

**NAME**

Sys::Virt::Secret – Represent & manage a libvirt secret

**DESCRIPTION**

The `Sys::Virt::Secret` module represents a secret managed by the virtual machine monitor.

**METHODS**

`my $uuid = $sec->get_uuid()`

Returns a 16 byte long string containing the raw globally unique identifier (UUID) for the secret.

`my $uuid = $sec->get_uuid_string()`

Returns a printable string representation of the raw UUID, in the format 'XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX'.

`my $type = $sec->get_usage_type()`

Returns the usage type of this secret. The usage type determines the format of the unique identifier for this secret.

`my $id = $sec->get_usage_id()`

Returns the identifier of the object with which the secret is to be used. For secrets with a usage type of volume, the identifier is the fully qualified path.

`my $xml = $sec->get_xml_description()`

Returns an XML document containing a complete description of the secret's configuration

`$sec->undefine()`

Remove the configuration associated with a secret previously defined with the `define_secret` method in `Sys::Virt`.

`$bytes = $sec->get_value()`

Returns the raw bytes for the value of this secret, or undef if there is no value stored with the secret.

`$sec->set_value($bytes)`

Sets the value for the secret to be `$bytes`.

**CONSTANTS**

This section documents constants that are used with various APIs described above

**SECRET USAGE TYPE**

The following constants refer to the different usage types

`Sys::Virt::Secret::USAGE_TYPE_NONE`

The constant for secrets which are not assigned for use with a particular object

`Sys::Virt::Secret::USAGE_TYPE_VOLUME`

The constant for secrets which are to be used for storage volume encryption. The usage ID for secrets will refer to the fully qualified volume path.

`Sys::Virt::Secret::USAGE_TYPE_CEPH`

The constant for secrets which are to be used for authenticating to CEPH storage volumes. The usage ID for secrets will refer to the server name.

`Sys::Virt::Secret::USAGE_TYPE_ISCSI`

The constant for secrets which are to be used for authenticating to iSCSI storage volumes. The usage ID for secrets will refer to the server name.

`Sys::Virt::Secret::USAGE_TYPE_TLS`

The constant for secrets which are to be used for authenticating for the TLS environment. The usage ID for the secret can be any name. The UUID for the secret will be used by the hypervisor to fetch the secret value.

`Sys::Virt::Secret::USAGE_TYPE_VTPM`

The constant for secrets which are to be used for encrypting virtual TPM state. The usage ID for the secret will refer to the TPM path. The UUID for the secret will be used by the hypervisor to fetch the secret value.

**LIST FILTERING**

The following constants are used to filter object lists

Sys::Virt::Secret::LIST\_EPHEMERAL

Include any secrets marked as ephemeral

Sys::Virt::Secret::LIST\_NO\_EPHEMERAL

Include any secrets not marked as ephemeral

Sys::Virt::Secret::LIST\_PRIVATE

Include any secrets marked as private

Sys::Virt::Secret::LIST\_NO\_PRIVATE

Include any secrets not marked as private

**SECRET DEFINE**

The following constants can be used to control the behaviour of secret define operations

Sys::Virt::Secret::DEFINE\_VALIDATE

Validate the XML document against the XML schema

**EVENT ID CONSTANTS**

Sys::Virt::Secret::EVENT\_ID\_LIFECYCLE

Secret lifecycle events

Sys::Virt::Secret::EVENT\_ID\_VALUE\_CHANGED

Secret value change events

**LIFECYCLE CHANGE EVENTS**

The following constants allow secret lifecycle change events to be interpreted. The events contain both a state change, and a reason though the reason is currently unused.

Sys::Virt::Secret::EVENT\_DEFINED

Indicates that a persistent configuration has been defined for the secret.

Sys::Virt::Secret::EVENT\_UNDEFINED

The persistent configuration has gone away

**AUTHORS**

Daniel P. Berrange <berrange@redhat.com>

**COPYRIGHT**

Copyright (C) 2006–2009 Red Hat Copyright (C) 2006–2007 Daniel P. Berrange

**LICENSE**

This program is free software; you can redistribute it and/or modify it under the terms of either the GNU General Public License as published by the Free Software Foundation (either version 2 of the License, or at your option any later version), or, the Artistic License, as specified in the Perl README file.

**SEE ALSO**

Sys::Virt, Sys::Virt::Error, <http://libvirt.org>