



# **Red Hat Subscription Management All Subscription Docs**

## **Introduction to Red Hat Subscription Management Workflows**

to better understand and manage subscriptions for an IT infrastructure

Edition 3



# Red Hat Subscription Management All Subscription Docs Introduction to Red Hat Subscription Management Workflows

---

to better understand and manage subscriptions for an IT infrastructure  
Edition 3

Red Hat Subscription Management Documentation Team  
[rhsm-docs@redhat.com](mailto:rhsm-docs@redhat.com)

## Legal Notice

Copyright © 2013 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

This guide covers background concepts about subscription management, tools, and common use scenarios.

# Table of Contents

<b>1. INTRODUCTION TO SUBSCRIPTION MANAGEMENT .....</b>	<b>3</b>
1.1. The Goals of Subscription Management	3
1.2. Subscriptions: The Relationship Between Systems, Software, and Support	3
1.3. About Relationships Between Subscriptions and Systems	6
1.3.1. Interactions with Subscriptions, Products, and Systems	6
1.3.2. Counting Subscriptions	6
1.4. Subscription Terms	7
1.5. RHN Classic, Satellite, and Channel Entitlements	9
<b>2. TOOLS AND APPLICATIONS FOR SUBSCRIPTION MANAGEMENT .....</b>	<b>10</b>
2.1. Local System Tools (Red Hat Subscription Manager)	10
2.1.1. Launching the Red Hat Subscription Manager UI	11
2.1.2. Running the subscription-manager Command-Line Tool	12
2.2. Customer Portal Subscription Management	14
2.3. Subscription Asset Manager	15
2.4. Hypervisor Processes	16
<b>3. SUBSCRIPTION WORKFLOWS .....</b>	<b>16</b>
3.1. Planning the Workflow to Use	16
3.2. A Note on User Accounts	18
3.3. Customer Portal: Autoattaching Systems	19
3.3.1. The Environment: Small Businesses	19
3.3.2. Workflow	19
3.3.3. Options and Details	20
3.4. Customer Portal: Registration and Manual Subscription	20
3.4.1. The Environment: Small Businesses	21
3.4.2. Workflow	21
3.4.3. Options and Details	22
3.5. Subscription Asset Manager: Direct Subscription Assignments	22
3.5.1. The Environment: Small Businesses to Large Enterprises for Locally-Defined Structure	22
3.5.2. Workflow	23
3.5.3. Details and Options	25
3.6. Subscription Asset Manager: Activation Keys	25
3.6.1. The Environment: Preconfigured Systems	25
3.6.2. Workflow	25
3.6.3. Details and Options	27
3.7. General Management: Firstboot	28
3.8. General Management: Kickstart	29
3.8.1. The Environment: Scripted Environments	30
3.8.2. Workflow	30
3.8.3. Options and Details	30
3.9. General Management: Hypervisors and Guests	30
3.9.1. The Environment: Data Centers, Cloud Environments, Any Environment Using Virtual Machines	31
3.9.2. Workflow	31
3.10. General Management: Disconnected Systems	32
3.10.1. The Environment: Security Environments and Backup Systems	32
3.10.2. Workflow	32
3.10.3. Options and Details	34
3.11. Legacy: Migration from RHN Classic	34
3.11.1. The Environment: Small Businesses with Older Red Hat Enterprise Linux Systems	34
3.11.2. Workflow	35
3.11.3. Options and Details	36

4. REVISION HISTORY ..... 36

# 1. INTRODUCTION TO SUBSCRIPTION MANAGEMENT

IT hardware needs to be managed and clearly inventoried, and the software installed on those machines also needs to be managed and clearly inventoried.

## 1.1. The Goals of Subscription Management

IT administrators face increasing pressure to have an accurate accounting of the software, not just from governmental regulations like Sarbanes-Oxley in the United States, but also to achieve critical industry certifications, such as Payment Card Industry Data Security Standard (PCI-DSS) or SAS-70. Generally, this accounting of software assets is called *software license management*; with Red Hat's subscription model, this is *subscription management*.

Effective subscription management helps enterprises achieve four primary goals:

- *Maintain regulatory compliance* by tracking software attached subscriptions and expiration periods.
- *Simplify IT audits.*
- *Be more effective at assigning subscriptions* by clarifying the relationships between subscriptions and systems.
- *Lower costs and streamline procurement.* While *under*-subscribing systems can run afoul of regulations, *over*-subscribing systems can cause a significant impact on IT budgets.

Red Hat Subscription Management tracks software counts by directly associating a purchased subscription with the machine the product is installed on. Subscription management establishes the *relationship* between the product subscriptions that are available and the elements of the IT infrastructure where those subscriptions are allocated.

This is very similar to license management, which is common in other software. However, licenses define who can use a piece of software. Red Hat is committed to free and open software, so we do not license our software and do not enforce usage requirements. Rather, our subscriptions allow:

- Access to support services
- Content delivery and hosted repositories
- Access to knowledgebases, forums, videos, and other resources

## 1.2. Subscriptions: The Relationship Between Systems, Software, and Support

An IT infrastructure tries to maintain balance between the products that are installed and the licenses or subscriptions that those products require.

There is a reliable pattern in subscription lifecycles:

1. A company (account) buys a subscription to a product. The subscription defines a number of times it can be used (the quantity), the support level, the content repositories, and the period that it is good for.

2. A system is added, or *registered*, to the *inventory* for the subscription service. This means that the subscription service can manage the server and attach subscriptions to it.

The subscription service collects certain facts about the system, including its hardware and installed products, which it uses to help determine what subscriptions the system requires.

3. A subscription is assigned or *attached* to a system.
4. The system downloads software packages and updates from the content delivery network for as long as the subscription is active.

An *inventory* is simply a means to track *what* software is installed and *where* it is installed and *how many* copies are actively being used.

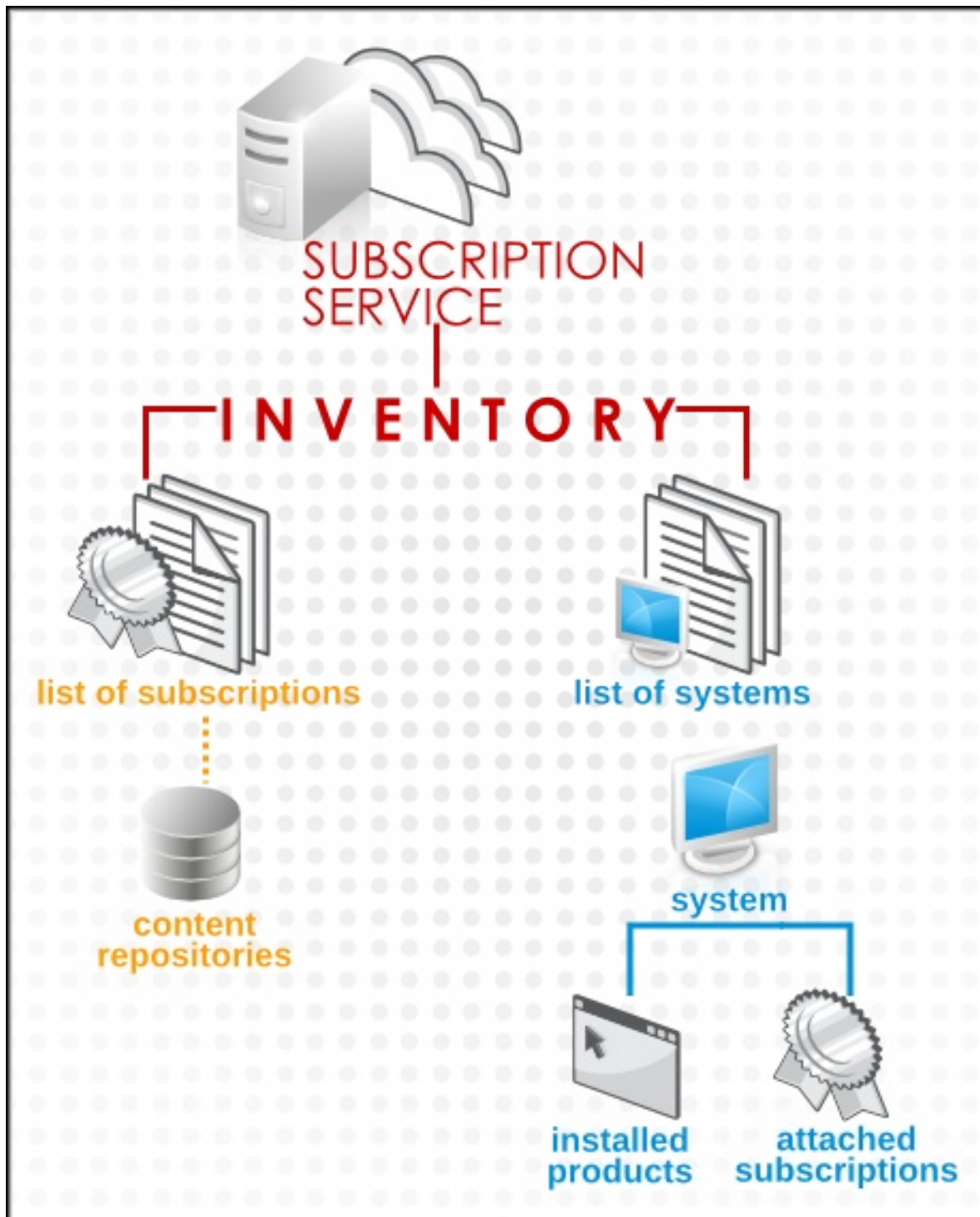


Figure 1. The Structure of the Subscription Service



Each element in the subscription service must be uniquely identified; this allows true relationships to be established between the system, the products, and the subscriptions. This is done using *public-key infrastructure (PKI) certificates*. The subscription service generates and installs these certificates on the local system:

- An **identity certificate** for the system; this is used by the system to authenticate to the subscription service periodically to check for updates.

This is created when the system is registered.

- A **product certificate** for each product installed on the system. Each product in Red Hat has an identifying certificate (but this is not unique to the system).

This is installed on the system as part of installing the product.

- A **subscription certificate** for each subscription attached to the system. This includes information about the subscription from the inventory.

This is installed on the system when a subscription is attached to the system.



## NOTE

Attaching a subscription does not actually perform any installation or updates. The software packages themselves are still maintained with the normal system tools, like **yum**. Conversely, having a product installed does not mean that the appropriate subscriptions have been attached to the system. A system does not require a valid certificate to install a product.

Ultimately, subscription management delivers better information and offers administrators better control over their infrastructures. The primary goals of subscription management are regulatory compliance, infrastructure management, better purchasing decisions, better system maintenance.

All of these goals come down to information — to providing administrators a better understanding of what subscriptions they have and how they are being used with their systems.

- For regulatory compliance and IT audits, Red Hat offers two account-wide views into systems and subscription associations: the utilization views in the Customer Portal and the reporting and dashboards in subscription management applications such as Subscription Asset Manager
- For better effectiveness at assigning subscriptions, system tools allow systems to have all relevant, compatible subscriptions automatically assigned (*attached*) to the system based on what products are installed — *or* for administrators to select and manually attach subscriptions ([Viewing Reports and Notifications](#)). Tools list both available subscriptions and the quantities still available for those subscriptions, so that the right subscription can be attached and the right number of subscriptions are used.
- For better purchasing strategies, subscription management tools help simultaneously track oversubscribing (using the same subscription on too many systems), undersubscribing (not having enough subscriptions for the installed products on a system), and subscription expirations.

For details, see [Viewing Reports and Notifications](#).

Having a time-based picture on subscriptions — not only what is covered but *how long* it is covered — helps administrators control identify when to purchase new subscriptions, so subscriptions are not purchased too early or too late.

Better information translates into better control. The long-range goals for subscription management for Red Hat software is to enhance how administrators understand and manage their infrastructures.

## 1.3. About Relationships Between Subscriptions and Systems

### 1.3.1. Interactions with Subscriptions, Products, and Systems

Products on a system have relationships, dependencies, and conflicts between each other. Likewise, subscriptions have relationships that parallel the relationships of the software it represents. Some subscriptions allow virtual guests, some require other subscriptions, some conflict with other subscriptions.

Subscription define the relationships between installed products and each other and the systems on which those products are installed. Likewise, subscriptions can also define relationships between systems and how they interact within an environment. This is particularly apparent with virtual environments, where subscriptions can define different relationships for physical hosts and virtual guests, but there are other ways that systems can interact, such as data centers and cloud infrastructures. Subscriptions are a part of those meta relationships.

Using subscriptions to define these relationships introduces a lot of flexibility in how products and systems interact:

- Associate a single quantity of a product with a single system (which is the most common relationship).
- Restrict one product so that it cannot be installed on the same system as a specific, different product.
- Keep a system on a consistent service level. Each subscription includes a definition for what service level (e.g., standard or premium) the product has. Subscription clients first try to assign subscriptions of the same service level (and this can be enforced) so that the system has consistent support levels.
- Allow virtual guests to inherit some subscriptions from their host.
- Allow some hosts to have unlimited guests for a data center deployment.
- Allow a single “subscription” to be broken across multiple systems. This works in something like Red Hat Cloud Infrastructure, where a single purchase actually covers four products — Red Hat Enterprise Linux, Red Hat OpenStack, Red Hat Virtualization, and Satellite 6 Management Engine — and those products each have their own subscription which can be used on different systems to create the stack.
- Stack or combine subscriptions of the same type to cover a system.

### 1.3.2. Counting Subscriptions

Part of the subscription service inventory is keeping track of subscriptions – not just what subscriptions are purchased but how many of those subscriptions are available.

When a subscription is first purchased, it defines the quantity of times that the subscription can be used. The subscription count is based on a certain element of the underlying system, most commonly its socket count (but it can be something else, such as the number of cores, depending on the specific subscription). The element of a system or software which is directly covered by a subscription is called an instance.

For example, for the subscription for Red Hat Enterprise Linux for 2 sockets, the product is Red Hat Enterprise Linux and the attribute is a physical socket pair. The socket pair is the instance.

A single subscription quantity is usually tied to a single socket pair (or other attribute). A system with eight sockets, then, requires more subscription quantities to cover its socket count than a four socket system. (This is called stacking.)

This simplistic arrangement, however, does not apply to all subscriptions.

Starting in October 2013, Red Hat began introducing other types of subscription relationships, such as:

- Multiple products with a single subscription (Red Hat Cloud Infrastructure)
- Inheritable subscriptions
- Data center subscriptions, which allow unlimited virtual guests (and only the host requires a specific subscription)

Additionally, the 2013 subscription changes altered how virtual guests are handled in subscriptions. There used to be subscriptions for physical systems and then different subscriptions for virtual guests. In the current subscription model, the same subscription is used for both physical and virtual systems – but the quantity used is different, depending on whether it is a physical system or a virtual one.

As stated previously, a single subscription quantity is used per socket pair on a physical system. A virtual guest counts as a single socket, not a socket pair – so it is essentially half of a subscription quantity. When virtual guests are added to the inventory, the total number of available subscriptions is multiplied by two (the instance multiplier). This allows the subscription count to stay in whole numbers, even with virtual guests taking only a “half” quantity.

However, with some subscriptions counts multiplied by two; data center virtual guests not consuming any individual subscriptions; some subscriptions (Cloud Infrastructure) relating to multiple products installed on different systems; *and* older, pre-2013-style subscriptions all in the same environment — the actual counts listed in the subscription utilization pages or subscription management tools may not appear to reflect the quantities purchased in the contract. The fundamental counts are the same; most of the differences reflect changes to keep the count whole or new, more flexible subscription types.

## 1.4. Subscription Terms

Certain terms are used repeatedly throughout the Red Hat Subscription Management documentation, messages, and tools, and it can be easier to understand the subscription lifecycle and management if those terms are clearly defined.

These are some of the most common, relevant terms for Red Hat Subscription Management.

### **account**

The primary, top-level account for a company. This is the entity that is used within the Customer Portal to define the company and it is where all user and subscription information originates.

### **attach**

Assigning a subscription to a system.

### **autoattach**

A setting on the local system (or an option with the Subscription Manager tools) that periodically checks for new subscriptions, changing subscription statuses for products, and then automatically assigns new subscriptions to cover the changes. The subscriptions are selected based on the best fit

for the system architecture and hardware, installed product, and defined preferences.

**certificate**

A specific file based on the X.509 certificate standard that is used for SSL communication and within a public key infrastructure. This is used to identify elements within Red Hat Subscription Management including identities (systems), installed products, and consumed subscriptions.

**identity**

An entity registered with a subscription management application. An identity usually correlates to a system, but it can also be a hypervisor, domain, an organization within a management application, or other servers such as RHUI and Satellite.

**instance**

An element within a system that is covered by a subscription. For physical systems, an instance is most commonly a socket pair. For virtual environments, an instance is a single virtual guest.

**inventory**

The list of systems, hypervisors, applications, and other entities registered with a subscription application. The inventory also contains a list of available and used subscriptions for the organization.

**organization**

A subdivision in an on-premise application such as Subscription Asset Manager. An organization has its own system inventory and a subset of subscriptions (defined in a manifest). This is a way to define a subscription structure that reflects the IT environment. An organization can be aligned with a physical location or an organizational division in a company.

**pools**

The complete set of subscriptions and quantities allocated to an organization or account.

**preference**

A defined criterion used by autoattach operations to select subscriptions. Preferences are set on the local system, but they actually define attributes within the potential subscriptions that should be evaluated for autoattach. There are two preferences: the service level for the subscription and the operating system minor release (the X.Y version, such as 5.10 or 6.5).

**quantity**

The number of subscriptions used to cover a product or system. A subscription covers a set amount of attributes, such as the number of sockets on a physical system. Multiple subscriptions may be required to cover a given system, based on its hardware and configuration.

**register**

To add a system or other entity (such as an organization or hypervisor) to the subscription management inventory.

**stacking**

Combining multiple subscriptions to cover a product or system. There are rules applied to what subscriptions can be stacked. For example, only subscriptions of the same service level can be stacked. Also, some subscriptions may restrict what products can be installed with it, so some subscriptions may not be available to some systems.

**status**

The cumulative state for a system of all of the installed products and associated valid subscriptions. If all installed products have active subscriptions, the status is *valid* (green). If some products lack subscriptions or do not have enough quantities of subscriptions to cover its configuration, the status is *insufficient* (yellow). If a product or system has no subscriptions, the status is *invalid* (red).

**subscription**

A definition of the products that are available, the support levels, the quantities (or number) of servers that the product can be installed on, architectures that the product is available for, content repositories which supply the product, and other information related to the products.

**subscription management application**

The backend server which interacts with the individual systems by creating an inventory of systems. It also keeps the inventory of subscriptions, including contracts, quantities, and expiration dates. When a new system is registered, when subscriptions are attached, or when products are installed, the subscription management service manages the changes and issues a corresponding certificate to the system to mark the change. The subscription management service also defines rules for products, such as hardware/architecture restrictions, to help with attaching subscriptions.

**system**

Any entity — a physical or virtual machine — which is in the subscription service inventory and which can have subscriptions attached to it.

**utilization**

A summary of the total number of subscriptions available to an organization, and the total number of subscriptions that are attached to Customer Portal Subscription Management, RHN Classic, and different subscription management applications.

## 1.5. RHN Classic, Satellite, and Channel Entitlements

Beginning in Red Hat Enterprise Linux 6.1, the way that subscriptions are defined was changed. In older subscription models — the model used by RHN Classic and Satellite 4 and 5 — a system required a *system entitlement* in order to register the system with the subscription service. Then, the system required *channel entitlements* which granted access to sets of content and software, whatever was defined in the channel.

There was no direct correlation between registered systems, the software installed on them, and the channels to which they were entitled access. There were two separate entitlements pools and even reporting didn't clarify or reveal any relationships.

However, clear subscription tracking is an increasingly critical part of IT maintenance. The older channel-based subscription management process in RHN Classic was great at providing access, but it was very difficult to get an accurate accounting of how subscriptions were applied, how many subscriptions were being used, where subscriptions were being used, and what any individual system was using.

Red Hat Subscription Management uses public-key infrastructure (PKI) certificates to uniquely identify the system, the products on the system, and its attached subscriptions. There is always a correlation, both on the system and in the subscription management service, between systems, products, and subscriptions.

The old and new subscription models are fundamentally different in how they view and structure systems and subscriptions within their inventories. Real-world infrastructures will commonly have a mix of

subscription services and system versions. Any infrastructure with Red Hat Enterprise Linux 4.x, 5.7, and 6.0 systems (and older), registered with RHN Classic or using Satellite 4 and 5 will use the older subscription model. Any systems registered to Customer Portal Subscription Management or using Subscription Asset Manager are using the new subscription model.

There are several different paths that administrators can take to manage these mixed environments.

- Use Satellite 5.6 and Subscription Asset Manager 1.3 for enhanced reporting.
- Migrate Red Hat Enterprise Linux 5 and 6 systems from the old subscription service to a new subscription service (Subscription Asset Manager or Customer Portal Subscription Management); this moves their registration, updates their attached subscriptions, and changes what client tools are used to manage the system from **rhn\_register** to **subscription-manager**.

Both have advantages and disadvantages which can be determined by administrators.

From a high-level, both migrating and enhanced reporting extend the benefits of better subscription tracking and maintenance to legacy systems:

- Consolidate channel entitlements and system entitlements into a single entitlement pool, which is more structurally similar to the new subscription model.
- Map channel entitlements to new product subscriptions.
- Correlate installed products on managed systems with corresponding subscriptions, based on their channel subscriptions.
- Display legacy systems and entitlements in the same style as modern subscription management, to make it easier to create cohesive audit reports.
- Provide unified views of subscriptions and products for legacy systems and systems registered with Red Hat Subscription Management.

## 2. TOOLS AND APPLICATIONS FOR SUBSCRIPTION MANAGEMENT

All Red Hat Enterprise Linux subscriptions automatically include some tools for managing the subscription configuration:

- Red Hat Subscription Manager client tools to manage local systems
- Customer Portal Subscription Management to manage systems and subscription application organizations for a single account globally through the Customer Portal
- Subscription Asset Manager to install an on-premise subscription service

The diversity of tools allows administrators to create a workflow that fits both the business and infrastructure demands of their organization. Since all of these elements are relatively independent of one another, that opens a lot of different potential configuration and deployment scenarios.

### 2.1. Local System Tools (Red Hat Subscription Manager)

Both registration and subscriptions are managed on the local system through UI and CLI tools called *Red Hat Subscription Manager*. The Subscription Manager tracks and displays what subscriptions are available to the local system and what subscriptions have been consumed by the local system. The

Subscription Manager works as a conduit back to the subscription service to synchronize changes like available product quantities or subscription expiration and renewals.



## NOTE

The Red Hat Subscription Manager tools are always run as **root** because of the nature of the changes to the system. However, Red Hat Subscription Manager connects to the subscription service as a user account for the subscription service.

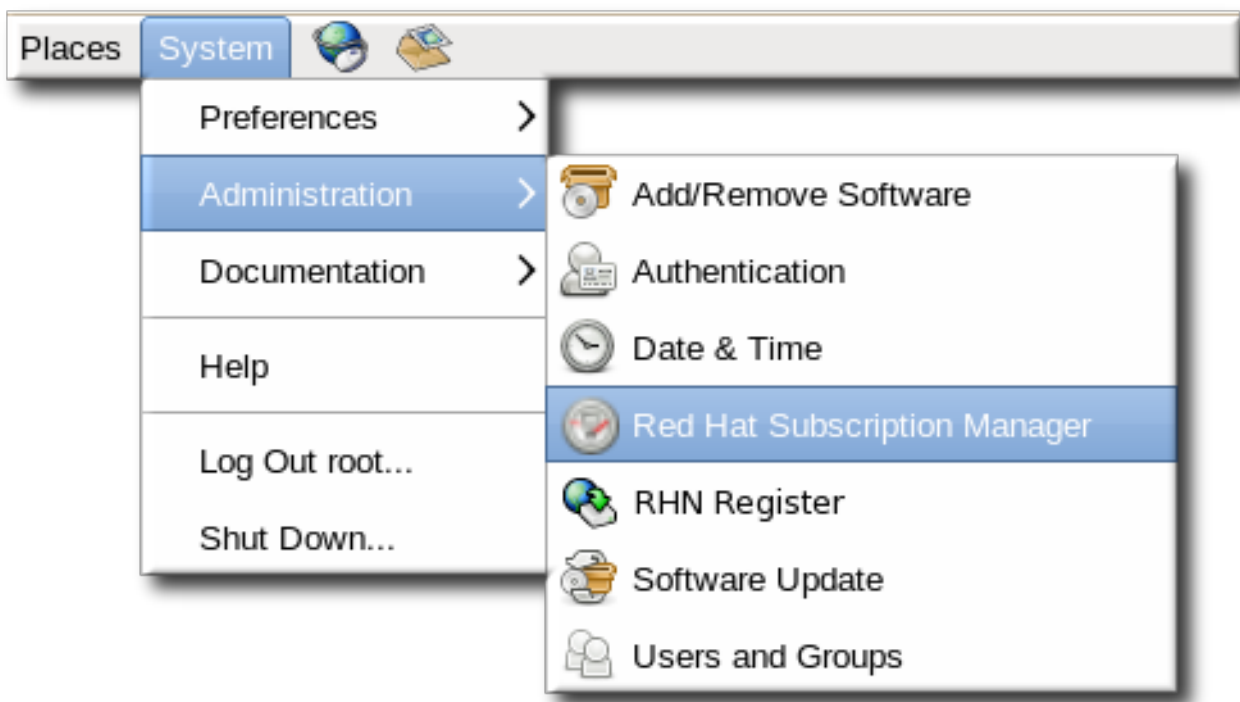
The Subscription Manager handles both registration and subscriptions for a system. The Subscription Manager is part of the **firstboot** process for configuring content and updates, but the system can be registered at any time through the Red Hat Subscription Manager UI or CLI. New subscriptions, new products, and updates can be viewed and applied to a system through the Red Hat Subscription Manager tools.

Red Hat Subscription Manager has two tools in its set, a UI-based client to manage the local machine and a CLI client for advanced users (which can be used to work with other applications or in scripting management tasks, like kickstarting machines).

These tools allow administrators to perform three major tasks directly related to managing subscriptions: registering machines, assigning (attaching) subscriptions to systems, and updating the certificates required for authentication. Some minor operations, like updating system facts, are available to help display and track what subscriptions are available.

### 2.1.1. Launching the Red Hat Subscription Manager UI

Red Hat Subscription Manager is listed as one of the administrative tools in the **System > Administration** menu in the top management bar.



**Figure 2. Red Hat Subscription Manager Menu Option**

Alternatively, the Red Hat Subscription Manager UI can be opened from the command line with a single command:

–

```
[root@server1 ~]# subscription-manager-gui
```

The Red Hat Subscription Manager UI has a single window with tabbed sections that offer quick views into the current state of the system, showing installed products, subscriptions for the system, and available subscriptions the system has access to. These tabs also allow administrators to manage subscriptions by subscribing and unsubscribing the system.

The Red Hat Subscription Manager has three tabs which manage products and subscriptions:

- The **My Subscriptions** tab shows all of the current subscriptions that the system is subscribed to.
- The **All Available Subscriptions** tab shows all of the subscriptions that are available to the system. The default displays only subscriptions that are compatible with the hardware, but these can be filtered to show any subscriptions which match the hardware, any subscriptions which match installed products, only subscriptions which do not overlap with a currently-attached subscription, or any subscription which matches a given string.
- The **My Installed Products** tab shows the currently installed products on the system, along with their subscription status. This does not allow administrators to install software, only to view installed software.

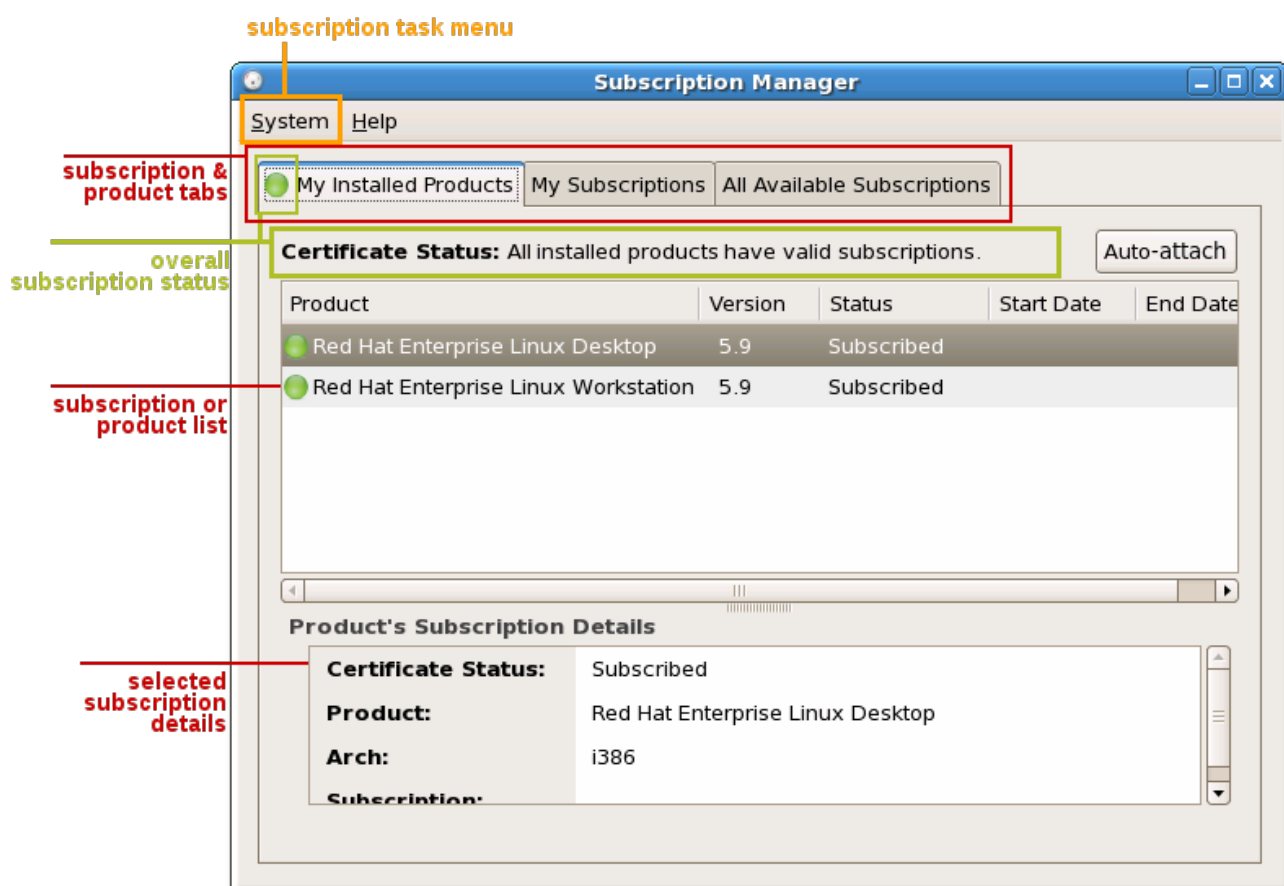


Figure 3. Red Hat Subscription Manager Main Screen

### 2.1.2. Running the subscription-manager Command-Line Tool

Any of the operations that can be performed through the Red Hat Subscription Manager UI can also be performed by running the **subscription-manager** tool. This tool has the following format:

■



```
[root@server1 ~]# subscription-manager command [options]
```

Each command has its own set of *options* that are used with it. The **subscription-manager** help and manpage have more information.

**Table 1. Frequently-Used subscription-manager Commands**

Command	Description
<b>Operational Commands</b>	
register	Registers or identifies a new system to the subscription service.
unregister	Unregisters a machine, which strips its subscriptions and removes the machine from the subscription service.
attach	Assigns a specific subscription to the machine.
remove	Removes a specific subscription or all subscriptions from the machine.
redeem	Autosubscribes a machine to a pre-specified subscription that was purchased from a vendor, based on its hardware and BIOS information.
import	Manually installs a subscription certificate, rather than contacting the subscription service with a request and then receiving the certificate.
list	Lists all of the subscriptions that are compatible with a machine, either subscriptions that are actually consumed by the machine or unused subscriptions that are available to the machine.
<b>Configuration Commands</b>	
config	Modifies a specified configuration parameter in the Red Hat Subscription Manager configuration file, <b>/etc/rhsm/rhsm.conf</b> . The parameters are passed in the form <i>configuration_area.parameter="value"</i> .
service-level	Sets the service-level preference for the system to use when selecting subscriptions in autoattach operations.
release	Sets the operating system release version preference for the system to use when selecting subscriptions in autoattach operations.

Command	Description
refresh	Pulls the latest subscription data from the server. Normally, the system polls the subscription server at a set interval (4 hours by default) to check for any changes in the available subscriptions. The <b>refresh</b> command checks with the subscription server immediately, outside the normal interval.
clean	Removes all of the subscription and identity data from the local system, <i>without affecting the consumer information in the subscription service</i> . Any of the subscriptions consumed by the system are still consumed and are not available for other systems to use. The <b>clean</b> command is useful in cases where the local subscription information is corrupted or lost somehow, and the system will be reregistered using the <b>register --consumerid=EXISTING_ID</b> command.
<b>Informative Commands</b>	
version	Returns the version of the local Red Hat Subscription Manager client, the name of the subscription service the system is registered with, and the version of the subscription service.
identity	Handles the identity certificate and registration ID for a system. This command can be used to return the current UUID or generate a new identity certificate.
facts	Lists the system information, like the release version, number of CPUs, and other architecture information.
orgs, repos, environments	Lists all of the configured organizations, environments, and content repositories that are available to the given user account or system. These commands are used to view information in a multi-org infrastructure. They are not used to configure the local machine or multi-org infrastructure.

## 2.2. Customer Portal Subscription Management

The ultimate goal of entitlement management is *to allow administrators to identify the relationship between their systems and the subscriptions used by those systems*. This can be done from two different perspectives: from the perspective of the local system looking externally to potential subscriptions and from the perspective of the organization, looking down at the total infrastructure of systems and all subscriptions.

The Red Hat Subscription Manager UI and CLI are both local clients which manage only the local machine. These tools are somewhat limited in their view; they only disclose information (such as available entitlements) from the perspective of that one system, so expired subscriptions, subscriptions

with no available quantity, or subscriptions for other architectures are not displayed.

Customer Portal Subscription Management is a *global* tool which is intended to give complete, organization-wide views into subscriptions and systems. It shows all subscriptions and all consumers for the entire organization. Customer Portal Subscription Management can perform many of the tasks of the local tools, like registering consumers, attaching subscriptions, and viewing system facts and UUID. It can also manage the subscriptions themselves, such as viewing contract information and renewing subscriptions — a task not possible in the local clients.

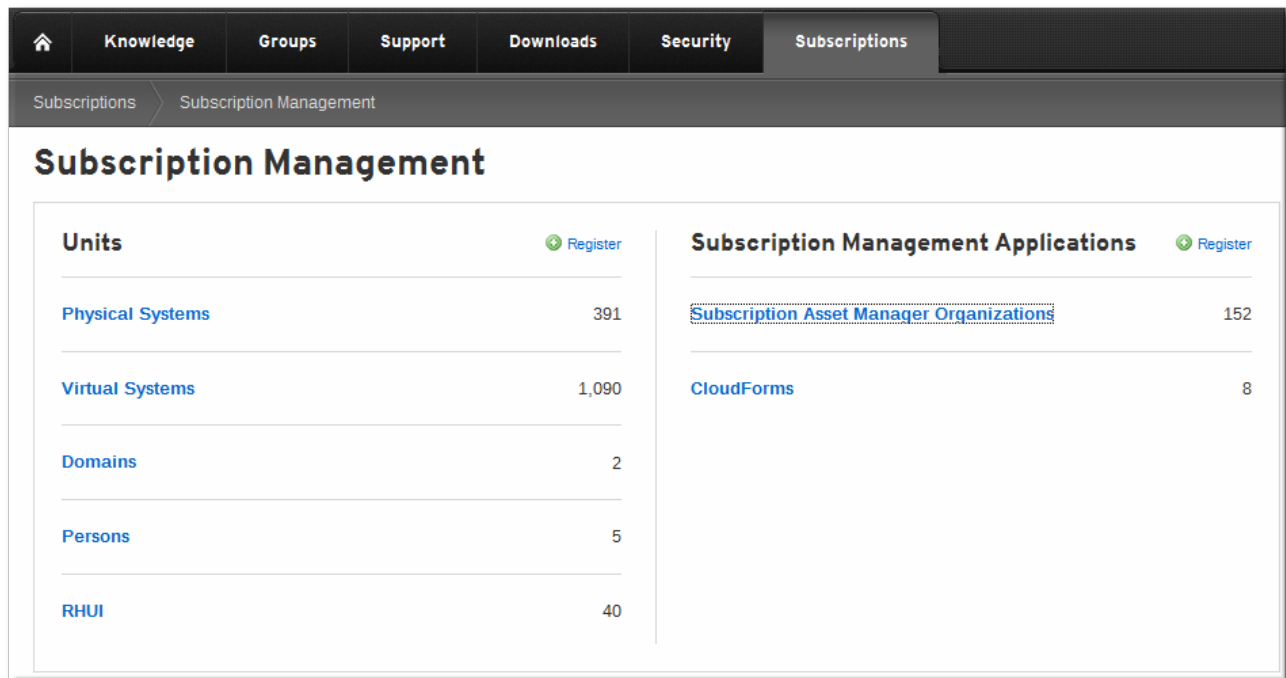


Figure 4. RHN Subscription Management in the Customer Portal



#### NOTE

Customer Portal Subscription Management gives a global view of all consumers, of all types, for an organization, which is crucial for planning and effectively attaching subscriptions. However, it does **not** provide any insight into what specific products are installed on a system and whether subscriptions are attached for those products. The Portal only gives an overall status for a system.

To track the individual status of installed products, you must use the local Subscription Manager tools.

Customer Portal Subscription Management also provides a view of systems and subscriptions managed under RHN Classic and provides access to the RHN Classic web tools.

All of the subscriptions for an entire organization — the subscriptions that have been purchased and the systems to which they have been allocated — are viewable through the account pages at <https://access.redhat.com/>. Additional information about Customer Portal Subscription Management is available in *Using the Customer Portal to Manage Subscriptions* at [https://access.redhat.com/site/documentation/Red\\_Hat\\_Subscription\\_Management/](https://access.redhat.com/site/documentation/Red_Hat_Subscription_Management/).

## 2.3. Subscription Asset Manager

Administrators may want to exert some control locally over subscription services or content delivery or both. Because each component in the subscription management framework is an independent application, with independent client configuration, different sources can be used.

Red Hat can delegate part or all of its subscription management functions to subscription management applications. Red Hat remains the primary source for subscription and content information, but it sends that information to on-premise applications, which then manage subscriptions and content locally.

Subscription management applications retain the global view and centralized control over subscriptions, but they allow administrators to define their IT environment in the subscription management setup. For example, IT administrators can create organizations that are independent of (and opaque to) one another to divide subscriptions, and can then create different environments within those organizations to manage content and repository access.

Using subscription management applications offers administrators control over configuration and the ability to improve performance based on network conditions or physical locations of machines.

Red Hat Subscription Management offers different on-premise subscription services:

- Subscription Asset Manager, which provides on-premise subscription services and proxies requests for Customer Portal Subscription Management content delivery
- Satellite 6, which provides both subscription services and content delivery on-premise

Subscription Asset Manager is included in the normal Red Hat Enterprise Linux subscription, so on-premise subscription services are available as part of any Red Hat Enterprise Linux infrastructure.

## 2.4. Hypervisor Processes

There are a couple of services available which can automatically detect guests on a virtual host system and register them as virtual systems. This allows subscriptions which are specific to virtual systems to be available to the guest and for subscriptions which are inherited from the host to be applied to the guest.

There is a special service that works with Subscription Manager to detect and identify hypervisors and their guests: **virt-who**.

The **virt-who** packages are not installed by default, but are available to Red Hat Enterprise Linux systems as part of their subscription.

## 3. SUBSCRIPTION WORKFLOWS

There are different types of systems and different ways of organizing systems. Subscription Manager, and the underlying concepts of a subscription server and content provider, are flexible enough to accommodate special types of systems and different infrastructure setups.

### 3.1. Planning the Workflow to Use

Subscription and content services work in tandem. The local Red Hat Subscription Manager client on a system must identify a subscription service and a content delivery service. These can both be hosted services, both locally-managed services, or a mix.



All subscription and content data originates with Red Hat. By default, all Red Hat Enterprise Linux systems are configured to point to the Red Hat hosted services. If administrators use an on-premise server for subscription or content management, that on-premise server first obtains its information from

the Red Hat hosted services, and then manages the local systems. Red Hat subscription inventories are aware of the on-premise subscription application, then, but not aware of any of the local systems that subscription application manages.

The Red Hat Subscription Manager configuration points to two servers, the subscription server and content server. These two settings can be changed (independently) to use any supported subscription server or content server. [Table 2, “Subscription and Content Services, by Source”](#) lists subscription and content servers.

**Table 2. Subscription and Content Services, by Source**

Server Type	Supports Subscription Services	Supports Content Delivery Services	Recommended Environment Types
Customer Portal Subscription Management	✓	✓	<ul style="list-style-type: none"> <li>• Small businesses (fewer than 20 servers)</li> <li>• Limited IT resources</li> <li>• No need to script subscription or content updates</li> </ul>
Subscription Asset Manager	✓	✗	<ul style="list-style-type: none"> <li>• Small and medium businesses</li> <li>• Business with security rules that preclude hosted services</li> <li>• Multiple virtual machines</li> <li>• Large enterprises which require auditing and reporting, without any additional management functions</li> </ul>

Server Type	Supports Subscription Services	Supports Content Delivery Services	Recommended Environment Types
Satellite 6			<ul style="list-style-type: none"> <li>• Medium businesses</li> <li>• Business with security rules that preclude hosted services</li> <li>• Custom content and deployment scripts</li> <li>• System management along with subscription/content management</li> </ul>

### 3.2. A Note on User Accounts

Systems in an inventory and subscriptions are managed through the subscription management service, and that service is connected to a user account.

**Each subscription management service defines its own set of users.** The user account to create and to use depends on which subscription management service is being accessed.

#### Customer Portal Subscription Management

When a subscription is initially purchased, it must be configured in Subscription Asset Manager. The user accounts used to connect to the Customer Portal cannot be used to manage those systems. The lo is created in Customer Portal Subscription Management. That administrator can create other user accounts, associated with the overall company account, which (with the proper permissions) can also manage systems and subscriptions. Creating and managing Customer Portal accounts is described in [Managing User Access to the Red Hat Customer Portal and the RHN Application](#).

#### Subscription Asset Manager

Subscription Asset Manager is an on-premise application. When an IT administrator is initially accessing subscription manifests and managing subscriptions across the Red Hat account, those actions are performed through the Customer Portal, so a Customer Portal user account is used to set up and manage the Subscription Asset Manager application itself.

Since the individual systems are managed by Subscription Asset Manager, the user accounts which are used to manage those systems must be configured in Subscription Asset Manager. The user accounts used to connect to the Customer Portal cannot be used to manage those systems. The local IT administrators are responsible for creating regular user accounts in Subscription Asset Manager.

#### Activation Keys

A user account is associated with an activation key and is used to redeem the activation key — but which user account to use depends on who issued the key. If a key was created by a vendor, then they will also define a user account to use with the key, as will Red Hat if the key were issued by Red Hat. For

activation keys created by local IT departments using Subscription Asset Manager, then the associated Subscription Asset Manager user account is to be used.

### 3.3. Customer Portal: Autoattaching Systems

Subscription management for a Red Hat Enterprise Linux system has two steps to it: registering the system to a subscription service and then applying subscriptions for the operating system and products installed on that system. By default, when attaching subscriptions to a system through the Red Hat Subscription Manager UI on a system or through firstboot, these two steps are performed at the same time.

The Red Hat Subscription Manager configuration can point to any subscription server or content server. By default, it points to Red Hat's hosted services.

So, using the default configuration, then system is registered with Customer Portal Subscription Management hosted services and the best-matched available subscriptions are automatically attached to the system.

#### 3.3.1. The Environment: Small Businesses

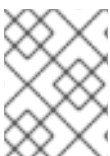
Hosted services are designed for IT environments where simplicity of deployment is the most important consideration. This is for small businesses or businesses with small Linux infrastructures:

- Fewer than 20 Linux servers
- Limited IT resources for system maintenance
- No business need to create custom subscription or content utilities
- Infrastructures already using RHN Classic hosted services for existing Linux systems

Hosted services make it easier to administer a small number of machines:

- Simple to implement, since it uses the default system configuration
- No additional software or hardware overhead
- Possible to migrate to on-premise subscription/content services based on organization configuration later
- The same subscription services for all systems, even if the systems are in different data centers or geographic locations

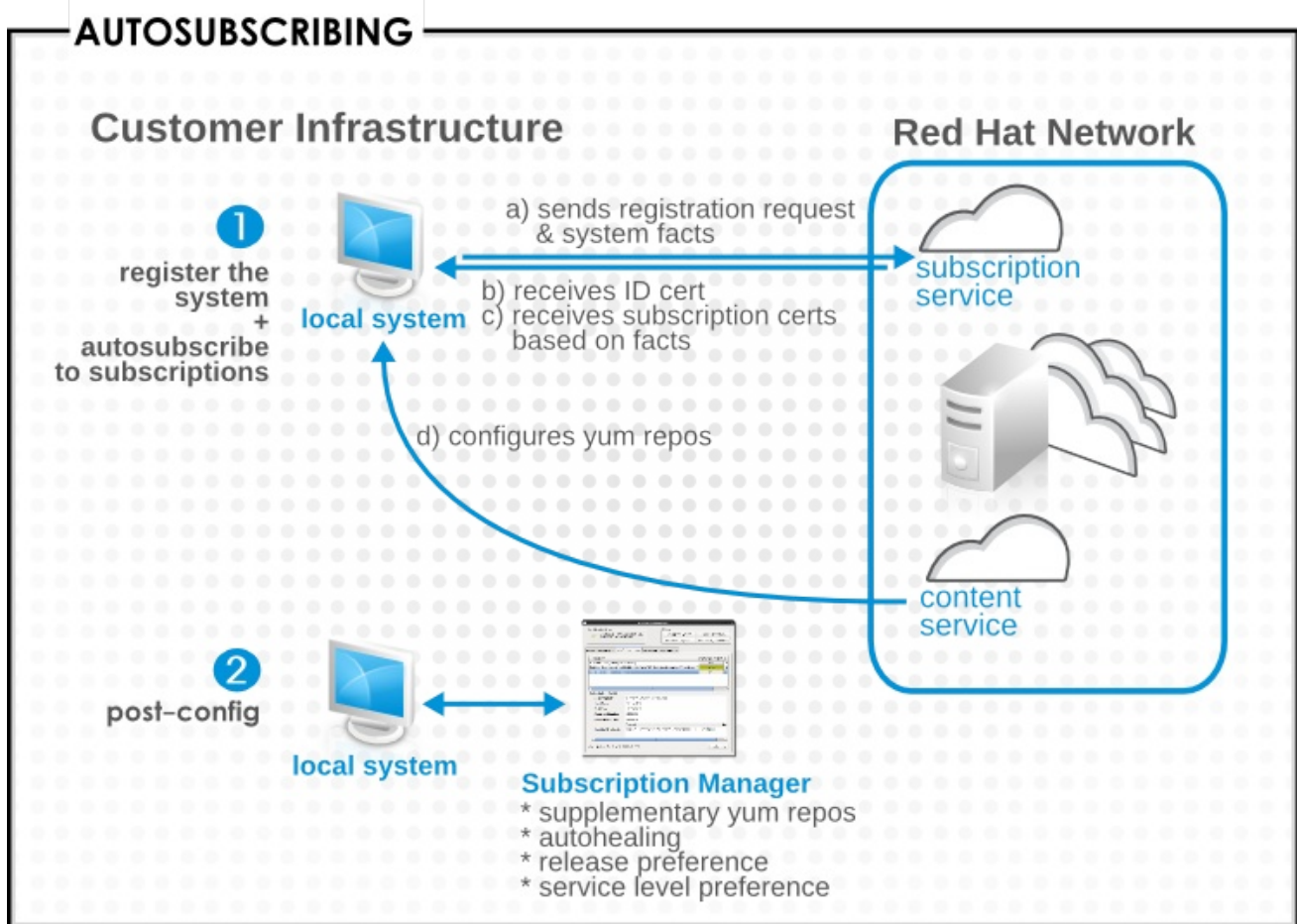
Hosted services do have some potential issues related to the overall network performance. As the IT infrastructure grows, then there could be local bandwidth or latency issues if a large number of systems are attempting to retrieve content or receive updates.



#### NOTE

Even if a small IT environment uses Red Hat hosted services at the start, it can be expanded to use on-premise subscription or content services in the future.

#### 3.3.2. Workflow



**Figure 5. Autoattach Process**

When automatically attaching matched subscriptions (the default setting in the Red Hat Subscription Manager UI or `--auto-attach` with the `subscription-manager` command), there is only a single step to the registration process.

### 3.3.3. Options and Details

Most of the options are configuration settings that can be set after registration.

- Attaching additional subscriptions, which is especially useful if the system is autoattached during firstboot, when subscriptions are only attached for the operating system.
- Overriding system facts, which is used by the autoattach and healing processes to determine what the system architecture and hardware is for finding compatible subscriptions.
- Setting a service level preference (this can also be done during registration, so it is used as one of the priorities when selecting subscriptions).
- Setting a release preference, so that the system only updates for software targeted to that release version and ignores any upgrades to a later operating system version.
- Enabling or disabling associated **yum** repos.

## 3.4. Customer Portal: Registration and Manual Subscription

While system registration and subscription can be performed in a single step (autoattaching), these are separate configuration areas. That means that the steps can be performed separately, which offers much more control over how systems are provisioned and how subscriptions are assigned.



With autoattaching, whatever subscriptions best match the system's architecture and currently installed products are attached automatically. This can be limiting in certain instances. For example, registering during firstboot would only attach the operating system subscriptions since no other products would yet be installed. Or, there may be a product which is not yet installed on a system, but will be soon, or a package listed for 32-bit systems which will be run on a 64-bit system.

In any of those cases, the autoattach process cannot detect the desired subscriptions because there are no settings on the system which would signal which subscriptions to include.

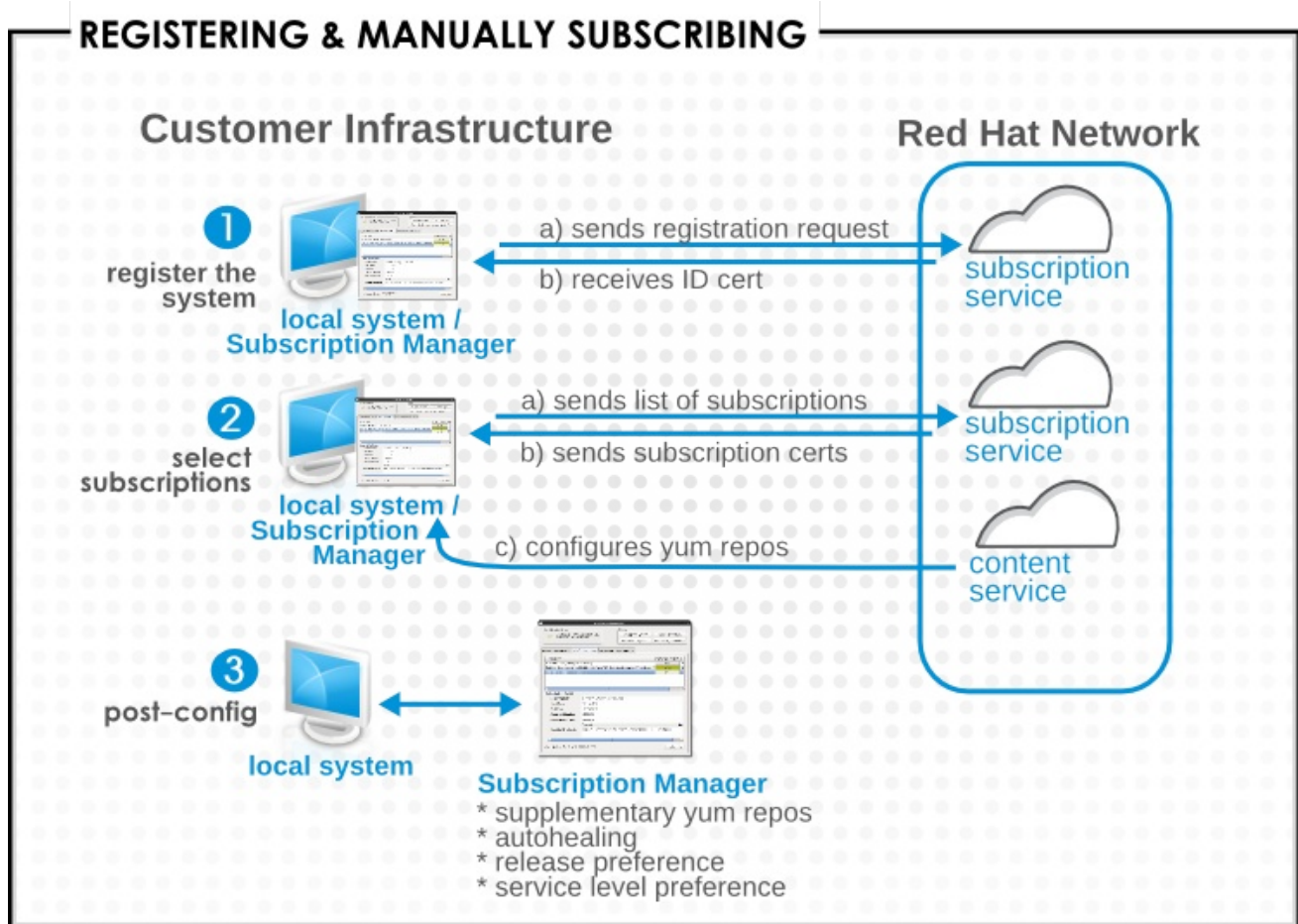
Dividing the registration and subscription processes allows administrators to manually select subscriptions and attach them to the systems.

### 3.4.1. The Environment: Small Businesses

There is nothing about registering and then applying subscriptions manually which requires that a system use Red Hat's hosted services.

There are some benefits, particularly for small businesses or infrastructures with relatively few Linux systems, to using hosted services. As with the autoattach process, using the hosted services minimizes the administrative overhead since it uses the default system configuration and requires no maintenance from IT staff.

### 3.4.2. Workflow



**Figure 6. Manual Subscription Process**

1. Register the system.

Using the **subscription-manager** CLI command, all this requires is sending the username and password for the Customer Portal Subscription Management account holder.

For the Red Hat Subscription Manager UI (both run on the system and in firstboot), an autoattach operation is run by default. To undo this, check the option to skip autoattaching subscriptions. Alternatively, some subscriptions could be autoattached for the operating system, and then additional subscriptions can be attached or other subscriptions removed later.

2. Select and attach the subscriptions, using the Red Hat Subscription Manager tools.

When the system is registered, the subscription service sends over a list of all subscriptions that are available to the account. Any of the available subscriptions can be attached to the system.

Using the Red Hat Subscription Manager UI, the available subscriptions are listed in the **Available Subscriptions** tab; they can be selected and attached by clicking a button.

In the CLI, subscriptions are listed first using the **list** subcommand and then attached by their pool ID, using the **attach** subcommand.

### 3.4.3. Options and Details

Most of the options are configuration settings that can be set after registration.

- Setting a service level preference (this can also be done during registration, so it is used as one of the priorities when selecting subscriptions).
- Setting a release preference, so that the system only updates for software targeted to that release version and ignores any upgrades to a later operating system version.
- Enabling or disabling associated **yum** repos.

## 3.5. Subscription Asset Manager: Direct Subscription Assignments

The Red Hat hosted services have a flat, undefined structure where all systems are in a single pool. While convenient in some ways, this loses the structure that real IT environments have, where systems are loosely or strictly associated with different organizational divisions, physical locations, and content streams. Subscription Asset Manager introduces an organizational structure, where administrators can create local organizations (for subscriptions) and system groups that allow them to arrange systems in a way that reflects real-life configurations.

### 3.5.1. The Environment: Small Businesses to Large Enterprises for Locally-Defined Structure

Subscription Asset Manager imposes a structure on the subscription service and content service that allows administrators to design those systems in a way that reflects the actual infrastructure configuration.

Subscriptions are grouped by organizations, which are top-level divisions. Each organization represents a separate subscription application entry to Customer Portal Subscription Management.

Within each organization, it is possible to create system groups which help structure systems for updates, access control, and content management.

Imposing an organizational structure on the subscription inventory gives Subscription Asset Manager the ability to better present information about the subscriptions and systems for an account. For large enterprises, this may be the only functionality from Subscription Asset Manager that is required, using

Subscription Asset Manager for reporting and auditing while custom or enterprise-level applications are used to manage systems.

For small and medium businesses, Subscription Asset Manager offers more control over system management than is possible through hosted services alone:

- Enact security rules that require on-premise services rather than hosted services.
- Better manage virtual environments, particularly in private clouds or data centers, which require systems to be created and removed on the fly.
- Define different content repositories for different types of systems, such as different sources for development and production systems.

The functionality of Subscription Asset Manager tracks closely with the functionality of Customer Portal Subscription Management and Red Hat Subscription Manager. Systems can be registered and have subscriptions automatically attached. Preferences like release versions and service levels can be used to govern software updates.

### 3.5.2. Workflow

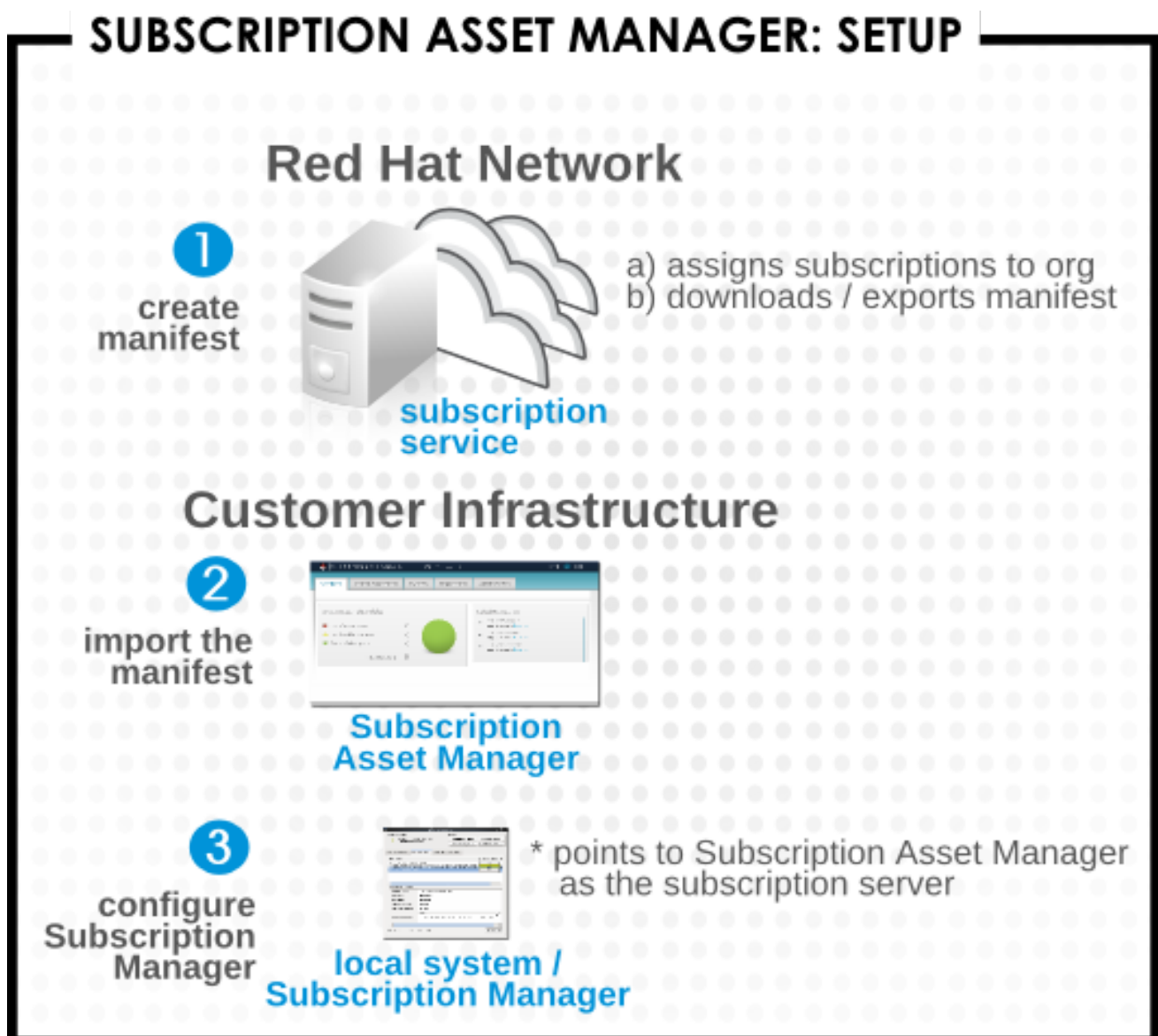


Figure 7. Subscription Asset Manager Setup

1. If necessary, create an entry in the Red Hat inventory for the organization. Every organization in Subscription Asset Manager must have a corresponding subscription service entry in the Red Hat inventory.
2. Assign a bloc of subscriptions to the organization. This bloc of subscriptions is the *manifest* of subscriptions for that Subscription Asset Manager organization.
3. Export the manifest.
4. Import the manifest into Subscription Asset Manager.
5. Configure the Red Hat Subscription Manager client on the local system to use the Subscription Asset Manager subscription service and, optionally, the Subscription Asset Manager content proxy.

The backend configuration for Subscription Asset Manager and its manifests only needs to be done once, and the Red Hat Subscription Manager configuration changes only need to be made once per system.

When that is complete, then the system can be registered and have subscriptions attached to it.

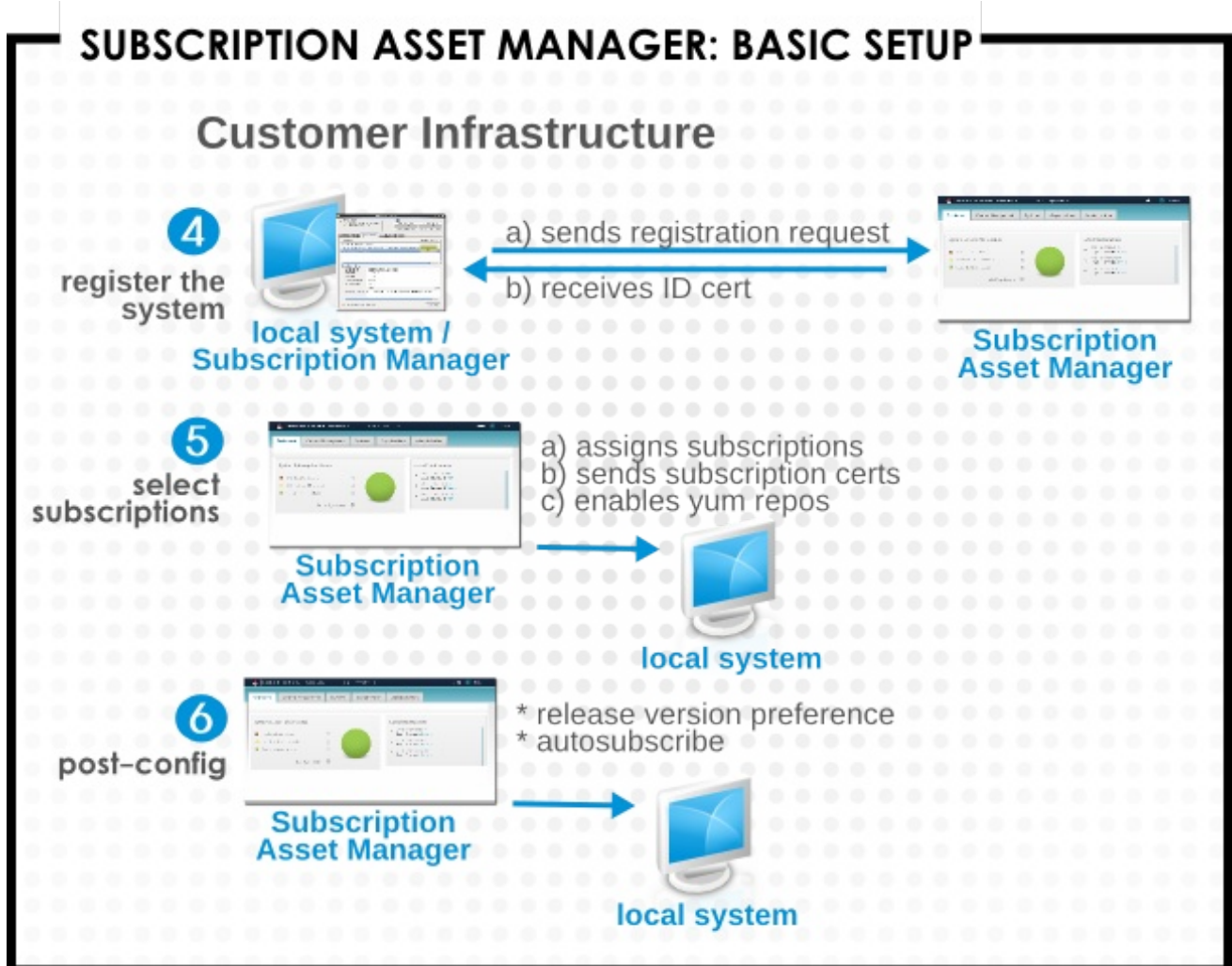


Figure 8. Registering with Subscription Asset Manager

1. Register the system.

Using the **subscription-manager** CLI command, use the **register** command with the username and password for the Customer Portal Subscription Management account holder and the hostname of the Subscription Asset Manager server.

For the Red Hat Subscription Manager UI, autoattaching subscriptions is performed by default. Check the option to attach subscriptions later.

2. Select and attach the subscriptions, using the Subscription Asset Manager UI.

### 3.5.3. Details and Options

After registering a system with Subscription Asset Manager, it can be managed using either Subscription Asset Manager or Red Hat Subscription Manager, but the options set in Red Hat Subscription Manager must match what is available in Subscription Asset Manager.

- Enable autoattaching for the system and, optionally, set a service level preference.

## 3.6. Subscription Asset Manager: Activation Keys

Activation keys are a way to preconfigure subscriptions for a system before it is registered (or, when used with provisioning systems like kickstart, before the system is even created). This grants a great deal of flexibility and control to administrators over which subscriptions to attach to new systems, while simplifying the registration process for users.

### 3.6.1. The Environment: Preconfigured Systems

[Section 3.5, “Subscription Asset Manager: Direct Subscription Assignments”](#) summarizes the benefits of using Subscription Asset Manager in general, and those same benefits apply, generally, to using Subscription Asset Manager to create activation keys.

Additionally, using activation keys makes the overall registration and subscription process simpler, almost transparent, for users. They can pass the activation key to register the system and have all of the configured subscriptions apply *and the user never has to know what those subscriptions are or what quantities are required*.

Preconfiguring systems allows administrators to define system configuration consistently. For example, using activation keys and pre-configured subscriptions is common when supplying company-configured laptops or workstations to employees.

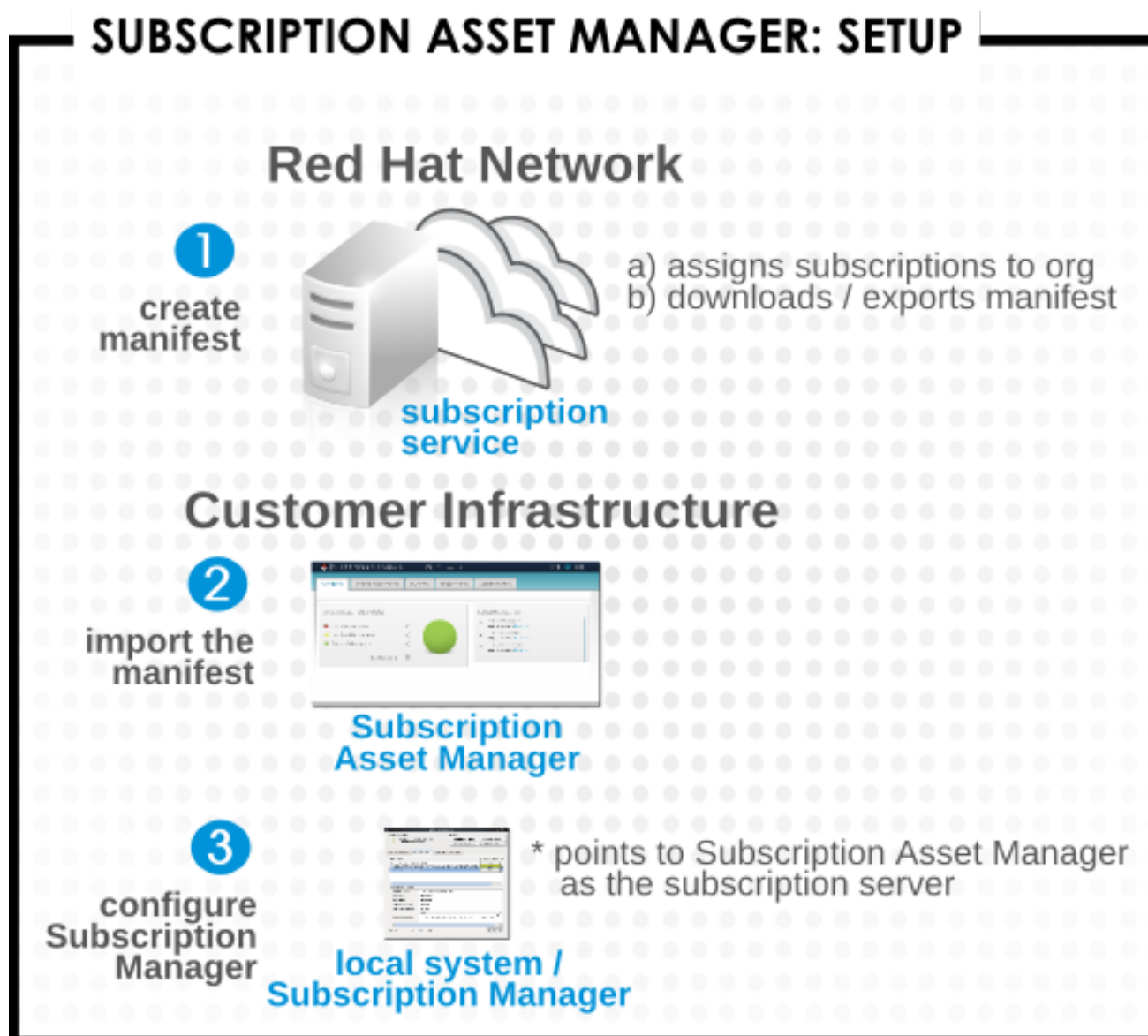
Activation keys create a set of subscriptions that can be attached to a system, without attaching them directly to a system. The activation key can be associated with a system group, within the Subscription Asset Manager configuration. From there, it can be used with any system. This results in several benefits for both users and administrators:

- Administrators have control over which subscriptions are installed to a system without having to create and configure every system first.
- Because activation keys are created within Subscription Asset Manager and do not rely on system settings or architecture, the target system does not have to exist yet.
- Users can register their system in a single step and automatically have all the proper subscriptions attached, without having to select and attach subscriptions manually and potentially miss a subscription.

### 3.6.2. Workflow



By default, systems are configured to use Red Hat hosted services. Before activation keys can be created and used, the appropriate backend infrastructure needs to be set up.



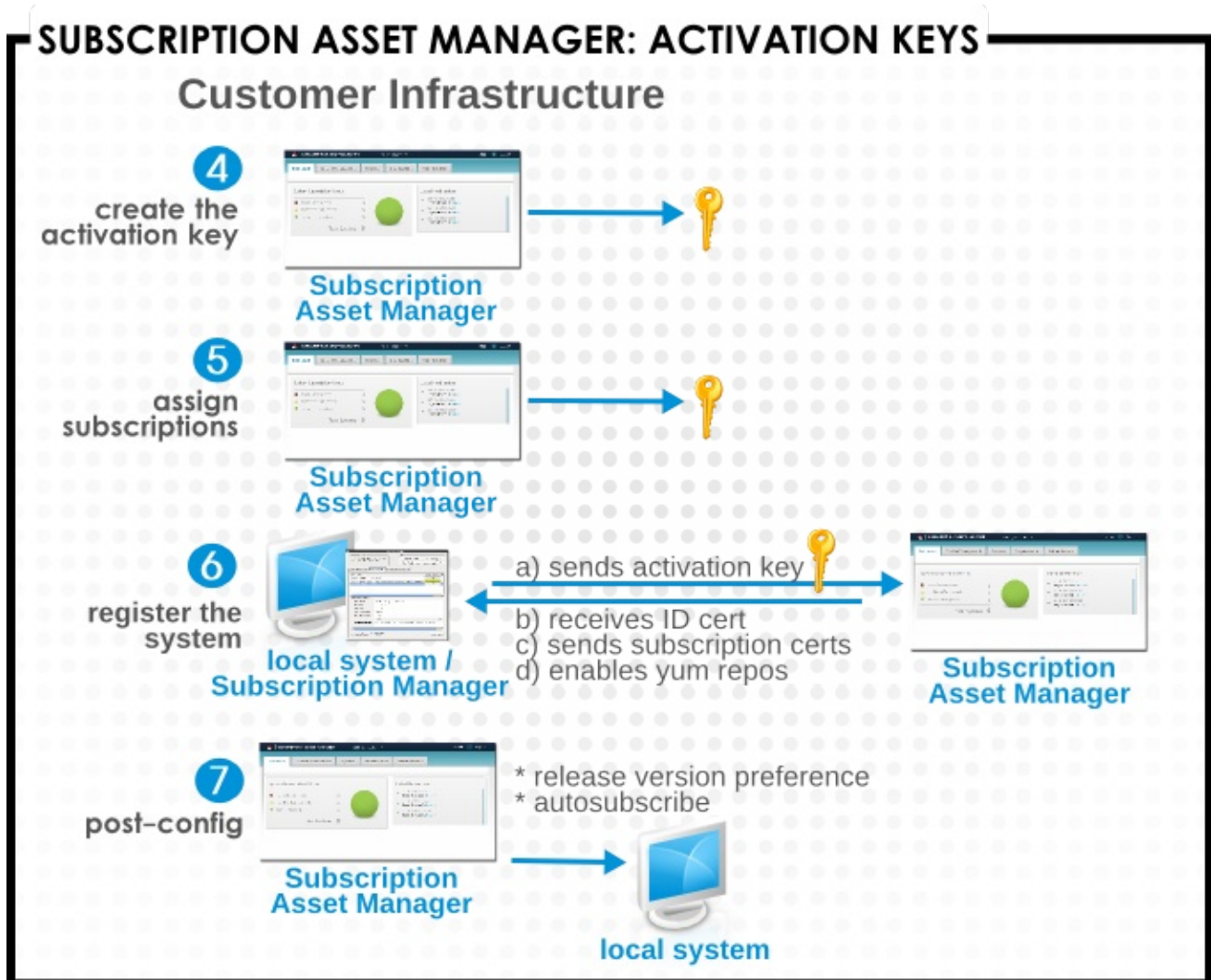
**Figure 9. Subscription Asset Manager Setup**

1. If necessary, create an entry in the Red Hat inventory for the organization. Every organization in Subscription Asset Manager must have a corresponding subscription service entry in the Red Hat inventory.
2. Assign a bloc of subscriptions to the organization. This bloc of subscriptions is the *manifest* of subscriptions for that Subscription Asset Manager organization.
3. Export the manifest.
4. Import the manifest into Subscription Asset Manager.
5. Configure the Red Hat Subscription Manager client on the local system to use the Subscription Asset Manager subscription service and, optionally, the Subscription Asset Manager content proxy.

This can be done at any point before the system is registered, so it can even be performed after the activation key is created.

The backend configuration for Subscription Asset Manager and its manifests only needs to be done once, and the Red Hat Subscription Manager configuration changes only need to be made once per system.

When that is complete, then activation keys can be made.



**Figure 10. Registering with Activation Keys**

1. Create the activation key. This is a container entry that subscriptions can be attached to.
2. Attach subscriptions to the key.
3. Register the local system using the activation key.

This is basically an autoattach operation, only instead of using the Red Hat Subscription Manager evaluation to select best-matched subscriptions, it attaches the pre-configured subscriptions associated with the key.

### 3.6.3. Details and Options

After registering a system with Subscription Asset Manager, it can be managed using either Subscription Asset Manager or Red Hat Subscription Manager, but the options set in Red Hat Subscription Manager must match what is available in Subscription Asset Manager.

- Enable autoattaching for the system and, optionally, set a service level preference.

### 3.7. General Management: Firstboot

Registering a system during firstboot is the same as registering a system using Red Hat Subscription Manager later, so there is no special workflow associated with it. The main reason for mentioning firstboot is that the firstboot wizard does provide several different ways to configure the system for subscriptions and content updates. The system always has some kind of subscription configuration, from the first time it is configured, and understanding those options can make it easier to administer the system later.

During firstboot, the subscription service and attached subscriptions are configured in the **Set Up Software Updates** screens. There are four choices:

- Red Hat Subscription Management, which covers product-driven subscription services in Customer Portal Subscription Management, Subscription Asset Manager, and Satellite 6
- RHN Classic, which uses channel-driven access to content
- Satellite or Proxy content delivery, which uses a channel-based system similar to RHN Classic
- Register later



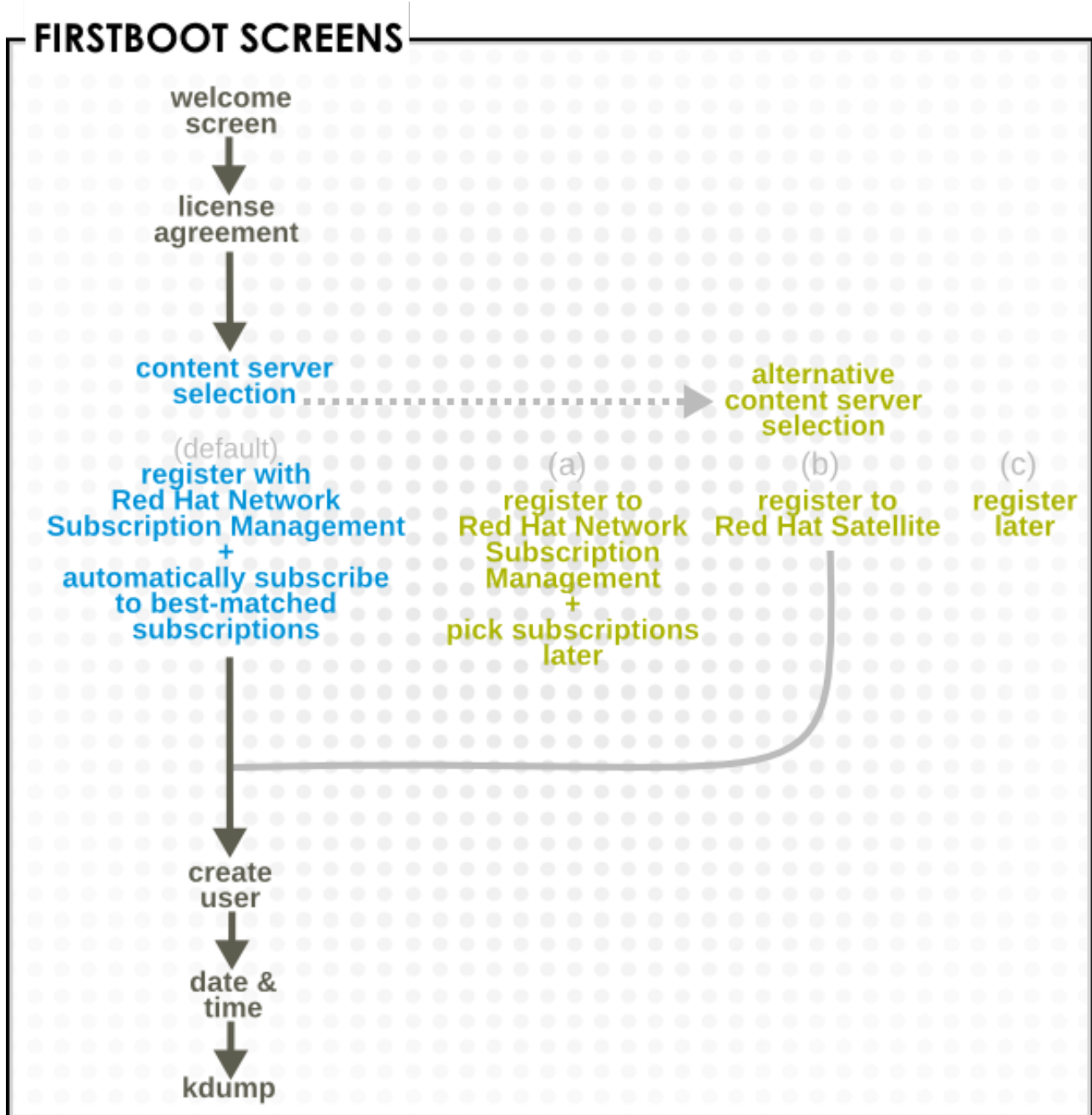


Figure 11. Firstboot

By default, firstboot registers the system to Customer Portal Subscription Management and autoattaches compatible subscriptions. Selecting the **Skip automatic subscription selection...** checkbox disables autosubscription, so that the system can have subscriptions attached manually.

The firstboot options for setting up the content server are the same as using Red Hat Subscription Manager to configure an existing system.

### 3.8. General Management: Kickstart

Kickstart is a way to script and automate system creation. Kickstart is a key component for provisioning systems. Subscription setup can be included in kickstart instructions as a post-install script.

Kickstart setups can follow any workflow an administrator wants to use: registering to hosted services, registering with a Subscription Asset Manager instance using an activation key, manually attaching subscriptions, whatever. First learn which subscription workflow to use, and then use the

**subscription-manager** command to run through each of the required steps.

### 3.8.1. The Environment: Scripted Environments

Kickstart is used to script system creation, so this is used in environments where instances may be created and destroyed routinely, such as test environments, or in data centers and private clouds.

### 3.8.2. Workflow

To register a system and attach subscriptions as part of kickstart, run the **subscription-manager** command as a post-install script.

```
%post --log=/root/ks-post.log
/usr/sbin/subscription-manager register --username rhn_username --password
rhn_password --auto-attach
```

In this example, the subscription configuration is the default configuration, meaning that the system is registered with the Customer Portal Subscription Management (hosted) system. Additionally, this uses the **--auto-attach** option to attach the best-matched subscriptions.

### 3.8.3. Options and Details

Kickstart can follow any of the supported configurations, but it can require using additional post-install scripts first to set up the environment.

- Set a service level preference using the **--servicelevel** option.
- To attach subscriptions manually, leave off the **--auto-attach** option and run a second script with the **attach** command or add the subscriptions later.
- By default, Red Hat Subscription Manager uses the Customer Portal hosted subscription and content services. To use an on-premise service like Subscription Asset Manager, first run the **subscription-manager config** command and reset the Red Hat Subscription Manager configuration, and then register the system using the new configuration.
- To register a system and attach subscriptions using an activation key, first configure Red Hat Subscription Manager to use the appropriate subscription service and then register with the activation key.

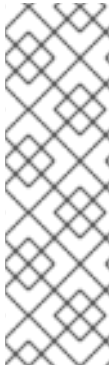
## 3.9. General Management: Hypervisors and Guests

Red Hat Enterprise Linux has an optional service which can automatically detect guests on a virtual host system, create a map of hosts to guests, and register guests as virtual systems. This allows subscriptions which are specific to virtual systems to be available to the guest and for subscriptions which are inherited from the host to be applied to the guest automatically.

The **virt-who** process can detect and associate guests on several different types of hypervisors:

- Red Hat Enterprise Virtualization Manager (KVM)
- Xen
- HyperV
- VMware ESX

### 3.9.1. The Environment: Data Centers, Cloud Environments, Any Environment Using Virtual Machines



#### NOTE

If your environment has a large number of virtual systems or depends on creating and destroying virtual systems (such as test environments or private clouds), then consider using Subscription Asset Manager to manage subscription services. Subscription Asset Manager allows administrators to define organizations to group subscriptions for systems and then groups within those organizations to control content flows within those organizations.

Subscription Asset Manager is available with any Red Hat Enterprise Linux subscription.

For subscriptions to be managed effectively — particularly with inheritable subscriptions or interactions between subscriptions — there has to be an internal awareness in the subscription service of the relationships between hosts and guests. This is a *host/guest mapping*, which is literally a list of all of the guest identifiers for a given hypervisor.

Hypervisors are registered as a special type of consumer in Subscription Asset Manager or Customer Portal Subscription Management. Hypervisors themselves are managed as regular physical systems, but the hypervisor type indicates that that particular system will have guests mapped to it, and that subscriptions may be inheritable or applied differently to those guests.

With a host/guest mapping to associate every guest with a specific host, a subscription service can properly attach a single subscription to a virtual host and then apply an included and inheritable subscription to its guest (for example), rather than consuming two separate subscriptions for each instance.

This association is done by extracting a *universally unique identifier* for each guest and associating it with its hypervisor. These UUIDs are part of the system facts for each virtual system.

The hypervisor is registered first, and then a related process on the system scans for any guests and submits the discovered UUIDs to the subscription service. This is done by the **virt-who** process on the hypervisor.

### 3.9.2. Workflow

1. If registering with a Subscription Asset Manager instance, install the Subscription Asset Manager RPM to configure Red Hat Subscription Manager.
2. Register the system as hypervisor, using the **--type=hypervisor** option with the **subscription-manager** command.
3. Install **virt-who** on a Red Hat Enterprise Linux system within the infrastructure. This Red Hat Enterprise Linux system manages the mapping and communicates with Red Hat Enterprise Linux guests, the hypervisor, and the backend subscription service, even if the hypervisor itself is not a Red Hat Enterprise Linux system.

If the Red Hat Enterprise Linux system is ever unavailable, the host/guest mapping is not available, and virtual systems are treated as physical systems.

4. For VMware or for environments using Subscription Asset Manager Edit the **virt-who** configuration file (**/etc/sysconfig/virt-who**) to recognize the appropriate virtualization and subscription services.

5. Start the **virt-who** process to create the host-guest mapping.

## 3.10. General Management: Disconnected Systems

Disconnected systems are a unique use case because the system is offline, either off the corporate network or possibly entirely lacking Internet access. This means that the system has no ability to access any subscription or content services, so any changes to the system must be manual.

### 3.10.1. The Environment: Security Environments and Backup Systems

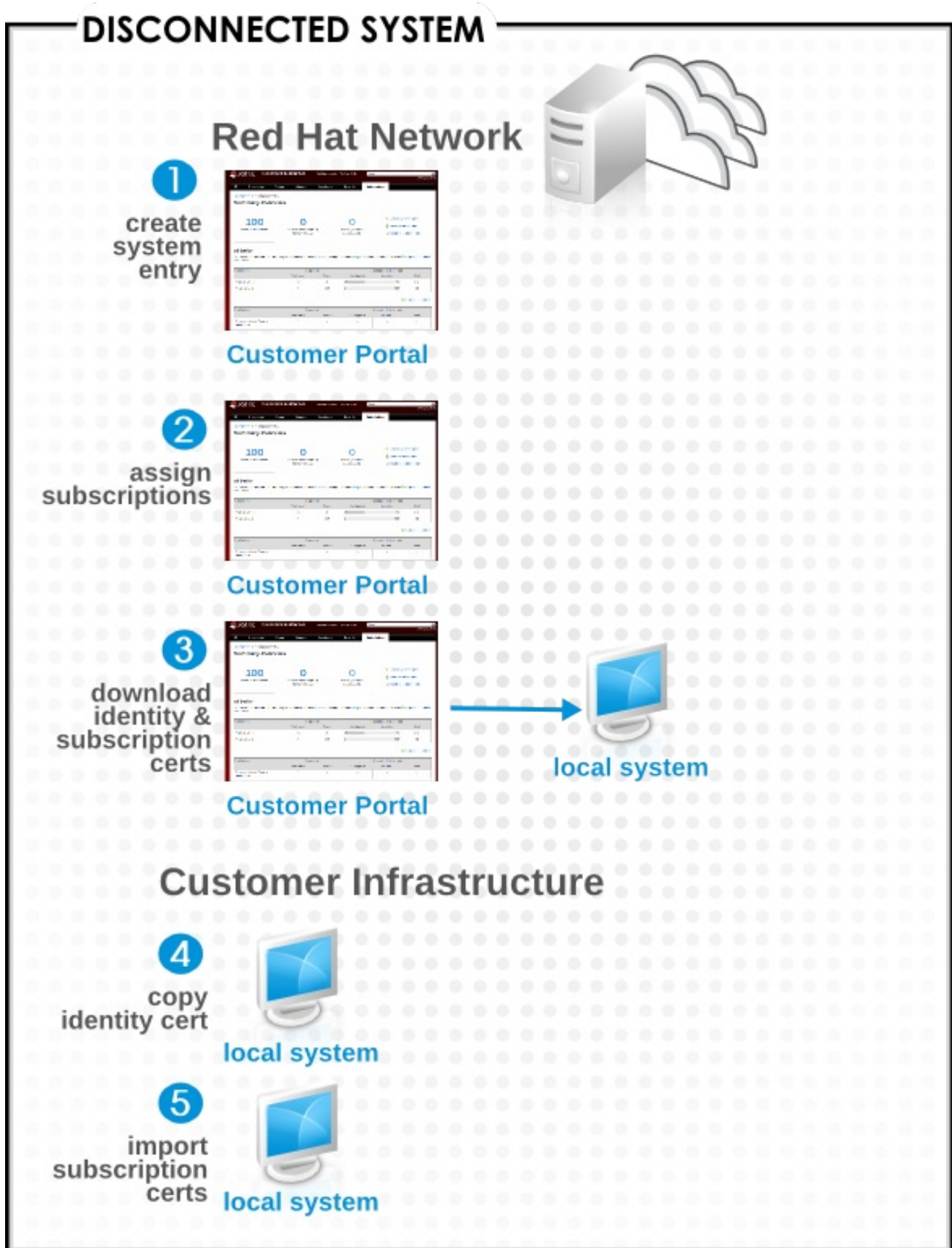
Disconnected systems are simply any system that cannot connect to the Internet or, possibly, even an intranet. The products and subscriptions for that system, and the system itself, should still be included in an inventory.

This frequently means servers in secure locations where they are prevented by business rules from connecting to the Internet. It can also include backup systems, which may be kept offline until needed.

Most of the subscription and content operations are performed over the network. For example, the **rhsmcertd** process checks for updated subscription information every four hours by connecting to the given subscription service. If a system cannot connect to the Internet, than almost all of those management tasks cannot be performed.

### 3.10.2. Workflow

The disconnected system workflow follows the same conceptual path as a workflow like [Section 3.3, “Customer Portal: Autoattaching Systems”](#), except that instead of using the tooling to connect to the required services and perform the services automatically, an administrator has to configure and copy certificates around manually.



**Figure 12. Registering Disconnected Systems**

1. Create the system's entry.

The simplest thing is to create this entry in the Customer Portal, which is a global view of all systems in the company. If a subscription service like Subscription Asset Manager is used, then

the disconnected system can be associated with a local organization and system group, which could be useful if the system will be brought online later.

2. Attach subscriptions to the system. If the system were online, the Red Hat Subscription Manager would pull in a list of available subscriptions and then communicate that back to the subscription service (much like a waiter taking an order in a restaurant). In that way, both the local system and the subscription service are aware of what is attached to the system.

With a disconnected system, the appropriate subscriptions need to be set aside and attached to the system first so that the subscription service is aware of the assignments.

3. Download all of the certificates for the system, both the identity (registration) certificate and all of the associated attached subscription certificates.
4. Copy the identity certificate into the appropriate location. This tells the system what its registration information is.
5. Copy the subscription certificates into the appropriate location.

This tells the system what subscriptions it has attached to it, without having to query the subscription service for available subscriptions.

### 3.10.3. Options and Details

Virtually all of the additional configuration options — both system-level configuration like setting a service level and infrastructure-level configuration like using a different subscription or content service — are not applicable to a disconnected system *as long as it is disconnected*. Most parameters, such as autoattaching, configure operations that occur automatically over the network.

One thing to plan is what configuration should be used if the system is ever brought online. For example, if the rest of the infrastructure is using Satellite 6 rather than Customer Portal Subscription Management hosted services, the disconnected system should probably also be registered to the Satellite 6 services rather than Customer Portal Subscription Management. For security rules, that system may never be brought online, so that configuration may always be irrelevant. Backup systems could come online at any time, which would make proper configuration important.

## 3.11. Legacy: Migration from RHN Classic

Certificate-based, or system-based, subscriptions were launched in Red Hat Enterprise Linux 6.1 and 5.7, and after those releases both system-based and channel-based subscription registration was supported. Older Red Hat Enterprise Linux systems exclusively used channel-based subscriptions.

A system's registration can be moved completely, with its subscriptions intact, from the channel-based RHN Classic to a system-based subscription service such as Customer Portal Subscription Management or Subscription Asset Manager.



### IMPORTANT

These migration scripts are intended to migrate systems which are registered with RHN Classic hosted services. They are not intended to migrate systems registered with an on-premise Satellite server.

### 3.11.1. The Environment: Small Businesses with Older Red Hat Enterprise Linux Systems

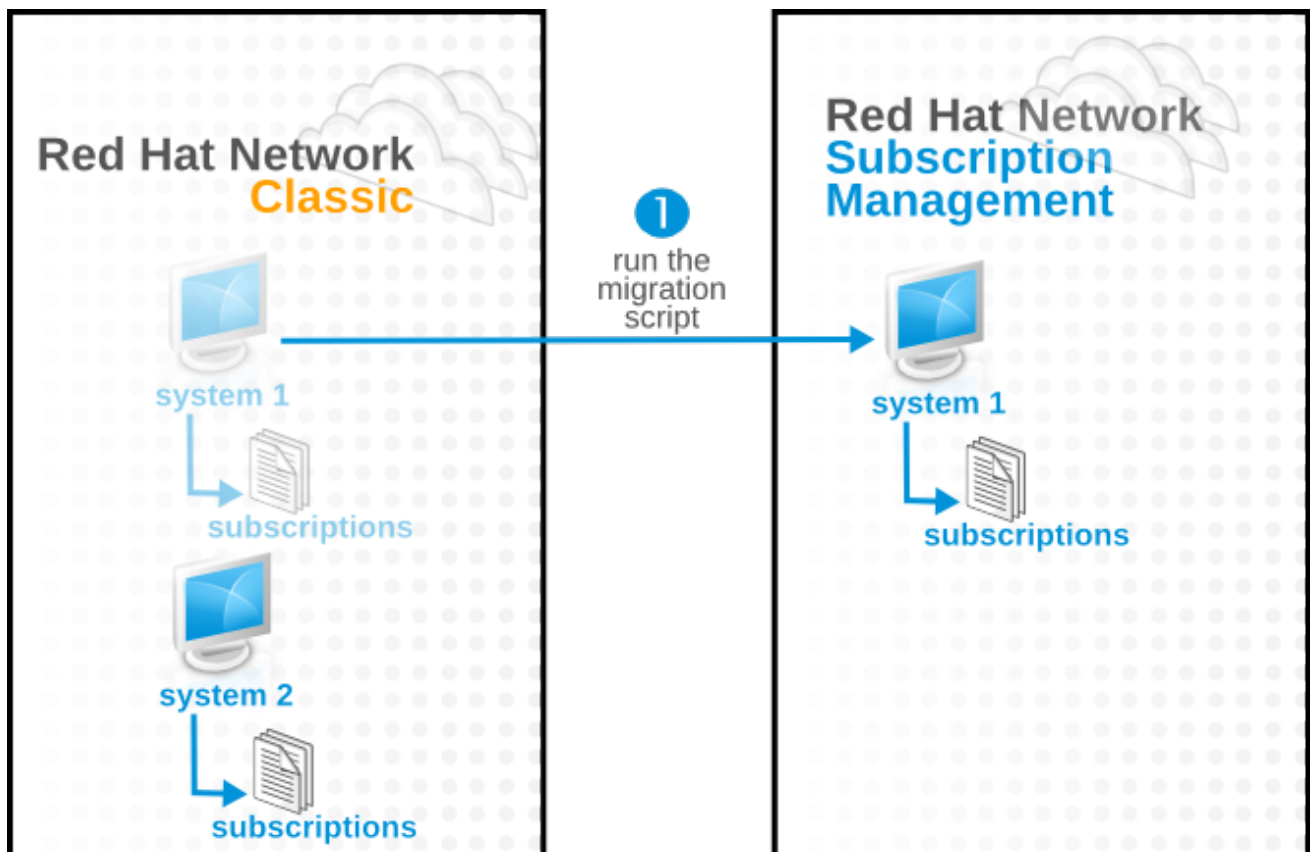
The migration scripts migrate registrations from the channel-based RHN Classic, which is a hosted subscription service. This migration can go to the newer hosted service, Customer Portal Subscription Management, or to an on-premise Subscription Asset Manager system.

For convenience and general performance, hosted services are usually used by small and medium-sized businesses because they have a relatively small number of servers.

Large environments, particularly environments with more than 1000 systems, use Satellite, which is not supported by the migration scripts.

### 3.11.2. Workflow

The migration process itself has only a single step for administrators: running the migration script.



**Figure 13. Migration Process**

There are two migration scripts available because there are two different ways that a system can be identified by the RHN Classic system: by its system ID or by an installation number. The migration script to use depends on which identifier is used for the system in RHN Classic:

- *Red Hat Enterprise Linux 5 and 6.* Migrating a system's registration and subscriptions from RHN Classic to Customer Portal Subscription Management (both hosted services).

This uses the **rhn-migrate-classic-to-rhsm** migration script.

- *Red Hat Enterprise Linux 5 only.* Migrating the local system configuration using RHN Classic-style channels to using Customer Portal Subscription Management certificates for installed products. This migration is based on the installation number for the system. Using the installation number as the basis for migrating information is particularly useful for a disconnected (offline) system, which has no way to connect to RHN Classic.

This uses the **install-num-migrate-to-rhsm** migration script.

### 3.11.3. Options and Details

After migrating, the local clients can be reconfigured to use Subscription Asset Manager or Satellite 6 to add more management options, such as the ability to define multiple sub-organizations and system groups to attach subscriptions based on local system rules.

## 4. REVISION HISTORY

<b>Revision 1.3-6</b>	<b>April 13, 2014</b>	<b>Ella Deon Ballard</b>
Updating instance-based and virtual setup sections.		

<b>Revision 1.3-4</b>	<b>September 18, 2013</b>	<b>Ella Deon Lackey</b>
New content and reorganization for the SAM 1.3 release.		