Management of the Access Control Module through the Xen-API

Stefan Berger, Reiner Sailer, Ronald Perez, Ramón Cáceres

IBM T. J. Watson Research Center, NY
Outline

- Background
- Motivation for ACM management API
- What is the Xen-API?
- Introducing our Contribution
- Conclusion
ACM: Controlled Sharing and Resource Access

Policy

Human Resources

Payroll

sHype
(control sharing)

Human Resources

Payroll

Work Load

Xen VMM
(virtualizes + isolates)

VM
Motivation for Management API

- Security needs to be manageable
  - Allow Systems management software to manage systems’ policies
  - Customers’ demand

- What to manage?
  - ACM policy for now, maybe others later
  - Labeling of VMs and resources

- Current requirements
  - Policy life-cycle support: Uploading, retrieving, activating, updating
  - Label assignments
  - Extensible design
What is the Xen-API?

- Programmatic RPC Interface for Xen system management
- Support for VM life cycle management:
  - Create / destroy
  - Suspend / resume
  - VM Configuration functions
- Manages VMs’ storage, network access etc.
- Some Systems management software integration through Xen-API
  - CIM providers use Xen-API
Tell me about the type of security, if any, is support on this system?

```python
types = get_xstype()
```

Methods and Classes for ACM support

- `activate_policy()`
- `get_xspolicy()`
- `set_xspolicy()`
- `get_xstype()`
- `set_resource_label()`
- `get_resource_label()`
- `get_labeled_resources()`
- `rm_xsbootpolicy()`
Methods and Classes for ACM support

Install a policy of a given type on the system, compile and activate it!

\[
policy\_state = \text{set}\_xspolicy(\text{type, repr, flags, overwrite})
\]

XSPolicy

activate_policy()
get_xspolicy()
set_xspolicy()
get_xstype()

get_labeled_resources()
get_resource_label()
set_resource_label()
rm-xsbootpolicy()
Methods and Classes for ACM support

Is there a policy on the system? Is it activated?

\[\text{policy\_state} = \text{get\_xspolicy()}\]

activate_policy()
get_xspolicy()
set_xspolicy()
get_xstype()
get_labeled_resources()
get_resource_label()
set_resource_label()
rm_xsbootpolicy()
Methods and Classes for ACM support

List all labeled resources along with their labels.

```
resourc_label_map = get_labeled_resources()
```

```
activate_policy()
get_xsPolicy()
set_xsPolicy()
get_xstype()
get_labeled_resources()
get_resource_label()
set_resource_label()
rm_xsbootpolicy()
```
Methods and Classes for ACM support

**XSPolicy**
- `activate_policy()`
- `get_xspolicy()`
- `set_xspolicy()`
- `get_xstype()`
- `get_labeled_resources()`
- `get_resource_label()`
- `set_resource_label()`
- `rm_xsbootpolicy()`

Labeling a resource

```
set_resource_label(resource, new_label, cur_label)
```
Methods and Classes for ACM support

ACMPolicy
- get_map()
- get_header()
- activate_xspolicy()
- get_xspolicy()
- set_xspolicy()
- get_xstype()

XSPolicy
- get_map()
- get_header()
- activate_xspolicy()
- get_xspolicy()
- set_xspolicy()
- get_xstype()

Specialization
- get_xspolicy()

Gives ACM policy-specific information
- get_xml()
- get_binary()
- get_labeled_resources()
- get_resource_label()
- set_resource_label()
- rm_xsbootpolicy()
Labeling of Domains – VM class extensions

Command line:

```
#xm addlabel Avis dom avis.xm  #old-style vm config file
#xm addlabel Avis mgt Avis      #managed domain, xen-api
```

Programmatic:

```
Python: session.xenapi.VM.{set/get}_security_label(…)
C:                 xen_vm_{set/get}_security_label(…)
```

→ VM record holds security label: security/label
Labeling of disks – VDI class extensions

Command line:

```bash
#xm addlabel Avis res /dev/sda3  #uses xen-api if possible
```

Programmatic:

- **Python:** `session.xenapi.VDI.{set/get}_security_label(...)`
- **C:** `xen_vdi_{set/get}_security_label(...)`
- System-wide resource file holds label for resource
- Hides exact location of image files
About the ACM Labels used with the Xen-API

- Flexible label format:
  
  `<PolicyType>:<PolicyName>:<Label>`

- **Policy type**: ‘ACM’
  - The only one currently supported

- **Policy name**: from the XML

- **Label**: a VM or Resource label from the XML

- Example:

  `ACM:test-policy:Avis`
Code & Documentation

- Extensions to libxen:
  - Prototypes: xen_xspolicy.h, xen_acmpolicy.h
  - C-bindings

- Test code to exercise API
  - test/test_xspolicy.c
  - Extended xm-test suite (with xen-api tests!)

- Documentation:
  - Xen-API docs
  - Detail parameters to API calls
  - User docs to follow

- Patches to follow
Conclusion

- Xen-API extended with programmatic interface for ACM management
- Customers want to manage security on virtualized systems

Questions or Comments?

Thank you!