

Red Hat Satellite API Documentation

Version 5.3.0

- activationkey
 - api
 - auth
 - channel
 - channel.access
 - channel.org
 - channel.software
 - configchannel
 - errata
 - kickstart
 - kickstart.filepreservation
 - kickstart.keys
 - kickstart.profile
 - kickstart.profile.keys
 - kickstart.profile.software
 - kickstart.profile.system
 - kickstart.tree
 - org
 - org.trusts
 - packages
 - packages.provider
 - packages.search
 - preferences.locale
 - proxy
 - satellite
 - schedule
 - system
 - system.config
 - system.custominfo
 - system.provisioning.snapshot
 - system.search
 - systemgroup
 - user
-

activationkey

(Return to Handler List)

Available methods

- addChildChannels
- addConfigChannels
- addEntitlements
- addPackageNames
- addPackages
- addServerGroups
- checkConfigDeployment
- create
- create
- delete
- disableConfigDeployment
- enableConfigDeployment
- getDetails
- listActivatedSystems
- listActivationKeys
- listConfigChannels
- removeChildChannels
- removeConfigChannels
- removeEntitlements
- removePackageNames
- removePackages
- removeServerGroups
- setConfigChannels
- setDetails

Description

Contains methods to access common activation key functions available from the web interface.

Namespace: activationkey

Method: addChildChannels

Description:

Add child channels to an activation key.

Parameters:

- string sessionKey
- string key
- array:
 - string - childChannelLabel

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addConfigChannels

Description:

Given a list of activation keys and configuration channels, this method adds given configuration channels to either the top or the bottom (whichever you specify) of an activation key's configuration channels list. The ordering of the configuration channels provided in the add list is maintained while adding. If one of the configuration channels in the 'add' list already exists in an activation key, the configuration channel will be re-ranked to the appropriate place.

Parameters:

- string sessionKey
- array:
 - string - activationKey
- array:
 - string - List of configuration channel labels in the ranked order.
- boolean addToTop
 - true - To prepend the given channels to the beginning of the activation key's config channel list
 - false - To append the given channels to the end of the activation key's config channel list

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addEntitlements

Description:

Add entitlements to an activation key. Currently only add-on entitlements are permitted. (monitoring_entitled, provisioning_entitled, virtualization_host, virtualization_host_platform)

Parameters:

- string sessionKey
- string key
- array:

-
- string - entitlement label
 - monitoring_entitled
 - provisioning_entitled
 - virtualization_host
 - virtualization_host_platform

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addPackageNames

Description:

Add packages to an activation key using package name only.

Deprecated as of version - being replaced by addPackages(string sessionKey, string key, array[packages])

Parameters:

- string sessionKey
- string key
- array:
 - string - packageName

Returns:

- int - 1 on success, exception thrown otherwise.

Available since: 10.2

Method: addPackages

Description:

Add packages to an activation key.

Parameters:

- string sessionKey
- string key
- array:
 - struct - packages
 - string "name" - Package name
 - string "arch" - Arch label - Optional

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: addServerGroups

Description:

Add server groups to an activation key.

Parameters:

- `string sessionKey`
- `string key`
- `array`:
 - `int - serverGroupId`

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: checkConfigDeployment

Description:

Check configuration file deployment status for the activation key specified.

Parameters:

- `string sessionKey`
- `string key`

Returns:

- `1` if enabled, `0` if disabled, exception thrown otherwise.
-

Method: create

Description:

Create a new activation key. The activation key parameter passed in will be prefixed with the organization ID, and this value will be returned from the create call. Eg. If the caller passes in the key "foo" and belong to an organization with the ID 100, the actual activation key will be "100-foo". This call allows for the setting of a usage limit on this activation key. If unlimited usage is desired see the similarly named API method with no usage limit argument.

Parameters:

- string sessionKey
- string key - Leave empty to have new key autogenerated.
- string description
- string baseChannelLabel - Leave empty to accept default.
- int usageLimit - If unlimited usage is desired, use the create API that does not include the parameter.
- array:
 - string - Add-on entitlement label to associate with the key.
 - monitoring_entitled
 - provisioning_entitled
 - virtualization_host
 - virtualization_host_platform
- boolean universalDefault

Returns:

- string - The new activation key.
-

Method: create

Description:

Create a new activation key with unlimited usage. The activation key parameter passed in will be prefixed with the organization ID, and this value will be returned from the create call. Eg. If the caller passes in the key "foo" and belong to an organization with the ID 100, the actual activation key will be "100-foo".

Parameters:

- string sessionKey
- string key - Leave empty to have new key autogenerated.
- string description
- string baseChannelLabel - Leave empty to accept default.
- array:
 - string - Add-on entitlement label to associate with the key.
 - monitoring_entitled
 - provisioning_entitled
 - virtualization_host
 - virtualization_host_platform
- boolean universalDefault

Returns:

- string - The new activation key.

Method: delete

Description:

Delete an activation key.

Parameters:

- string sessionKey
- string key

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: disableConfigDeployment

Description:

Disable configuration file deployment for the specified activation key.

Parameters:

- string sessionKey
- string key

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: enableConfigDeployment

Description:

Enable configuration file deployment for the specified activation key.

Parameters:

- string sessionKey
- string key

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: `getDetails`

Description:

Lookup an activation key's details.

Parameters:

- string `sessionKey`
- string `key`

Returns:

- struct - activation key
 - string "key"
 - string "description"
 - int "usage_limit"
 - string "base_channel_label"
 - array "child_channel_labels"
 - string `childChannelLabel`
 - array "entitlements"
 - string `entitlementLabel`
 - array "server_group_ids"
 - string `serverGroupId`
 - array "package_names"
 - string `packageName` - (deprecated by `packages`)
 - array "packages"
 - struct - package
 - string "name" - `packageName`
 - string "arch" - `archLabel` - optional
 - boolean "universal_default"

Available since: 10.2

Method: `listActivatedSystems`

Description:

List the systems activated with the key provided.

Parameters:

- string `sessionKey`
- string `key`

Returns:

- array:
 -
 - int "id" - System id
 - string "hostname"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: listActivationKeys

Description:

List activation keys that are visible to the user.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - activation key
 - string "key"
 - string "description"
 - int "usage_limit"
 - string "base_channel_label"
 - array "child_channel_labels"
 - string childChannelLabel
 - array "entitlements"
 - string entitlementLabel
 - array "server_group_ids"
 - string serverGroupId
 - array "package_names"
 - string packageName - (deprecated by packages)
 - array "packages"
 - struct - package
 - string "name" - packageName
 - string "arch" - archLabel - optional
 - boolean "universal_default"

Available since: 10.2

Method: listConfigChannels

Description:

List configuration channels associated to an activation key.

Parameters:

- string sessionKey
- string key

Returns:

- array:
 - struct - Configuration Channel information
 - int "id"
 - int "orgId"
 - string "label"
 - string "name"
 - string "description"
 - struct "configChannelType" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"
-

Method: removeChildChannels

Description:

Remove child channels from an activation key.

Parameters:

- string sessionKey
- string key
- array:
 - string - childChannelLabel

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removeConfigChannels

Description:

Remove configuration channels from the given activation keys.

Parameters:

- string sessionKey
- array:
 - string - activationKey
- array:
 - string - configChannelLabel

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removeEntitlements

Description:

Remove entitlements (by label) from an activation key. Currently only add-on entitlements are permitted. (monitoring_entitled, provisioning_entitled, virtualization_host, virtualization_host_platform)

Parameters:

- string sessionKey
- string key
- array:
 - string - entitlement label
 - monitoring_entitled
 - provisioning_entitled
 - virtualization_host
 - virtualization_host_platform

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removePackageNames

Description:

Remove package names from an activation key.

Deprecated as of version - being replaced by `removePackages(string sessionKey, string key, array[packages])`

Parameters:

- string sessionKey
- string key
- array:
 - string - packageName

Returns:

- int - 1 on success, exception thrown otherwise.

Available since: 10.2

Method: removePackages

Description:

Remove package names from an activation key.

Parameters:

- string sessionKey
- string key
- array:
 - struct - packages
 - string "name" - Package name
 - string "arch" - Arch label - Optional

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removeServerGroups

Description:

Remove server groups from an activation key.

Parameters:

- string sessionKey
- string key
- array:
 - int - serverGroupId

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: `setConfigChannels`

Description:

Replace the existing set of configuration channels on the given activation keys. Channels are ranked by their order in the array.

Parameters:

- `string sessionKey`
- `array:`
 - `string - activationKey`
- `array:`
 - `string - configChannelLabel`

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: `setDetails`

Description:

Update the details of an activation key.

Parameters:

- `string sessionKey`
- `string key`
- `struct - activation key`
 - `string "description"`
 - `string "base_channel_label"`
 - `int "usage_limit"`
 - `boolean "universal_default"`

Returns:

- `int - 1` on success, exception thrown otherwise.
-

api

[\(Return to Handler List\)](#)

Available methods

- getVersion
- systemVersion

Description

Methods providing information about the API.

Namespace: api

Method: getVersion

Description:

Returns the version of the API. Since Spacewalk 0.4 (Satellie 5.3) it is no more related to server version.

Parameters:

Returns:

- `string`
-

Method: systemVersion

Description:

Returns the server version.

Parameters:

Returns:

- `string`
-

auth

[\(Return to Handler List\)](#)

Available methods

- login
- login
- logout

Description

This namespace provides methods to authenticate with the system's management server.

Namespace: auth

Method: login

Description:

Login using a username and password. Returns the session key used by most other API methods.

Parameters:

- string username
- string password

Returns:

- `string sessionKey`
-

Method: login

Description:

Login using a username and password. Returns the session key used by other methods.

Parameters:

- string username
- string password
- int duration - Length of session.

Returns:

- `string sessionKey`
-

Method: logout

Description:

Logout the user with the given session key.

Parameters:

- string sessionKey

Returns:

- int - 1 on success, exception thrown otherwise.
-

channel

(Return to Handler List)

Available methods

- listAllChannels
- listMyChannels
- listPopularChannels
- listRedHatChannels
- listRetiredChannels
- listSharedChannels
- listSoftwareChannels

Description

Provides method to get back a list of Software Channels.

Namespace: channel

Method: listAllChannels

Description:

List all software channels that the user's organization is entitled to.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - channel info
 - int "id"
 - string "label"
 - string "name"
 - string "provider_name"
 - int "packages"
 - int "systems"
-

Method: listMyChannels

Description:

List all software channels that belong to the user's organization.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - channel info
 - int "id"
 - string "label"
 - string "name"
 - string "provider_name"
 - int "packages"
 - int "systems"
-

Method: listPopularChannels

Description:

List the most popular software channels. Channels that have at least the number of systems subscribed as specified by the popularity count will be returned.

Parameters:

- string sessionKey
-
- int "popularityCount"

Returns:

- array:
 - struct - channel info
 - int "id"
 - string "label"
 - string "name"
 - string "provider_name"
 - int "packages"
 - int "systems"
-

Method: listRedHatChannels

Description:

List all Red Hat software channels that the user's organization is entitled to.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - channel info
 - int "id"
 - string "label"
 - string "name"
 - string "provider_name"
 - int "packages"
 - int "systems"
-

Method: listRetiredChannels

Description:

List all retired software channels. These are channels that the user's organization is entitled to, but are no longer supported because they have reached their 'end-of-life' date.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - channel info
 - int "id"
 - string "label"

- string "name"
 - string "provider_name"
 - int "packages"
 - int "systems"
-

Method: listSharedChannels

Description:

List all software channels that may be shared by the user's organization.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - channel info
 - int "id"
 - string "label"
 - string "name"
 - string "provider_name"
 - int "packages"
 - int "systems"
-

Method: listSoftwareChannels

Description:

List all visible software channels.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - channel
 - string "label"
 - string "name"
 - string "parent_label"
 - string "end_of_life"
 - string "arch"

channel.access

(Return to Handler List)

Available methods

- disableUserRestrictions
- enableUserRestrictions
- getOrgSharing
- setOrgSharing

Description

Provides methods to retrieve and alter channel access restrictions.

Namespace: channel.access

Method: disableUserRestrictions

Description:

Disable user restrictions for the given channel. If disabled, all users within the organization may subscribe to the channel.

Parameters:

- string sessionKey
- string channelLabel - label of the channel

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: enableUserRestrictions

Description:

Enable user restrictions for the given channel. If enabled, only selected users within the organization may subscribe to the channel.

Parameters:

- string sessionKey
- string channelLabel - label of the channel

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: `getOrgSharing`

Description:

Get organization sharing access control.

Parameters:

- `string sessionKey`
- `string channelLabel` - label of the channel

Returns:

- `string` - The access value (one of the following: `'public'`, `'private'`, or `'protected'`).
-

Method: `setOrgSharing`

Description:

Set organization sharing access control.

Parameters:

- `string sessionKey`
- `string channelLabel` - label of the channel
- `string "access"` - Access (one of the following: `'public'`, `'private'`, or `'protected'`)

Returns:

- `int - 1` on success, exception thrown otherwise.
-

channel.org

(Return to Handler List)

Available methods

- `disableAccess`
- `enableAccess`
- `list`

Description

Provides methods to retrieve and alter organization trust relationships for a channel.

Namespace: channel.org

Method: disableAccess

Description:

Disable access to the channel for the given organization.

Parameters:

- string sessionKey
- string channelLabel - label of the channel
- int orgId - id of org being removed access

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: enableAccess

Description:

Enable access to the channel for the given organization.

Parameters:

- string sessionKey
- string channelLabel - label of the channel
- int orgId - id of org being granted access

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: list

Description:

List the organizations associated with the given channel that may be trusted.

Parameters:

- string sessionKey
- string channelLabel - label of the channel

Returns:

- struct - org
 - int "org_id"
 - string "org_name"
 - boolean "access_enabled"
-

channel.software

(Return to Handler List)

Available methods

- addPackages
- availableEntitlements
- clone
- create
- delete
- getDetails
- getDetails
- isGloballySubscribable
- isUserSubscribable
- listAllPackages
- listAllPackages
- listAllPackages
- listAllPackages
- listAllPackages
- listAllPackagesByDate
- listAllPackagesByDate
- listAllPackagesByDate
- listArches
- listErrata
- listErrata
- listErrata
- listErrata
- listErrata
- listErrata
- listErrataByType
- listLatestPackages
- listPackagesWithoutChannel
- listSubscribedSystems
- listSystemChannels
- mergeErrata
- mergeErrata

- mergePackages
- regenerateNeededCache
- regenerateNeededCache
- removePackages
- setContactDetails
- setGloballySubscribable
- setSystemChannels
- setUserSubscribable
- subscribeSystem

Description

Provides methods to access and modify many aspects of a channel.

Namespace: channel.software

Method: addPackages

Description:

Adds a given list of packages to the given channel.

Parameters:

- string sessionKey
- string channelLabel - target channel.
- array:
 - int - packageId - id of a package to add to the channel.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: availableEntitlements

Description:

Returns the number of available subscriptions for the given channel

Parameters:

- string sessionKey
- string channelLabel - channel to query

Returns:

- int number of available subscriptions for the given channel

Method: clone

Description:

Clone a channel. If `arch_label` is omitted, the arch label of the original channel will be used. If `parent_label` is omitted, the clone will be a base channel.

Parameters:

- string `sessionKey`
- string `original_label`
- struct - channel details
 - string "name"
 - string "label"
 - string "summary"
 - string "parent_label" - (optional)
 - string "arch_label" - (optional)
 - string "gpg_url" - (optional)
 - string "gpg_id" - (optional)
 - string "gpg_fingerprint" - (optional)
 - string "description" - (optional)
- boolean `original_state`

Returns:

- int the cloned channel ID
-

Method: create

Description:

Creates a software channel

Parameters:

- string `sessionKey`
- string `label` - label of the new channel
- string `name` - name of the new channel
- string `summary` - summary of the channel
- string `archLabel` - the label of the architecture the channel corresponds to
 - `channel-ia32` - For 32 bit channel architecture
 - `channel-ia64` - For 64 bit channel architecture
 - `channel-sparc` - For Sparc channel architecture
 - `channel-alpha` - For Alpha channel architecture
 - `channel-s390` - For s390 channel architecture
 - `channel-s390x` - For s390x channel architecture

- channel-iSeries - For i-Series channel architecture
 - channel-pSeries - For p-Series channel architecture
 - channel-x86_64 - For x86_64 channel architecture
 - channel-ppc - For PPC channel architecture
 - channel-sparc-sun-solaris - For Sparc Solaris channel architecture
 - channel-i386-sun-solaris - For i386 Solaris channel architecture
- string parentLabel - label of the parent of this channel, an empty string if it does not have one

Returns:

- int - 1 if the creation operation succeeded, 0 otherwise
-

Method: delete

Description:

Deletes a custom software channel

Parameters:

- string sessionKey
- string channelLabel - channel to delete

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getDetails

Description:

Returns details of the given channel as a map

Parameters:

- string sessionKey
- string channelLabel - channel to query

Returns:

- struct - channel
- int "id"
 - string "name"
 - string "label"
 - string "arch_name"
 - string "summary"
 - string "description"
 - string "maintainer_name"
 - string "maintainer_email"
 - string "maintainer_phone"

- string "support_policy"
 - string "gpg_key_url"
 - string "gpg_key_id"
 - string "gpg_key_fp"
 - string "end_of_life"
 - string "parent_channel_label"
-

Method: getDetails

Description:

Returns details of the given channel as a map

Parameters:

- string sessionKey
- int id - channel to query

Returns:

- struct - channel
 - int "id"
 - string "name"
 - string "label"
 - string "arch_name"
 - string "summary"
 - string "description"
 - string "maintainer_name"
 - string "maintainer_email"
 - string "maintainer_phone"
 - string "support_policy"
 - string "gpg_key_url"
 - string "gpg_key_id"
 - string "gpg_key_fp"
 - string "end_of_life"
 - string "parent_channel_label"
-

Method: isGloballySubscribable

Description:

Returns whether the channel is subscribable by any user in the organization

Parameters:

- string sessionKey
- string channelLabel - channel to query

Returns:

- int - 1 if true, 0 otherwise
-

Method: isUserSubscribable

Description:

Returns whether the channel may be subscribed to by the given user.

Parameters:

- string sessionKey
- string channelLabel - label of the channel
- string login - login of the target user

Returns:

- int - 1 if subscribable, 0 if not
-

Method: listAllPackages

Description:

Lists all packages in the channel, regardless of package version, between the given dates.

Parameters:

- string sessionKey
- string channelLabel - channel to query
- dateTime.iso8601 startDate
- dateTime.iso8601 endDate

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "last_modified_date"
 - string "last_modified" - (Deprecated)
-

Method: listAllPackages

Description:

Lists all packages in the channel, regardless of version whose last modified date is greater than given date.

Parameters:

- string sessionKey
- string channelLabel - channel to query
- dateTime.iso8601 startDate

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "last_modified_date"
 - string "last_modified" - (Deprecated)
-

Method: listAllPackages

Description:

Lists all packages in the channel, regardless of the package version

Parameters:

- string sessionKey
- string channelLabel - channel to query

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "last_modified_date"
 - string "last_modified" - (Deprecated)

Method: listAllPackages

Description:

Lists all packages in the channel, regardless of package version, between the given dates. Example Date: '2008-08-20 08:00:00'

Deprecated as of version - being replaced by listAllPackages(string sessionKey, string channelLabel, dateTime.iso8601 startDate, dateTime.iso8601 endDate)

Parameters:

- string sessionKey
- string channelLabel - channel to query
- string startDate
- string endDate

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "last_modified_date"
 - string "last_modified" - (Deprecated)

Method: listAllPackages

Description:

Lists all packages in the channel, regardless of version whose last modified date is greater than given date. Example Date: '2008-08-20 08:00:00'

Deprecated as of version - being replaced by listAllPackages(string sessionKey, string channelLabel, dateTime.iso8601 startDate)

Parameters:

- string sessionKey
- string channelLabel - channel to query
- string startDate

Returns:

- array:

- struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "last_modified_date"
 - string "last_modified" - (Deprecated)
-

Method: listAllPackagesByDate

Description:

Lists all packages in the channel, regardless of the package version, between the given dates. Example Date: '2008-08-20 08:00:00'

Deprecated as of version - being replaced by listAllPackages(string sessionKey, string channelLabel, dateTime.iso8601 startDate, dateTime.iso8601 endDate)

Parameters:

- string sessionKey
- string channelLabel - channel to query
- string startDate
- string endDate

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - string "id"
 - string "arch_label"
 - string "last_modified"
-

Method: listAllPackagesByDate

Description:

Lists all packages in the channel, regardless of the package version, whose last modified date is greater than given date. Example Date: '2008-08-20 08:00:00'

Deprecated as of version - being replaced by `listAllPackages(string sessionKey, string channelLabel, dateTime.iso8601 startDate)`

Parameters:

- string `sessionKey`
- string `channelLabel` - channel to query
- string `startDate`

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - string "id"
 - string "arch_label"
 - string "last_modified"
-

Method: `listAllPackagesByDate`

Description:

Lists all packages in the channel, regardless of the package version

Deprecated as of version - being replaced by `listAllPackages(string sessionKey, string channelLabel)`

Parameters:

- string `sessionKey`
- string `channelLabel` - channel to query

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - string "id"
 - string "arch_label"
 - string "last_modified"

Method: listArches

Description:

Lists the potential software channel architectures that can be created

Parameters:

- string sessionKey

Returns:

- array:
 - struct - channel arch
 - string "name"
 - string "label"

Method: listErrata

Description:

List the errata applicable to a channel after given startDate

Parameters:

- string sessionKey
- string channelLabel - channel to query
- dateTime.iso8601 startDate

Returns:

- array:
 - struct - errata
 - int "id" - Errata ID.
 - string "date" - Date erratum was created.
 - string "advisory_synopsis" - Summary of the erratum.
 - string "advisory_type" - Type label such as Security, Bug Fix
 - string "advisory_name" - Name such as RHSA, etc

Method: listErrata

Description:

List the errata applicable to a channel between startDate and endDate.

Parameters:

- string sessionKey
- string channelLabel - channel to query
- dateTime.iso8601 startDate

- `dateTime.iso8601 endDate`

Returns:

- `array:`
 - `struct - errata`
 - `int "id"` - Errata ID.
 - `string "date"` - Date erratum was created.
 - `string "advisory_synopsis"` - Summary of the erratum.
 - `string "advisory_type"` - Type label such as Security, Bug Fix
 - `string "advisory_name"` - Name such as RHSA, etc
-

Method: `listErrata`

Description:

List the errata applicable to a channel

Parameters:

- `string sessionKey`
- `string channelLabel` - channel to query

Returns:

- `array:`
 - `struct - errata`
 - `int "id"` - Errata Id
 - `string "date"` - Date erratum was created.
 - `string "advisory_synopsis"` - Summary of the erratum.
 - `string "advisory_type"` - Type label such as Security, Bug Fix
 - `string "advisory_name"` - Name such as RHSA, etc
 - `string "advisory"` - name of the advisory (Deprecated)
 - `string "issue_date"` - date format follows YYYY-MM-DD HH24:MI:SS (Deprecated)
 - `string "update_date"` - date format follows YYYY-MM-DD HH24:MI:SS (Deprecated)
 - `string "synopsis (Deprecated)"`
 - `string "last_modified_date"` - date format follows YYYY-MM-DD HH24:MI:SS (Deprecated)
-

Method: `listErrata`

Description:

List the errata applicable to a channel after given `startDate`

Deprecated as of version - being replaced by listErrata(string sessionKey, string channelLabel, dateTime.iso8601 startDate)

Parameters:

- string sessionKey
- string channelLabel - channel to query
- string startDate

Returns:

- array:
 - struct - errata
 - string "advisory" - name of the advisory
 - string "issue_date" - date format follows YYYY-MM-DD HH24:MI:SS
 - string "update_date" - date format follows YYYY-MM-DD HH24:MI:SS
 - string "synopsis"
 - string "advisory_type"
 - string "last_modified_date" - date format follows YYYY-MM-DD HH24:MI:SS
-

Method: listErrata

Description:

List the errata applicable to a channel between startDate and endDate.

Deprecated as of version - being replaced by listErrata(string sessionKey, string channelLabel, dateTime.iso8601 startDate, dateTime.iso8601)

Parameters:

- string sessionKey
- string channelLabel - channel to query
- string startDate
- string startDate

Returns:

- array:
 - struct - errata
 - string "advisory" - name of the advisory
 - string "issue_date" - date format follows YYYY-MM-DD HH24:MI:SS
 - string "update_date" - date format follows YYYY-MM-DD HH24:MI:SS
 - string "synopsis"
 - string "advisory_type"
 - string "last_modified_date" - date format follows YYYY-MM-DD HH24:MI:SS

Method: listErrataByType

Description:

List the errata of a specific type that are applicable to a channel

Parameters:

- string sessionKey
- string channelLabel - channel to query
- string advisoryType - type of advisory (one of the following: 'Security Advisory', 'Product Enhancement Advisory', 'Bug Fix Advisory')

Returns:

- array:
 - struct - errata
 - string "advisory" - name of the advisory
 - string "issue_date" - date format follows YYYY-MM-DD HH24:MI:SS
 - string "update_date" - date format follows YYYY-MM-DD HH24:MI:SS
 - string "synopsis"
 - string "advisory_type"
 - string "last_modified_date" - date format follows YYYY-MM-DD HH24:MI:SS

Method: listLatestPackages

Description:

Lists the packages with the latest version (including release and epoch) for the given channel

Parameters:

- string sessionKey
- string channelLabel - channel to query

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"

Method: listPackagesWithoutChannel

Description:

Lists all packages that are not associated with a channel. Typically these are custom packages.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "path" - The path on that file system that the package resides
 - string "provider" - The provider of the package, determined by the gpg key it was signed with.
 - dateTime.iso8601 "last_modified"

Method: listSubscribedSystems

Description:

Returns list of subscribed systems for the given channel label

Parameters:

- string sessionKey
- string channelLabel - channel to query

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"

Method: listSystemChannels

Description:

Returns a list of channels that a system is subscribed to for the given system id

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - channel
 - string "label"
 - string "name"
-

Method: mergeErrata

Description:

Merges all errata from one channel into another

Parameters:

- string sessionKey
- string mergeFromLabel - the label of the channel to pull errata from
- string mergeToLabel - the label to push the errata into

Returns:

- array:
 - struct - errata
 - int "id" - Errata Id
 - string "date" - Date erratum was created.
 - string "advisory_type" - Type of the advisory.
 - string "advisory_name" - Name of the advisory.
 - string "advisory_synopsis" - Summary of the erratum.
-

Method: mergeErrata

Description:

Merges all errata from one channel into another based upon a given start/end date.

Parameters:

- string sessionKey
- string mergeFromLabel - the label of the channel to pull errata from
- string mergeToLabel - the label to push the errata into

- string startDate
- string endDate

Returns:

- array:
 - struct - errata
 - int "id" - Errata Id
 - string "date" - Date erratum was created.
 - string "advisory_type" - Type of the advisory.
 - string "advisory_name" - Name of the advisory.
 - string "advisory_synopsis" - Summary of the erratum.
-

Method: mergePackages

Description:

Merges all packages from one channel into another

Parameters:

- string sessionKey
- string mergeFromLabel - the label of the channel to pull packages from
- string mergeToLabel - the label to push the packages into

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "path" - The path on that file system that the package resides
 - string "provider" - The provider of the package, determined by the gpg key it was signed with.
 - dateTime.iso8601 "last_modified"
-

Method: regenerateNeededCache

Description:

Completely clear and regenerate the needed Errata and Package cache for all systems subscribed to the specified channel. This should be used only if you believe your cache is incorrect for all the systems in a given channel.

Parameters:

- string sessionKey
- string channelLabel - the label of the channel

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: regenerateNeededCache

Description:

Completely clear and regenerate the needed Errata and Package cache for all systems subscribed. You must be a Satellite Admin to perform this action.

Parameters:

- string sessionKey

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removePackages

Description:

Removes a given list of packages from the given channel.

Parameters:

- string sessionKey
- string channelLabel - target channel.
- array:
 - int - packageId - id of a package to remove from the channel.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setContactDetails

Description:

Set contact/support information for given channel.

Parameters:

- string sessionKey
- string channelLabel - label of the channel
- string maintainerName - name of the channel maintainer
- string maintainerEmail - email of the channel maintainer
- string maintainerPhone - phone number of the channel maintainer

- string supportPolicy - channel support policy

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setGloballySubscribable

Description:

Set globally subscribable attribute for given channel.

Parameters:

- string sessionKey
- string channelLabel - label of the channel
- boolean subscribable - true if the channel is to be globally subscribable. False otherwise.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setSystemChannels

Description:

Change a systems subscribed channels to the list of channels passed in.

Deprecated as of version - being replaced by system.setBaseChannel(string sessionKey, int serverId, string channelLabel) and system.setChildChannels(string sessionKey, int serverId, array[string channelLabel])

Parameters:

- string sessionKey
- int serverId
- array:
 - string - channelLabel - labels of the channels to subscribe the system to.

Returns:

- int - 1 on success, 0 otherwise
-

Method: setUserSubscribable

Description:

Set the subscribable flag for a given channel and user. If value is set to 'true', this method will give the user subscribe permissions to the channel. Otherwise, that privilege is revoked.

Parameters:

- string sessionKey
- string channelLabel - label of the channel

- string login - login of the target user
- boolean value - value of the flag to set

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: subscribeSystem

Description:

Subscribes a system to a list of channels. If a base channel is included, that is set before setting child channels. When setting child channels the current child channel subscriptions are cleared. To fully unsubscribe the system from all channels, simply provide an empty list of channel labels.

Deprecated as of version - being replaced by `system.setBaseChannel(string sessionKey, int serverId, string channelLabel)` and `system.setChildChannels(string sessionKey, int serverId, array[string channelLabel])`

Parameters:

- string sessionKey
- int serverId
- array:
 - string - label - channel label to subscribe the system to.

Returns:

- int - 1 on success, exception thrown otherwise.
-

configchannel

(Return to Handler List)

Available methods

- channelExists
- create
- createOrUpdatePath
- deleteChannels
- deleteFiles
- getDetails
- getDetails
- listFiles
- listGlobals
- lookupChannelInfo
- lookupFileInfo
- scheduleFileComparisons

- update

Description

Provides methods to access and modify many aspects of configuration channels.

Namespace: configchannel

Method: channelExists

Description:

Check for the existence of the config channel provided.

Parameters:

- string sessionKey
- string channelLabel - Channel to check for.

Returns:

- 1 if exists, 0 otherwise.
-

Method: create

Description:

Create a new global config channel. Caller must be at least a config admin or an organization admin.

Parameters:

- string sessionKey
- string channelLabel
- string channelName
- string channelDescription

Returns:

- struct - Configuration Channel information
 - int "id"
 - int "orgId"
 - string "label"
 - string "name"
 - string "description"
 - struct "configChannelType" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"

Method: createOrUpdatePath

Description:

Create a new file or directory with the given path, or update an existing path.

Parameters:

- string sessionKey
- string configChannelLabel
- string path
- boolean isDir - True if the path is a directory, False if it is a file.
- struct - path info
 - string "contents" - Contents of the file (text or base64 encoded if binary). (ignored for directories)
 - string "owner" - Owner of the file/directory.
 - string "group" - Group name of the file/directory.
 - string "permissions" - Octal file/directory permissions (eg: 644)
 - string "macro-start-delimiter" - Config file macro start delimiter. Use null or empty string to accept the default. (ignored if working with a directory)
 - string "macro-end-delimiter" - Config file macro end delimiter. Use null or empty string to accept the default. (ignored if working with a directory)

Returns:

- struct - Configuration Revision information
 - string "type"
 - file
 - directory
 - string "path" - File Path
 - string "channel" - Channel Name
 - string "contents" - File contents for text files only.
 - int "revision" - File Revision
 - dateTime.iso8601 "creation" - Creation Date
 - dateTime.iso8601 "modified" - Last Modified Date
 - string "owner" - File Owner
 - string "group" - File Group
 - int "permissions" - File Permissions (Deprecated)
 - string "permissions_mode" - File Permissions
 - boolean "binary" - true/false , Present for files only.
 - string "md5" - File's md5 signature. Present for files only.
 - string "macro-start-delimiter" - Macro start delimiter for a config file. Present for files only.
 - string "macro-end-delimiter" - Macro end delimiter for a config file. Present for files only.

Available since: 10.2

Method: deleteChannels

Description:

Delete a list of global config channels. Caller must be a config admin.

Parameters:

- string sessionKey
- array:
 - string - configuration channel labels to delete.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deleteFiles

Description:

Remove file paths from a global channel.

Parameters:

- string sessionKey
- string channelLabel - Channel to remove the files from.
- array:
 - string - file paths to remove.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getDetails

Description:

Lookup config channel details.

Parameters:

- string sessionKey
- string channelLabel

Returns:

- struct - Configuration Channel information

- int "id"
 - int "orgId"
 - string "label"
 - string "name"
 - string "description"
 - struct "configChannelType" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"
-

Method: getDetails

Description:

Lookup config channel details.

Parameters:

- string sessionKey
- int channelId

Returns:

- struct - Configuration Channel information
 - int "id"
 - int "orgId"
 - string "label"
 - string "name"
 - string "description"
 - struct "configChannelType" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"
-

Method: listFiles

Description:

Return a list of files in a channel.

Parameters:

- string sessionKey
- string channelLabel - label of config channel to list files on.

Returns:

- array:
 - struct - Configuration File information
 - string "type"
 - file
 - directory
 - string "path" - File Path
 - dateTime.iso8601 "last_modified" - Last Modified Date
-

Method: listGlobals

Description:

List all the global config channels accessible to the logged-in user.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - Configuration Channel information
 - int "id"
 - int "orgId"
 - string "label"
 - string "name"
 - string "description"
 - string "type"
 - struct "configChannelType" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"
-

Method: lookupChannelInfo

Description:

Lists details on a list channels given their channel labels.

Parameters:

- string sessionKey
- array:

- string - configuration channel label

Returns:

- array:
 - struct - Configuration Channel information
 - int "id"
 - int "orgId"
 - string "label"
 - string "name"
 - string "description"
 - struct "configChannelType" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"
-

Method: lookupFileInfo

Description:

Given a list of paths and a channel, returns details about the latest revisions of the paths.

Parameters:

- string sessionKey
- string channelLabel - label of config channel to lookup on
- array:
 - string - List of paths to examine.

Returns:

- array:
 - struct - Configuration Revision information
 - string "type"
 - file
 - directory
 - string "path" - File Path
 - string "channel" - Channel Name
 - string "contents" - File contents for text files only.
 - int "revision" - File Revision
 - dateTime.iso8601 "creation" - Creation Date
 - dateTime.iso8601 "modified" - Last Modified Date
 - string "owner" - File Owner
 - string "group" - File Group

- int "permissions" - File Permissions (Deprecated)
- string "permissions_mode" - File Permissions
- boolean "binary" - true/false , Present for files only.
- string "md5" - File's md5 signature. Present for files only.
- string "macro-start-delimiter" - Macro start delimiter for a config file. Present for files only.
- string "macro-end-delimiter" - Macro end delimiter for a config file. Present for files only.

Available since: 10.2

Method: scheduleFileComparisons

Description:

Schedule a comparison of the latest revision of a file against the version deployed on a list of systems.

Parameters:

- string sessionKey
- string channelLabel - Label of config channel
- string path - File path
- array:
 - long - The list of server id that the comparison will be performed on

Returns:

- int actionId - The action id of the scheduled action
-

Method: update

Description:

Update a global config channel. Caller must be at least a config admin or an organization admin, or have access to a system containing this config channel.

Parameters:

- string sessionKey
- string channelLabel
- string channelName
- string description

Returns:

- struct - Configuration Channel information
 - int "id"
 - int "orgId"
 - string "label"

- string "name"
 - string "description"
 - struct "configChannelType" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"
-

errata

(Return to Handler List)

Available methods

- addPackages
- applicableToChannels
- bugzillaFixes
- clone
- create
- delete
- findByCve
- getDetails
- listAffectedSystems
- listByDate
- listCves
- listKeywords
- listPackages
- publish
- removePackages
- setDetails

Description

Provides methods to access and modify errata.

Namespace: errata

Method: addPackages

Description:

Add a set of packages to an erratum with the given advisory name. This method will only allow for modification of custom errata created either through the UI or API.

Parameters:

- string sessionKey
- string advisoryName
- array:
 - int - packageId

Returns:

- int - representing the number of packages added, exception otherwise
-

Method: applicableToChannels

Description:

Returns a list of channels applicable to the erratum with the given advisory name.

Parameters:

- string sessionKey
- string advisoryName

Returns:

- array:
 - struct - channel
 - int "channel_id"
 - string "label"
 - string "name"
 - string "parent_channel_label"
-

Method: bugzillaFixes

Description:

Get the Bugzilla fixes for an erratum matching the given advisoryName. The bugs will be returned in a struct where the bug id is the key. i.e. 208144="errata.bugzillaFixes Method Returns different results than docs say"

Parameters:

- string sessionKey
- string advisoryName

Returns:

- struct - Bugzilla info
 - string "bugzilla_id" - actual bug number is the key into the struct
 - string "bug_summary" - summary who's key is the bug id

Method: clone

Description:

Clone a list of errata into the specified channel.

Parameters:

- string sessionKey
- string channel_label
- array:
 - string - advisory - The advisory name of the errata to clone.

Returns:

- array:
 - struct - errata
 - int "id" - Errata Id
 - string "date" - Date erratum was created.
 - string "advisory_type" - Type of the advisory.
 - string "advisory_name" - Name of the advisory.
 - string "advisory_synopsis" - Summary of the erratum.

Method: create

Description:

Create a custom errata. If "publish" is set to true, the errata will be published as well

Parameters:

- string sessionKey
- struct - errata info
 - string "synopsis"
 - string "advisory_name"
 - int "advisory_release"
 - string "advisory_type" - Type of advisory (one of the following: 'Security Advisory', 'Product Enhancement Advisory', or 'Bug Fix Advisory')
 - string "product"
 - string "topic"
 - string "description"
 - string "references"
 - string "notes"
 - string "solution"

- array:
 - struct - bug
 - int "id" - Bug Id
 - string "summary"
- array:
 - string - keyword - List of keywords to associate with the errata.
- array:
 - int - packageId
- boolean publish - Should the errata be published.
- array:
 - string - channelLabel - list of channels the errata should be published too, ignored if publish is set to false

Returns:

- struct - errata
 - int "id" - Errata Id
 - string "date" - Date erratum was created.
 - string "advisory_type" - Type of the advisory.
 - string "advisory_name" - Name of the advisory.
 - string "advisory_synopsis" - Summary of the erratum.
-

Method: delete

Description:

Delete an erratum. This method will only allow for deletion of custom errata created either through the UI or API.

Parameters:

- string sessionKey
- string advisoryName

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: findByCve

Description:

Lookup the details for errata associated with the given CVE (e.g. CVE-2008-3270)

Parameters:

- string sessionKey
- string cveName

Returns:

- array:
 - struct - errata
 - int "id" - Errata Id
 - string "date" - Date erratum was created.
 - string "advisory_type" - Type of the advisory.
 - string "advisory_name" - Name of the advisory.
 - string "advisory_synopsis" - Summary of the erratum.
-

Method: getDetails

Description:

Retrieves the details for the erratum matching the given advisory name.

Parameters:

- string sessionKey
- string advisoryName

Returns:

- struct - erratum
 - string "issue_date"
 - string "update_date"
 - string "last_modified_date" - This date is only included for published erratum and it represents the last time the erratum was modified.
 - string "synopsis"
 - int "release"
 - string "type"
 - string "product"
 - string "topic"
 - string "description"
 - string "references"
 - string "notes"
 - string "solution"

Method: listAffectedSystems

Description:

Return the list of systems affected by the erratum with advisory name.

Parameters:

- string sessionKey
- string advisoryName

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in

Method: listByDate

Description:

List errata that have been applied to a particular channel by date.

Deprecated as of version - being replaced by `channel.software.listErrata(string sessionKey, string channelLabel)`

Parameters:

- string sessionKey
- string channelLabel

Returns:

- array:
 - struct - errata
 - int "id" - Errata Id
 - string "date" - Date erratum was created.
 - string "advisory_type" - Type of the advisory.
 - string "advisory_name" - Name of the advisory.
 - string "advisory_synopsis" - Summary of the erratum.

Method: listCves

Description:

Returns a list of CVEs applicable to the erratum with the given advisory name.

Parameters:

- string sessionKey
- string advisoryName

Returns:

- array:
 - string - cveName
-

Method: listKeywords

Description:

Get the keywords associated with an erratum matching the given advisory name.

Parameters:

- string sessionKey
- string advisoryName

Returns:

- array:
 - string - Keyword associated with erratum.
-

Method: listPackages

Description:

Returns a list of the packages affected by the erratum with the given advisory name.

Parameters:

- string sessionKey
- string advisoryName

Returns:

- array:
 - struct - package
 - int "id"
 - string "name"
 - string "epoch"
 - string "version"
 - string "release"

- string "arch_label"
 - array "providing_channels"
 - string - Channel label providing this package.
 - string "build_host"
 - string "description"
 - string "md5sum"
 - string "vendor"
 - string "summary"
 - string "cookie"
 - string "license"
 - string "file"
 - string "path"
 - string "build_date"
 - string "last_modified_date"
 - string "size"
 - string "payload_size"
-

Method: publish

Description:

Publish an existing (unpublished) errata to a set of channels.

Parameters:

- string sessionKey
- string advisoryName
- array:
 - string - channelLabel - list of channel labels to publish to

Returns:

- struct - errata
 - int "id" - Errata Id
 - string "date" - Date erratum was created.
 - string "advisory_type" - Type of the advisory.
 - string "advisory_name" - Name of the advisory.
 - string "advisory_synopsis" - Summary of the erratum.
-

Method: removePackages

Description:

Remove a set of packages from an erratum with the given advisory name. This method will only allow for modification of custom errata created either through the UI or API.

Parameters:

- string sessionKey
- string advisoryName
- array:
 - int - packageId

Returns:

- int - representing the number of packages removed, exception otherwise
-

Method: setDetails

Description:

Set erratum details. All arguments are optional and will only be modified if included in the struct. This method will only allow for modification of custom errata created either through the UI or API.

Parameters:

- string sessionKey
- string advisoryName
- struct - errata details
 - string "synopsis"
 - string "advisory_name"
 - int "advisory_release"
 - string "advisory_type" - Type of advisory (one of the following: 'Security Advisory', 'Product Enhancement Advisory', or 'Bug Fix Advisory')
 - string "product"
 - string "topic"
 - string "description"
 - string "references"
 - string "notes"
 - string "solution"
 - array "bugs" - 'bugs' is the key into the struct array:
 - struct - bug
 - int "id" - Bug Id
 - string "summary"
 - array "keywords" - 'keywords' is the key into the struct array:
 - string - keyword - List of keywords to associate with the errata.

Returns:

- int - 1 on success, exception thrown otherwise.
-

kickstart

(Return to Handler List)

Available methods

- createProfile
- createProfileWithCustomUrl
- deleteProfile
- findKickstartForIp
- importFile
- importFile
- importRawFile
- listAllIpRanges
- listKickstartableChannels
- listKickstartableTrees
- listKickstarts
- renameProfile

Description

Provides methods to create kickstart files

Namespace: kickstart

Method: createProfile

Description:

Import a kickstart profile into RHN.

Parameters:

- string sessionKey
- string profileLabel - Label for the new kickstart profile.
- string virtualizationType - none, para_host, qemu, xenfv or xenpv.
- string kickstartableTreeLabel - Label of a kickstartable tree to associate the new profile with.
- string kickstartHost - Kickstart hostname (of a satellite or proxy) used to construct the default download URL for the new kickstart profile.
- string rootPassword - Root password.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: createProfileWithCustomUrl

Description:

Import a kickstart profile into RHN.

Parameters:

- string sessionKey
- string profileLabel - Label for the new kickstart profile.
- string virtualizationType - none, para_host, qemu, xenfv or xenpv.
- string kickstartableTreeLabel - Label of a kickstartable tree to associate the new profile with.
- boolean downloadUrl - Download URL, or 'default' to use the kickstart tree's default URL.
- string rootPassword - Root password.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deleteProfile

Description:

Delete a kickstart profile

Parameters:

- string sessionKey
- string ksLabel - The label of the kickstart profile you want to remove

Returns:

- int - 1 on successful deletion, 0 if kickstart wasn't found or couldn't be deleted.
-

Method: findKickstartForIp

Description:

Find an associated kickstart for a given ip address.

Parameters:

- string sessionKey
- string ipAddress - The ip address to search for (i.e. 192.168.0.1)

Returns:

- string - label of the kickstart. Empty string (") if not found.
-

Method: importFile

Description:

Import a kickstart profile into RHN.

Parameters:

- string sessionKey
- string profileLabel - Label for the new kickstart profile.
- string virtualizationType - none, para_host, qemu, xenvf or xenpv.
- string kickstartableTreeLabel - Label of a kickstartable tree to associate the new profile with.
- string kickstartFileContents - Contents of the kickstart file to import.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: importFile

Description:

Import a kickstart profile into RHN.

Parameters:

- string sessionKey
- string profileLabel - Label for the new kickstart profile.
- string virtualizationType - none, para_host, qemu, xenvf or xenpv.
- string kickstartableTreeLabel - Label of a kickstartable tree to associate the new profile with.
- string kickstartHost - Kickstart hostname (of a satellite or proxy) used to construct the default download URL for the new kickstart profile. Using this option signifies that this default URL will be used instead of any url/nfs/cdrom/harddrive commands in the kickstart file itself.
- string kickstartFileContents - Contents of the kickstart file to import.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: importRawFile

Description:

Import a raw kickstart file into satellite.

Parameters:

- string sessionKey
- string profileLabel - Label for the new kickstart profile.
- string virtualizationType - none, para_host, qemu, xenvf or xenpv.
- string kickstartableTreeLabel - Label of a kickstartable tree to associate the new profile with.
- string kickstartFileContents - Contents of the kickstart file to import.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: listAllIpRanges

Description:

List all Ip Ranges and their associated kickstarts available in the user's org.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - Kickstart Ip Range
 - string "ksLabel" - The kickstart label associated with the ip range
 - string "max" - The max ip of the range
 - string "min" - The min ip of the range
-

Method: listKickstartableChannels

Description:

List kickstartable channels for the logged in user.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - channel
 - int "id"
 - string "name"
 - string "label"
 - string "arch_name"
 - string "summary"
 - string "description"
 - string "maintainer_name"
 - string "maintainer_email"
 - string "maintainer_phone"
 - string "support_policy"
 - string "gpg_key_url"
 - string "gpg_key_id"
 - string "gpg_key_fp"

- string "end_of_life"
 - string "parent_channel_label"
-

Method: listKickstartableTrees

Description:

List the available kickstartable trees for the given channel.

Deprecated as of version - being replaced by `kickstart.tree.list(string sessionKey, string channelLabel)`

Parameters:

- string sessionKey
- string channelLabel - Label of channel to search.

Returns:

- array:
 - struct - kickstartable tree
 - int "id"
 - string "label"
 - string "base_path"
 - int "channel_id"
-

Method: listKickstarts

Description:

Provides a list of kickstart profiles visible to the user's org

Parameters:

- string sessionKey

Returns:

- array:
 - struct - kickstart
 - string "label"
 - string "tree_label"
 - string "name"
 - boolean "advanced_mode"
 - boolean "org_default"
 - boolean "active"
-

Method: renameProfile

Description:

Rename a Kickstart Tree (Distribution) in Satellite

Parameters:

- string sessionKey
- string originalLabel - Label for the kickstart tree you want to rename
- string newLabel - new label to change too

Returns:

- int - 1 on success, exception thrown otherwise.
-

kickstart.filepreservation

(Return to Handler List)

Available methods

- create
- delete
- getDetails
- listAllFilePreservations

Description

Provides methods to retrieve and manipulate kickstart file preservation lists.

Namespace: kickstart.filepreservation

Method: create

Description:

Create a new file preservation list.

Parameters:

- string session_key
- string name - name of the file list to create
- array:
 - string - name - file names to include

Returns:

- int - 1 on success, exception thrown otherwise.

Method: delete

Description:

Delete a file preservation list.

Parameters:

- string session_key
- string name - name of the file list to delete

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getDetails

Description:

Returns all of the data associated with the given file preservation list.

Parameters:

- string session_key
- string name - name of the file list to retrieve details for

Returns:

- struct - file list
 - string "name"
 - array "file_names"
 - string name
-

Method: listAllFilePreservations

Description:

List all file preservation lists for the organization associated with the user logged into the given session

Parameters:

- string sessionKey

Returns:

- array:
 - struct - file preservation
 - int "id"
 - string "name"
 - dateTime.iso8601 "created"
 - dateTime.iso8601 "last_modified"

kickstart.keys

(Return to Handler List)

Available methods

- create
- delete
- getDetails
- listAllKeys

Description

Provides methods to manipulate kickstart keys.

Namespace: kickstart.keys

Method: create

Description:

creates a new key with the given parameters

Parameters:

- string session_key
- string description
- string type - valid values are GPG or SSL
- string content

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: delete

Description:

deletes the key identified by the given parameters

Parameters:

- string session_key
- string description

Returns:

- int - 1 on success, exception thrown otherwise.

Method: getDetails

Description:

returns all of the data associated with the given key

Parameters:

- string session_key
- string description

Returns:

- struct - key
 - string "description"
 - string "type"
 - string "content"

Method: listAllKeys

Description:

list all keys for the org associated with the user logged into the given session

Parameters:

- string sessionKey

Returns:

- array:
 - struct - key
 - string "description"
 - string "type"

kickstart.profile

(Return to Handler List)

Available methods

- addIpRange
- addScript
- compareActivationKeys
- compareAdvancedOptions
- comparePackages
- downloadKickstart
- getAdvancedOptions

- getChildChannels
- getCustomOptions
- getKickstartTree
- listIpRanges
- listScripts
- removeIpRange
- removeScript
- setAdvancedOptions
- setChildChannels
- setCustomOptions
- setKickstartTree

Description

Provides methods to access and modify many aspects of a kickstart profile.

Namespace: kickstart.profile

Method: addIpRange

Description:

Add an ip range to a kickstart profile.

Parameters:

- string sessionKey
- string label - The label of the kickstart
- string min - The ip address making up the minimum of the range (i.e. 192.168.0.1)
- string max - The ip address making up the maximum of the range (i.e. 192.168.0.254)

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addScript

Description:

Add a pre/post script to a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - The kickstart label to add the script to.
- string contents - The full script to add.
- string interpreter - The path to the interpreter to use (i.e. /bin/bash). An empty string will use the kickstart default interpreter.
- string type - The type of script (either 'pre' or 'post').

- boolean chroot - Whether to run the script in the chrooted install location (recommended) or not.

Returns:

- int id - the id of the added script
-

Method: compareActivationKeys

Description:

Returns a list for each kickstart profile; each list will contain activation keys not present on the other profile.

Parameters:

- string sessionKey
- string kickstartLabel1
- string kickstartLabel2

Returns:

- struct - Comparison Info
 - array "kickstartLabel1" - Actual label of the first kickstart profile is the key into the struct array:
 - struct - activation key
 - string "key"
 - string "description"
 - int "usage_limit"
 - string "base_channel_label"
 - array "child_channel_labels"
 - string childChannelLabel
 - array "entitlements"
 - string entitlementLabel
 - array "server_group_ids"
 - string serverGroupId
 - array "package_names"
 - string packageName - (deprecated by packages)
 - array "packages"
 - struct - package
 - string "name" - packageName
 - string "arch" - archLabel - optional
 - boolean "universal_default"
 - array "kickstartLabel2" - Actual label of the second kickstart profile is the key into the struct array:
 - struct - activation key
 - string "key"
 - string "description"

- int "usage_limit"
 - string "base_channel_label"
 - array "child_channel_labels"
 - string childChannelLabel
 - array "entitlements"
 - string entitlementLabel
 - array "server_group_ids"
 - string serverGroupId
 - array "package_names"
 - string packageName - (deprecated by packages)
 - array "packages"
 - struct - package
 - string "name" - packageName
 - string "arch" - archLabel - optional
 - boolean "universal_default"
-

Method: compareAdvancedOptions

Description:

Returns a list for each kickstart profile; each list will contain the properties that differ between the profiles and their values for that specific profile .

Parameters:

- string sessionKey
- string kickstartLabel1
- string kickstartLabel2

Returns:

- struct - Comparison Info
 - array "kickstartLabel1" - Actual label of the first kickstart profile is the key into the struct array:
 - struct - value
 - string "name"
 - string "value"
 - boolean "enabled"
 - array "kickstartLabel2" - Actual label of the second kickstart profile is the key into the struct array:
 - struct - value
 - string "name"
 - string "value"
 - boolean "enabled"

Method: comparePackages

Description:

Returns a list for each kickstart profile; each list will contain package names not present on the other profile.

Parameters:

- string sessionKey
- string kickstartLabel1
- string kickstartLabel2

Returns:

- struct - Comparison Info
 - array "kickstartLabel1" - Actual label of the first kickstart profile is the key into the struct array:
 - string "package name"
 - array "kickstartLabel2" - Actual label of the second kickstart profile is the key into the struct array:
 - string "package name"

Method: downloadKickstart

Description:

Download the full contents of a kickstart file.

Parameters:

- string sessionKey
- string ksLabel - The label of the kickstart to download.
- string host - The host to use when referring to the satellite itself (Usually this should be the FQDN of the satellite, but could be the ip address or shortname of it as well).

Returns:

- string - The contents of the kickstart file. Note: if an activation key is not associated with the kickstart file, registration will not occur in the satellite generated %post section. If one is associated, it will be used for registration.
-

Method: getAdvancedOptions

Description:

Get advanced options for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - Label of kickstart profile to be changed.

Returns:

- array:
 - struct - option
 - string "name"
 - string "arguments"
-

Method: getChildChannels

Description:

Get the child channels for a kickstart profile.

Parameters:

- string sessionKey
- string kslabel - Label of kickstart profile.

Returns:

- array:
 - string - channelLabel
-

Method: getCustomOptions

Description:

Get custom options for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel

Returns:

- array:
 - struct - option
 - int "id"
 - string "arguments"

Method: getKickstartTree

Description:

Get the kickstart tree for a kickstart profile.

Parameters:

- string sessionKey
- string kslabel - Label of kickstart profile to be changed.

Returns:

- - string "kstreeLabel" - Label of the kickstart tree.
-

Method: listIpRanges

Description:

List all ip ranges for a kickstart profile.

Parameters:

- string sessionKey
- string label - The label of the kickstart

Returns:

- array:
 - struct - Kickstart Ip Range
 - string "ksLabel" - The kickstart label associated with the ip range
 - string "max" - The max ip of the range
 - string "min" - The min ip of the range
-

Method: listScripts

Description:

List the pre and post scripts for a kickstart profile. profile

Parameters:

- string sessionKey
- string ksLabel - The label of the kickstart

Returns:

- array:

- struct - kickstart script
 - int "id"
 - string "contents"
 - string "script_type" - Which type of script ('pre' or 'post').
 - string "interpreter" - The scripting language interpreter to use for this script. An empty string indicates the default kickstart shell.
 - boolean "chroot" - True if the script will be executed within the chroot environment.
-

Method: removeIpRange

Description:

Remove an ip range from a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - The kickstart label of the ip range you want to remove
- string ip_address - An Ip Address that falls within the range that you are wanting to remove. The min or max of the range will work.

Returns:

- int - 1 on successful removal, 0 if range wasn't found for the specified kickstart, exception otherwise.
-

Method: removeScript

Description:

Remove a script from a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - The kickstart from which to remove the script from.
- int scriptId - The id of the script to remove.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setAdvancedOptions

Description:

Set advanced options for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel
- array:
 - struct - advanced options
 - string "name" - Name of the advanced option. Valid Option names: autostep, interactive, install, upgrade, text, network, cdrom, harddrive, nfs, url, lang, langsupport keyboard, mouse, device, deviceprobe, zerombr, clearpart, bootloader, timezone, auth, rootpw, selinux, reboot, firewall, xconfig, skipx, key, ignoredisk, autopart, cmdline, firstboot, graphical, iscsi, iscsiname, logging, monitor, multipath, poweroff, halt, service, shutdown, user, vnc, zfc
 - string "arguments" - Arguments of the option

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setChildChannels

Description:

Set the child channels for a kickstart profile.

Parameters:

- string sessionKey
- string kslabel - Label of kickstart profile to be changed.
- string[] channelLabels - List of labels of child channels

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setCustomOptions

Description:

Set custom options for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel
- string[] options

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setKickstartTree

Description:

Set the kickstart tree for a kickstart profile.

Parameters:

- string sessionKey
- string kslabel - Label of kickstart profile to be changed.
- string kstreeLabel - Label of new kickstart tree.

Returns:

- int - 1 on success, exception thrown otherwise.
-

kickstart.profile.keys

(Return to Handler List)

Available methods

- addActivationKey
- getActivationKeys
- removeActivationKey

Description

Provides methods to access and modify the list of activation keys associated with a kickstart profile.

Namespace: kickstart.profile.keys

Method: addActivationKey

Description:

Add an activation key association to the kickstart profile

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label
- string key - the activation key

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getActivationKeys

Description:

Lookup the activation keys associated with the kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

- array:
 - struct - activation key
 - string "key"
 - string "description"
 - int "usage_limit"
 - string "base_channel_label"
 - array "child_channel_labels"
 - string childChannelLabel
 - array "entitlements"
 - string entitlementLabel
 - array "server_group_ids"
 - string serverGroupId
 - array "package_names"
 - string packageName - (deprecated by packages)
 - array "packages"
 - struct - package
 - string "name" - packageName
 - string "arch" - archLabel - optional
 - boolean "universal_default"
-

Method: removeActivationKey

Description:

Remove an activation key association from the kickstart profile

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label
- string key - the activation key

Returns:

- int - 1 on success, exception thrown otherwise.

kickstart.profile.software

(Return to Handler List)

Available methods

- appendToSoftwareList
- getSoftwareList
- setSoftwareList

Description

Provides methods to access and modify the software list associated with a kickstart profile.

Namespace: kickstart.profile.software

Method: appendToSoftwareList

Description:

Append the list of software packages to a kickstart profile. Duplicate packages will be ignored.

Parameters:

- string sessionKey
- string ksLabel - The label of a kickstart profile.
- string[] packageList - A list of package names to be added to the profile.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getSoftwareList

Description:

Get a list of a kickstart profile's software packages.

Parameters:

- string sessionKey
- string ksLabel - The label of a kickstart profile.

Returns:

- string[] - Get a list of a kickstart profile's software packages.
-

Method: setSoftwareList

Description:

Set the list of software packages for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - The label of a kickstart profile.
- string[] packageList - A list of package names to be set on the profile.

Returns:

- int - 1 on success, exception thrown otherwise.
-

kickstart.profile.system

(Return to Handler List)

Available methods

- addFilePreservations
- addKeys
- checkConfigManagement
- checkRemoteCommands
- disableConfigManagement
- disableRemoteCommands
- enableConfigManagement
- enableRemoteCommands
- getLocale
- getPartitioningScheme
- getSELinux
- listFilePreservations
- listKeys
- removeFilePreservations
- removeKeys
- setLocale
- setPartitioningScheme
- setSELinux

Description

Provides methods to set various properties of a kickstart profile.

Namespace: kickstart.profile.system

Method: addFilePreservations

Description:

Adds the given list of file preservations to the specified kickstart profile.

Parameters:

- string sessionKey
- string kickstartLabel
- array:
 - string - filePreservations

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addKeys

Description:

Adds the given list of keys to the specified kickstart profile.

Parameters:

- string sessionKey
- string kickstartLabel
- array:
 - string - keyDescription

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: checkConfigManagement

Description:

Check the configuration management status for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

-
- boolean "enabled" - true if configuration management is enabled; otherwise, false

Method: checkRemoteCommands

Description:

Check the remote commands status flag for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

- - boolean "enabled" - true if remote commands support is enabled; otherwise, false
-

Method: disableConfigManagement

Description:

Disables the configuration management flag in a kickstart profile so that a system created using this profile will be NOT be configuration capable.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: disableRemoteCommands

Description:

Disables the remote command flag in a kickstart profile so that a system created using this profile will be capable of running remote commands

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: enableConfigManagement

Description:

Enables the configuration management flag in a kickstart profile so that a system created using this profile will be configuration capable.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: enableRemoteCommands

Description:

Enables the remote command flag in a kickstart profile so that a system created using this profile will be capable of running remote commands

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getLocale

Description:

Retrieves the locale for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

- struct - locale info
 - string "locale"
 - boolean "useUtc"
 - true - the hardware clock uses UTC
 - false - the hardware clock does not use UTC

Method: getPartitioningScheme

Description:

Get the partitioning scheme for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - The label of a kickstart profile.

Returns:

- string[] - A list of partitioning commands used to setup the partitions, logical volumes and volume groups."
-

Method: getSELinux

Description:

Retrieves the SELinux enforcing mode property of a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

Returns:

- string enforcingMode
 - enforcing
 - permissive
 - disabled
-

Method: listFilePreservations

Description:

Returns the set of all file preservations associated with the given kickstart profile.

Parameters:

- string sessionKey
- string kickstartLabel

Returns:

- array:
 - struct - file list
 - string "name"
 - array "file_names"
 - string name

Method: listKeys

Description:

Returns the set of all keys associated with the given kickstart profile.

Parameters:

- string sessionKey
- string kickstartLabel

Returns:

- array:
 - struct - key
 - string "description"
 - string "type"
 - string "content"

Method: removeFilePreservations

Description:

Removes the given list of file preservations from the specified kickstart profile.

Parameters:

- string sessionKey
- string kickstartLabel
- array:

- string - filePreservations

Returns:

- int - 1 on success, exception thrown otherwise.

Method: removeKeys

Description:

Removes the given list of keys from the specified kickstart profile.

Parameters:

- string sessionKey
- string kickstartLabel
- array:

- string - keyDescription

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setLocale

Description:

Sets the locale for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label
- string locale - the locale
- boolean useUtc
 - true - the hardware clock uses UTC
 - false - the hardware clock does not use UTC

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setPartitioningScheme

Description:

Set the partitioning scheme for a kickstart profile.

Parameters:

- string sessionKey
- string ksLabel - The label of the kickstart profile to update.
- string[] scheme - The partitioning scheme is a list of partitioning command strings used to setup the partitions, volume groups and logical volumes.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setSELinux

Description:

Sets the SELinux enforcing mode property of a kickstart profile so that a system created using this profile will be have the appropriate SELinux enforcing mode.

Parameters:

- string sessionKey
- string ksLabel - the kickstart profile label

- string enforcingMode - the selinux enforcing mode
 - enforcing
 - permissive
 - disabled

Returns:

- int - 1 on success, exception thrown otherwise.
-

kickstart.tree

(Return to Handler List)

Available methods

- create
- delete
- deleteTreeAndProfiles
- list
- listInstallTypes
- rename
- update

Description

Provides methods to access and modify the kickstart trees.

Namespace: kickstart.tree

Method: create

Description:

Create a Kickstart Tree (Distribution) in Satellite.

Parameters:

- string sessionKey
- string treeLabel - The new kickstart tree label.
- string basePath - Path to the base or root of the kickstart tree.
- string channelLabel - Label of channel to associate with the kickstart tree.
- string installType - Label for KickstartInstallType (rhel_2.1, rhel_3, rhel_4, rhel_5, fedora_9).

Returns:

- int - 1 on success, exception thrown otherwise.

Method: delete

Description:

Delete a Kickstart Tree (Distribution) in Satellite.

Parameters:

- string sessionKey
- string treeLabel - Label for the kickstart tree to delete.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deleteTreeAndProfiles

Description:

Delete a kickstarttree and any profiles associated with this kickstart tree. **WARNING:** This will delete all profiles associated with this kickstart tree!

Parameters:

- string sessionKey
- string treeLabel - Label for the kickstart tree to delete.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: list

Description:

List the available kickstartable trees for the given channel.

Parameters:

- string sessionKey
- string channelLabel - Label of channel to search.

Returns:

- array:
 - struct - kickstartable tree
 - int "id"
 - string "label"
 - string "base_path"
 - int "channel_id"

Method: listInstallTypes

Description:

List the available kickstartable install types (rhel2,3,4,5 and fedora9+).

Parameters:

- string sessionKey

Returns:

- array:
 - struct - kickstart install type
 - int "id"
 - string "label"
 - string "name"

Method: rename

Description:

Rename a Kickstart Tree (Distribution) in Satellite.

Parameters:

- string sessionKey
- string originalLabel - Label for the kickstart tree to rename.
- string newLabel - The kickstart tree's new label.

Returns:

- int - 1 on success, exception thrown otherwise.

Method: update

Description:

Edit a Kickstart Tree (Distribution) in Satellite.

Parameters:

- string sessionKey
- string treeLabel - Label for the kickstart tree.
- string basePath - Path to the base or root of the kickstart tree.
- string channelLabel - Label of channel to associate with kickstart tree.
- string installType - Label for KickstartInstallType (rhel_2.1, rhel_3, rhel_4, rhel_5, fedora_9).

Returns:

- int - 1 on success, exception thrown otherwise.

org

(Return to Handler List)

Available methods

- create
- delete
- getDetails
- getDetails
- listOrgs
- listSoftwareEntitlements
- listSoftwareEntitlements
- listSoftwareEntitlements
- listSoftwareEntitlementsForOrg
- listSystemEntitlements
- listSystemEntitlements
- listSystemEntitlements
- listSystemEntitlementsForOrg
- listUsers
- migrateSystems
- setSoftwareEntitlements
- setSystemEntitlements
- updateName

Description

Contains methods to access common organization management functions available from the web interface.

Namespace: org

Method: create

Description:

Create a new organization and associated administrator account.

Parameters:

- string sessionKey
- string orgName - Organization name. Must meet same criteria as in the web UI.
- string adminLogin - New administrator login name.
- string adminPassword - New administrator password.
- string prefix - New administrator's prefix. Must match one of the values available in the web UI. (i.e. Dr., Mr., Mrs., Sr., etc.)

- string firstName - New administrator's first name.
- string lastName - New administrator's first name.
- string email - New administrator's e-mail.
- boolean usePamAuth - True if PAM authentication should be used for the new administrator account.

Returns:

- struct - organization info
 - int "id"
 - string "name"
 - int "active_users" - Number of active users in the organization.
 - int "systems" - Number of systems in the organization.
 - int "trusts" - Number of trusted organizations.
 - int "system_groups" - Number of system groups in the organization.
 - int "activation_keys" - Number of activation keys in the organization. (optional)
 - int "kickstart_profiles" - Number of kickstart profiles in the organization. (optional)
 - int "configuration_channels" - Number of configuration channels in the organization. (optional)
-

Method: delete

Description:

Delete an organization. The default organization (i.e. orgId=1) cannot be deleted.

Parameters:

- string sessionKey
- int orgId

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getDetails

Description:

The detailed information about an organization given the organization ID.

Parameters:

- string sessionKey
- int orgId

Returns:

- struct - organization info

- int "id"
 - string "name"
 - int "active_users" - Number of active users in the organization.
 - int "systems" - Number of systems in the organization.
 - int "trusts" - Number of trusted organizations.
 - int "system_groups" - Number of system groups in the organization.
 - int "activation_keys" - Number of activation keys in the organization. (optional)
 - int "kickstart_profiles" - Number of kickstart profiles in the organization. (optional)
 - int "configuration_channels" - Number of configuration channels in the organization. (optional)
-

Method: getDetails

Description:

The detailed information about an organization given the organization name.

Parameters:

- string sessionKey
- string name

Returns:

- struct - organization info
 - int "id"
 - string "name"
 - int "active_users" - Number of active users in the organization.
 - int "systems" - Number of systems in the organization.
 - int "trusts" - Number of trusted organizations.
 - int "system_groups" - Number of system groups in the organization.
 - int "activation_keys" - Number of activation keys in the organization. (optional)
 - int "kickstart_profiles" - Number of kickstart profiles in the organization. (optional)
 - int "configuration_channels" - Number of configuration channels in the organization. (optional)
-

Method: listOrgs

Description:

Returns the list of organizations.

Parameters:

- string sessionKey

Returns:

- struct - organization info
 - int "id"
 - string "name"
 - int "active_users" - Number of active users in the organization.
 - int "systems" - Number of systems in the organization.
 - int "trusts" - Number of trusted organizations.
 - int "system_groups" - Number of system groups in the organization.
 - int "activation_keys" - Number of activation keys in the organization. (optional)
 - int "kickstart_profiles" - Number of kickstart profiles in the organization. (optional)
 - int "configuration_channels" - Number of configuration channels in the organization. (optional)
-

Method: listSoftwareEntitlements

Description:

List software entitlement allocation information across all organizations. Caller must be a satellite administrator.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - entitlement usage
 - string "label"
 - string "name"
 - int "free"
 - int "used"
 - int "allocated"
 - int "unallocated"
-

Method: listSoftwareEntitlements

Description:

List each organization's allocation of a given software entitlement. Organizations with no allocation will not be present in the list. A value of -1 indicates unlimited entitlements.

Deprecated as of version - being replaced by `listSoftwareEntitlements(string sessionKey, string label, boolean includeUnentitled)`

Parameters:

- string sessionKey
- string label - Software entitlement label.

Returns:

- array:
 - struct - entitlement usage
 - int "org_id"
 - string "org_name"
 - int "allocated"
 - int "unallocated"
 - int "used"
 - int "free"
-

Method: listSoftwareEntitlements

Description:

List each organization's allocation of a given software entitlement. A value of -1 indicates unlimited entitlements.

Parameters:

- string sessionKey
- string label - Software entitlement label.
- boolean includeUnentitled - If true, the result will include both organizations that have the entitlement as well as those that do not; otherwise, the result will only include organizations that have the entitlement.

Returns:

- array:
 - struct - entitlement usage
 - int "org_id"
 - string "org_name"
 - int "allocated"
 - int "unallocated"
 - int "used"
 - int "free"

Available since: 10.4

Method: listSoftwareEntitlementsForOrg

Description:

List an organization's allocation of each software entitlement. A value of -1 indicates unlimited entitlements.

Parameters:

- string sessionKey
- int orgId

Returns:

- array:
 - struct - entitlement usage
 - string "label"
 - string "name"
 - int "allocated"
 - int "unallocated"
 - int "free"
 - int "used"

Method: listSystemEntitlements

Description:

Lists system entitlement allocation information across all organizations. Caller must be a satellite administrator.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - entitlement usage
 - string "label"
 - string "name"
 - int "free"
 - int "used"
 - int "allocated"
 - int "unallocated"
-

Method: listSystemEntitlements

Description:

List each organization's allocation of a system entitlement. If the organization has no allocation for a particular entitlement, it will not appear in the list.

Deprecated as of version - being replaced by listSystemEntitlements(string sessionKey, string label, boolean includeUnentitled)

Parameters:

- string sessionKey
- string label

Returns:

- array:
 - struct - entitlement usage
 - int "org_id"
 - string "org_name"
 - int "allocated"
 - int "unallocated"
 - int "used"
 - int "free"
-

Method: listSystemEntitlements

Description:

List each organization's allocation of a system entitlement.

Parameters:

- string sessionKey
- string label
- boolean includeUnentitled - If true, the result will include both organizations that have the entitlement as well as those that do not; otherwise, the result will only include organizations that have the entitlement.

Returns:

- array:
 - struct - entitlement usage
 - int "org_id"
 - string "org_name"
 - int "allocated"
 - int "unallocated"
 - int "used"
 - int "free"

Available since: 10.4

Method: listSystemEntitlementsForOrg

Description:

List an organization's allocation of each system entitlement.

Parameters:

- string sessionKey
- int orgId

Returns:

- array:
 - struct - entitlement usage
 - string "label"
 - string "name"
 - int "free"
 - int "used"
 - int "allocated"
 - int "unallocated"

Method: listUsers

Description:

Returns the list of users in a given organization.

Parameters:

- string sessionKey
- int orgId

Returns:

- array:
 - struct - user
 - string "login"
 - string "login_uc"
 - string "name"
 - string "email"
 - boolean "is_org_admin"

Method: migrateSystems

Description:

Migrate systems from one organization to another. If executed by a Satellite administrator, the systems will be migrated from their current organization to the organization specified by the toOrgId. If executed by an organization administrator, the systems must exist in the same organization as that administrator and the systems will be migrated to the organization specified by the toOrgId. In any scenario, the origination and destination organizations must be defined in a trust.

Parameters:

- string sessionKey
- int toOrgId - ID of the organization where the system(s) will be migrated to.
- array:
 - int - systemId

Returns:

- array:
 - int - serverIdMigrated
-

Method: setSoftwareEntitlements

Description:

Set an organization's entitlement allocation for the given software entitlement. If increasing the entitlement allocation, the default organization (i.e. orgId=1) must have a sufficient number of free entitlements.

Parameters:

- string sessionKey
- int orgId
- string label - Software entitlement label.
- int allocation

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setSystemEntitlements

Description:

Set an organization's entitlement allocation for the given software entitlement. If increasing the entitlement allocation, the default organization (i.e. orgId=1) must have a sufficient number of free entitlements.

Parameters:

- string sessionKey
- int orgId
- string label - System entitlement label. Valid values include:
 - enterprise_entitled
 - monitoring_entitled
 - provisioning_entitled
 - virtualization_host
 - virtualization_host_platform

- int allocation

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: updateName

Description:

Updates the name of an organization

Parameters:

- string sessionKey
- int orgId
- string name - Organization name. Must meet same criteria as in the web UI.

Returns:

- struct - organization info
 - int "id"
 - string "name"
 - int "active_users" - Number of active users in the organization.
 - int "systems" - Number of systems in the organization.
 - int "trusts" - Number of trusted organizations.
 - int "system_groups" - Number of system groups in the organization.
 - int "activation_keys" - Number of activation keys in the organization. (optional)
 - int "kickstart_profiles" - Number of kickstart profiles in the organization. (optional)
 - int "configuration_channels" - Number of configuration channels in the organization. (optional)
-

org.trusts

(Return to Handler List)

Available methods

- addTrust
- getDetails
- listChannelsConsumed
- listChannelsProvided
- listOrgs
- listSystemsAffected
- listTrusts
- removeTrust

Description

Contains methods to access common organization trust information available from the web interface.

Namespace: org.trusts

Method: addTrust

Description:

Add an organization to the list of trusted organizations.

Parameters:

- string sessionKey
- int orgId
- int trustOrgId

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getDetails

Description:

The trust details about an organization given the organization's ID.

Parameters:

- string sessionKey
- int trustOrgId - Id of the trusted organization

Returns:

- struct - org trust details
 - dateTime.iso8601 "created" - Date the organization was created
 - dateTime.iso8601 "trusted_since" - Date the organization was defined as trusted
 - int "channels_provided" - Number of channels provided by the organization.
 - int "channels_consumed" - Number of channels consumed by the organization.
 - int "systems_migrated_to" - Number of systems migrated to the organization.
 - int "systems_migrated_from" - Number of systems migrated from the organization.
-

Method: listChannelsConsumed

Description:

Lists all software channels that the organization given may consume from the user's organization.

Parameters:

- string sessionKey
- int trustOrgId - Id of the trusted organization

Returns:

- array:
 - struct - channel info
 - int "channel_id"
 - string "channel_name"
 - int "packages"
 - int "systems"
-

Method: listChannelsProvided

Description:

Lists all software channels that the organization given is providing to the user's organization.

Parameters:

- string sessionKey
- int trustOrgId - Id of the trusted organization

Returns:

- array:

- struct - channel info
 - int "channel_id"
 - string "channel_name"
 - int "packages"
 - int "systems"
-

Method: listOrgs

Description:

List all organizations trusted by the user's organization.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - trusted organizations
 - int "org_id"
 - string "org_name"
 - int "shared_channels"
-

Method: listSystemsAffected

Description:

Get a list of systems within the *trusted* organization that would be affected if the *trust* relationship was removed. This basically lists systems that are sharing at least (1) package.

Parameters:

- string sessionKey
- int orgId
- string trustOrgId

Returns:

- array:
 - struct - affected systems
 - int "systemId"
 - string "systemName"
-

Method: listTrusts

Description:

Returns the list of trusted organizations.

Parameters:

- string sessionKey
- int orgId

Returns:

- array:
 - struct - trusted organizations
 - int "orgId"
 - string "orgName"
 - bool "trustEnabled"
-

Method: removeTrust

Description:

Remove an organization to the list of trusted organizations.

Parameters:

- string sessionKey
- int orgId
- int trustOrgId

Returns:

- int - 1 on success, exception thrown otherwise.
-

packages

(Return to Handler List)

Available methods

- findByNvrea
- getDetails
- getPackage
- getPackageUrl
- listChangelog
- listDependencies
- listFiles
- listProvidingChannels

- listProvidingErrata
- removePackage

Description

Methods to retrieve information about the Packages contained within this server.

Namespace: packages

Method: findByNvrea

Description:

Lookup the details for packages with the given name, version, release, architecture label, and (optionally) epoch.

Parameters:

- string sessionKey
- string name
- string version
- string release
- string epoch - If set to something other than empty string, strict matching will be used and the epoch string must be correct. If set to an empty string, if the epoch is null or there is only one NVRA combination, it will be returned. (Empty string is recommended.)
- string archLabel

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "path" - The path on that file system that the package resides
 - string "provider" - The provider of the package, determined by the gpg key it was signed with.
 - dateTime.iso8601 "last_modified"
-

Method: getDetails

Description:

Retrieve details for the package with the ID.

Parameters:

- string sessionKey
- int packageId

Returns:

- struct - package
 - int "id"
 - string "name"
 - string "epoch"
 - string "version"
 - string "release"
 - string "arch_label"
 - array "string"
 - label Channel label providing this package.
 - string "build_host"
 - string "description"
 - string "md5sum"
 - string "vendor"
 - string "summary"
 - string "cookie"
 - string "license"
 - string "file"
 - string "build_date"
 - string "last_modified_date"
 - string "size"
 - string "path" - The path on the Satellite's file system that the package resides.
 - string "payload_size"
-

Method: getPackage

Description:

Retrieve the package file associated with a package.

Parameters:

- string sessionKey
- int package_id

Returns:

- `base64` - base64 encoded package
-

Method: `getPackageUrl`

Description:

Retrieve the url that can be used to download a package. This will expire after a certain time period.

Parameters:

- `string` `sessionKey`
- `int` `package_id`

Returns:

- `string` - the download url
-

Method: `listChangelog`

Description:

List the change log for a package.

Parameters:

- `string` `sessionKey`
- `int` `packageId`

Returns:

- `array`:
 - `struct` - changelog entry
 - `string` `"author"`
 - `string` `"date"`
 - `string` `"text"`
-

Method: `listDependencies`

Description:

List the dependencies for a package.

Parameters:

- `string` `sessionKey`
- `int` `packageId`

Returns:

- `array`:

- struct - dependency
 - string "dependency"
 - string "dependency_type" - One of the following:
 - requires
 - conflicts
 - obsoletes
 - provides
 - string "dependency_modifier"
-

Method: listFiles

Description:

List the files associated with a package.

Parameters:

- string sessionKey
- int packageId

Returns:

- array:
 - struct - file info
 - string "path"
 - string "type"
 - string "last_modified_date"
 - string "md5sum"
 - int "size"
 - string "linkto"
-

Method: listProvidingChannels

Description:

List the channels that provide the a package.

Parameters:

- string sessionKey
- int packageId

Returns:

- array:
 - struct - channel
 - string "label"
 - string "parent_label"
 - string "name"

Method: listProvidingErrata

Description:

List the errata providing the a package.

Parameters:

- string sessionKey
- int packageId

Returns:

- array:
 - struct - errata
 - string "advisory"
 - string "issue_date"
 - string "last_modified_date"
 - string "update_date"
 - string "synopsis"
 - string "type"

Method: removePackage

Description:

Remove a package from the satellite.

Parameters:

- string sessionKey
- int packageId

Returns:

- int - 1 on success, exception thrown otherwise.

packages.provider

(Return to Handler List)

Available methods

- associateKey
- list
- listKeys

Description

Methods to retrieve information about Package Providers associated with packages.

Namespace: packages.provider

Method: associateKey

Description:

Associate a package security key and with the package provider. If the provider or key doesn't exist, it is created. User executing the request must be a Satellite administrator.

Parameters:

- string sessionKey
- string providerName - The provider name
- string key - The actual key
- string type - The type of the key. Currently, only 'gpg' is supported

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: list

Description:

List all Package Providers. User executing the request must be a Satellite administrator.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - package provider
 - string "name"
 - array "keys"
 - struct - package security key
 - string "key"
 - string "type"
-

Method: listKeys

Description:

List all security keys associated with a package provider. User executing the request must be a Satellite administrator.

Parameters:

- string sessionKey
- string providerName - The provider name

Returns:

- array:
 - struct - package security key
 - string "key"
 - string "type"
-

packages.search

(Return to Handler List)

Available methods

- advanced
- advancedWithActKey
- advancedWithChannel
- name
- nameAndDescription
- nameAndSummary

Description

Methods to interface to package search capabilities in search server..

Namespace: packages.search

Method: advanced

Description:

Advanced method to search lucene indexes with a passed in query written in Lucene Query Parser syntax. Lucene Query Parser syntax is defined here.

Fields searchable for Packages: name, epoch, version, release, arch, description, summary

Lucene Query Example: "name:kernel AND version:2.6.18 AND -description:devel"

Parameters:

- string sessionKey
- string luceneQuery - a query written in the form of Lucene QueryParser Syntax

Returns:

- array:

- struct - package overview
 - int "id"
 - string "name"
 - string "summary"
 - string "description"
 - string "version"
 - string "release"
 - string "arch"
 - string "epoch"
 - string "provider"
-

Method: advancedWithActKey

Description:

Advanced method to search lucene indexes with a passed in query written in Lucene Query Parser syntax, additionally this method will limit results to those which are associated with a given activation key.

Lucene Query Parser syntax is defined here.

Fields searchable for Packages: name, epoch, version, release, arch, description, summary

Lucene Query Example: "name:kernel AND version:2.6.18 AND -description:devel"

Parameters:

- string sessionKey
- string luceneQuery - a query written in the form of Lucene QueryParser Syntax
- string actKey - activation key to look for packages in

Returns:

- array:
 - struct - package overview
 - int "id"
 - string "name"
 - string "summary"
 - string "description"
 - string "version"
 - string "release"
 - string "arch"
 - string "epoch"
 - string "provider"
-

Method: advancedWithChannel

Description:

Advanced method to search lucene indexes with a passed in query written in Lucene Query Parser syntax, additionally this method will limit results to those which are in the passed in channel label.

Lucene Query Parser syntax is defined here.

Fields searchable for Packages: name, epoch, version, release, arch, description, summary

Lucene Query Example: "name:kernel AND version:2.6.18 AND -description:devel"

Parameters:

- string sessionKey
- string luceneQuery - a query written in the form of Lucene QueryParser Syntax
- string channelLabel - Channel Label

Returns:

- array:
 - struct - package overview
 - int "id"
 - string "name"
 - string "summary"
 - string "description"
 - string "version"
 - string "release"
 - string "arch"
 - string "epoch"
 - string "provider"

Method: name

Description:

Search the lucene package indexes for all packages which match the given name.

Parameters:

- string sessionKey
- string name - package name to search for

Returns:

- array:
 - struct - package overview
 - int "id"
 - string "name"
 - string "summary"
 - string "description"

- string "version"
 - string "release"
 - string "arch"
 - string "epoch"
 - string "provider"
-

Method: nameAndDescription

Description:

Search the lucene package indexes for all packages which match the given query in name or description

Parameters:

- string sessionKey
- string query - text to match in package name or description

Returns:

- array:
 - struct - package overview
 - int "id"
 - string "name"
 - string "summary"
 - string "description"
 - string "version"
 - string "release"
 - string "arch"
 - string "epoch"
 - string "provider"
-

Method: nameAndSummary

Description:

Search the lucene package indexes for all packages which match the given query in name or summary.

Parameters:

- string sessionKey
- string query - text to match in package name or summary

Returns:

- array:
 - struct - package overview
 - int "id"
 - string "name"
 - string "summary"

- string "description"
 - string "version"
 - string "release"
 - string "arch"
 - string "epoch"
 - string "provider"
-

preferences.locale

(Return to Handler List)

Available methods

- listLocales
- listTimeZones
- setLocale
- setTimeZone

Description

Provides methods to access and modify user locale information

Namespace: preferences.locale

Method: listLocales

Description:

Returns a list of all understood locales. Can be used as input to setLocale.

Parameters:

Returns:

- array:
 - string - Locale code.
-

Method: listTimeZones

Description:

Returns a list of all understood timezones. Results can be used as input to setTimeZone.

Parameters:

Returns:

- array:
 - struct - timezone
 - int "time_zone_id" - Unique identifier for timezone.
 - string "olson_name" - Name as identified by the Olson database.
-

Method: setLocale

Description:

Set a user's locale.

Parameters:

- string sessionKey
- string login - User's login name.
- string locale - Locale to set. (from listLocales)

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setTimeZone

Description:

Set a user's timezone.

Parameters:

- string sessionKey
- string login - User's login name.
- int tzid - Timezone ID. (from listTimeZones)

Returns:

- int - 1 on success, exception thrown otherwise.
-

proxy

(Return to Handler List)

Available methods

- activateProxy
- createMonitoringScout
- deactivateProxy
- isProxy
- listAvailableProxyChannels

Description

Provides methods to activate/deactivate a proxy server.

Namespace: proxy

Method: activateProxy

Description:

Activates the proxy identified by the given client certificate i.e. systemid file.

Parameters:

- string systemid - systemid file
- string version - Version of proxy to be registered.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: createMonitoringScout

Description:

Create Monitoring Scout for proxy.

Parameters:

- string systemid - systemid file

Returns:

- string

Available since: 10.7

Method: deactivateProxy

Description:

Deactivates the proxy identified by the given client certificate i.e. systemid file.

Parameters:

- string systemid - systemid file

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: isProxy

Description:

Test, if the system identified by the given client certificate i.e. systemid file, is proxy.

Parameters:

- string systemid - systemid file

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: listAvailableProxyChannels

Description:

List available version of proxy channel for system identified by the given client certificate i.e. systemid file.

Parameters:

- string systemid - systemid file

Returns:

- array:
 - string - version

Available since: 10.5

satellite

(Return to Handler List)

Available methods

- getCertificateExpirationDate
- listEntitlements
- listProxies

Description

Provides methods to obtain details on the Satellite.

Namespace: satellite

Method: `getCertificateExpirationDate`

Description:

Retrieves the certificate expiration date of the activated certificate.

Parameters:

- `string sessionKey`

Returns:

- `dateTime.iso8601`
-

Method: `listEntitlements`

Description:

Lists all channel and system entitlements for the organization associated with the user executing the request.

Parameters:

- `string sessionKey`

Returns:

- `struct - channel/system entitlements`
 - `array "system"`
 - `struct - system entitlement`
 - `string "label"`
 - `string "name"`
 - `int "used_slots"`
 - `int "free_slots"`
 - `int "total_slots"`
 - `array "channel"`
 - `struct - channel entitlement`
 - `string "label"`
 - `string "name"`
 - `int "used_slots"`
 - `int "free_slots"`
 - `int "total_slots"`
-

Method: `listProxies`

Description:

List the proxies within the user's organization.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

schedule

(Return to Handler List)

Available methods

- cancelActions
- listAllActions
- listArchivedActions
- listCompletedActions
- listCompletedSystems
- listFailedActions
- listFailedSystems
- listInProgressActions
- listInProgressSystems

Description

Methods to retrieve information about scheduled actions.

Namespace: schedule

Method: cancelActions

Description:

Cancel all actions in given list. If an invalid action is provided, none of the actions given will canceled.

Parameters:

- string sessionKey
- array:

- int - action id

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: listAllActions

Description:

Returns a list of all actions. This includes completed, in progress, failed and archived actions.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - action
 - int "id" - Action Id.
 - string "name" - Action name.
 - string "type" - Action type.
 - string "scheduler" - The user that scheduled the action.
 - dateTime.iso8601 "earliest" - The earliest date and time the action will be performed
-

Method: listArchivedActions

Description:

Returns a list of actions that have been archived.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - action
 - int "id" - Action Id.
 - string "name" - Action name.
 - string "type" - Action type.
 - string "scheduler" - The user that scheduled the action.
 - dateTime.iso8601 "earliest" - The earliest date and time the action will be performed

Method: listCompletedActions

Description:

Returns a list of actions that have completed successfully.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - action
 - int "id" - Action Id.
 - string "name" - Action name.
 - string "type" - Action type.
 - string "scheduler" - The user that scheduled the action.
 - dateTime.iso8601 "earliest" - The earliest date and time the action will be performed

Method: listCompletedSystems

Description:

Returns a list of systems that have completed a specific action.

Parameters:

- string sessionKey
- string actionId

Returns:

- array:
 - struct - system
 - int "server_id"
 - string "server_name" - Server name.
 - string "base_channel" - Base channel used by the server.
 - dateTime.iso8601 "timestamp" - The time the action was completed
 - string "message" - Optional message containing details on the execution of the action. For example, if the action failed, this will contain the failure text.

Method: listFailedActions

Description:

Returns a list of actions that have failed.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - action
 - int "id" - Action Id.
 - string "name" - Action name.
 - string "type" - Action type.
 - string "scheduler" - The user that scheduled the action.
 - dateTime.iso8601 "earliest" - The earliest date and time the action will be performed
-

Method: listFailedSystems

Description:

Returns a list of systems that have failed a specific action.

Parameters:

- string sessionKey
- string actionId

Returns:

- array:
 - struct - system
 - int "server_id"
 - string "server_name" - Server name.
 - string "base_channel" - Base channel used by the server.
 - dateTime.iso8601 "timestamp" - The time the action was completed
 - string "message" - Optional message containing details on the execution of the action. For example, if the action failed, this will contain the failure text.
-

Method: listInProgressActions

Description:

Returns a list of actions that are in progress.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - action
 - int "id" - Action Id.
 - string "name" - Action name.
 - string "type" - Action type.
 - string "scheduler" - The user that scheduled the action.
 - dateTime.iso8601 "earliest" - The earliest date and time the action will be performed
-

Method: listInProgressSystems

Description:

Returns a list of systems that have a specific action in progress.

Parameters:

- string sessionKey
- string actionId

Returns:

- array:
 - struct - system
 - int "server_id"
 - string "server_name" - Server name.
 - string "base_channel" - Base channel used by the server.
 - dateTime.iso8601 "timestamp" - The time the action was completed
 - string "message" - Optional message containing details on the execution of the action. For example, if the action failed, this will contain the failure text.
-

system

(Return to Handler List)

Available methods

- addEntitlements
- addNote
- applyErrata
- comparePackageProfile
- comparePackages
- createPackageProfile
- deleteCustomValues
- deleteNote
- deleteNotes
- deleteSystems
- downloadSystemId
- getConnectionPath
- getCpu
- getCustomValues
- getDetails
- getDevices
- getDmi
- getEntitlements
- getEventHistory
- getId
- getMemory
- getName
- getNetwork
- getNetworkDevices
- getRegistrationDate
- getRelevantErrata
- getRelevantErrataByType
- getRunningKernel
- getScriptResults
- getSubscribedBaseChannel
- getUnscheduledErrata
- isNvreInstalled
- isNvreInstalled
- listActivationKeys
- listActiveSystems
- listAdministrators
- listBaseChannels
- listChildChannels

- listGroups
- listInactiveSystems
- listInactiveSystems
- listLatestInstallablePackages
- listLatestUpgradablePackages
- listNewerInstalledPackages
- listNotes
- listOlderInstalledPackages
- listOutOfDateSystems
- listPackages
- listPackagesFromChannel
- listSubscribableBaseChannels
- listSubscribableChildChannels
- listSubscribedChildChannels
- listSystemEvents
- listSystems
- listSystemsWithPackage
- listSystemsWithPackage
- listUngroupedSystems
- listUserSystems
- listUserSystems
- listVirtualGuests
- listVirtualHosts
- obtainReactivationKey
- provisionSystem
- provisionSystem
- provisionVirtualGuest
- provisionVirtualGuest
- removeEntitlements
- scheduleApplyErrata
- scheduleApplyErrata
- scheduleGuestAction
- scheduleGuestAction
- scheduleHardwareRefresh
- schedulePackageInstall
- schedulePackageRefresh
- schedulePackageRemove
- scheduleReboot
- scheduleScriptRun
- scheduleSyncPackagesWithSystem
- searchByName
- setBaseChannel
- setBaseChannel

- setChildChannels
- setCustomValues
- setDetails
- setGroupMembership
- setGuestCpus
- setGuestMemory
- setLockStatus
- setProfileName
- upgradeEntitlement
- whoRegistered

Description

Provides methods to access and modify registered system.

Namespace: system

Method: addEntitlements

Description:

Add addon entitlements to a server. Entitlements a server already has are quietly ignored.

Parameters:

- string sessionKey
- int serverId
- array:
 - string - entitlementLabel

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addNote

Description:

Add a new note to the given server.

Parameters:

- string sessionKey
- int serverId
- string subject - What the note is about.
- string body - Content of the note.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: applyErrata

Description:

Schedules an action to apply errata updates to a system.

Deprecated as of version - being replaced by `system.scheduleApplyErrata(string sessionKey, int serverId, array[int errataId])`

Parameters:

- string sessionKey
- int serverId
- array:
 - int - errataId

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: comparePackageProfile

Description:

Compare a system's packages against a package profile. In the result returned, 'this_system' represents the server provided as an input and 'other_system' represents the profile provided as an input.

Parameters:

- string sessionKey
- int serverId
- string profileLabel

Returns:

- array:
 - struct - Package Metadata
 - int "package_name_id"
 - string "package_name"
 - string "this_system" - Version of package on this system.
 - string "other_system" - Version of package on the other system.
 - int "comparison"
 - 0 - No difference.
 - 1 - Package on this system only.
 - 2 - Newer package version on this system.
 - 3 - Package on other system only.

- 4 - Newer package version on other system.
-

Method: comparePackages

Description:

Compares the packages installed on two systems.

Parameters:

- string sessionKey
- int thisServerId
- int otherServerId

Returns:

- array:
 - struct - Package Metadata
 - int "package_name_id"
 - string "package_name"
 - string "this_system" - Version of package on this system.
 - string "other_system" - Version of package on the other system.
 - int "comparison"
 - 0 - No difference.
 - 1 - Package on this system only.
 - 2 - Newer package version on this system.
 - 3 - Package on other system only.
 - 4 - Newer package version on other system.
-

Method: createPackageProfile

Description:

Create a new stored Package Profile from a systems installed package list.

Parameters:

- string sessionKey
- int serverId
- string profileLabel
- string description

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deleteCustomValues

Description:

Delete the custom values defined for the custom system information keys provided from the given system.

Parameters:

- string sessionKey
- int serverId
- array:
 - string - customInfoLabel

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deleteNote

Description:

Deletes the given note from the server.

Parameters:

- string sessionKey
- int serverId
- int noteId

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deleteNotes

Description:

Deletes all notes from the server.

Parameters:

- string sessionKey
- int serverId

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deleteSystems

Description:

Delete systems given a list of system ids.

Parameters:

- string sessionKey
- array:
 - int - serverId

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: downloadSystemId

Description:

Get the system ID file for a given server.

Parameters:

- string sessionKey
- int serverId

Returns:

- string
-

Method: getConnectionPath

Description:

Get the list of proxies that the given system connects through in order to reach the server.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - proxy connection path details
 - int "position" - Position of proxy in chain. The proxy that the system connects directly to is listed in position 1.
 - int "id" - Proxy system id
 - string "hostname" - Proxy host name

Method: getCpu

Description:

Gets the CPU information of a system.

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - CPU
 - string "cache"
 - string "family"
 - string "mhz"
 - string "flags"
 - string "model"
 - string "vendor"
 - string "arch"
 - string "stepping"
 - string "count"

Method: getCustomValues

Description:

Get the custom data values defined for the server.

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - custom value
 - string "custom info label"

Method: getDetails

Description:

Get system details.

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - server details
 - int "id" - System id
 - string "profile_name"
 - string "base_entitlement" - System's base entitlement label. (enterprise_entitled or sw_mgr_entitled)
 - array "string"
 - addon_entitlements System's addon entitlements labels, including monitoring_entitled, provisioning_entitled, virtualization_host, virtualization_host_platform
 - boolean "auto_update" - True if system has auto errata updates enabled.
 - string "release" - The Operating System release (i.e. 4AS, 5Server)
 - string "address1"
 - string "address2"
 - string "city"
 - string "state"
 - string "country"
 - string "building"
 - string "room"
 - string "rack"
 - string "description"
 - string "hostname"
 - string "osa_status" - Either 'unknown', 'offline', or 'online'.
 - boolean "lock_status" - True indicates that the system is locked. False indicates that the system is unlocked.
-

Method: getDevices

Description:

Gets a list of devices for a system.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - device
 - string "device" - optional
 - string "device_class" - Includes CDROM, FIREWIRE, HD, USB, VIDEO, OTHER, etc.
 - string "driver"

- string "description"
 - string "bus"
 - string "pcitype"
-

Method: getDmi

Description:

Gets the DMI information of a system.

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - DMI
 - string "vendor"
 - string "system"
 - string "product"
 - string "asset"
 - string "board"
 - string "bios_release"
 - string "bios_vendor"
 - string "bios_version"
-

Method: getEntitlements

Description:

Gets the entitlements for a given server.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - string - entitlement_label
-

Method: `getEventHistory`

Description:

Returns a list history items associated with the system, ordered from newest to oldest. Note that the details may be empty for events that were scheduled against the system (as compared to instant). For more information on such events, see the `system.listSystemEvents` operation.

Parameters:

- string `sessionKey`
- int `serverId`

Returns:

- array:
 - struct - History Event
 - `dateTime.iso8601 "completed"` - Date that the event occurred (optional)
 - string `"summary"` - Summary of the event
 - string `"details"` - Details of the event
-

Method: `getId`

Description:

Get system IDs and last check in information for the given system name.

Parameters:

- string `sessionKey`
- string `systemName`

Returns:

- array:
 - struct - system
 - int `"id"`
 - string `"name"`
 - `dateTime.iso8601 "last_checkin"` - Last time server successfully checked in
-

Method: `getMemory`

Description:

Gets the memory information for a system.

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - memory
 - int "ram" - The amount of physical memory in MB.
 - int "swap" - The amount of swap space in MB.
-

Method: getName

Description:

Get system name and last check in information for the given system ID.

Parameters:

- string sessionKey
- string serverId

Returns:

- struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: getNetwork

Description:

Get the IP address and hostname for a given server.

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - network info
 - string "ip" - IP address of server
 - string "hostname" - Hostname of server
-

Method: getNetworkDevices

Description:

Returns the network devices for the given server.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - network device
 - string "ip" - IP address assigned to this network device
 - string "interface" - Network interface assigned to device e.g. eth0
 - string "netmask" - Network mask assigned to device
 - string "hardware_address" - Hardware Address of device.
 - string "module" - Network driver used for this device.
 - string "broadcast" - Broadcast address for device.
-

Method: getRegistrationDate

Description:

Returns the date the system was registered.

Parameters:

- string sessionKey
- int serverId

Returns:

- dateTime.iso8601 - The date the system was registered, in local time.
-

Method: getRelevantErrata

Description:

Returns a list of all errata that are relevant to the system.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:

- struct - errata
 - int "id" - Errata ID.
 - string "date" - Date erratum was created.
 - string "advisory_synopsis" - Summary of the erratum.
 - string "advisory_type" - Type label such as Security, Bug Fix
 - string "advisory_name" - Name such as RHSA, etc
-

Method: getRelevantErrataByType

Description:

Returns a list of all errata of the specified type that are relevant to the system.

Parameters:

- string sessionKey
- int serverId
- string advisoryType - type of advisory (one of the following: 'Security Advisory', 'Product Enhancement Advisory', 'Bug Fix Advisory')

Returns:

- array:
 - struct - errata
 - int "id" - Errata ID.
 - string "date" - Date erratum was created.
 - string "advisory_synopsis" - Summary of the erratum.
 - string "advisory_type" - Type label such as Security, Bug Fix
 - string "advisory_name" - Name such as RHSA, etc
-

Method: getRunningKernel

Description:

Returns the running kernel of the given system.

Parameters:

- string sessionKey
- int serverId

Returns:

- string
-

Method: getScriptResults

Description:

Fetch results from a script execution. Returns an empty array if no results are yet available.

Parameters:

- string sessionKey
- int actionId - ID of the script run action.

Returns:

- array:
 - struct - script result
 - dateTime.iso8601 "startDate" - Time script began execution.
 - dateTime.iso8601 "stopDate" - Time script stopped execution.
 - int "returnCode" - Script execution return code.
 - string "output" - Output of the script
-

Method: getSubscribedBaseChannel

Description:

Provides the base channel of a given system

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - channel
 - int "id"
 - string "name"
 - string "label"
 - string "arch_name"
 - string "summary"
 - string "description"
 - string "maintainer_name"
 - string "maintainer_email"
 - string "maintainer_phone"
 - string "support_policy"
 - string "gpg_key_url"
 - string "gpg_key_id"
 - string "gpg_key_fp"
 - string "end_of_life"
 - string "parent_channel_label"

Method: getUnscheduledErrata

Description:

Provides an array of errata that are applicable to a given system.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - errata
 - int "id" - Errata Id
 - string "date" - Date erratum was created.
 - string "advisory_type" - Type of the advisory.
 - string "advisory_name" - Name of the advisory.
 - string "advisory_synopsis" - Summary of the erratum.

Method: isNvreInstalled

Description:

Check if the package with the given NVRE is installed on given system.

Parameters:

- string sessionKey
- int serverId
- string name - Package name.
- string version - Package version.
- string release - Package release.

Returns:

- int - 1 on success, exception thrown otherwise.

Method: isNvreInstalled

Description:

Is the package with the given NVRE installed on given system.

Parameters:

- string sessionKey
- int serverId
- string name - Package name.
- string version - Package version.

- string release - Package release.
- string epoch - Package epoch.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: listActivationKeys

Description:

List the activation keys the system was registered with. An empty list will be returned if an activation key was not used during registration.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - string - key
-

Method: listActiveSystems

Description:

Returns a list of active servers visible to the user.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: listAdministrators

Description:

Returns a list of users which can administer the system.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - user
 - int "id"
 - string "login"
 - string "login_uc" - upper case version of the login
 - boolean "enabled" - true if user is enabled, false if the user is disabled
-

Method: listBaseChannels

Description:

Returns a list of subscribable base channels.

Deprecated as of version - being replaced by listSubscribableBaseChannels(string sessionKey, int serverId)

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - channel
 - int "id" - Base Channel ID.
 - string "name" - Name of channel.
 - string "label" - Label of Channel
 - int "current_base" -1 indicates it is the current base channel
-

Method: listChildChannels

Description:

Returns a list of subscribable child channels. This only shows channels the system is **not** currently subscribed to.

Deprecated as of version - being replaced by listSubscribableChildChannels(string sessionKey, int serverId)

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - child channel
 - int "id"
 - string "name"
 - string "label"
 - string "summary"
 - string "has_license"
 - string "gpg_key_url"
-

Method: listGroups

Description:

List the available groups for a given system.

Parameters:

- string sessionKey
- int serverId

Returns:

- array struct - system group
 - int "id" - server group id
 - int "subscribed" - 1 if the given server is subscribed to this server group, 0 otherwise
 - string "system_group_name" - Name of the server group
 - string "sgid" - server group id (Deprecated)
-

Method: listInactiveSystems

Description:

Lists systems that have been inactive for the default period of inactivity

Parameters:

- string sessionKey

Returns:

- array struct - system

- int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: listInactiveSystems

Description:

Lists systems that have been inactive for the specified number of days..

Parameters:

- string sessionKey
- int days

Returns:

- array struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: listLatestInstallablePackages

Description:

Get the list of latest installable packages for a given system.

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
-

Method: listLatestUpgradablePackages

Description:

Get the list of latest upgradable packages for a given system.

Parameters:

- string sessionKey
- int serverId

Returns:

- struct - package
 - string "name"
 - string "from_version"
 - string "from_release"
 - string "from_epoch"
 - string "to_version"
 - string "to_release"
 - string "to_epoch"
 - string "to_package_id"
-

Method: listNewerInstalledPackages

Description:

Given a package name, version, release, and epoch, returns the list of packages installed on the system w/ the same name that are newer.

Parameters:

- string sessionKey
- int serverId
- string name - Package name.
- string version - Package version.
- string release - Package release.
- string epoch - Package epoch.

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"

Method: listNotes

Description:

Provides a list of notes associated with a system.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - note details
 - int "id"
 - string "subject" - Subject of the note
 - string "note" - Contents of the note
 - int "system_id" - The id of the system associated with the note
 - string "creator" - Creator of the note

Method: listOlderInstalledPackages

Description:

Given a package name, version, release, and epoch, returns the list of packages installed on the system with the same name that are older.

Parameters:

- string sessionKey
- int serverId
- string name - Package name.
- string version - Package version.
- string release - Package release.
- string epoch - Package epoch.

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"

Method: listOutOfDateSystems

Description:

Returns list of systems needing package updates.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in

Method: listPackages

Description:

List the installed packages for a given system.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - string "arch"

Method: listPackagesFromChannel

Description:

Provides a list of packages installed on a system that are also contained in the given channel. The installed package list did not include arch information before RHEL 5, so it is arch unaware. RHEL 5 systems do upload the arch information, and thus are arch aware.

Parameters:

- string sessionKey
- int serverId
- string channelLabel

Returns:

- array:
 - struct - package
 - string "name"
 - string "version"
 - string "release"
 - string "epoch"
 - int "id"
 - string "arch_label"
 - string "path" - The path on that file system that the package resides
 - string "provider" - The provider of the package, determined by the gpg key it was signed with.
 - dateTime.iso8601 "last_modified"
-

Method: listSubscribableBaseChannels

Description:

Returns a list of subscribable base channels.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - channel
 - int "id" - Base Channel ID.
 - string "name" - Name of channel.
 - string "label" - Label of Channel
 - int "current_base" - 1 indicates it is the current base channel
-

Method: listSubscribableChildChannels

Description:

Returns a list of subscribable child channels. This only shows channels the system is **not** currently subscribed to.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - child channel
 - int "id"
 - string "name"
 - string "label"
 - string "summary"
 - string "has_license"
 - string "gpg_key_url"
-

Method: listSubscribedChildChannels

Description:

Returns a list of subscribed child channels.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - channel
 - int "id"
 - string "name"
 - string "label"
 - string "arch_name"
 - string "summary"
 - string "description"
 - string "maintainer_name"
 - string "maintainer_email"
 - string "maintainer_phone"
 - string "support_policy"
 - string "gpg_key_url"
 - string "gpg_key_id"
 - string "gpg_key_fp"
 - string "end_of_life"
 - string "parent_channel_label"

Method: listSystemEvents

Description:

List all system events for given server. This includes *all* events for the server since it was registered. This may require the caller to filter the results to fetch the specific events they are looking for.

Parameters:

- string sessionKey
- int serverId - ID of system.

Returns:

- array:
 - struct - action
 - int "failed_count" - Number of times action failed.
 - string "modified" - Date modified. (Deprecated by modified_date)
 - dateTime.iso8601 "modified_date" - Date modified.
 - string "created" - Date created. (Deprecated by created_date)
 - dateTime.iso8601 "created_date" - Date created.
 - string "action_type"
 - int "successful_count" - Number of times action was successful.
 - string "earliest_action" - Earliest date this action will occur.
 - int "archived" - If this action is archived. (1 or 0)
 - string "scheduler_user"
 - string "prerequisite" - Pre-requisite action. (optional)
 - string "name" - Name of this action.
 - int "id" - Id of this action.
 - string "version" - Version of action.
 - string "completion_time" - The date/time the event was completed. Format ->YYYY-MM-dd hh:mm:ss.ms Eg ->2007-06-04 13:58:13.0. (optional) (Deprecated by completed_date)
 - dateTime.iso8601 "completed_date" - The date/time the event was completed. (optional)
 - string "pickup_time" - The date/time the action was picked up. Format ->YYYY-MM-dd hh:mm:ss.ms Eg ->2007-06-04 13:58:13.0. (optional) (Deprecated by pickup_date)
 - dateTime.iso8601 "pickup_date" - The date/time the action was picked up. (optional)
 - string "result_msg" - The result string after the action executes at the client machine. (optional)
 - array "additional_info" - This array contains additional information for the event, if available.
 - struct - info
 - string "detail" - The detail provided depends on the

specific event. For example, for a package event, this will be the package name, for an errata event, this will be the advisory name and synopsis, for a config file event, this will be path and optional revision information...etc.

- string "result" - The result (if included) depends on the specific event. For example, for a package or errata event, no result is included, for a config file event, the result might include an error (if one occurred, such as the file was missing) or in the case of a config file comparison it might include the differences found.

Available since: 10.8

Method: listSystems

Description:

Returns a list of all servers visible to the user.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: listSystemsWithPackage

Description:

Lists the systems that have the given installed package

Parameters:

- string sessionKey
- int pid - the package id

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in

Method: listSystemsWithPackage

Description:

Lists the systems that have the given installed package

Parameters:

- string sessionKey
- string name - the package name
- string version - the package version
- string release - the package release

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in

Method: listUngroupedSystems

Description:

List systems that are not associated with any system groups.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - system
 - int "id" - server id
 - string "name"

Method: listUserSystems

Description:

List systems for a given user.

Parameters:

- string sessionKey
- string login - User's login name.

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: listUserSystems

Description:

List systems for the logged in user.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: listVirtualGuests

Description:

Lists the virtual guests for given virtual host

Parameters:

- string sessionKey
- int sid - the virtual host's id

Returns:

- array:
 - struct - virtual system
 - int "id"
 - string "name"
 - string "guest_name" - The virtual guest name as provided by the virtual host
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in.

- string "uuid"
-

Method: listVirtualHosts

Description:

Lists the virtual hosts visible to the user

Parameters:

- string sessionKey

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: obtainReactivationKey

Description:

Obtains a reactivation key for this server.

Parameters:

- string sessionKey
- int serverId

Returns:

- string
-

Method: provisionSystem

Description:

Provision a system using the specified kickstart profile.

Parameters:

- string sessionKey
- int serverId - ID of the system to be provisioned.
- string profileName - Kickstart profile to use.

Returns:

- int - ID of the action scheduled, otherwise exception thrown on error

Method: provisionSystem

Description:

Provision a system using the specified kickstart profile.

Parameters:

- string sessionKey
- int serverId - ID of the system to be provisioned.
- string profileName - Kickstart profile to use.
- dateTime.iso8601 earliestDate

Returns:

- int - ID of the action scheduled, otherwise exception thrown on error
-

Method: provisionVirtualGuest

Description:

Provision a guest on the host specified. Defaults to: memory=256MB, vcpu=1, storage=2048MB.

Parameters:

- string sessionKey
- int serverId - ID of host to provision guest on.
- string guestName
- string profileName - Kickstart profile to use.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: provisionVirtualGuest

Description:

Provision a guest on the host specified. This schedules the guest for creation and will begin the provisioning process when the host checks in or if OSAD is enabled will begin immediately.

Parameters:

- string sessionKey
- int serverId - ID of host to provision guest on.
- string guestName
- string profileName - Kickstart Profile to use.
- int memoryMb - Memory to allocate to the guest
- int vcpus - Number of virtual CPUs to allocate to the guest.
- int storageMb - Size of the guests disk image.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removeEntitlements

Description:

Remove addon entitlements from a server. Entitlements a server does not have are quietly ignored.

Parameters:

- string sessionKey
- int serverId
- array:
 - string - entitlement_label

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: scheduleApplyErrata

Description:

Schedules an action to apply errata updates to a system.

Parameters:

- string sessionKey
- int serverId
- array:
 - int - errataId

Returns:

- int - 1 on success, exception thrown otherwise.

Available since: 10.6

Method: scheduleApplyErrata

Description:

Schedules an action to apply errata updates to a system at a given date/time.

Parameters:

- string sessionKey
- int serverId
- array:

- int - errataId
- dateTime.iso8601 earliestOccurrence

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: scheduleGuestAction

Description:

Schedules a guest action for the specified virtual guest for a given date/time.

Parameters:

- string sessionKey
- int sid - the system Id of the guest
- string state - One of the following actions 'start', 'suspend', 'resume', 'restart', 'shutdown'.
- dateTime.iso8601 date - the time/date to schedule the action

Returns:

- int actionId - The action id of the scheduled action
-

Method: scheduleGuestAction

Description:

Schedules a guest action for the specified virtual guest for the current time.

Parameters:

- string sessionKey
- int sid - the system Id of the guest
- string state - One of the following actions 'start', 'suspend', 'resume', 'restart', 'shutdown'.

Returns:

- int actionId - The action id of the scheduled action
-

Method: scheduleHardwareRefresh

Description:

Schedule a hardware refresh for a system.

Parameters:

- string sessionKey
- int serverId
- dateTime.iso8601 earliestOccurrence

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: schedulePackageInstall

Description:

Schedule package installation for a system.

Parameters:

- string sessionKey
- int serverId
- array:
 - int - packageId
- dateTime.iso8601 earliestOccurrence

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: schedulePackageRefresh

Description:

Schedule a package list refresh for a system.

Parameters:

- string sessionKey
- int serverId
- dateTime.iso8601 earliestOccurrence

Returns:

- int - ID of the action scheduled, otherwise exception thrown on error
-

Method: schedulePackageRemove

Description:

Schedule package removal for a system.

Parameters:

- string sessionKey
- int serverId
- array:

- int - packageId
- dateTime.iso8601 earliestOccurrence

Returns:

- int - ID of the action scheduled, otherwise exception thrown on error
-

Method: scheduleReboot

Description:

Schedule a reboot for a system.

Parameters:

- string sessionKey
- int serverId
- dateTime.iso8601 earliestOccurrence

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: scheduleScriptRun

Description:

Schedule a script to run.

Parameters:

- string sessionKey
- int serverId - ID of the server to run the script on.
- string username - User to run script as.
- string groupname - Group to run script as.
- int timeout - Seconds to allow the script to run before timing out.
- string script - Contents of the script to run.
- dateTime.iso8601 earliestOccurrence - Earliest the script can run.

Returns:

- int - ID of the script run action created. Can be used to fetch results with `system.getScriptResults`.
-

Method: scheduleSyncPackagesWithSystem

Description:

Sync packages from a source system to a target.

Parameters:

- string sessionKey
- int targetServerId - Target system to apply package changes to.
- int sourceServerId - Source system to retrieve package state from.
- array:
 - int - packageId - Package IDs to be synced.
- dateTime.iso8601 date - Date to schedule action for

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: searchByName

Description:

Returns a list of system IDs whose name matches the supplied regular expression.

Parameters:

- string sessionKey
- string regexp - A regular expression. (as defined by <http://java.sun.com/j2se/1.4.2/docs/api/java/util/regex/Pattern.html>)

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
-

Method: setBaseChannel

Description:

Assigns the server to a new baseChannel.

Deprecated as of version - being replaced by system.setBaseChannel(string sessionKey, int serverId, string channelLabel)

Parameters:

- string sessionKey
- int serverId
- int channelId

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: `setBaseChannel`

Description:

Assigns the server to a new base channel. If the user provides an empty string for the `channelLabel`, the current base channel and all child channels will be removed from the system.

Parameters:

- `string sessionKey`
- `int serverId`
- `string channelLabel`

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: `setChildChannels`

Description:

Subscribe the given server to the child channels provided. This method will unsubscribe the server from any child channels that the server is currently subscribed to, but that are not included in the list. The user may provide either a list of channel ids (`int`) or a list of channel labels (`string`) as input.

Parameters:

- `string sessionKey`
- `int serverId`
- `array:`
 - `int` (deprecated) or `string - channelId` (deprecated) or `channelLabel`

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: `setCustomValues`

Description:

Set custom values for the specified server.

Parameters:

- `string sessionKey`
- `int serverId`
- `struct - Map` of custom labels to custom values

- string "custom info label"
- string "value"

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setDetails

Description:

Set server details. All arguments are optional and will only be modified if included in the struct.

Parameters:

- string sessionKey
- int serverId - ID of server to lookup details for.
- struct - server details
 - string "profile_name" - System's profile name
 - string "base_entitlement" - System's base entitlement label. (enterprise_entitled or sw_mgr_entitled)
 - boolean "auto_errata_update" - True if system has auto errata updates enabled
 - string "description" - System description
 - string "address1" - System's address line 1.
 - string "address2" - System's address line 2.
 - string "city"
 - string "state"
 - string "country"
 - string "building"
 - string "room"
 - string "rack"

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setGroupMembership

Description:

Set a servers membership in a given group.

Parameters:

- string sessionKey
- int serverId
- int serverGroupId
- boolean member - '1' to assign the given server to the given server group, '0' to remove the given server from the given server group.

Returns:

- `int` - 1 on success, exception thrown otherwise.
-

Method: `setGuestCpus`

Description:

Schedule an action of a guest's host, to set that guest's CPU allocation

Parameters:

- `string` `sessionKey`
- `int` `sid` - The guest's system id
- `int` `numOfCpus` - The number of virtual cpus to allocate to the guest

Returns:

- `int` `actionID` - the action Id for the schedule action on the host system.
-

Method: `setGuestMemory`

Description:

Schedule an action of a guest's host, to set that guest's memory allocation

Parameters:

- `string` `sessionKey`
- `int` `sid` - The guest's system id
- `int` `memory` - The amount of memory to allocate to the guest

Returns:

- `int` `actionID` - the action Id for the schedule action on the host system.
-

Method: `setLockStatus`

Description:

Set server lock status.

Parameters:

- `string` `sessionKey`
- `int` `serverId`
- `boolean` `lockStatus` - true to lock the system, false to unlock the system.

Returns:

- `int` - 1 on success, exception thrown otherwise.

Method: setProfileName

Description:

Set the profile name for the server.

Parameters:

- string sessionKey
- int serverId
- string name - Name of the profile.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: upgradeEntitlement

Description:

Adds an entitlement to a given server.

Parameters:

- string sessionKey
- int serverId
- string entitlementName - One of: 'enterprise_entitled', 'provisioning_entitled', 'monitoring_entitled', 'nonlinux_entitled', 'virtualization_host', or 'virtualization_host_platform'.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: whoRegistered

Description:

Returns information about the user who registered the system

Parameters:

- string sessionKey
- int sid - Id of the system in question

Returns:

- struct - user
 - int "id"
 - string "login"
 - string "login_uc" - upper case version of the login
 - boolean "enabled" - true if user is enabled, false if the user is disabled

system.config

(Return to Handler List)

Available methods

- addChannels
- createOrUpdatePath
- deleteFiles
- deployAll
- listChannels
- listFiles
- lookupFileInfo
- removeChannels
- setChannels

Description

Provides methods to access and modify many aspects of configuration channels and server association. basically system.config name space

Namespace: system.config

Method: addChannels

Description:

Given a list of servers and configuration channels, this method appends the configuration channels to either the top or the bottom (whichever you specify) of a system's subscribed configuration channels list. The ordering of the configuration channels provided in the add list is maintained while adding. If one of the configuration channels in the 'add' list has been previously subscribed by a server, the subscribed channel will be re-ranked to the appropriate place.

Parameters:

- string sessionKey
- array:
 - int - IDs of the systems to add the channels to.
- array:
 - string - List of configuration channel labels in the ranked order.
- boolean addToTop

- true - to prepend the given channels list to the top of the configuration channels list of a server to the system's local override configuration channel")
- false - to append the given channels list to the bottom of the configuration channels list of a server

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: createOrUpdatePath

Description:

Create a new file (text or binary) or directory with the given path, or update an existing path on a server.

Parameters:

- string sessionKey
- int serverId
- string path - the configuration file/directory path
- boolean isDir
 - True - if the path is a directory
 - False - if the path is a file
- struct - path info
 - string "contents" - Contents of the file (text or base64 encoded if binary). (ignored for directories)
 - string "owner" - Owner of the file/directory.
 - string "group" - Group name of the file/directory.
 - string "permissions" - Octal file/directory permissions (eg: 644)
 - string "macro-start-delimiter" - Config file macro end delimiter. Use null or empty string to accept the default. (ignored if working with a directory)
 - string "macro-end-delimiter" - Config file macro end delimiter. Use null or empty string to accept the default. (ignored if working with a directory)
- int commitToLocal
 - 1 - to commit configuration files to the system's local override configuration channel
 - 0 - to commit configuration files to the system's sandbox configuration channel

Returns:

- struct - Configuration Revision information
 - string "type"
 - file
 - directory
 - string "path" - File Path
 - string "channel" - Channel Name
 - string "contents" - File contents for text files only.

- int "revision" - File Revision
- dateTime.iso8601 "creation" - Creation Date
- dateTime.iso8601 "modified" - Last Modified Date
- string "owner" - File Owner
- string "group" - File Group
- int "permissions" - File Permissions (Deprecated)
- string "permissions_mode" - File Permissions
- boolean "binary" - true/false , Present for files only.
- string "md5" - File's md5 signature. Present for files only.
- string "macro-start-delimiter" - Macro start delimiter for a config file. Present for files only.
- string "macro-end-delimiter" - Macro end delimiter for a config file. Present for files only.

Available since: 10.2

Method: deleteFiles

Description:

Removes file paths from a local or sandbox channel of a server.

Parameters:

- string sessionKey
- int serverId
- array:
 - string - paths to remove.
- boolean deleteFromLocal
 - True - to delete configuration file paths from the system's local override configuration channel
 - False - to delete configuration file paths from the system's sandbox configuration channel

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deployAll

Description:

Schedules a deploy action for all the configuration files on the given list of systems.

Parameters:

- string sessionKey
- array:

- int - id of the systems to schedule configuration files deployment
- dateTime.iso8601 date - Earliest date for the deploy action.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: listChannels

Description:

List all global configuration channels associated to a system in the order of their ranking.

Parameters:

- string sessionKey
- int serverId

Returns:

- array:
 - struct - Configuration Channel information
 - int "id"
 - int "orgId"
 - string "label"
 - string "name"
 - string "description"
 - struct "configChannelType" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"
-

Method: listFiles

Description:

Return the list of files in a given channel.

Parameters:

- string sessionKey
- int serverId
- int listLocal
 - 1 - to return configuration files in the system's local override configuration channel
 - 0 - to return configuration files in the system's sandbox configuration channel

Returns:

- array:
 - struct - Configuration File information
 - string "type"
 - file
 - directory
 - string "path" - File Path
 - string "channel_label" - the label of the central configuration channel that has this file. Note this entry only shows up if the file has not been overridden by a central channel.
 - struct "channel_type" struct - Configuration Channel Type information
 - int "id"
 - string "label"
 - string "name"
 - int "priority"
 - dateTime.iso8601 "last_modified" - Last Modified Date
-

Method: lookupFileInfo

Description:

Given a list of paths and a server, returns details about the latest revisions of the paths.

Parameters:

- string sessionKey
- int serverId
- array:
 - string - paths to lookup on.
- int searchLocal
 - 1 - to search configuration file paths in the system's local override configuration or systems subscribed central channels
 - 0 - to search configuration file paths in the system's sandbox configuration channel

Returns:

- array:
 - struct - Configuration Revision information
 - string "type"
 - file
 - directory
 - string "path" - File Path

- string "channel" - Channel Name
- string "contents" - File contents for text files only.
- int "revision" - File Revision
- dateTime.iso8601 "creation" - Creation Date
- dateTime.iso8601 "modified" - Last Modified Date
- string "owner" - File Owner
- string "group" - File Group
- int "permissions" - File Permissions (Deprecated)
- string "permissions_mode" - File Permissions
- boolean "binary" - true/false , Present for files only.
- string "md5" - File's md5 signature. Present for files only.
- string "macro-start-delimiter" - Macro start delimiter for a config file. Present for files only.
- string "macro-end-delimiter" - Macro end delimiter for a config file. Present for files only.

Available since: 10.2

Method: removeChannels

Description:

Remove config channels from the given servers.

Parameters:

- string sessionKey
- array:
 - int - the IDs of the systems from which you would like to remove configuration channels..
- array:
 - string - List of configuration channel labels to remove.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: setChannels

Description:

Replace the existing set of config channels on the given servers. Channels are ranked according to their order in the configChannelLabels array.

Parameters:

- string sessionKey
- array:

- int - IDs of the systems to set the channels on.
- array:
 - string - List of configuration channel labels in the ranked order.

Returns:

- int - 1 on success, exception thrown otherwise.
-

system.custominfo

(Return to Handler List)

Available methods

- createKey
- deleteKey
- listAllKeys

Description

Provides methods to access and modify custom system information.

Namespace: system.custominfo

Method: createKey

Description:

Create a new custom key

Parameters:

- string sessionKey
- string keyLabel - new key's label
- string keyDescription - new key's description

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: deleteKey

Description:

Delete an existing custom key and all systems' values for the key.

Parameters:

- string sessionKey
- string keyLabel - new key's label

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: listAllKeys

Description:

List the custom information keys defined for the user's organization.

Parameters:

- string sessionKey

Returns:

- array struct - custom info
 - int "id"
 - string "label"
 - string "description"
 - int "system_count"
 - dateTime.iso8601 "last_modified"
-

system.provisioning.snapshot

(Return to Handler List)

Available methods

- deleteSnapshot
- deleteSnapshots
- deleteSnapshots
- listSnapshotConfigFiles
- listSnapshotPackages
- listSnapshots

Description

Provides methods to access and delete system snapshots.

Namespace: system.provisioning.snapshot

Method: deleteSnapshot

Description:

Deletes a snapshot with the given snapshot id

Parameters:

- string sessionKey
- int snapshotId - Id of snapshot to delete

Returns:

- int - 1 on success, exception thrown otherwise.

Available since: 10.1

Method: deleteSnapshots

Description:

Deletes all snapshots across multiple systems based on the given date criteria. For example,

- If the user provides startDate only, all snapshots created either on or after the date provided will be removed.
- If user provides startDate and endDate, all snapshots created on or between the dates provided will be removed.
- If the user doesn't provide a startDate and endDate, all snapshots will be removed.

Parameters:

- string sessionKey
- struct - date details
 - dateTime.iso8601 "startDate" - Optional, unless endDate is provided.
 - dateTime.iso8601 "endDate" - Optional.

Returns:

- int - 1 on success, exception thrown otherwise.

Available since: 10.1

Method: deleteSnapshots

Description:

Deletes all snapshots for a given system based on the date criteria. For example,

- If the user provides startDate only, all snapshots created either on or after the date provided will be removed.
- If user provides startDate and endDate, all snapshots created on or between the dates provided will be removed.
- If the user doesn't provide a startDate and endDate, all snapshots associated with the server will be

removed.

Parameters:

- string sessionKey
- int sid - system id of system to delete snapshots for
- struct - date details
 - dateTime.iso8601 "startDate" - Optional, unless endDate is provided.
 - dateTime.iso8601 "endDate" - Optional.

Returns:

- int - 1 on success, exception thrown otherwise.

Available since: 10.1

Method: listSnapshotConfigFiles

Description:

List the config files associated with a snapshot.

Parameters:

- string sessionKey
- int snapId

Returns:

- array:
 - struct - Configuration Revision information
 - string "type"
 - file
 - directory
 - string "path" - File Path
 - string "channel" - Channel Name
 - string "contents" - File contents for text files only.
 - int "revision" - File Revision
 - dateTime.iso8601 "creation" - Creation Date
 - dateTime.iso8601 "modified" - Last Modified Date
 - string "owner" - File Owner
 - string "group" - File Group
 - int "permissions" - File Permissions (Deprecated)
 - string "permissions_mode" - File Permissions
 - boolean "binary" - true/false , Present for files only.
 - string "md5" - File's md5 signature. Present for files only.
 - string "macro-start-delimiter" - Macro start delimiter for a config file. Present for files only.

- string "macro-end-delimiter" - Macro end delimiter for a config file. Present for files only.

Available since: 10.2

Method: listSnapshotPackages

Description:

List the packages associated with a snapshot.

Parameters:

- string sessionKey
- int snapId

Returns:

- array:
 - struct - package nvera
 - string "name"
 - string "epoch"
 - string "version"
 - string "release"
 - string "arch"

Available since: 10.1

Method: listSnapshots

Description:

List snapshots for a given system. A user may optionally provide a start and end date to narrow the snapshots that will be listed. For example,

- If the user provides startDate only, all snapshots created either on or after the date provided will be returned.
- If user provides startDate and endDate, all snapshots created on or between the dates provided will be returned.
- If the user doesn't provide a startDate and endDate, all snapshots associated with the server will be returned.

Parameters:

- string sessionKey
- int serverId
- struct - date details

- `dateTime.iso8601 "startDate"` - Optional, unless `endDate` is provided.
- `dateTime.iso8601 "endDate"` - Optional.

Returns:

- `array:`
 - `struct` - server snapshot
 - `int "id"`
 - `string "reason"` - the reason for the snapshot's existence
 - `dateTime.iso8601 "created"`
 - `array "channels"`
 - `string` labels of channels associated with the snapshot
 - `array "groups"`
 - `string` Names of server groups associated with the snapshot
 - `array "entitlements"`
 - `string` Names of system entitlements associated with the snapshot
 - `array "config_channels"`
 - `string` Labels of config channels the snapshot is associated with.
 - `array "tags"`
 - `string` Tag names associated with this snapshot.
 - `string "Invalid_reason"` - If the snapshot is invalid, this is the reason (optional).

Available since: 10.1

system.search

(Return to Handler List)

Available methods

- `deviceDescription`
- `deviceDriver`
- `deviceId`
- `deviceVendorId`
- `hostname`
- `ip`
- `nameAndDescription`

Description

Provides methods to perform system search requests using the search server.

Namespace: system.search

Method: deviceDescription

Description:

List the systems which match the device description.

Parameters:

- string sessionKey
- string searchTerm

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
 - string "hostname"
 - string "ip"
 - string "hw_description" - hw description if not null
 - string "hw_device_id" - hw device id if not null
 - string "hw_vendor_id" - hw vendor id if not null
 - string "hw_driver" - hw driver if not null
-

Method: deviceDriver

Description:

List the systems which match this device driver.

Parameters:

- string sessionKey
- string searchTerm

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"

- `dateTime.iso8601 "last_checkin"` - Last time server successfully checked in
 - `string "hostname"`
 - `string "ip"`
 - `string "hw_description"` - hw description if not null
 - `string "hw_device_id"` - hw device id if not null
 - `string "hw_vendor_id"` - hw vendor id if not null
 - `string "hw_driver"` - hw driver if not null
-

Method: deviceId

Description:

List the systems which match this device id

Parameters:

- `string sessionKey`
- `string searchTerm`

Returns:

- `array:`
 - `struct - system`
 - `int "id"`
 - `string "name"`
 - `dateTime.iso8601 "last_checkin"` - Last time server successfully checked in
 - `string "hostname"`
 - `string "ip"`
 - `string "hw_description"` - hw description if not null
 - `string "hw_device_id"` - hw device id if not null
 - `string "hw_vendor_id"` - hw vendor id if not null
 - `string "hw_driver"` - hw driver if not null
-

Method: deviceVendorId

Description:

List the systems which match this device vendor_id

Parameters:

- `string sessionKey`
- `string searchTerm`

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
 - string "hostname"
 - string "ip"
 - string "hw_description" - hw description if not null
 - string "hw_device_id" - hw device id if not null
 - string "hw_vendor_id" - hw vendor id if not null
 - string "hw_driver" - hw driver if not null
-

Method: hostname

Description:

List the systems which match this hostname

Parameters:

- string sessionKey
- string searchTerm

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
 - string "hostname"
 - string "ip"
 - string "hw_description" - hw description if not null
 - string "hw_device_id" - hw device id if not null
 - string "hw_vendor_id" - hw vendor id if not null
 - string "hw_driver" - hw driver if not null
-

Method: ip

Description:

List the systems which match this ip.

Parameters:

- string sessionKey
- string searchTerm

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
 - string "hostname"
 - string "ip"
 - string "hw_description" - hw description if not null
 - string "hw_device_id" - hw device id if not null
 - string "hw_vendor_id" - hw vendor id if not null
 - string "hw_driver" - hw driver if not null
-

Method: nameAndDescription

Description:

List the systems which match this name or description

Parameters:

- string sessionKey
- string searchTerm

Returns:

- array:
 - struct - system
 - int "id"
 - string "name"
 - dateTime.iso8601 "last_checkin" - Last time server successfully checked in
 - string "hostname"
 - string "ip"
 - string "hw_description" - hw description if not null
 - string "hw_device_id" - hw device id if not null

- string "hw_vendor_id" - hw vendor id if not null
 - string "hw_driver" - hw driver if not null
-

systemgroup

(Return to Handler List)

Available methods

- addOrRemoveAdmins
- addOrRemoveSystems
- create
- delete
- getDetails
- getDetails
- listActiveSystemsInGroup
- listAdministrators
- listAllGroups
- listGroupsWithNoAssociatedAdmins
- listInactiveSystemsInGroup
- listInactiveSystemsInGroup
- listSystems
- update

Description

Provides methods to access and modify system groups.

Namespace: systemgroup

Method: addOrRemoveAdmins

Description:

Add or remove administrators to/from the given group. Satellite and Organization administrators are granted access to groups within their organization by default; therefore, users with those roles should not be included in the array provided. Caller must be an organization administrator.

Parameters:

- string sessionKey
- string systemGroupName
- array:

- string - loginName - User's loginName
- int add - 1 to add administrators, 0 to remove.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addOrRemoveSystems

Description:

Add/remove the given servers to a system group.

Parameters:

- string sessionKey
 - string systemGroupName
 - array:
 - int - serverId
- boolean add - True to add to the group, False to remove.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: create

Description:

Create a new system group.

Parameters:

- string sessionKey
- string name - Name of the system group.
- string description - Description of the system group.

Returns:

- struct - Server Group
- int "id"
 - string "name"
 - string "description"
 - int "org_id"
 - int "system_count"
-

Method: delete

Description:

Delete a system group.

Parameters:

- string sessionKey
- string systemGroupName

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: getDetails

Description:

Retrieve details of a ServerGroup based on it's id

Parameters:

- string sessionKey
- int systemGroupId

Returns:

- struct - Server Group
 - int "id"
 - string "name"
 - string "description"
 - int "org_id"
 - int "system_count"
-

Method: getDetails

Description:

Retrieve details of a ServerGroup based on it's name

Parameters:

- string sessionKey
- string systemGroupName

Returns:

- struct - Server Group
 - int "id"
 - string "name"
 - string "description"

- int "org_id"
 - int "system_count"
-

Method: listActiveSystemsInGroup

Description:

Lists active systems within a server group

Parameters:

- string sessionKey
- string systemGroupName

Returns:

- array:
 - int server_id
-

Method: listAdministrators

Description:

Returns the list of users who can administer the given group. Caller must be a system group admin or an organization administrator.

Parameters:

- string sessionKey
- string systemGroupName

Returns:

- array:
 - struct - user
 - int "id"
 - string "login"
 - string "login_uc" - upper case version of the login
 - boolean "enabled" - true if user is enabled, false if the user is disabled
-

Method: listAllGroups

Description:

Retrieve a list of system groups that are accessible by the logged in user.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - Server Group
 - int "id"
 - string "name"
 - string "description"
 - int "org_id"
 - int "system_count"
-

Method: listGroupsWithNoAssociatedAdmins

Description:

Returns a list of system groups that do not have an administrator. (who is not an organization administrator, as they have implicit access to system groups) Caller must be an organization administrator.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - Server Group
 - int "id"
 - string "name"
 - string "description"
 - int "org_id"
 - int "system_count"
-

Method: listInactiveSystemsInGroup

Description:

Lists inactive systems within a server group using a specified inactivity time.

Parameters:

- string sessionKey
- string systemGroupName
- int daysInactive - Number of days a system must not check in to be considered inactive.

Returns:

- array:
 - int server_id

Method: listInactiveSystemsInGroup

Description:

Lists inactive systems within a server group using the default 1 day threshold.

Parameters:

- string sessionKey
- string systemGroupName

Returns:

- array:
 - int server_id

Method: listSystems

Description:

Return a list of systems associated with this system group. User must have access to this system group.

Parameters:

- string sessionKey
- string systemGroupName

Returns:

- array:
 - struct - server details
 - int "id" - System id
 - string "profile_name"
 - string "base_entitlement" - System's base entitlement label. (enterprise_entitled or sw_mgr_entitled)
 - array "string"
 - addon_entitlements System's addon entitlements labels, including monitoring_entitled, provisioning_entitled, virtualization_host, virtualization_host_platform
 - boolean "auto_update" - True if system has auto errata updates enabled.
 - string "release" - The Operating System release (i.e. 4AS, 5Server)
 - string "address1"
 - string "address2"
 - string "city"
 - string "state"
 - string "country"
 - string "building"

- string "room"
 - string "rack"
 - string "description"
 - string "hostname"
 - string "osa_status" - Either 'unknown', 'offline', or 'online'.
 - boolean "lock_status" - True indicates that the system is locked. False indicates that the system is unlocked.
-

Method: update

Description:

Update an existing system group.

Parameters:

- string sessionKey
- string systemGroupName
- string description

Returns:

- struct - Server Group
 - int "id"
 - string "name"
 - string "description"
 - int "org_id"
 - int "system_count"
-

user

(Return to Handler List)

Available methods

- addAssignedSystemGroup
- addAssignedSystemGroups
- addDefaultSystemGroup
- addDefaultSystemGroups
- addRole
- create
- create
- delete
- disable
- enable
- getDetails

- `getLoggedInTime`
- `listAssignableRoles`
- `listAssignedSystemGroups`
- `listDefaultSystemGroups`
- `listRoles`
- `listUsers`
- `removeAssignedSystemGroup`
- `removeAssignedSystemGroups`
- `removeDefaultSystemGroup`
- `removeDefaultSystemGroups`
- `removeRole`
- `setDetails`
- `usePamAuthentication`

Description

User namespace contains methods to access common user functions available from the web user interface.

Namespace: `user`

Method: `addAssignedSystemGroup`

Description:

Add system group to user's list of assigned system groups.

Parameters:

- `string sessionKey`
- `string login` - User's login name.
- `string serverGroupName`
- `boolean setDefault` - Should system group also be added to user's list of default system groups.

Returns:

- `int` - 1 on success, exception thrown otherwise.
-

Method: `addAssignedSystemGroups`

Description:

Add system groups to user's list of assigned system groups.

Parameters:

- `string sessionKey`
- `string login` - User's login name.
- `array:`

- string - serverGroupName
- boolean setDefault - Should system groups also be added to user's list of default system groups.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addDefaultSystemGroup

Description:

Add system group to user's list of default system groups.

Parameters:

- string sessionKey
- string login - User's login name.
- string serverGroupName

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addDefaultSystemGroups

Description:

Add system groups to user's list of default system groups.

Parameters:

- string sessionKey
- string login - User's login name.
- array:

- string - serverGroupName

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: addRole

Description:

Adds a role to a user.

Parameters:

- string sessionKey
- string login - User login name to update.
- string role - Role label to add. Can be any of: satellite_admin, org_admin, channel_admin, config_admin, system_group_admin, activation_key_admin, or monitoring_admin.

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: create

Description:

Create a new user.

Parameters:

- `string sessionKey`
- `string desiredLogin` - Desired login name, will fail if already in use.
- `string desiredPassword`
- `string firstName`
- `string lastName`
- `string email` - User's e-mail address.

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: create

Description:

Create a new user.

Parameters:

- `string sessionKey`
- `string desiredLogin` - Desired login name, will fail if already in use.
- `string desiredPassword`
- `string firstName`
- `string lastName`
- `string email` - User's e-mail address.
- `int usePamAuth` - 1 if you wish to use PAM authentication for this user, 0 otherwise.

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: delete

Description:

Delete a user.

Parameters:

- `string sessionKey`
- `string login` - User login name to delete.

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: disable

Description:

Disable a user.

Parameters:

- `string sessionKey`
- `string login` - User login name to disable.

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: enable

Description:

Enable a user.

Parameters:

- `string sessionKey`
- `string login` - User login name to enable.

Returns:

- `int - 1` on success, exception thrown otherwise.
-

Method: getDetails

Description:

Returns the details about a given user.

Parameters:

- `string sessionKey`
- `string login` - User's login name.

Returns:

- `struct` - user details
 - `string "first_names"` - deprecated, use `first_name`
 - `string "first_name"`
 - `string "last_name"`
 - `string "email"`
 - `int "org_id"`
 - `string "prefix"`

- string "last_login_date"
 - string "created_date"
 - boolean "enabled" - true if user is enabled, false if the user is disabled
-

Method: getLoggedInTime

Description:

Returns the time user last logged in.

Parameters:

- string sessionKey
- string login - User's login name.

Returns:

- dateTime.iso8601
-

Method: listAssignableRoles

Description:

Returns a list of user roles that this user can assign to others.

Parameters:

- string sessionKey

Returns:

- array:
 - string - (role label)
-

Method: listAssignedSystemGroups

Description:

Returns the system groups that a user can administer.

Parameters:

- string sessionKey
- string login - User's login name.

Returns:

- array:
 - struct - system group
 - int "id"
 - string "name"

- string "description"
 - int "system_count"
 - int "org_id" - Organization ID for this system group.
-

Method: listDefaultSystemGroups

Description:

Returns a user's list of default system groups.

Parameters:

- string sessionKey
- string login - User's login name.

Returns:

- array:
 - struct - system group
 - int "id"
 - string "name"
 - string "description"
 - int "system_count"
 - int "org_id" - Organization ID for this system group.
-

Method: listRoles

Description:

Returns a list of the user's roles.

Parameters:

- string sessionKey
- string login - User's login name.

Returns:

- array:
 - string - (role label)
-

Method: listUsers

Description:

Returns a list of users in your organization.

Parameters:

- string sessionKey

Returns:

- array:
 - struct - user
 - int "id"
 - string "login"
 - string "login_uc" - upper case version of the login
 - boolean "enabled" - true if user is enabled, false if the user is disabled
-

Method: removeAssignedSystemGroup

Description:

Remove system group from the user's list of assigned system groups.

Parameters:

- string sessionKey
- string login - User's login name.
- string serverGroupName
- boolean setDefault - Should system group also be removed from the user's list of default system groups.

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removeAssignedSystemGroups

Description:

Remove system groups from a user's list of assigned system groups.

Parameters:

- string sessionKey
- string login - User's login name.
- array:
 - string - serverGroupName
- boolean setDefault - Should system groups also be removed from the user's list of default system groups.

Returns:

- int - 1 on success, exception thrown otherwise.

Method: removeDefaultSystemGroup

Description:

Remove a system group from user's list of default system groups.

Parameters:

- string sessionKey
- string login - User's login name.
- string serverGroupName

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removeDefaultSystemGroups

Description:

Remove system groups from a user's list of default system groups.

Parameters:

- string sessionKey
- string login - User's login name.
- array:
 - string - serverGroupName

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: removeRole

Description:

Remove a role from a user.

Parameters:

- string sessionKey
- string login - User login name to update.
- string role - Role label to remove. Can be any of: satellite_admin, org_admin, channel_admin, config_admin, system_group_admin, activation_key_admin, or monitoring_admin.

Returns:

- int - 1 on success, exception thrown otherwise.

Method: setDetails

Description:

Updates the details of a user.

Parameters:

- string sessionKey
- string login - User's login name.
- struct - user details
 - string "first_names" - deprecated, use first_name
 - string "first_name"
 - string "last_name"
 - string "email"
 - string "prefix"
 - string "password"

Returns:

- int - 1 on success, exception thrown otherwise.
-

Method: usePamAuthentication

Description:

Toggles whether or not a user uses PAM authentication or basic RHN authentication.

Parameters:

- string sessionKey
- string login - User's login name.
- int pam_value
 - 1 to enable PAM authentication
 - 0 to disable.

Returns:

- int - 1 on success, exception thrown otherwise.
-