

Xen Security Modules (XSM)

George Coker
National Information Assurance Research Lab
National Security Agency (NSA)
gscoker@alpha.ncsc.mil

XSM Background

- Motivation
 - New usage models for Xen have different security goals
 - Generalized security framework for Xen
 - Creates general security interfaces for Xen
 - Allows custom security functionality in modules
 - Removes security model specific code from Xen
- Modules
 - Flask, ACM (sHype), dummy (default)

XSM Status

- Hypervisor
 - Hook targets
 - Privileged hypercall commands
 - Cross-domain interactions
 - Domain::Domain
 - Domains, event channels, grant tables
 - Domain::IO resource
 - IO memory, IO ports, interrupts
 - Security hypercall

XSM Status (cont.)

- Hypervisor
 - Future work
 - Track Xen changes
 - Documentation
 - Continue analysis

XSM Status (cont.)

- Control Plane
 - Hook targets
 - Facilitate usage of hypervisor-XSM interfaces
 - Limited access control over control plane resources
 - Future work
 - XSM enabling control plane
 - Facilitate new module development
 - Unification of security policy management

XSM Status (cont.)

- Performance
 - XSM vs. unmodified Xen
 - Nominal performance impact
 - dom0, domU workload
 - Measured from dom0 and domU
 - Imbench, kbench
 - Performance impact of modules
 - Variation in microbenchmarks
 - Consistent performance for macrobenchmarks

Flask Module Status

- Background
 - Fine grain, flexible MAC similar to SELinux
 - Developing new usage models for Xen

Flask Module Status (cont.)

- Capabilities
 - Comprehensive user of XSM
 - RCU-enabled cache
 - Labeling
 - Hypervisor managed SIDs for domains and event channels
 - Flask managed SIDs for IO Resources
 - Page SIDs inferred from page owner SID

Flask Module Status (cont.)

- Future work
 - Develop demonstrable policies
 - Dom0 decomposition
 - Policy support for domain groups
 - Build new security platforms
 - Xen for security

XSM Submission

- To date submissions
 - Productive feedback and discussion
 - Reorganization of XSM and module code
 - /xsm, /xsm{flask, acm}, /include/xsm
 - Discussion of proposed hooks
- Ratify a plan for submission and acceptance
 - Strong patch dependencies

XSM Submission (cont.)

- Patch breakdown proposal
 - XSM patches
 - XSM infrastructure
 - Hooks by hypercall

XSM Submission (cont.)

- Patch breakdown proposal
 - Module patches
 - Flask module
 - Flask module
 - Xen support modifications
 - Add vsprintf, sscanf to vsscanf.c
 - Flask userland
 - ACM module
 - XSM-enabling
 - Remove legacy interfaces
 - Refactoring to /xsm/acm

XSM Submission (cont.)

- Patch breakdown proposal
 - Tool patches
 - Control plane modifications
 - XSM infrastructure
 - Dummy module
 - Flask module
 - ACM module
 - Refactoring patch
 - Documentation
 - TBD

Questions?