



Full credit is given to the above companies including the OS that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'subscription-manager.8'

\$ man subscription-manager.8

subscription-manager(8) Subscription Management subscription-manager(8)

NAME

subscription-manager - Registers systems to a subscription management service and then attaches and manages subscriptions for software products.

SYNOPSIS

subscription-manager command [options]

DESCRIPTION

subscription-manager is a client program that registers a system with a subscription management service such as the Customer Portal Subscription Management service or on-premise Subscription Asset Manager. Red Hat provides content updates and support by issuing subscriptions for its products. These subscriptions are applied to systems; once a subscription for a product is attached to a system, that system is allowed to install, update, and receive support for that software product. IT administrators need to track these subscriptions and how they are attached. This subscription management is a feature available for Red Hat platforms version 5.7 (and later) and version 6.1 (and later).

For RHEL systems, content is delivered through the Red Hat Customer Portal. Subscriptions and systems are managed globally through the Red Hat subscription management service, which is integrated with the Customer Portal. Subscriptions are managed for the local system by using the Red Hat Subscription Manager tool. Subscription Manager is a local client which connects a system with the subscription management service.

subscription-manager is the command-line based client for the Red Hat Subscription Manager tool.

Subscription Manager performs several key operations:

- * It registers systems to the Red Hat subscription management service and adds the system to the inventory. Once a system is registered, it can receive updates based on its subscriptions to any kind of software products.
- * It lists both available and used subscriptions.
- * It allows administrators to both attach specific subscriptions to a system and remove those subscriptions.

Subscription Manager can be used to auto-attach subscriptions to a system, as well. The subscription-manager command can even be invoked as part of a kickstart process.

Available subscriptions are based on the specific information about the system's architecture. A subscription is only considered available if the platform and hardware can support that specific product.

Subscription Manager also collects and summarizes system facts related to its hardware, operating system, and other characteristics. These facts can be edited in the Subscription Manager configuration and displayed through Subscription Manager.

There is also a Subscription Manager GUI, which can be invoked simply by running subscription-manager-gui from the command line.

Subscription management is only available for RHEL 5.7/6.1 and later systems. Older systems should register to Red Hat Network Classic using the rhn_register command.

subscription-manager has specific options available for each command, depending on what operation is being performed. Subscription Manager commands are related to the different subscription operations:

Note: Please note that using commands that require providing a username using `--username` , a password using `--password` , an organization using `--org` , or environments using `--environments` must be passed as system arguments in a non-interactive session.

1. register
2. unregister
3. attach
4. auto-attach
5. remove
6. release
7. import
8. redeem
9. list
10. refresh
11. environments
12. repos
13. orgs
14. plugins
15. identity
16. facts
17. clean
18. config
19. version
20. status
21. syspurpose
22. repo-override

Following commands were deprecated: addons, role, service-level, sub? scribe, unsubscribe, usage, and activate

COMMON OPTIONS

`-h, --help`

Prints the specific help information for the given command.

`--proxy=PROXY`

Uses an HTTP proxy. The PROXY name has the format hostname:port.

`--proxyuser=PROXYUSERNAME`

Gives the username to use to authenticate to the HTTP proxy.

`--proxypass=PROXYPASSWORD`

Gives the password to use to authenticate to the HTTP proxy.

`--noproxy=NOPROXY`

Specifies a list of domain suffixes which should bypass the HTTP proxy.

`--no-progress-messages`

Disables progress messages that are being displayed when waiting for server response.

REGISTER OPTIONS

The register command registers a new system to the subscription management service.

Note: Please note that using commands that require providing a username using `--username` , a password using `--password` , an organization using `--org` , or environments using `--environments` must be passed as system arguments in a non-interactive session.

`--username=USERNAME`

Gives the username for the account which is registering the system; this user account is usually tied to the user account for the content delivery system which supplies the content. Optional, for user-based authentication.

`--password=PASSWORD`

Gives the user account password.

`--token=TOKEN`

Token to use when authorizing against the server.

`--serverurl=SERVER_HOSTNAME`

Passes the name of the subscription service with which to register the system. The default value, if this is not given, is the Customer Portal Subscription Management service, `subscrip?`

tion.rhsm.redhat.com. If there is an on-premise subscription service such as Subscription Asset Manager, this parameter can be used to submit the hostname of the subscription service. For Subscription Asset Manager, if the Subscription Manager tool is configured with the Subscription Asset Manager RPM, then the default value for the `--serverurl` parameter is for the on-premise Subscription Asset Manager server.

`--baseurl=https://CONTENT_SERVICE:PORT/PREFIX`

Passes the name of the content delivery service to configure the yum service to use to pull down packages. If there is an on-premise subscription service such as Subscription Asset Manager or CloudForms System Engine, this parameter can be used to submit the URL of the content repository, in the form `https://server_name:port/prefix`. PREFIX in particular depends on the service type. For example, `https://sam.example.com:8088/sam` is the baseurl for a SAM service. `https://sat6.example.com/pulp/repos` is the baseurl for a Satellite 6 service with the hostname `sat6.example.com`. `https://cdn.redhat.com` is the baseurl for the Red Hat CDN.

`--name=SYSTEM_NAME`

Sets the name of the system to register. This defaults to the hostname.

`--consumerid=CONSUMERID`

References an existing system inventory ID to resume using a previous registration for this system. The ID is used as an inventory number for the system in the subscription management service database. If the system's identity is lost or corrupted, this option allows it to resume using its previous identity and subscriptions.

`--activationkey=KEYS`

Gives a comma-separated list of product keys to use to redeem or apply specific subscriptions to the system. This is used for preconfigured systems which may already have products installed.

Activation keys are issued by an on-premise subscription management service, such as Subscription Asset Manager.

When the `--activationkey` option is used, it is not necessary to use the `--username` and `--password` options, because the authentication information is implicit in the activation key.

For example:

```
subscription-manager register --org="IT Dept" --activationkey=1234abcd
```

`--auto-attach`

Automatically attaches compatible subscriptions to this system.

`--servicelevel=LEVEL`

Sets the preferred service level to use with subscriptions added to the system. Service levels are commonly premium, standard, and none, though other levels may be available depending on the product and the contract.

`--force`

When the system is already registered, a new attempt to register will fail with a message reminding the user that the system is already registered. However, passing the `--force` option will implicitly attempt to unregister the system first. Beware that the `--force` option does not guarantee a successful registration.

For example, if the registration with `--force` includes a different `--serverurl` than was used for the original registration, the implicit call to unregister from the original entitlement server will fail with invalid credentials and the registration with `force` will be aborted. In this case, the user should explicitly unregister from the original entitlement server. If unregistering is not possible, then running `subscription-manager clean` will effectively abandon the original registration identity and entitlements. Once cleaned, registering a new system identity should succeed with or without `force`.

`--org=ORG`

Assigns the system to an organization. Infrastructures which are managed on-site may be multi-tenant, meaning that there are mul?

multiple organizations within one customer unit. A system may be assigned manually to one of these organizations. When a system is registered with the Customer Portal, this is not required.

When a system is registered with an on-premise application such as Subscription Asset Manager, this argument is required, unless there is only a single organization configured.

`--environments=ENV`

Registers the system to one or more environments within an organization. This is a comma-separated list and the order is maintained.

`--release=VERSION`

Shortcut for "release --set=VERSION"

UNREGISTER OPTIONS

The unregister command does two important things. Firstly, it will implicitly remove all of the currently attached subscriptions thereby returning the consumed quantity of entitlements back to their subscription pools making them available for other consumers. Secondly, it will remove the system's consumer identity thereby removing its contact with the currently configured subscription management service.

This command has no options.

ATTACH OPTIONS

The attach command applies a specific subscription to the system. This command is not possible to use, when the content access mode of the organization to which the system is registered is simple content access mode.

`--auto` Automatically attaches the best-matched compatible subscription or subscriptions to the system. This is the default unless `--pool` or `--file` are used.

`--pool=POOLID`

Gives the ID for the subscriptions pool (collection of products) to attach to the system. This overrides the default of `--auto`.

`--file=FILE`

Specifies a file from which to read whitespace-delimited pool

IDs. If FILE is "-", the pool IDs will be read from stdin. This overrides the default of --auto.

--quantity=NUMBER

Attaches a specified number of subscriptions to the system. Subscriptions may have certain limits on them, like the number of sockets on the system or the number of allowed virtual guests. It is possible to attach multiple subscriptions (or stacking subscriptions) to cover the number of sockets, guests, or other characteristics. May not be used with an auto-attach.

--servicelevel=LEVEL

Sets the preferred service level to use with subscriptions automatically attached to the system. Service levels are commonly premium, standard, and none, though other levels may be available depending on the product and the contract. This option cannot be used when attaching specific pools via --pool or --file.

AUTO-ATTACH OPTIONS

The auto-attach command sets whether the ability to check, attach, and update subscriptions occurs automatically on the system. Auto-attaching subscriptions checks the currently-installed products, attached subscriptions, and any changes in available subscriptions every four hours using the rhsmcertd daemon.

--enable

Enables the auto-attach option for the system. If there is any change in the subscriptions for the system, any subscriptions expire, or any new products are installed, then subscription-manager detects the changes and automatically attaches the appropriate subscriptions so that the system remains covered.

--disable

Disables the auto-attach option for the system. If auto-attach is disabled, then any changes in installed products or subscriptions for the system (including expired subscriptions) must be addressed manually by the administrator.

--show Shows whether auto-attach is enabled on the systems.

REMOVE OPTIONS

The `remove` command removes a subscription from the system. (This does not uninstall the associated products.)

`--serial=SERIALNUMBER`

Gives the serial number of the subscription certificate for the specific product to remove from the system. Subscription certificates attached to a system are in a certificate, in `/etc/pki/entitlement/<serial_number>.pem`. To remove multiple subscriptions, use the `--serial` option multiple times.

`--pool=POOLID`

Removes all subscription certificates for the specified pool id from the system. To remove multiple sets of subscriptions, use the `--pool` option multiple times.

`--all` Removes all of the subscriptions attached to a system.

RELEASE OPTIONS

The `release` command sets a sticky OS version to use when installing or updating packages. This sets a preference for the minor version of the OS, such as 6.2 or 6.3. This can prevent unplanned or unsupported operating system version upgrades when an IT environment must maintain a certified configuration.

`--list` Lists the available OS versions. If a release preference is not set, then there is a message saying it is not set.

`--set=RELEASE`

Sets the minor (Y-stream) release version to use, such as 6.3.

`--unset`

Removes any previously set release version preference.

SYSPURPOSE OPTIONS

The `syspurpose` command displays the current configured syspurpose preferences for the system.

The `syspurpose` command has subcommands for all the various syspurpose preferences and attributes:

1. `addons`
2. `role`

3. service-level

4. usage

--show Shows the system's current set of syspurpose preference format?
ted as JSON. Single-valued entries for which there is no value
will be included in the output with a value of "". List entries
which have no value will be included in the output with a value
of "[]" (less the quotes).

addons options

The addons subcommand displays the current configured addons system
purpose attribute preference for products installed on the system. For
example, if the addons preference is ADDON1, then a subscription with a
ADDON1 addon is selected when auto-attaching subscriptions to the sys?
tem.

--show Shows the system's current addons preference. If a addons is not
set, then there is a message saying it is not set.

--list Lists the available addons system purpose values.

--username=USERNAME

Gives the username for the account to use to connect to the or?
ganization account [Usable with --list on unregistered systems].

--password=PASSWORD

Gives the user account password [Usable with --list on unregis?
tered systems].

--token=TOKEN

Token to use when authorizing against the server [Usable with
--list on unregistered systems].

--org=ORG

Identifies the organization for which the addons apply [Usable
with --list on unregistered systems].

--add=ADDON

Addon to add to the list of requested addons for this system

--remove=ADDON

Remove the addon from the list of requested addons.

--unset

Removes all addons from the list of requested addons.

role options

The role subcommand displays the current configured role preference for products installed on the system. For example, if the role preference is "Red Hat Enterprise Linux Server", then a subscription with a "Red Hat Enterprise Linux Server" role is selected when auto-attaching subscriptions to the system.

`--show` Shows the system's current role preference. If a role is not set, then there is a message saying it is not set.

`--list` Lists the available role system purpose values.

`--username=USERNAME`

Gives the username for the account to use to connect to the organization account [Usable with `--list` on unregistered systems].

`--password=PASSWORD`

Gives the user account password [Usable with `--list` on unregistered systems].

`--token=TOKEN`

Token to use when authorizing against the server [Usable with `--list` on unregistered systems].

`--org=ORG`

Identifies the organization for which the role applies [Usable with `--list` on unregistered systems].

`--set=ROLE`

Role to apply to this system

`--unset`

Removes any previously set role preference.

service-level options

The service-level subcommand displays the current configured service level preference for products installed on the system. For example, if the service-level preference is standard, then a subscription with a standard service level is selected when auto-attaching subscriptions to the system.

`--serverurl=SERVER_URL`

Server URL in the form of https://hostname:port/prefix [Usable on unregistered systems].

--insecure

Do not check the server SSL certificate against available certificate authorities

--show Shows the system's current service-level preference. If a service level is not set, then there is a message saying it is not set.

--list Lists the available service levels.

--username=USERNAME

Gives the username for the account to use to connect to the organization account [Usable with --list on unregistered systems].

--password=PASSWORD

Gives the user account password [Usable with --list on unregistered systems].

--token=TOKEN

Token to use when authorizing against the server [Usable with --list on unregistered systems].

--set=SERVICE_LEVEL

Service level to apply to this system

--unset

Removes any previously set service-level preference.

usage options

The usage subcommand displays the current configured usage preference for products installed on the system. For example, if the usage preference is "Production", then a subscription with a "Production" usage is selected when auto-attaching subscriptions to the system.

--show Shows the system's current usage preference. If a usage is not set, then there is a message saying it is not set.

--list Lists the available usage system purpose values.

--username=USERNAME

Gives the username for the account to use to connect to the organization account [Usable with --list on unregistered systems].

--password=PASSWORD

Gives the user account password [Usable with --list on unregistered systems].

--token=TOKEN

Token to use when authorizing against the server [Usable with --list on unregistered systems].

--org=ORG

Identifies the organization for which the usage applies [Usable with --list on unregistered systems].

--set=USAGE

Usage to apply to this system

--unset

Removes any previously set usage preference.

IMPORT OPTIONS

The import command imports and applies a subscription certificate for the system which was generated externally, such as in the Customer Portal, and then copied over to the system. Importing can be necessary if a system is preconfigured in the subscription management service or if it is offline or unable to access the subscription management service but it has the proper, relevant subscriptions attached to the system.

--certificate=CERTIFICATE_FILE

Points to a certificate PEM file which contains the subscription certificate. This can be used multiple times to import multiple subscription certificates.

REDEEM OPTIONS

The redeem command is used for systems that are purchased from third-party vendors that include a subscription. The redemption process essentially auto-attaches the preselected subscription that the vendor supplied to the system.

--email=EMAIL

Gives the email account to send the redemption notification message to.

--locale=LOCALE

Sets the locale to use for the message. If none is given, then it defaults to the local system's locale.

LIST OPTIONS

The list command lists all of the subscriptions that are compatible with a system. The options allow the list to be filtered by subscriptions that are used by the system or unused subscriptions that are available to the system.

`--afterdate=YYYY-MM-DD`

Shows pools that are active on or after the given date. This is only used with the `--available` option.

`--all` Lists all possible subscriptions that have been purchased, even if they don't match the architecture of the system. This is used with the `--available` option.

`--available`

Lists available subscriptions which are not yet attached to the system.

`--consumed`

Lists all of the subscriptions currently attached to the system.

`--installed`

Lists products which are currently installed on the system which may (or may not) have subscriptions associated with them, as well as products with attached subscriptions which may (or may not) be installed. (default)

`--ondate=YYYY-MM-DD`

Sets the date to use to search for active and available subscriptions. The default (if not explicitly passed) is today's date; using a later date looks for subscriptions which will be active then. This is only used with the `--available` option.

`--no-overlap`

Shows pools which provide products that are not already covered; only used with `--available` option.

`--match-installed`

Shows only subscriptions matching products that are currently

installed; only used with --available option.

--matches=SEARCH

Limits the output of --installed, --available and --consumed to only subscriptions or products which contain SEARCH in the subscription or product information, varying with the list requested and the server version.

SEARCH may contain the wildcards ? or * to match a single character or zero or more characters, respectively. The wildcard characters may be escaped with a backslash to represent a literal question mark or asterisk. Likewise, to represent a backslash, it must be escaped with another backslash.

--pool-only

Limits the output of --available and --consumed such that only the pool IDs are displayed. No labels or errors will be printed if this option is specified.

REFRESH OPTIONS

The refresh command pulls the latest subscription data from the server. Normally, the system polls the subscription management service at a set interval (4 hours by default) to check for any changes in the available subscriptions. The refresh command checks with the subscription management service right then, outside the normal interval. Use of the refresh command will clear caches related to the content access mode of the system and allow the system to retrieve fresh data as necessary.

--force

Force regeneration of entitlement certificates on the server before these certificates are pulled from the server.

ENVIRONMENTS OPTIONS

The environments command lists all of the environments that have been configured for an organization. This command is only used for organizations which have a locally-hosted subscription or content service of some kind, like Subscription Asset Manager. The concept of environments -- and therefore this command -- have no meaning for environments which use the Customer Portal Subscription Management services.

--username=USERNAME

Gives the username for the account to use to connect to the organization account.

--password=PASSWORD

Gives the user account password.

--token=TOKEN

Token to use when authorizing against the server.

--org=ORG

Identifies the organization for which to list the configured environments.

--list Lists all of the environments that have been configured for an organization.

--list-enabled

Lists the environments in the order that they have been enabled for this consumer.

--list-disabled

Lists all of the environments that have been configured for an organization but not enabled for this consumer.

--set=SET

Sets an ordered list of one or more comma-separated environments for this consumer.

REPOS OPTIONS

The repos command lists all of the repositories that are available to a system. This command is only used for organizations which have a locally-hosted content service of some kind, like Subscription Asset Manager. With Red Hat's hosted content service, there is only one central repository.

--list Lists all of the repositories that are provided by the content service used by the system.

--list-enabled

Lists all of the enabled repositories that are provided by the content service used by the system.

--list-disabled

Lists all of the disabled repositories that are provided by the content service used by the system.

`--enable=REPO_ID`

Enables the specified repository, which is made available by the content sources identified in the system subscriptions. To enable multiple repositories, use this argument multiple times. Wild cards * and ? are supported. The repositories enabled by this option and disabled by `--disable` are processed in the same order they are specified.

`--disable=REPO_ID`

Disables the specified repository, which is made available by the content sources identified in the system subscriptions. To disable multiple repositories, use this argument multiple times. Wild cards * and ? are supported. The repositories disabled by this option and enabled by `--enable` are processed in the same order they are specified.

ORGS OPTIONS

The `orgs` command lists all of the organizations which are available to the specified user account. A multi-tenant infrastructure may have multiple organizations within a single customer, and users may be restricted to access only a subset of the total number of organizations.

`--username=USERNAME`

Gives the username for the account to use to connect to the organization account.

`--password=PASSWORD`

Gives the user account password.

`--token=TOKEN`

Token to use when authorizing against the server.

`--serverurl=SERVER_HOSTNAME`

Passes the name of the subscription service to use to list all available organizations. The `orgs` command will list all organizations for the specified service for which the user account is granted access. The default value, if this is not given, is the

Customer Portal Subscription Management service, <https://subscription.rhsm.redhat.com:443>. If there is an on-premise subscription service such as Subscription Asset Manager, this parameter can be used to submit the hostname of the subscription service, in the form `[protocol://]servername[:port][/prefix]`.

For Subscription Asset Manager, if the Subscription Manager tool is configured with the Subscription Asset Manager RPM, then the default value for the `--serverurl` parameter is for the on-premise Subscription Asset Manager server.

PLUGIN OPTIONS

The `plugins` command lists the available subscription-manager plugins.

`--list` List the available subscription-manager plugins.

`--listslots`

List the available plugin slots

`--listhooks`

List the available plugin slots and the hooks that handle them.

`--verbose`

Show additional info about the plugins, such as the plugin configuration values.

REPO-OVERRIDE OPTIONS

The `repo-override` command allows the user to manage custom content repository settings

`--repo` The repository to modify (can be specified more than once)

`--add=NAME:VALUE`

Adds a named override with the provided value to repositories specified with the `--repo` option

`--remove=NAME`

Removes a named override from the repositories specified with the `--repo` option

`--remove-all`

Removes all overrides from repositories specified with the `--repo` option

`--list` Lists all overrides from repositories specified with the `--repo`

option

IDENTITY OPTIONS

The identity command handles the UUID of a system, which identifies the system to the subscription management service after registration. This command can simply return the UUID or it can be used to restore the registration of a previously-registered system to the subscription management service.

--regenerate

Requests that the subscription management service issue a new identity certificate for the system, using an existing UUID in the original identity certificate. If this is used alone, then the identity command also uses the original identity certificate to bind to the subscription management service, using certificate-based authentication.

--username=USERNAME

Gives the username for the account which is registering the system; this user account is usually tied to the user account for the content delivery system which supplies the content. Optional, for user-based authentication.

--password=PASSWORD

Gives the user account password. Optional, for user-based authentication.

--token=TOKEN

Token to use when authorizing against the server.

--force

Regenerates the identity certificate for the system using username/password or token authentication. This is used with the --regenerate option. --regenerate alone will use an existing identity certificate to authenticate to the subscription management service. If the certificate is missing or corrupted or in other circumstances, then it may be better to use user authentication rather than certificate-based authentication. In that case, the --force option requires the username or password or

token to be given either as an argument or in response to a prompt.

FACTS OPTIONS

The `facts` command lists the system information, like the release version, number of CPUs, and other architecture information.

`--list` Lists the system information. These are simple attribute: value pairs that reflect much of the information in the `/etc/sysconfig` directory

`cpu.architecture: x86_64`

`cpu.core(s)_per_socket: 1`

`cpu.cpu(s): 2`

`cpu.cpu_family: 6`

`cpu.cpu_mhz: 1861.776`

`cpu.cpu_op-mode(s): 64-bit`

`cpu.cpu_socket(s): 2`

`cpu.hypervisor_vendor: KVM`

`cpu.model: 2`

`cpu.numa_node(s): 1`

`cpu.numa_node0_cpu(s): 0,1`

`cpu.stepping: 3`

`cpu.thread(s)_per_core: 1`

`cpu.vendor_id: GenuineIntel`

`cpu.virtualization_type: full`

`distribution.id: Santiago`

`distribution.name: Red Hat Enterprise Linux Workstation`

`distribution.version: 6.1`

`--update`

Updates the system information. This is particularly important whenever there is a hardware change (such as adding a CPU) or a system upgrade because these changes can affect the subscriptions that are compatible with the system.

CLEAN OPTIONS

The `clean` command removes all of the subscription and identity data from the local system without affecting the system information in the subscription management service. This means that any of the subscriptions applied to the system are not available for other systems to use.

The `clean` command is useful in cases where the local subscription information is corrupted or lost somehow, and the system will be re-registered using the `register --consumerid=EXISTING_ID` command.

This command has no options.

CONFIG OPTIONS

The `config` command changes the `rhsm.conf` configuration file used by Subscription Manager. Almost all of the connection information used by Subscription Manager to access the subscription management service, content server, and any proxies is set in the configuration file, as well as general configuration parameters like the frequency Subscription Manager checks for subscriptions updates. There are major divisions in the `rhsm.conf` file, such as `[server]` which is used to configure the subscription management service. When changing the Subscription Manager configuration, the settings are identified with the format `section.name` and then the new value. For example:

```
server.hostname=newsubscription.example.com
```

`--list` Prints the current configuration for Subscription Manager.

`--remove=section.name`

Deletes the current value for the parameter without supplying a new parameter. A blank value tells Subscription Manager to use service default values for that parameter. If there are no defaults, then the feature is ignored.

`--section.name=VALUE`

Sets a parameter to a new, specified value. This is commonly used for connection settings:

* `server.hostname` (subscription management service)

* `server.proxy`

* `server.proxy_port`

* `server.proxy_user`

- * server.proxy_password
- * rhsm.baseurl (content server)
- * rhsm.certFrequency

VERSION OPTIONS

The `version` command displays information about the current Subscription Manager package, the subscription service the system is registered to (if it is currently registered), and the subscription management server that the system is configured to use. For example:

```
[root@server ~]# subscription-manager version
server type: Red Hat Subscription Management
subscription management server: 0.9.18-1
subscription management rules: 5.9
subscription-manager: 1.12.1-1.git.28.5cd97a5.fc20
python-rhsm: 1.11.4-1.git.1.2f38ded.fc20
```

This command has no options.

STATUS OPTIONS

The `status` command shows the current status of the products and attached subscriptions for the system. If some products are not fully covered or subscriptions have expired, then the `status` command shows why subscriptions are not current and returns an error code.

```
[root@server ~]# subscription-manager status
```

```
+-----+
```

```
System Status Details
```

```
+-----+
```

```
Overall Status: Current
```

`--ondate=DATE`

Shows the system status for a specific date in the future. The format of the date is YYYY-MM-DD.

```
[root@server ~]# subscription-manager status --ondate=2014-01-01
```

```
+-----+
```

```
System Status Details
```

```
+-----+
```

```
Overall Status: Insufficient
```

DEPRECATED COMMANDS

As the structures of subscription configuration have changed, some of the original management commands have become obsolete. These commands have been replaced with updated commands.

subscribe

This has been replaced with `attach`. A similar registration operation, `--subscribe`, has also been replaced with `--auto-attach`.

unsubscribe

This has been replaced with `remove`.

activate

This has been replaced with `redeem`.

addons This has been replaced with `syspurpose addons`.

role This has been replaced with `syspurpose role`.

service-level

This has been replaced with `syspurpose service-level`.

usage This has been replaced with `syspurpose usage`.

USAGE

`subscription-manager` has two major tasks:

1. Handling the registration for a given system to a subscription management service
2. Handling the product subscriptions for installed products on a system

`subscription-manager` makes it easier for network administrators to maintain parity between software subscriptions and updates and their installed products by tracking and managing what subscriptions are attached to a system and when those subscriptions expire or are exceeded.

REGISTERING AND UNREGISTERING MACHINES

A system is either registered to a subscription management service -- which makes all of the subscriptions available to the system -- or it is not registered. Unregistered systems necessarily lack valid software subscriptions because there is no way to record that the subscriptions have been used nor any way to renew them.

The default subscription management service in the Subscription Manager

configuration is the Customer Portal Subscription Management service.

The configuration file can be edited before the system is registered to point to an on-premise subscription management service like Subscription Asset Manager.

Systems are usually registered to a subscription management service as part of their initial configuration, such as the kickstart process.

However, systems can be registered manually after they are configured, can be removed from a content service, or re-registered.

If a system has never been registered (not even during first boot), then the register command will register the system with whatever subscription management service is configured in the `/etc/rhsm/rhsm.conf` file. This command requires, at a minimum, the username and password or token for an account to connect to the subscription management service.

If the credentials aren't passed with the command, then subscription-manager prompts for the username and password interactively.

When there is a single organization or when using the Customer Portal Subscription Management service, all that is required is the username/password set or the token is used. For example:

```
subscription-manager register --username=admin --password=secret or subscription-manager register
--token=eyJhbGciOiJSUzI1NiIsI...stGc_2bFDQC8CENE0o
```

With on-premise subscription services, such as Subscription Asset Manager, the infrastructure is more complex. The local administrator can define independent groups called organizations which represent physical or organizational divisions (`--org`). Those organizations can be subdivided into environments (`--environment`). Optionally, the information about what subscription service (`--serverurl`) and content delivery network (`--baseurl`) to use for the system registration can also be passed (which overrides the Red Hat Subscription Manager settings). The server and content URLs are usually configured in the Subscription Manager configuration before registering a system.

```
subscription-manager register --username=admin --password=secret
--org="IT Dept" --environment="local dev" --serverurl=local-cloudforms.example.com
--baseurl=https://local-cloudforms.example.com:8088/cfFe
```


If a system is in a multi-tenant environment and the organization is not provided with the registration request, registration fails with a remote server error. In the rhsm.log, there will be errors about being unable to load the owners interface.

If a system is registered and then somehow its subscription information is lost -- a drive crashes or the certificates are deleted or corrupted -- the system can be re-registered, with all of its subscriptions re? stored, by registering with the existing ID.

```
subscription-manager register --username=admin  
--password=secret --consumerid=1234abcd
```

A system uses an SSL client certificate (its identity certificate) to authenticate to the subscriptions system to check for updates or changes to subscriptions. If the identity certificate is lost or corrupted, it can be regenerated using the identity command.

```
subscription-manager identity --regenerate
```

Using the --force option will prompt for the username and password for the account, if one isn't given, and then return the new inventory ID and the hostname of the registered system.

```
subscription-manager identity --force
```

Username: jsmith

Password:

```
eff9a4c9-3579-49e5-a52f-83f2db29ab52 server.example.com
```

A system is unregistered and removed from the subscription management service simply by running the unregister command. Unregistering a system and removing its attached subscriptions can free up subscriptions when a system is taken offline or moved to a different department.

```
subscription-manager unregister
```

An option with registration, --auto-attach, will automatically attach the subscriptions pool which best matches the system architecture and configuration to the newly-registered system. This option attaches subscriptions as part of the registration process, rather than separately managing subscriptions.

```
subscription-manager register --username=admin --password=secret
```

```
--auto-attach
```

Auto-attach also supports an option to set a preferred service level with the selected subscriptions, the `--servicelevel` option. In this case, the `--servicelevel` option sets a preference that helps the auto-attach process select appropriate subscriptions. For example, if the preferred service level for a production server is premium, and there are three matching subscriptions with different service levels (none, standard, and premium), the auto-attach process selects the subscription which offers a premium service level.

```
subscription-manager register --username=admin --password=secret
--auto-attach --servicelevel=premium
```

LISTING, ATTACHING, AND REMOVING SUBSCRIPTIONS FOR PRODUCTS

A subscription is essentially the right to install, use, and receive updates for a Red Hat product. (Sometimes multiple individual software products are bundled together into a single subscription.) When a system is registered, the subscription management service is aware of the system and has a list of all of the possible product subscriptions that the system can install and use. A subscription is applied to a system when the system is attached to the subscription pool that makes that product available. A system releases or removes that subscription (meaning, it removes that subscription so that another system can use that subscription count).

`list` command shows you what subscriptions are available specifically to the system (meaning subscriptions which are active, have available quantities, and match the hardware and architecture) or all subscriptions for the organization. Using the `--ondate` option shows subscriptions that are or will be active at a specific time (otherwise, it shows subscriptions which are active today).

```
subscription-manager list --available --ondate=2012-01-31
```

```
+-----+
```

```
Available Subscriptions
```

```
+-----+
```

```
Subscription Name: Red Hat Enterprise Linux
```

SKU: SYS0395
Pool Id: 8a85f981302cbaf201302d899adf05a9
Quantity: 249237
Service Level: None
Service Type: None
Multi-Entitlement: No
Starts: 01/01/2021
Ends: 01/01/2022
Machine Type: physical

The list command can also be used to show what products you currently have installed, as a way of tracking what products you have versus what subscriptions you have on the system.

```
subscription-manager list --installed
+-----+
    Installed Product Status
+-----+
ProductName: Red Hat Enterprise Linux Server
Product ID: 69
Version: 6.3
Arch: x86_64
Status: Subscribed
Started: 07/26/2012
Ends: 08/31/2015
```

The list can be filtered to only include products or subscriptions that match the query string provided to --matches option.

```
subscription-manager list --installed --matches="*Server*"
+-----+
    Installed Product Status
+-----+
ProductName: Red Hat Enterprise Linux Server
Product ID: 69
Version: 6.3
Arch: x86_64
```

Status: Subscribed

Started: 07/26/2012

Ends: 08/31/2015

Attaching a subscription requires the ID for the subscription pool (the --pool option). For example:

```
subscription-manager attach
--pool=ff8080812bc382e3012bc3845da100d2
```

As with the register command, the system can be auto-attached to the best-fitting subscriptions. This is the default action and is equivalent to using the --auto option:

```
subscription-manager attach
```

Auto-attach also supports an option to set a preferred service level with the selected subscriptions, the --servicelevel option. In this case, the --servicelevel option sets a preference that helps the auto-attach process select appropriate subscriptions. For example, if the preferred service level for a production server is premium, and there are three matching subscriptions with different service levels (none, standard, and premium), the auto-attach process selects the subscription which offers a premium subscription.

```
subscription-manager attach --servicelevel=premium
```

Some subscriptions define a count based on attributes of the system itself, like the number of sockets or the number of virtual guests on a host. You can combine multiple subscriptions together to cover the count. For example, if there is a four socket server, you can use two subscriptions for "RHEL Server for Two Sockets" to cover the socket count. To specify the number of subscriptions to use, use the --quantity option. For example:

```
subscription-manager attach
--pool=ff8080812bc382e3012bc3845da100d2
--quantity=2
```

Removing subscription from a system releases the subscription back into the pool. The system remains registered with the subscription management service. Each product has an identifying X.509 certificate in?

stalled with it. To remove a subscription for a specific product, specify the serial number (or numbers, in multiple --serial options) of the certificate:

```
subscription-manager remove --serial=1128750306742160
```

Giving the remove command with the --all option removes every subscription the system has used.

REDEEMING EXISTING SUBSCRIPTIONS

Sometimes, a system may come preconfigured with products and subscriptions. Rather than attaching a pool and claiming a subscription, this system simply needs to redeem its existing subscriptions.

After registration, subscriptions on preconfigured systems can be claimed using the redeem command, which essentially auto-attaches the system to its preexisting subscriptions.

```
subscription-manager redeem --email=admin@example.com --org="IT Dept"
```

VIEWING LOCAL SUBSCRIPTION & CONTENT PROVIDER INFORMATION

Red Hat has a hosted environment, through the Customer Portal, that provides centralized access to subscription management and content repositories. However, organizations can use other tools -- like Subscription Manager -- for content hosting and subscription management. With a local content provider, the organization, environments, repositories, and other structural configuration is performed in the content provider. Red Hat Subscription Manager can be used to display this information, using the environments, orgs, and repos commands.

```
subscription-manager repos --list
```

```
subscription-manager environments --username=jsmith
```

```
--password=secret --org=prod
```

or

```
subscription-manager environments --token=eyJhbGciOiJSUzI1NiIsI ... stGc_2bFDQC8CENE0o --org=prod
```

```
subscription-manager orgs --username=jsmith
```

```
--password=secret
```

or

```
subscription-manager orgs --token=eyJhbGciOiJSUzI1NiIsI ... stGc_2bFDQC8CENE0o
```

CHANGING SUBSCRIPTION MANAGER CONFIGURATION

The Subscription Manager CLI and GUI both use the `/etc/rhsm/rhsm.conf` file for configuration, including what content and subscription management services to use and management settings like auto-attaching. This configuration file can be edited directly, or it can be edited using the `config` command. Parameters and values are passed as arguments with the `config` command in the format `--section.parameter=value`, where `section` is the configuration section in the file: `server`, `rhsm`, `rhsmcertd` or `logging`.

For example, to change the hostname of the subscription management service host:

```
subscription-manager config --server.hostname=myserver.example.com
```

The entries in the logging section are somewhat special. The keys in this section are a name of a logger. The values are the logging level. Valid levels are one of: `DEBUG`, `INFO`, `WARNING`, `ERROR`, or `CRITICAL`. Valid logger names are the full module path of any Subscription Manager module. For example: `subscription_manager` or `subscription_manager.managercli`.

There are three main top-level loggers: `subscription_manager`, `rhsm`, and `rhsm-app`. All logger names begin with one of the above.

To set the default log level for all loggers (that are not otherwise set in the logging section), edit the `default_log_level` key in `/etc/rhsm/rhsm.conf`

UPDATING FACTS

The information about a system, such as its hardware and CPU, its operating system versions, and memory, are collected by Subscription Manager in a list of facts. Subscription Manager uses these facts to determine what purchased subscriptions are compatible with the system. Whenever these facts change (such as installing an additional CPU), the facts can be updated immediately using the `facts` command.

```
subscription-manager facts --update
```

The collected facts can also be overridden by creating a JSON file in the `/etc/rhsm/facts/` directory. These have simple formats that define a fact and value:

```
{"fact1": "value1", "fact2": "value2"}
```

Any fact override file must have a .facts extension.

When these fact files are added, running the facts command will update the collected facts with the new, manual facts or values.

SUBSCRIPTIONS AND KICKSTART

The subscription-manager tool can be run as a post-install script as part of the kickstart installation process. This allows subscription management (registering and applying subscriptions) to be automated along with installation. For example:

```
%post --log=/root/ks-post.log
```

```
    /usr/sbin/subscription-manager register --username admin --password secret --org 'east colo' --auto-attach  
--servicelevel=premium --force
```

NETWORK INFORMATION

The subscription-manager tool uses outgoing HTTPS requests. In the default configuration it will use HTTPS on port 443 to the subscription servers `subscription.rhsm.redhat.com` and to the content delivery service `cdn.redhat.com`.

For information about the network addresses that subscription-manager and the subscription-manager yum plugin use see <https://access.redhat.com/site/solutions/59586>

PROXY CONFIGURATION

subscription-manager can be configured to use a proxy in several ways:

- * via standard `HTTP_PROXY`, `HTTPS_PROXY`, `NO_PROXY` environment variables (environment-level settings)
- * via options in `/etc/rhsm/rhsm.conf` (application-level settings)
- * via command-line arguments (command-level overrides)

Although subscription-manager respects environment variables for proxy configuration, this should be avoided in favor of the configuration file, because the daemons (ex. `rhsmcertd`) do not provide ways to modify their environments.

Each option of the proxy configuration (hostname, port, host/domain pattern blacklist, username, password) is read independently, with

precedence being command-line over configuration over environment, and then the resulting set of options is used to configure the proxy configuration.

For example, if the HTTP_PROXY environment variable is set and no_proxy is set in /etc/rhsm/rhsm.conf then both are present in the effective proxy configuration.

If two equivalent options are set in different places, then the precedence determines which value is effective.

For example, the NO_PROXY environment variable is set and the no_proxy configuration file option is set, then the value from the configuration file is the effective value.

LOG FILES

Default log location of the subscription-manager is /var/log/rhsm/rhsm.log. When the program is run under non-root user (e.g. as dnf plugin) the logs are written to \$XDG_CACHE_HOME/rhsm/rhsm.log.

If the directory isn't writable, the logs are printed to stderr.

FILES

- * /etc/pki/consumer/*.pem
- * /etc/pki/entitlement/<serial>.pem
- * /etc/pki/product/*.pem
- * /etc/rhsm/rhsm.conf
- * /etc/rhsm/facts/*.facts
- * /var/log/rhsm/rhsm.log

AUTHORS

Deon Lackey, <dlackey@redhat.com>, and Pradeep Kilambi, <pkilambi@redhat.com>

subscription-manager(8)