

src.ielo.net - Zinn XMLRPC API 1.0.0 Documentation

Table of Contents

<u>src.ielo.net - Zinn XMLRPC API 1.0.0 Documentation</u>	1
<u>Table of Contents</u>	2
<u>Zinn XMLRPC API 1.0.0</u>	4
<u>Methods Index</u>	4
<u>Methods List</u>	6
<u>api.version</u>	6
<u>client.accounting</u>	6
<u>client.attributes</u>	7
<u>client.create</u>	7
<u>client.delete</u>	7
<u>client.invitation.acknowledge</u>	8
<u>client.invitation.create</u>	8
<u>client.invitation.list</u>	9
<u>client.list</u>	9
<u>client.update</u>	9
<u>image.accounting</u>	10
<u>image.attributes</u>	10
<u>image.create</u>	10
<u>image.delete</u>	11
<u>image.list</u>	11
<u>image.update</u>	11
<u>interface.accounting</u>	12
<u>interface.attributes</u>	12
<u>interface.create</u>	12
<u>interface.delete</u>	13
<u>interface.ipaddress.attach</u>	13
<u>interface.ipaddress.detach</u>	14
<u>interface.ipaddress.list</u>	14
<u>interface.list</u>	14
<u>interface.types.list</u>	15
<u>interface.update</u>	15
<u>ipaddress.accounting</u>	15
<u>ipaddress.attributes</u>	15
<u>ipaddress.create</u>	16
<u>ipaddress.delete</u>	16
<u>ipaddress.list</u>	16
<u>ipaddress.update</u>	17
<u>machine.accounting</u>	17
<u>machine.attributes</u>	18
<u>machine.boot</u>	18
<u>machine.cputype.list</u>	18
<u>machine.create</u>	19
<u>machine.delete</u>	19
<u>machine.interface.attach</u>	19
<u>machine.interface.detach</u>	20
<u>machine.interface.list</u>	20
<u>machine.list</u>	20

Table of Contents

Table of Contents

<u>machine.migrate</u>	21
<u>machine.shutdown</u>	21
<u>machine.storage.attach</u>	21
<u>machine.storage.bustype.list</u>	22
<u>machine.storage.detach</u>	22
<u>machine.storage.list</u>	22
<u>machine.update</u>	22
<u>machine.vnc.keymap.list</u>	23
<u>permission.list</u>	23
<u>role.accounting</u>	23
<u>role.attributes</u>	24
<u>role.create</u>	24
<u>role.delete</u>	24
<u>role.list</u>	25
<u>role.permission.attach</u>	25
<u>role.permission.detach</u>	25
<u>role.permission.list</u>	26
<u>role.update</u>	26
<u>role.user.attach</u>	26
<u>role.user.attributes</u>	27
<u>role.user.detach</u>	27
<u>role.user.list</u>	27
<u>role.user.update</u>	28
<u>sitegroup.list</u>	28
<u>storage.accounting</u>	28
<u>storage.attributes</u>	28
<u>storage.clone</u>	29
<u>storage.clone config</u>	29
<u>storage.clone data</u>	29
<u>storage.create</u>	30
<u>storage.delete</u>	30
<u>storage.list</u>	30
<u>storage.update</u>	31
<u>system.listMethods</u>	31
<u>system.methodHelp</u>	31
<u>system.methodSignature</u>	31
<u>user.list</u>	32
<u>vlan.accounting</u>	32
<u>vlan.attributes</u>	32
<u>vlan.create</u>	33
<u>vlan.delete</u>	33
<u>vlan.get next tag</u>	33
<u>vlan.interface.attach</u>	33
<u>vlan.interface.detach</u>	34
<u>vlan.interface.list</u>	34
<u>vlan.is tag free</u>	34
<u>vlan.list</u>	35

Table of Contents

Table of Contents

<u>vlan.update</u>	35
<u>vsite.attributes</u>	35
<u>vsite.create</u>	36
<u>vsite.delete</u>	36
<u>vsite.list</u>	36

src.ielo.net - Zinn XMLRPC API 1.0.0 **Documentation**

Table of Contents

- Zinn XMLRPC API 1.0.0
 - ◆ Methods Index
 - ◆ Methods List
 - ◇ api.version
 - ◇ client.accounting
 - ◇ client.attributes
 - ◇ client.create
 - ◇ client.delete
 - ◇ client.invitation.acknowledge
 - ◇ client.invitation.create
 - ◇ client.invitation.list
 - ◇ client.list
 - ◇ client.update
 - ◇ image.accounting
 - ◇ image.attributes
 - ◇ image.create
 - ◇ image.delete
 - ◇ image.list
 - ◇ image.update
 - ◇ interface.accounting
 - ◇ interface.attributes
 - ◇ interface.create
 - ◇ interface.delete
 - ◇ interface.ipaddress.attach
 - ◇ interface.ipaddress.detach
 - ◇ interface.ipaddress.list
 - ◇ interface.list
 - ◇ interface.types.list
 - ◇ interface.update
 - ◇ ipaddress.accounting
 - ◇ ipaddress.attributes
 - ◇ ipaddress.create
 - ◇ ipaddress.delete
 - ◇ ipaddress.list
 - ◇ ipaddress.update
 - ◇ machine.accounting
 - ◇ machine.attributes
 - ◇ machine.boot
 - ◇ machine.cputype.list
 - ◇ machine.create
 - ◇ machine.delete
 - ◇ machine.interface.attach
 - ◇ machine.interface.detach
 - ◇ machine.interface.list
 - ◇ machine.list
 - ◇ machine.migrate
 - ◇ machine.shutdown
 - ◇ machine.storage.attach

- ◇ [machine.storage.bustype.list](#)
 - ◇ [machine.storage.detach](#)
 - ◇ [machine.storage.list](#)
 - ◇ [machine.update](#)
 - ◇ [machine.vnc.keymap.list](#)
 - ◇ [permission.list](#)
 - ◇ [role.accounting](#)
 - ◇ [role.attributes](#)
 - ◇ [role.create](#)
 - ◇ [role.delete](#)
 - ◇ [role.list](#)
 - ◇ [role.permission.attach](#)
 - ◇ [role.permission.detach](#)
 - ◇ [role.permission.list](#)
 - ◇ [role.update](#)
 - ◇ [role.user.attach](#)
 - ◇ [role.user.attributes](#)
 - ◇ [role.user.detach](#)
 - ◇ [role.user.list](#)
 - ◇ [role.user.update](#)
 - ◇ [sitegroup.list](#)
 - ◇ [storage.accounting](#)
 - ◇ [storage.attributes](#)
 - ◇ [storage.clone](#)
 - ◇ [storage.clone_config](#)
 - ◇ [storage.clone_data](#)
 - ◇ [storage.create](#)
 - ◇ [storage.delete](#)
 - ◇ [storage.list](#)
 - ◇ [storage.update](#)
 - ◇ [system.listMethods](#)
 - ◇ [system.methodHelp](#)
 - ◇ [system.methodSignature](#)
 - ◇ [user.list](#)
 - ◇ [vlan.accounting](#)
 - ◇ [vlan.attributes](#)
 - ◇ [vlan.create](#)
 - ◇ [vlan.delete](#)
 - ◇ [vlan.get_next_tag](#)
 - ◇ [vlan.interface.attach](#)
 - ◇ [vlan.interface.detach](#)
 - ◇ [vlan.interface.list](#)
 - ◇ [vlan.is_tag_free](#)
 - ◇ [vlan.list](#)
 - ◇ [vlan.update](#)
 - ◇ [vsite.attributes](#)
 - ◇ [vsite.create](#)
 - ◇ [vsite.delete](#)
 - ◇ [vsite.list](#)
-

Zinn XMLRPC API 1.0.0

Methods Index

A

- [api.version](#)

C

- [client.accounting](#)
- [client.attributes](#)
- [client.create](#)
- [client.delete](#)
- [client.invitation.acknowledge](#)
- [client.invitation.create](#)
- [client.invitation.list](#)
- [client.list](#)
- [client.update](#)

I

- [image.accounting](#)
- [image.attributes](#)
- [image.create](#)
- [image.delete](#)
- [image.list](#)
- [image.update](#)
- [interface.accounting](#)
- [interface.attributes](#)
- [interface.create](#)
- [interface.delete](#)
- [interface.ipaddress.attach](#)
- [interface.ipaddress.detach](#)
- [interface.ipaddress.list](#)
- [interface.list](#)
- [interface.types.list](#)
- [interface.update](#)
- [ipaddress.accounting](#)
- [ipaddress.attributes](#)
- [ipaddress.create](#)
- [ipaddress.delete](#)
- [ipaddress.list](#)
- [ipaddress.update](#)

M

- [machine.accounting](#)
- [machine.attributes](#)
- [machine.boot](#)

- [machine.cputype.list](#)
- [machine.create](#)
- [machine.delete](#)
- [machine.interface.attach](#)
- [machine.interface.detach](#)
- [machine.interface.list](#)
- [machine.list](#)
- [machine.migrate](#)
- [machine.shutdown](#)
- [machine.storage.attach](#)
- [machine.storage.bustype.list](#)
- [machine.storage.detach](#)
- [machine.storage.list](#)
- [machine.update](#)
- [machine.vnc.keymap.list](#)

P

- [permission.list](#)

S

- [sitegroup.list](#)
- [storage.accounting](#)
- [storage.attributes](#)
- [storage.clone](#)
- [storage.clone_config](#)
- [storage.clone_data](#)
- [storage.create](#)
- [storage.delete](#)
- [storage.list](#)
- [storage.update](#)
- [system.listMethods](#)
- [system.methodHelp](#)
- [system.methodSignature](#)

R

- [role.accounting](#)
- [role.attributes](#)
- [role.create](#)
- [role.delete](#)
- [role.list](#)
- [role.permission.attach](#)
- [role.permission.detach](#)
- [role.permission.list](#)
- [role.update](#)
- [role.user.attach](#)
- [role.user.attributes](#)
- [role.user.detach](#)
- [role.user.list](#)

- [role.user.update](#)

U

- [user.list](#)

V

- [vlan.accounting](#)
- [vlan.attributes](#)
- [vlan.create](#)
- [vlan.delete](#)
- [vlan.get next tag](#)
- [vlan.interface.attach](#)
- [vlan.interface.detach](#)
- [vlan.interface.list](#)
- [vlan.is tag free](#)
- [vlan.list](#)
- [vlan.update](#)
- [vsite.attributes](#)
- [vsite.create](#)
- [vsite.delete](#)
- [vsite.list](#)

Methods List

[go to index](#)

api.version

string api.version()

Parameters

Description

Get the API version.

[go to index](#)

client.accounting

none client.accounting()

Description

[go to index](#)

client.attributes

array client.attributes(bool onlyMutables)

Parameters

- onlyMutables bool (optionnal) filters (un)mutable attributes. default=False.

Description

List of attributes of a client.

possible attributes are :

- label : unmutable, string the client label.
- address : mutable, string the client postal address.
- country : mutable, string the client country name.

[go to index](#)

client.create

none client.create(string label , array attributes)

Parameters

- label string wanted label for the client (client name).
- attributes array (optionnal) of attributes.

Description

Create a Client.

A Client is a geographical container entity (eg: a company, a physical person, ...).

A Client handle one/many users via Client's Role(s).

By default when a User create a Client, he's attached to the client via the default 'owner' Role and have all permissions granted over the Client.

See also:

- [client.attributes](#)

[go to index](#)

client.delete

none client.delete(string label)

Parameters

- label string the client label.

Description

client.attributes

Destroy a client.

The client must be empty (no vsite(s) and no other user(s)) prior to allow deletion.

[go to index](#)

client.invitation.acknowledge

string client.invitation.acknowledge(string key)

Parameters

- key string the invitation key.

Description

Acknowledge an invitation with a key.

Return the label of the client.

[go to index](#)

client.invitation.create

array client.invitation.create(string label)

Parameters

- label string label of the client.

Description

Generate an invitation key for a given client.

Invitation key principle is :

1. One User U1 attached to a Client C create an invitation Key for this Client.
2. U1 sends this generated Key to a User U2 (eg via email).
3. U2 acknowledge the invitation with the Key.
4. Client C users can now see User U2. U2 has no rights at this time so U1 or another Client C user with right permissions can give some credentials to U2.

Returned array contains a struct compound of :

- key : a md5 style string.
- expiration : expiration date of the key.

See also:

- [client.invitation.acknowledge](#)
- [client.invitation.list](#)

[go to index](#)

client.delete

client.invitation.list

array client.invitation.list(string label)

Parameters

- label string label of the client.

Description

List all invitation keys for a given client.

Returned array contains a struct compound of :

- key : a md5 style string.
- expiration : expiration date of the key.

[go to index](#)

client.list

array client.list(string label)

Parameters

- label string (optionnal) the client label.

Description

List clients.

If no label passed or empty : return the names of a user's clients.

Else, return a detailed list of the user's client :

- vsites : struct indexed by vsite label (included Vsite attribute)
- roles : struct indexed by role label (included public attribute)
- users : array of client's users labels

[go to index](#)

client.update

none client.update(string label , struct attributes)

Parameters

- label string label of associated Client.
- attributes struct (optionnal) attributes.

Description

Update mutables attributes of a client.

See also:

client.invitation.list

- [client.attributes](#)

[go to index](#)

image.accounting

none image.accounting()

Description

[go to index](#)

image.attributes

array image.attributes(bool onlyMutables)

Parameters

- onlyMutables bool (optionnal) filters unmutable attributes.

Description

List of attributes of a image.

Possible attributes are :

- label : unmutable
- size : unmutable, string, default=1G

[go to index](#)

image.create

none image.create(string vsite , string label , attributes attributes)

Parameters

- vsite string associated vsite.
- label string label of the new storage.
- attributes attributes struct.

Description

Create an image.

See also:

- [storage.attributes](#)

[go to index](#)

client.update

image.delete

none image.delete(string vsite , string label)

Parameters

- vsite string associated vsite.
- label string label of the image.

Description

Delete an image, must not be attached to any machine.

[go to index](#)

image.list

array|struct image.list(string vsite , (optionnal) label)

Parameters

- vsite string associated vsite.
- label (optionnal) string|array label(s) of the image(s).

Description

List image(s) : detailed if label(s), general if not.

[go to index](#)

image.update

none image.update(string vsite , string label , struct attributes)

Parameters

- vsite string associated vsite.
- label string label of the image.
- attributes struct (optionnal) attributes.

Description

Update mutables attributes of a image.

See also:

- [image.attributes](#)
- [image.create](#)

[go to index](#)

interface.accounting

array interface.accounting(string vsite, string label)

Parameters

- vsite string associated vsite
- label string label of the interface.

Description

Get Accounting list for an interface.

list interfaces accounting period :

- start : string of the starting period
- stop : string of the stopping period
- byteIn : BigInt number of bytes In
- byteOut : BigInt number of bytes Out
- packetIn : BigInt number of packets In
- packetOut : BigInt number of packets Out

[go to index](#)

interface.attributes

array interface.attributes(bool onlyMutables)

Parameters

- onlyMutables bool (optionnal) filters unmutable attributes.

Description

List attributes of an interface.

Possible attributes are :

- label : unmutable, string
- macAddr : unmutable, string
- model : mutable, string, default=virtio

See also:

- [interface.types.list](#)

[go to index](#)

interface.create

none interface.create(string vsite , string label , struct attributes)

Parameters

interface.accounting

- vsite string associated vsite.
- label string label of the new interface.
- attributes struct (optionnal) attributes.

Description

Create an interface.

An interface is a virtual network card.

See also:

- [interface.attributes](#)
- [interface.ipaddress.attach](#)
- [interface.ipaddress.detach](#)
- [machine.interface.attach](#)
- [machine.interface.detach](#)
- [vlan.interface.detach](#)
- [vlan.interface.detach](#)

[go to index](#)

interface.delete

none interface.delete(string vsite, string label)

Parameters

- vsite string associated vsite.
- label string label of the interface.

Description

Delete an interface.

[go to index](#)

interface.ipaddress.attach

none interface.ipaddress.attach(string vsite , string label , string ipaddress)

Parameters

- vsite string associated vsite.
- label string label of the interface.
- ipaddress string label of ipaddress to attach.

Description

Attach an ipaddress to an interface.

[go to index](#)

interface.ipaddress.detach

none interface.ipaddress.detach(string vsite , string label , string ipaddress)

Parameters

- vsite string associated vsite.
- label string label of the interface.
- ipaddress string label of ipaddress to detach.

Description

Detach an ipaddress from an interface.

[go to index](#)

interface.ipaddress.list

array interface.ipaddress.list(string vsite , string label , nonelstringlarray ipaddress)

Parameters

- vsite string associated vsite.
- label string label of the interface.
- ipaddress nonelstringlarray (optionnal) label(s) of the ipaddress(es).

Description

List interface's attached ipaddress(es) : detailed if label(s), general if not.

[go to index](#)

interface.list

array interface.list(string vsite, nonelstringlarray label)

Parameters

- vsite string associated vsite
- label nonelstringlarray (optionnal) label(s) of the interface(s).

Description

List interface(s).

If no label, list all vsite interfaces labels.

Else, list interfaces indexed by label in a struct compound of :

- Vsite : string name of asociated Vsite.
- machine : struct compound of label and current status of attached machine(s).
- vlan : struct compound of label and tag of attached vlan.
- ip : array of attached ip label(s)

[go to index](#)

interface.types.list

array interface.types.list(string vsite)

Parameters

- vsite string to have the list related to.

Description

List network interface types (eg: virtio, ne2000, ...).

[go to index](#)

interface.update

none interface.update(string vsite , string label , struct attributes)

Parameters

- vsite string associated vsite.
- label string label of the interface.
- attributes struct (optionnal) attributes to update.

Description

Update mutable attributes of an interface.

See also:

- [interface.attributes](#)
- [interface.create](#)

[go to index](#)

ipaddress.accounting

none ipaddress.accounting()

Description

[go to index](#)

ipaddress.attributes

array ipaddress.attributes(bool onlyMutables)

Parameters

- onlyMutables bool (optionnal) filters unmutable attributes.

Description

List of attributes of a `ipaddress`.

Possible attributes are :

- `label` : immutable, string
- `addr` : immutable string, IP address
- `reverse` : mutable, string, default=None

[go to index](#)

ipaddress.create

none `ipaddress.create(string vsite, string label, struct attributes)`

Parameters

- `vsite` string associated vsite.
- `label` string label of the new `ipaddress`.
- `attributes` struct (optionnal) attributes.

Description

Create an `ipaddress`.

Note: `ipaddress` is dynamic, so you have to specify kind of IP Family you want to obtain in attributes : `family=ipv4` or `family=ipv6`.

See also:

- [ipaddress.attributes](#)

[go to index](#)

ipaddress.delete

none `ipaddress.delete(string vsite , string label)`

Parameters

- `vsite` string associated vsite.
- `label` string label of `ipaddress`.

Description

Destroy an `ipaddress`

[go to index](#)

ipaddress.list

array `ipaddress.list(string vsite , string label)`

Parameters

`ipaddress.attributes`

- vsite string associated vsite.
- label string (optionnal) label of ipaddress.

Description

List network ipaddresses.

[go to index](#)

ipaddress.update

none ipaddress.update(string vsite, string label, struct attributes)

Parameters

- vsite string associated vsite.
- label string label of the ipaddress.
- attributes struct attributes.

Description

Update mutable attributes of an ipaddress.

See also:

- [ipaddress.attributes](#)
- [ipaddress.create](#)

[go to index](#)

machine.accounting

array machine.accounting(string vsite, string label)

Parameters

- vsite string associated vsite
- label string label of the machine.

Description

Get Accounting list for a machine.

list machine accounting period :

- start : string of the starting period
- stop : string of the stopping period
- vcpu : integer number of virtual cpu
- cpu : string type of cpu
- memory : integer memory in bytes

[go to index](#)

machine.attributes

array machine.attributes(boolean onlyMutables)

Parameters

- onlyMutables boolean (optionnal) filters unmutable attributes.

Description

List attributes of a machine.

Possible attributes are :

- label : unmutable, string
- vcpu : mutable, integer, default=1
- cpu : mutable, integer, default=x86-64bit
- memory : mutable, string, default=1024M
- image : mutable, string, default=None

- vnc (substruct) : password(mutable, string, default=None), keymap(mutable, string, default=en-us)
[go to index](#)

machine.boot

none machine.boot(string vsite , string label)

Parameters

- vsite string associated vsite.
- label string label of the machine.

Description

Boot a machine.

The machine can boot only if it has :

- An interface.
- At least a bootable storage.
- All of its storages are ready.

[go to index](#)

machine.cputype.list

array machine.cputype.list(string vsite)

Parameters

- vsite string associated vsite.

Description

List cpu types.

[go to index](#)

machine.create

none machine.create(string vsite , string label , struct attributes)

Parameters

- vsite string associated vsite.
- label string label of the new machine.
- attributes struct (optionnal) attributes.

Description

Create a machine.

The machine is a GNU/Linux kvm virtual machine instance.

See also:

- [machine.attributes](#)

[go to index](#)

machine.delete

none machine.delete(string vsite , string label)

Parameters

- vsite string associated vsite.
- label string label of the machine.

Description

Delete a machine.

Attached resources (interface, storage, ...) must be detached first.

[go to index](#)

machine.interface.attach

none machine.interface.attach(string vsite , string label , string interface)

Parameters

- vsite string associated vsite.
- label string label of the machine.
- interface string label of interface to attach.

Description

Attach an interface to a machine.

machine.cputype.list

[go to index](#)

machine.interface.detach

none machine.interface.detach(string vsite , string label , string interface)

Parameters

- vsite string associated vsite.
- label string label of the machine.
- interface string label of interface to detach.

Description

Detach an interface from a machine.

[go to index](#)

machine.interface.list

array machine.interface.list(string vsite , string label , nonelstringlarray interface)

Parameters

- vsite string associated vsite.
- label string label of the machine.
- interface nonelstringlarray (optionnal) label(s) of the interface(s).

Description

List machine's attached interface(s).
Detailed if label(s), general if not.

[go to index](#)

machine.list

array machine.list(string vsite , nonelstringlarray label)

Parameters

- vsite string associated vsite.
- label nonelstringlarray (optionnal) label(s) of the machine(s).

Description

List machine(s).
Detailed if label(s), general if not.

[go to index](#)

machine.migrate

none machine.migrate(string vsite , string label , string hostnode)

Parameters

- vsite string associated vsite.
- label string machine's label.
- hostnode string machine destination node label.

Description

Migrate a machine.

The machine can only migrate to another node than the one it runs.

[go to index](#)

machine.shutdown

none machine.shutdown(string vsite , string label)

Parameters

- vsite string associated vsite.
- label string label of the machine.

Description

Shutdown a machine.

[go to index](#)

machine.storage.attach

none machine.storage.attach(string vsite , string label , string storage , nonelstruct attributes)

Parameters

- vsite string associated vsite.
- label string label of the machine.
- storage string label of storage to attach.
- attributes nonelstruct (optionnal) array attributes.

Description

Attach a storage volume to a virtual machine.

Possible attributes are :

- bus (default : virtio)
- media (default : disk)
- readOnly (default : False)
- bootOrder (default : 1)
- bootable (default: False)

[go to index](#)

machine.storage.bustype.list

array machine.storage.bustype.list(string vsite)

Parameters

- vsite string label of associated vsite.

Description

List storage bus types.

[go to index](#)

machine.storage.detach

none machine.storage.detach(string vsite , string label , string storage)

Parameters

- vsite string associated vsite.
- label string label of the machine.
- storage string label of storage to detach.

Description

Detach a storage from a machine.

[go to index](#)

machine.storage.list

array machine.storage.list(string vsite , string label , nonelstringlarray storage)

Parameters

- vsite string associated vsite.
- label string label of the machine.
- storage nonelstringlarray (optionnal) label(s) of the storage(s).

Description

List machine's attached storage(s)

Detailed if label(s), general if not.

[go to index](#)

machine.update

none machine.update(string vsite , string label , struct attributes)

Parameters

machine.storage.attach

- vsite string associated vsite.
- label string label of the machine.
- attributes struct (optionnal) attributes.

Description

Update mutable attributes of a machine.

See also:

- [machine.attributes](#)
- [machine.create](#)

[go to index](#)

machine.vnc.keymap.list

array machine.vnc.keymap.list(string vsite)

Parameters

- vsite string associated vsite.

Description

List vnc keymaps.

[go to index](#)

permission.list

struct permission.list()

Parameters

Description

List permissions.

[go to index](#)

role.accounting

none role.accounting()

Description

[go to index](#)

role.attributes

array role.attributes(bool onlyMutables)

Parameters

- onlyMutables bool (optionnal) filters unmutable attributes.

Description

List of attributes of a role.

possible attributes are :

- label : mutable, string
- public : mutable, bool

[go to index](#)

role.create

none role.create(string client , string label , struct attributes)

Parameters

- client string associated client label.
- label string role label.
- attributes struct (optionnal) attributes.

Description

Create a role.

See also:

- [role.attributes](#)

[go to index](#)

role.delete

none role.delete(string client , string label)

Parameters

- client string associated client label.
- label string role label.

Description

Destroy a role.

[go to index](#)

role.attributes

role.list

array role.list(string client , nonelstringlarray label)

Parameters

- client string associated client label.
- label nonelstringlarray role label.

Description

List roles.

[go to index](#)

role.permission.attach

none role.permission.attach(string client , string label , string vsite , string family , string permission)

Parameters

- client string associated client.
- label string label of the role.
- vsite string label of the vsite.
- family string family of the service.
- permission string permission related to service family.

Description

Attach a user to a role.

See also:

- [role.permission.list](#)

[go to index](#)

role.permission.detach

none role.permission.detach(string client , string label , string vsite , string family , string permission)

Parameters

- client string associated client.
- label string label of the role.
- vsite string label of the vsite.
- family string family of the service.
- permission string permission related to service family.

Description

Detach a user from a role.

See also:

[role.list](#)

- [role.permission.list](#)

[go to index](#)

role.permission.list

array role.permission.list(string client , string label , string vsite)

Parameters

- client string associated client.
- label string label of the role.
- vsite string label of the vsite.

Description

List role's attached permission(s)

[go to index](#)

role.update

none role.update(string client , string label , struct attributes)

Parameters

- client string associated client label.
- label string role label.
- attributes struct (optionnal) attributes.

Description

Update role attributes.

[go to index](#)

role.user.attach

none role.user.attach(string client , string label , string user)

Parameters

- client string associated client label.
- label string role label.
- user string user label.

Description

Attach a user to a role.

[go to index](#)

role.user.attributes

array role.user.attributes(bool onlyMutables)

Parameters

- onlyMutables bool (optionnal) filters unmutable attributes.

Description

List of attributes of a userrole relation.

possible attributes are :

- label : unmutable, string
- public : mutable, bool, default=False

[go to index](#)

role.user.detach

none role.user.detach(string client , string label , string user)

Parameters

- client string associated client label.
- label string role label.
- user string user label.

Description

Detach a user from a role.

[go to index](#)

role.user.list

array role.user.list(string client , string label , nonelstringlabel user)

Parameters

- client string associated client label.
- label string label of the role.
- user nonelstringlabel (optionnal) label(s) of user(s).

Description

List role's attached user(s) : detailed if label(s), general if not.

[go to index](#)

role.user.update

none role.user.update(string client , string label , struct attributes)

Parameters

- client string associated client label.
- label string role label.
- attributes struct (optionnal) attributes.

Description

Update userrole relation attributes.

See also:

- [role.user.attributes](#)

[go to index](#)

sitegroup.list

none sitegroup.list(nonelstringlarray label)

Parameters

- label nonelstringlarray (optionnal) sitegroup label

Description

List sitegroups.

[go to index](#)

storage.accounting

none storage.accounting()

Description

[go to index](#)

storage.attributes

array storage.attributes(bool onlyMutables)

Parameters

- onlyMutables bool (optionnal) filters unmutable attributes.

Description

List of attributes of a storage.

Possible attributes are :

- label : immutable
- size : immutable, string, default=1G

[go to index](#)

storage.clone

none storage.clone(string vsite , string label , string from_label)

Parameters

- vsite string associated vsite.
- label string label of the storage.
- from_label string label of the storage to clone.

Description

Composite call of storage_cloneconfig and storage_clonedata

See also:

- [storage.clone_config](#)
- [storage.clone_data](#)

[go to index](#)

storage.clone_config

none storage.clone_config(string vsite , string label , string from_label)

Parameters

- vsite string associated vsite.
- label string label of the storage.
- from_label string label of the storage to clone.

Description

Create from an existant storage a new empty one with same size/vsite.

[go to index](#)

storage.clone_data

none storage.clone_data(string vsite , string label , string from_label)

Parameters

storage.attributes

- vsite string associated vsite.
- label string label of the storage.
- from_label string label of the storage to clone.

Description

Clone data of a storage to another storage.

[go to index](#)

storage.create

none storage.create(string vsite , string label , attributes attributes)

Parameters

- vsite string associated vsite.
- label string label of the new storage.
- attributes attributes struct.

Description

Create a storage volume.

See also:

- [storage.attributes](#)

[go to index](#)

storage.delete

none storage.delete(string vsite , string label)

Parameters

- vsite string associated vsite.
- label string label of the storage.

Description

Delete a storage, must not be attached to any machine.

[go to index](#)

storage.list

array|struct storage.list(string vsite , (optionnal) label)

Parameters

- vsite string associated vsite.
- label (optionnal) string|array label(s) of the storage(s).

Description

List storage(s) : detailed if label(s), general if not.

[go to index](#)

storage.update

none storage.update(string vsite , string label , struct attributes)

Parameters

- vsite string associated vsite.
- label string label of the storage.
- attributes struct (optionnal) attributes.

Description

Update mutables attributes of a storage.

See also:

- [storage.attributes](#)
- [storage.create](#)

[go to index](#)

system.listMethods

none system.listMethods()

Description

Return a list of the method names implemented by this server.

[go to index](#)

system.methodHelp

none system.methodHelp()

Description

Return a documentation string describing the use of the given method.

[go to index](#)

system.methodSignature

none system.methodSignature()

Description

Return a list of type signatures.

storage.list

Each type signature is a list of the form [rtype, type1, type2, ...] where rtype is the return type and typeN is the type of the Nth argument. If no signature information is available, the empty string is returned.

[go to index](#)

user.list

none user.list(nonelstring|array label)

Parameters

- label nonelstring|array (optionnal) user label

Description

List user(s).

[go to index](#)

vlan.accounting

none vlan.accounting()

Description

[go to index](#)

vlan.attributes

array vlan.attributes(bool onlyMutables)

Parameters

- onlyMutables bool (optionnal) filters unmutable attributes.

Description

List attributes of a vlan.

Possible attributes are :

- label : unmutable, string
- tag : unmutable, integer, default=1

[go to index](#)

vlan.create

none vlan.create(string vsite , string label , struct attributes)

Parameters

- vsite string associated vsite label.
- label string vlan label.
- attributes struct (optionnal) attributes.

Description

Create a tagged vlan.

[go to index](#)

vlan.delete

none vlan.delete(string vsite , string label)

Parameters

- vsite string associated vsite label.
- label string vlan label.

Description

Destroy a vlan.

[go to index](#)

vlan.get_next_tag

int vlan.get_next_tag(string vsite)

Parameters

- vsite string associated vsite label.

Description

Get the next available tag.

[go to index](#)

vlan.interface.attach

none vlan.interface.attach(string vsite , string label , string interface)

Parameters

- vsite string associated vsite label.
- label string vlan label.
- interface string interface label.

Description

Attach an interface to a vlan.

[go to index](#)

vlan.interface.detach

none vlan.interface.detach(string vsite , string label , string interface)

Parameters

- vsite string associated vsite label.
- label string vlan label.
- interface string interface label.

Description

Restore interface attachment to default vlan.

[go to index](#)

vlan.interface.list

array vlan.interface.list(string vsite , string label , nonelstringlarray interface)

Parameters

- vsite string associated vsite.
- label string label of the vlan.
- interface nonelstringlarray label(s) of the interface(s).

Description

List vlan's attached interface(s)
Detailed if interface(s) label(s), general if not.

[go to index](#)

vlan.is_tag_free

True/False vlan.is_tag_free(int tag)

Parameters

- tag int tag.

Description

Check if a tag id is free to use.

[go to index](#)

vlan.list

array vlan.list(string vsite , nonelstringlarray label)

Parameters

- vsite string associated vsite label.
- label nonelstringlarray vlan label.

Description

List virtual networks.

[go to index](#)

vlan.update

none vlan.update(string vsite , string label , struct attributes)

Parameters

- vsite string associated vsite.
- label string label of the vlan.
- attributes struct (optionnal) attributes.

Description

Update mutable attributes of a vlan.

See also:

- [vlan.attributes](#)
- [vlan.create](#)

[go to index](#)

vsite.attributes

array vsite.attributes(boolean onlyMutables)

Parameters

- onlyMutables boolean (optionnal) filters unmutable attributes.

Description

List of attributes of a vsite.

possible attributes are :

- label : unmutable, string name of the vsite

[go to index](#)

vsite.create

none vsite.create(string label, struct attributes)

Parameters

- label string label of the vsite.
- attributes struct (optionnal) attributes.

Description

Create a vsite.

[go to index](#)

vsite.delete

none vsite.delete(string label)

Parameters

- label string vsite label.

Description

Destroy a vsite.

[go to index](#)

vsite.list

array vsite.list(nonelstringlarray label)

Parameters

- label nonelstringlarray (optionnal) vsite label.

Description

List vsite(s).