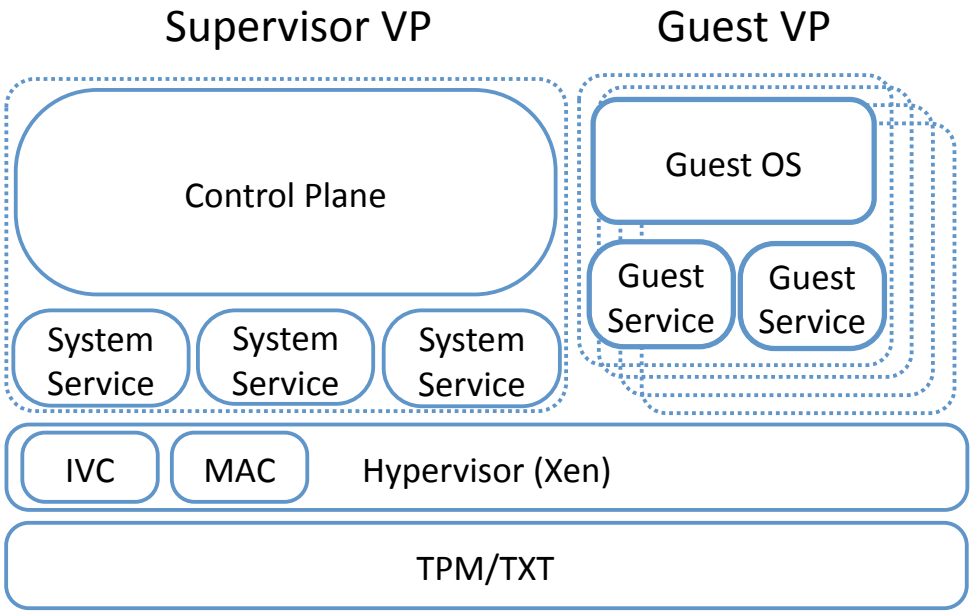
- Advance security architecture for secure virtualization system
  - Adhere to identified security principles and desired properties when assigning functionality to components
  - Analyze component interdependencies, ensuring all assumptions about self, providers and clients are sound
- Provide robustness in presence of inevitable flaws
  - Understand the attack surface of each component and the potential consequence of its complete compromise
  - Design to contain damage and/or react appropriately

SVP Strategy - Assurance

- Incorporate robust detection mechanisms to warn of component compromises, including those to the detection mechanisms
  - Minimize likelihood of effective exploitation
  - Prevent execution of sensitive operations from bad states
  - Enable appropriate response to compromises
- Apply assurance activities where most effective
  - Assurance activities focused on proving properties of architecture and its components
  - Minimize assurance burden on any given component by constructing arguments that leverage system assurance argument
  - High assurance implementations where feasible and necessary

A High-level View of the System

- Secure Hypervisor
  - Provides solid foundation leveraged to meet many system security objectives
  - SVP focused on Xen
- Virtual Platforms (VP)
  - Logical groups of Virtual Machines (VM) supporting mission or function
  - Isolates groups of components from each other
  - Supervisor VP for control plane and system-level services
  - Guest VPs for all guest-specific services



Secure Hypervisor Desired Properties

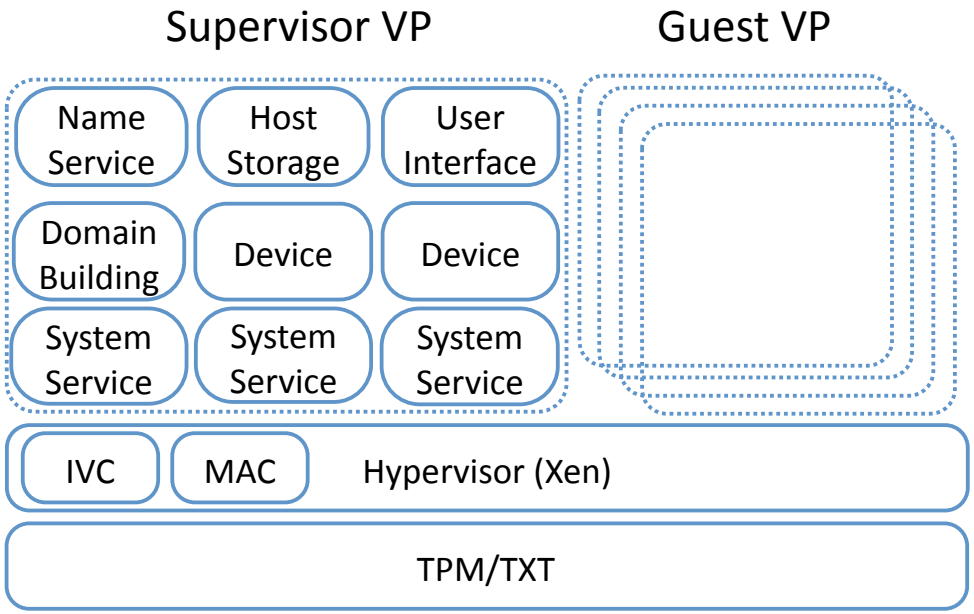
- Use of HW virtualization for VM isolation
- Supports DRTM to reach initial known state
- Supports IOMMU for explicit memory protections
- MAC to control VM resources and interactions
- Secure Inter-VM Communication (IVC) to support MAC over VM services
- Minimal size and function to facilitate high-assurance implementation
- Supports lightweight VMs to facilitate high-assurance VM implementations
- VM grouping to support VPs function and security

Access Control in SVP

- Discrete components enforce MAC focused at proper abstraction level
  - Components leverage MAC from others
  - Union embodies system MAC policy and helps guarantee important security properties
- XSM/Flask for VM isolation and controlled interaction
- VMs enforce MAC over objects/services they provide
  - Leverage secure IVC
- SELinux MAC for intra-VM protections

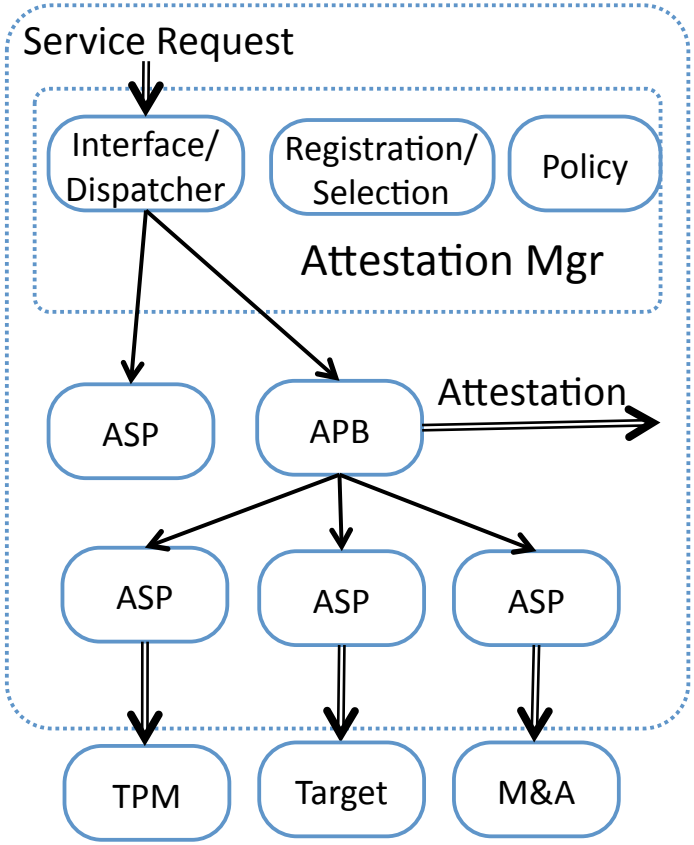
Disaggregation of Control Plane

- Isolates privileged functions in VMs
  - Targeting least privilege
  - Facilitating assurance
  - Examples: Host Storage, Name Service, Domain Building, User Interface
- Grouped in Supervisor VP
- Minimizes need for Dom0
- Feasibility demonstrated by Xoar



Measurement and Attestation (M&A)

- Detection strategy centered on M&A for total system
  - OS, User space, and Hypervisor components
    - At load and run times
    - LKIM developed to prove feasibility
  - Chained together for multi-realm attestations
    - Measures all components including M&A
- General Framework (Maat)
  - Multiple measurement agents, appraisers, and protocols
  - Local selection policy
  - Negotiated mechanisms and protocols
  - Enforces MAC internal policy



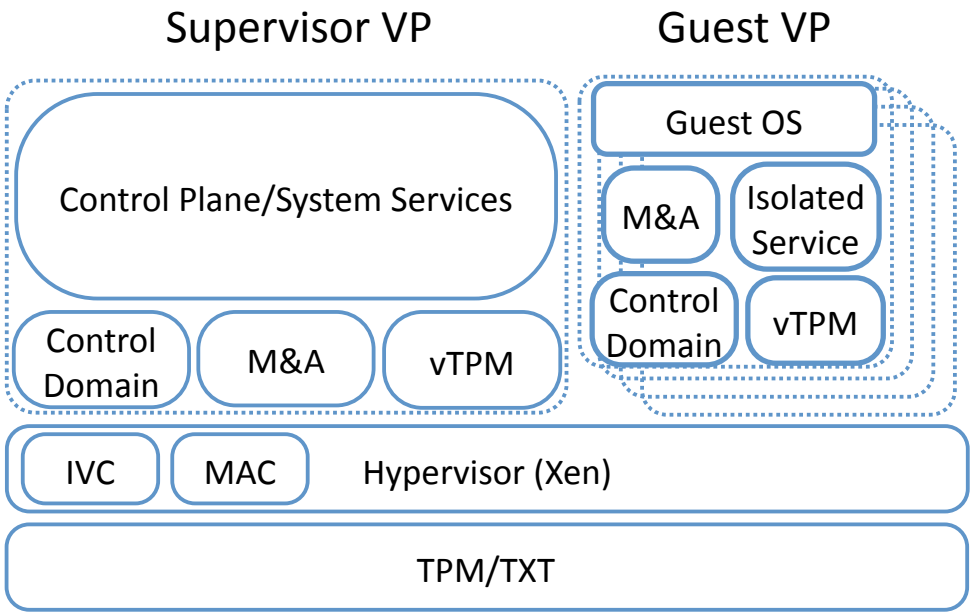
Maat M&A Service

Virtual TPMs (vTPM)

- Support TPM functionality in any VM while reserving dedicated TPM use for virtualization system
  - In SVP, vTPM manager and one vTPM per VP
  - Support M&A architecture at all layers
- Make vTPMs suitable for storage of long-term secrets
  - SVP influence in TCG
- Some specific vTPM security properties
  - Only approved images can function as vTPM
  - vTPM secrets only released to authorized vTPM
  - only vTPM can possess both secrets and keys to use
- vTPM manager domain for TPM access and vTPM data protection
- vTPM provides MAC over services and resources
  - Can support locality enforcement for multiple clients
- Technology like Intel SGX to address run-time confidentiality
- Xen vTPM good start to realizing these goals

Virtual Platforms

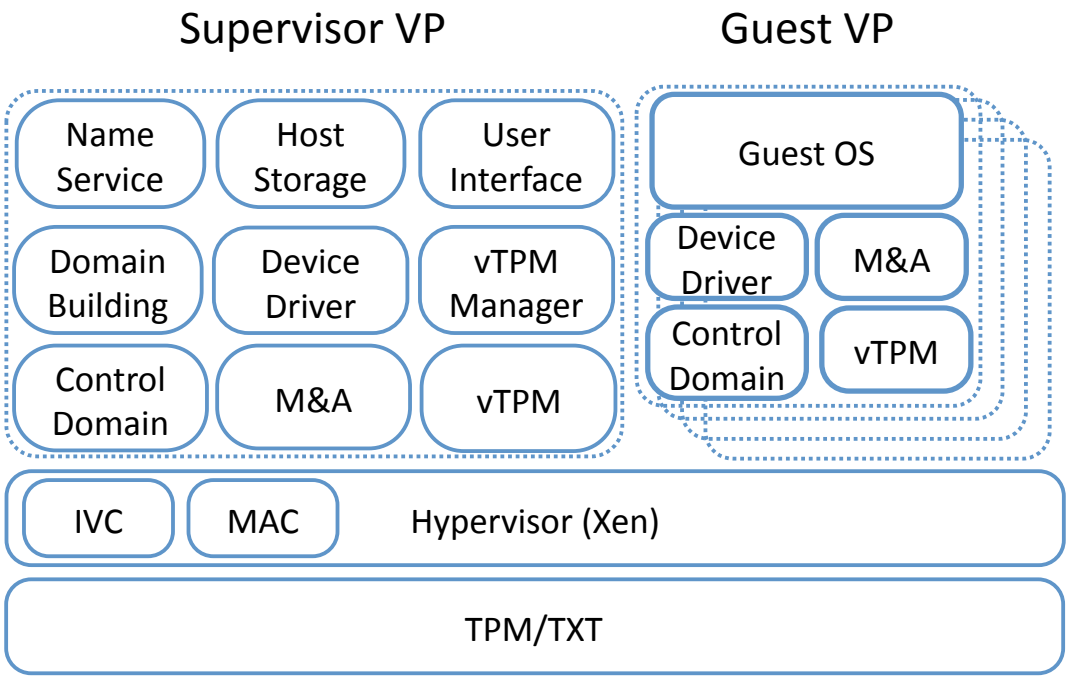
- Common structure for each Virtual Platform
  - Control Domain for VP booting, error handling, shutdown, and migration
  - M&A and vTPM
- Guest domain for main VP function
- Helper domains provide mutual protection for services and guest
  - E.g. Isolated devices, security services, or mission functionality



Using Trusted Computing Concepts

- Leveraging DRTM and (v)TPMs to support
  - Measurement & Attestation
  - Protection and controlled use of secrets
- Secure boot of platform
  - Each component measured at boot prior to use
    - Boot configuration constrained by owner
  - Carefully prescribed boot order enforced
    - Hypervisor, Supervisor and Guest VP components
    - Secrets and functions unlocked as component boot succeeds
      - i.e. Successful boot of vTPM Manager required for vTPM boot
  - Reliance on 3rd party attestations as use case requires
    - Supported by M&A infrastructure

A possible SVP Instance



Increasing Assurance

- Invest in high assurance implementations where needed most
  - i.e. Hypervisor or vTPM
  - Enables evolution
  - Lightweight VMs facilitate
- Assurance through sound architecture
  - Careful analysis
  - Formal modeling/proofs about properties
    - i.e. Boot process, M&A, component interactions

A Vision for OpenXT

- Already shares many SVP goals and characteristics
  - Roots in XenClient XT, an early commercial adopter of SVP concepts
- Would benefit from adopting more of the SVP Vision
  - Goal of highly configurable system
    - Adapt easily to wide variety of use cases
    - Bring common security architecture across all
  - OpenXT Core that evolves to include SVP ideas
    - Careful adherence to achieving identified component properties
    - Adopting/maturing key concepts like full system M&A and secure boot
    - Embrace assurance strategy focusing on sound security architecture
- Should become focal point for advancing system security
  - Enhanced security with virtualization and trusted computing
  - Hold OpenXT as an example of making security work
  - Encourage participation and development by those outside community to advance technologies as well as OpenXT

SVP Research Partners

- The MITRE Corporation
- Johns Hopkins University Applied Physics Lab
- Galois Inc.
  - University of Kansas
- University of British Columbia

Questions?

Peter A. Loscocco

loscocco@tycho.nsa.gov

NSA Trusted Systems Research