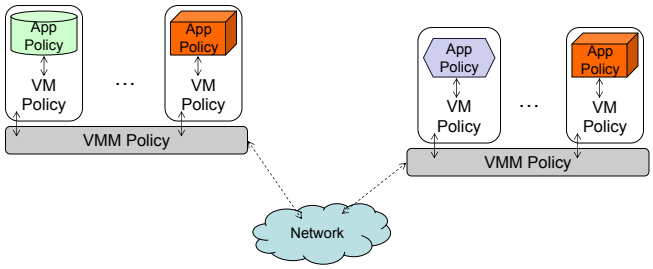
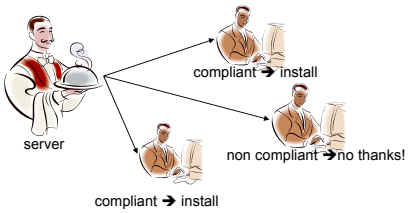
Compliance Analysis of Mandatory Access Control Policies

Sandra Rueda, Dave King, Trent Jaeger, Patrick McDaniel

- Administrators define security policies at various layers: Application, Operating System, Virtual Machine Monitors and Network.
- *Since these policies are independently developed and enforced, there is no guarantee about their compliance with a global security goal.*
- We developed a formal definition of compliance based on information flows and tools to check, based on our compliance definition, whether an application policy and a system policy are compliant or not.



Compliance of Trusted Programs

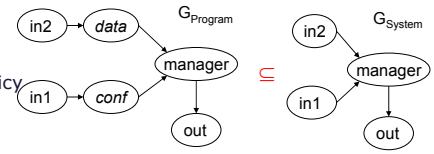


Trusted Programs (TPs) are expected to only perform safe operations even though they have the rights to perform unsafe operations.

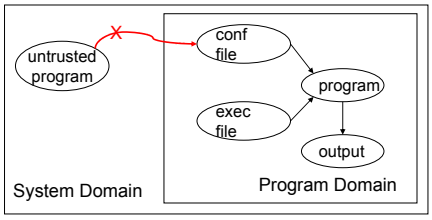
Historically, TPs were blindly trusted. Nowadays we have mechanisms to generate proofs about TPs behavior: security typed languages and user-level reference monitors.

A trusted program and OS policies are compliant if the program enforces system policy and the system protects program's components.

Program enforces system policy

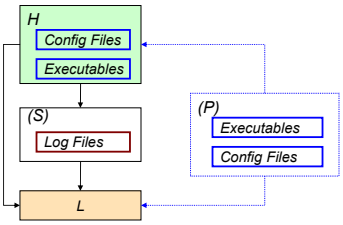


System protects program's components



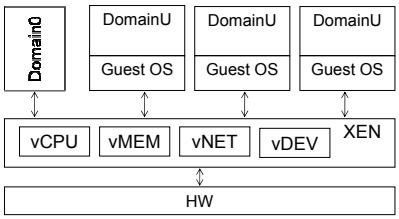
Evaluating compliance of application and operating system security policies is difficult. If you do not have a mapping, how do you relate elements in one policy to elements in the other policy?

We do not map element to element, we decided to map to an intermediate representation. We are able to automate such mapping based on the fact that trusted program components are higher integrity than the data they process.



Compliance of XSM policies

Xen Security Module (XSM) provides a mechanism to define policies that control inter-VM communication and access to virtual and physical resources.

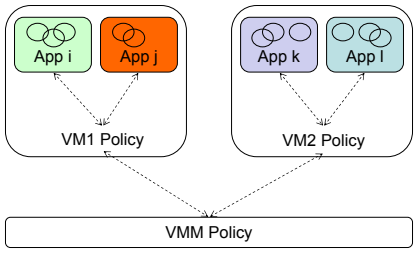


We say that the components of a VM system are policy compliant if:

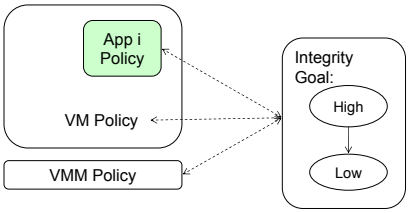
1. The XSM policy protects the domains (Dom0 and DomU) and guest operating systems protect internal application's components (tamperproof goal).

2. Applications, operating systems and XSM enforce system policies (general system goal).

Tamperproof Goal



System Goal



Issues to address:

- Mechanism to enable administrators define high level information-flow-based security goals (i.e. what VMs are allowed to exchange information).
- Mechanism to generate the set of information flows allowed by a given XSM policy specification.
- Evaluate whether a given XSM policy meets security requirements defined by an administrator.

Publications

S. Rueda, D. King, T. Jaeger. Verifying Compliance of Trusted Programs. USENIX Security Symposium 2008.

S. Rueda, Y. Sreenivasan, T. Jaeger. Flexible Security Configuration for Virtual Machines. ACM Computer Security Architecture Workshop. CSAW, 2008.

S. Rueda, D. King, T. Jaeger. Reduction of the Compliance Problem to the Graph Isomorphism Problem. SIIS Lab, Technical Report NAS-TR-0081-2007, 2007.

B. Hicks, S. Rueda, L. StClair, T. Jaeger, P. McDaniel. A Logical Specification and Analysis for SELinux MLS Policy. ACM Symposium on Access Control Models and Technologies, SACMAT, 2007.

Sponsored by IARPA and AFRL