IBM T.J. Watson Research Center

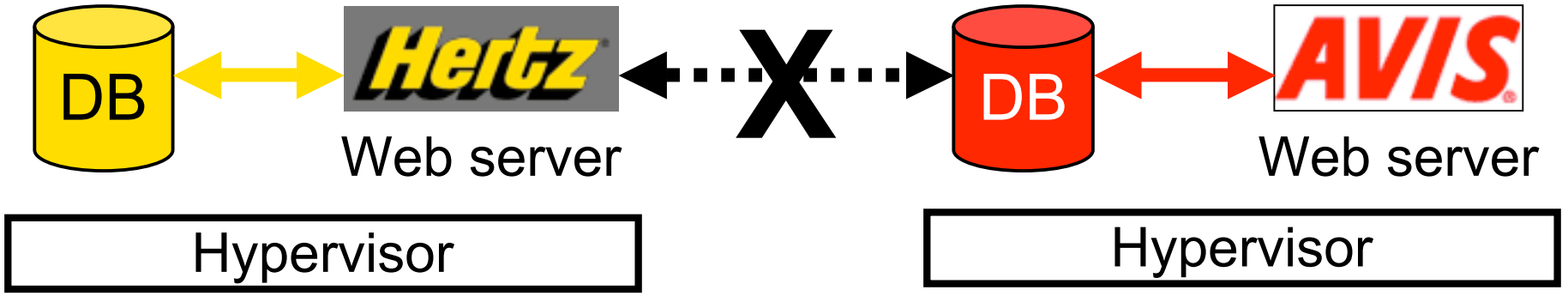
# Shamon: A System for Distributed Mandatory Access Control

Jonathan McCune

Joint work with Stefan Berger, Ramón Cáceres, Trent Jaeger, and Reiner Sailer

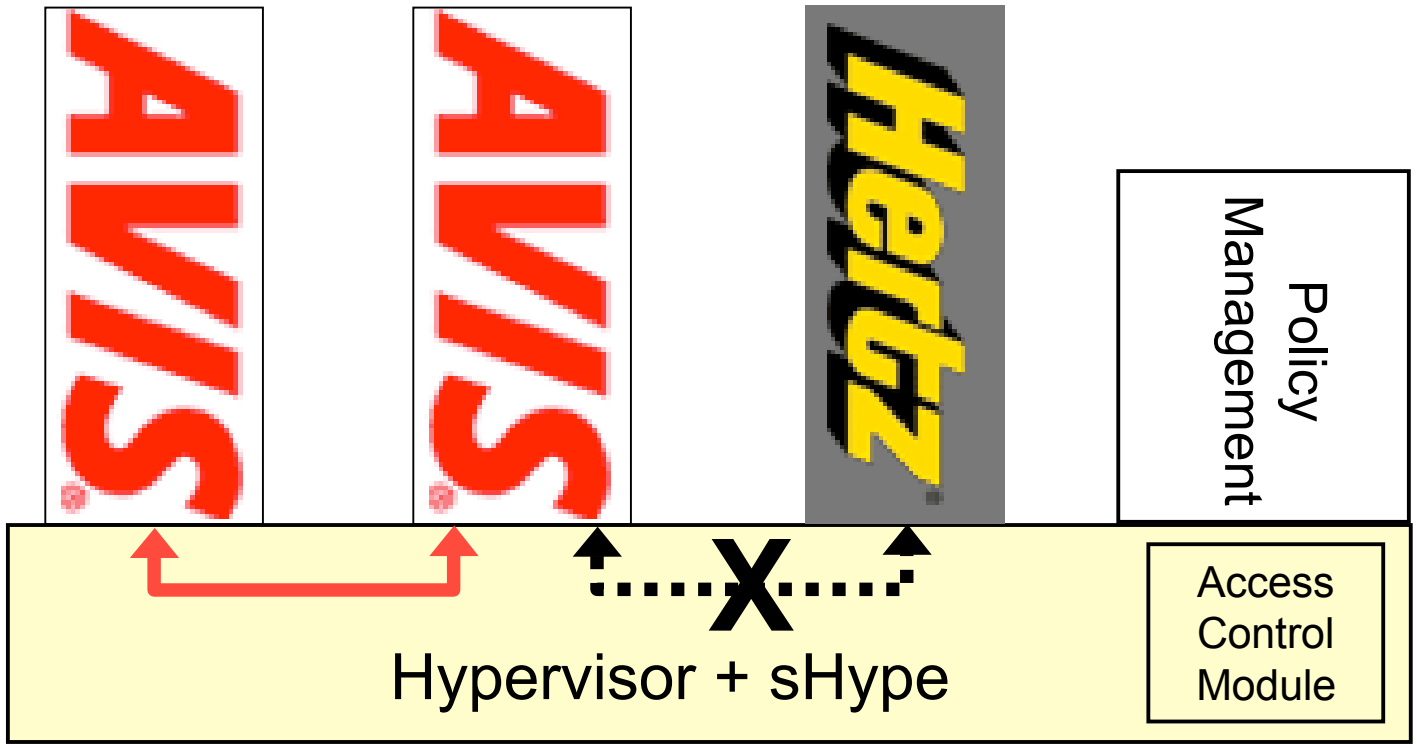
# Security issues with distributed computing

- **Mutually distrustful data center customers**
  - Need isolation guarantees to share machines
- **Data centers want to share physical machines**
  - Virtualization to move workloads between machines
  - Fault tolerance, load balancing, power saving, A/C costs



# Starting point: Secure hypervisor (sHype)




sHype prevents virtual machines from communicating or sharing resources subject to policy, e.g., unless they share a type (color)

# Introduction

- **Goal: achieve local-hypervisor security properties for distributed applications**
- **Strategy: enforce Mandatory Access Control (MAC) across a distributed set of machines**
- **Implementation: *Shamon* (Shared Reference Monitor)**
  - Achieves modification-detection, mediation, and isolation of distributed software

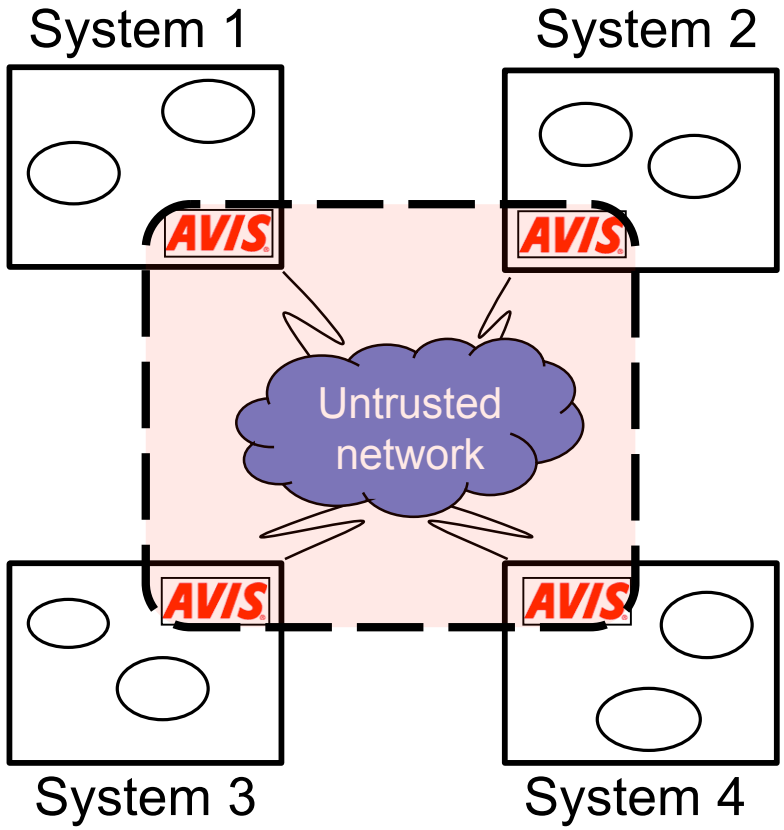
# Talk outline

- **Coalitions**
- **Shared reference monitor architecture**
- **Prototype implementation**
- **Next steps**
- **Related work and conclusions**

# Coalition

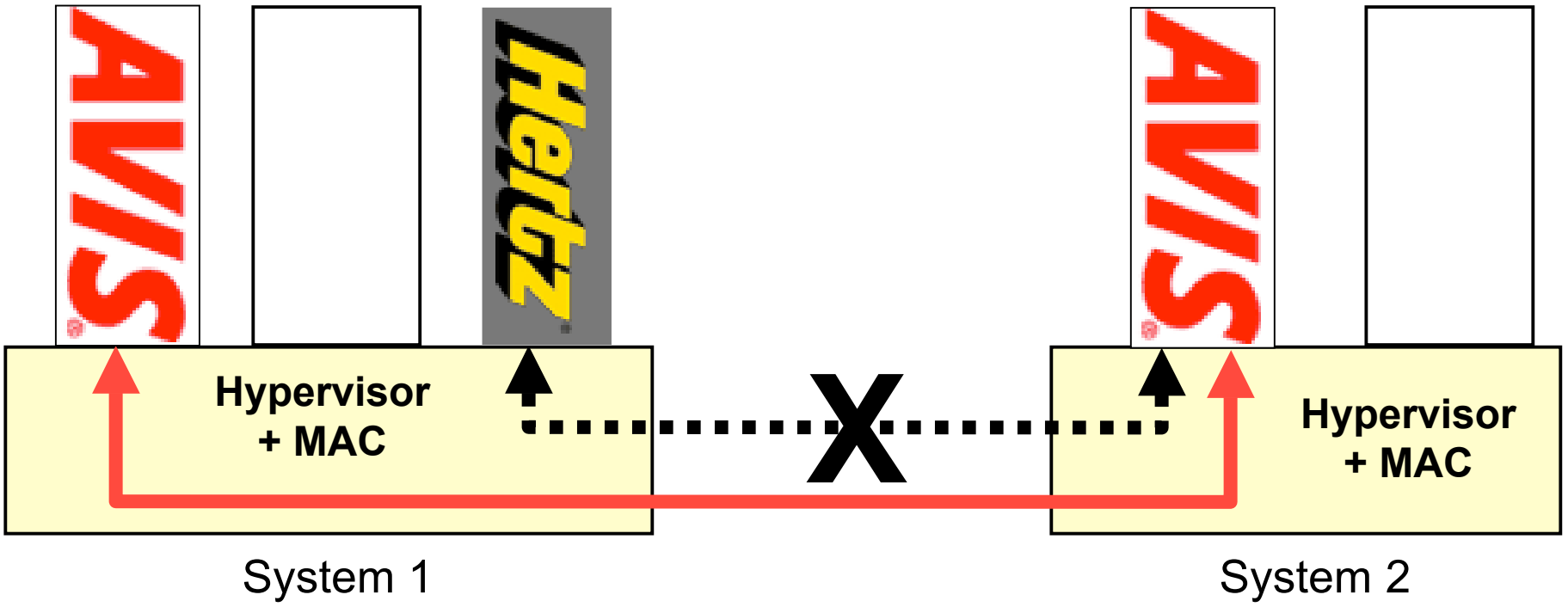
- **Coalition properties**
  - Compatible security policies
  - Isolated application workloads
  - Attested enforcement capabilities
  - Secure communication
- **Promises to reduce security-related complexity of applications**
- **Higher layers may add their own policy**



# To build a coalition: Extend MAC across machines

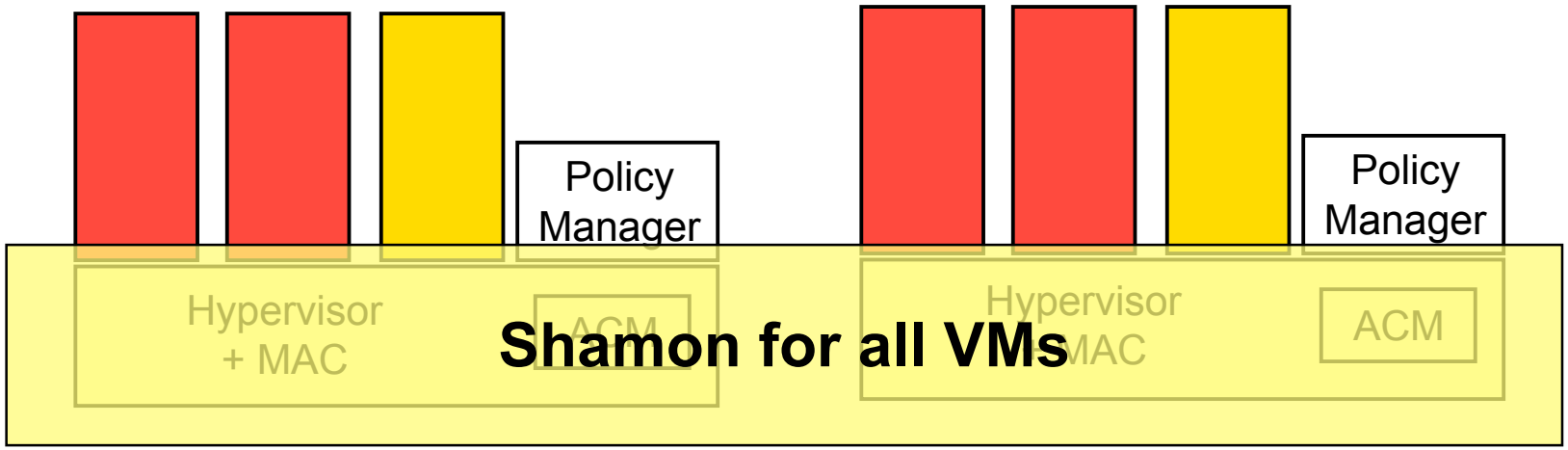
- ***Compatible*** **VMs are able to communicate**
- **Incompatible VMs cannot communicate**



# Shamon concept

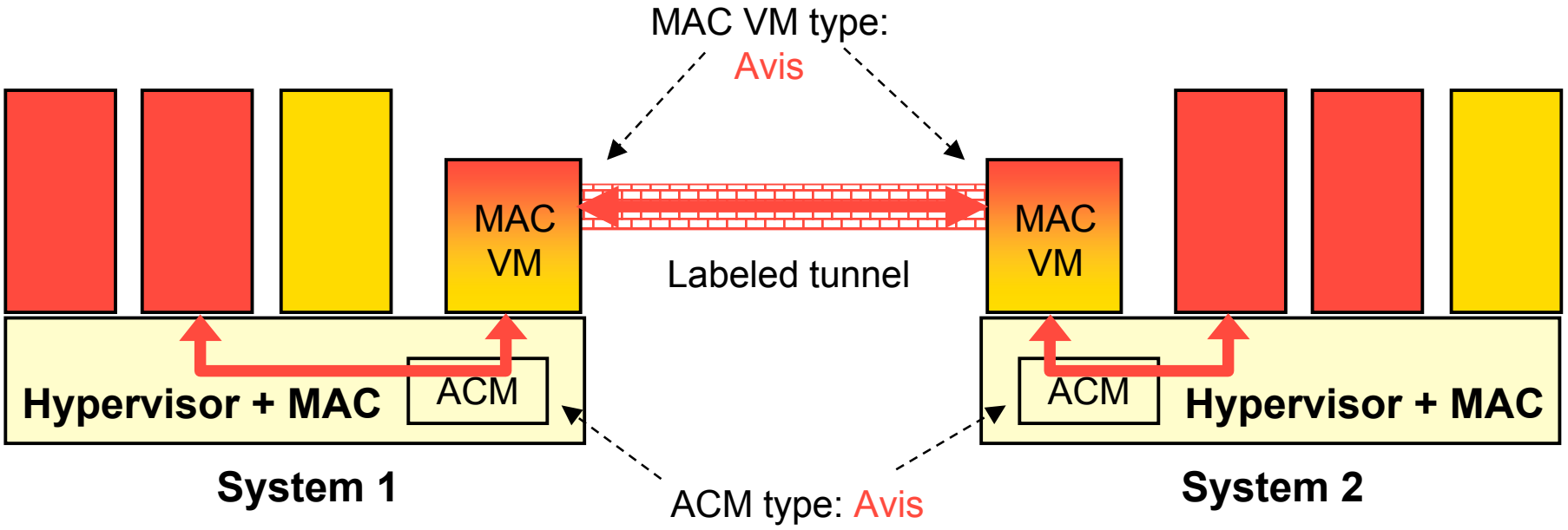
- **Reference monitors behave as a single reference monitor across all machines**
- **Individual virtual machines verify that peers are enforcing the desired MAC policy**



# Bridging MAC between systems

- **Trusted VMs with MAC responsibilities manage access to network**
- **MAC VMs map between ACM- and OS-level types**
- **Labeled tunnels connect MAC VMs and communicate types**
- **MAC VMs relay traffic between App VMs on different machines**



# MAC VM responsibilities

- **Translate between hypervisor and OS type labels**
  - Hypervisor performs local enforcement only
  - MAC VM must understand network types
- **Perform mutual attestation**
  - Bootloader, hypervisor, MAC VM image
  - Hypervisor MAC policy
  - MAC VM policy
  - Network security policy
- **Attestation is TPM-based, load-time attestation**
  - Details are in the paper

# Building blocks for Shamon prototype

- **Hypervisor Security Architecture (sHype)**

  Isolates virtual machines on a single system using Mandatory Access Control (MAC)

  Ability to quarantine, shutdown, or replace misbehaving VM

- **Labeled IPsec (Internet Protocol Security) for SELinux (Security Enhanced Linux)**

  Establishes authenticated and encrypted communication channels subject to MAC policy on end systems

  Ability to teardown tunnel to exclude misbehaving machines

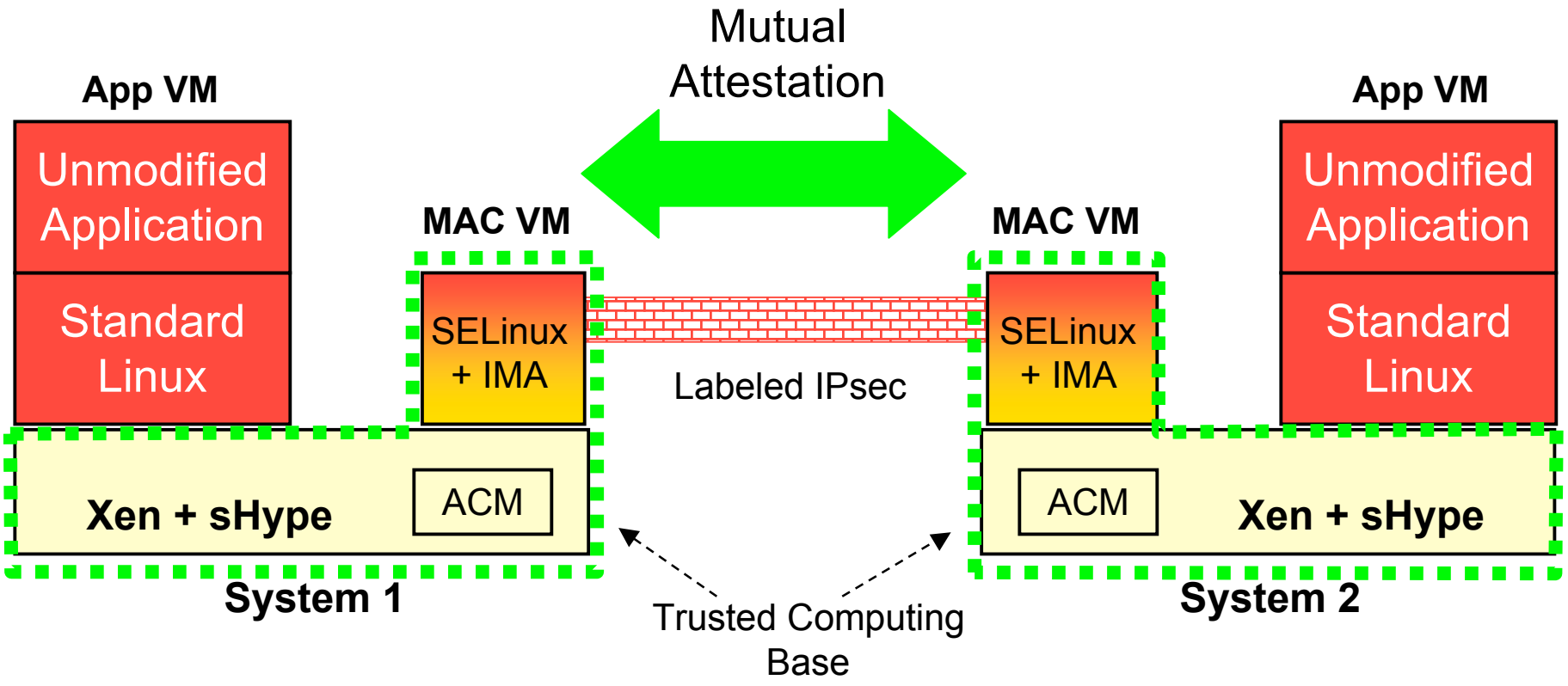
- **Integrity Measurement Architecture (IMA)**

  Load-time, TPM-based chain of measurements

  Detects misbehaving VMs or incompatible policies

# Shamon prototype implementation



# Status

- **MAC is working across systems**
  - Isolates VMs and mediates access to resources, e.g., network
  - Labeled IPsec tunnels are automatically created subject to policy
- **Integrity attestation is working across systems**
  - Periodically attests security properties to remote systems
  - Detects when incorrect software is loaded
  - Shuts down communication when trouble is detected
- **Can detect, confine and replace misbehaving VMs**
  - Can quarantine a VM based on attestation results
  - Can replace VM with a clean/patched VM image

# Next steps

- **Refine ability to determine policy compatibility**
  - Presently compatible means equal
- **Automate mechanisms for making systems compatible**
  - Adding new types
- **Need to establish common policy semantics across systems**
  - Describe / define universal type semantics

# Related work

- **OS-based MAC**
  - SELinux, TrustedBSD, TrustedSolaris
- **Virtualization-based security**
  - Terra, NetTop, …
- **Distributed system security**
  - Taos, Kerberos, trust management, grid computing
- **Trusted Virtual Domains (TVDs)**
- **Trusted Computing (TCG / OpenTC)**

# Conclusions

- **Shamon enables creation of coalitions with MAC across networked machines**
- **MAC VMs bridge individual reference monitors into a Shamon**
- **Attestation conveys modification-detection, mediation, and isolation properties**

- **jonmccune@cmu.edu**

# Thank you!

- Questions?
- jonmccune@cmu.edu
- **For more details:**

  **http://www.research.ibm.com/secure_systems_department**