



Red Hat Subscription Management All Subscription Docs

Using Notification and Reports with Red Hat Subscription Management

to view and respond to system notifications, infrastructure reports, and other
subscription information

Edition 3

Last Updated: 2018-05-01

Red Hat Subscription Management All Subscription Docs Using Notification and Reports with Red Hat Subscription Management

to view and respond to system notifications, infrastructure reports, and other subscription information

Edition 3

Red Hat Subscription Management Documentation Team
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

Red Hat's Subscription Management tools and applications provide different ways to view system-level and organization-level notifications and statuses and to respond to changing subscription needs. This guide details the different reporting and notification mechanisms and some quick paths to remediating insufficient and expired subscriptions.

Table of Contents

1. UNDERSTANDING SUBSCRIPTION STATUS, NOTIFICATIONS, AND COMPLIANCE	3
1.1. About Relationships Between Subscriptions and Systems	3
1.1.1. Interactions with Subscriptions, Products, and Systems	4
1.1.2. Counting Subscriptions	4
1.2. Status: Expiration and Validity Ranges	5
1.3. Different Subscription States	6
1.4. Notifications and Messages	7
1.5. Compliance: Ways to Remediate Changing Subscription States	8
2. VIEWING STATUS AND NOTIFICATIONS	8
2.1. Red Hat Subscription Manager: Status and Notifications	8
2.1.1. Status	8
2.1.2. Notifications for Subscription Expiration	9
2.2. Portal: Status and Utilization	10
2.2.1. Checking Status	10
2.2.2. Viewing Subscriptions for a System	12
2.2.3. Subscription Utilization	13
2.2.4. Expired and Expiring Subscriptions	14
2.3. Subscription Asset Manager: System-Level Subscription Information	16
2.3.1. High-Level Information in the Dashboard	16
2.3.2. Viewing System Notifications	18
2.3.3. Checking Individual System Status	19
2.4. Subscription Asset Manager: Organizational Usage Reports (TECH PREVIEW)	21
2.4.1. Prerequisites	21
2.4.2. Setting up Reporting	21
2.4.3. Creating Report Filters	22
2.4.4. Running Reports	24
2.4.5. Subscription Asset Manager Reports Results and Data	25
2.4.6. Enhanced Reporting Logs	27
2.5. Subscription Asset Manager: Satellite 5.6 Usage Reports	27
2.5.1. About Satellite Consolidated Reports	27
2.5.2. Prerequisites	30
2.5.3. Configuring Reporting	31
2.5.4. Running Reports and Getting Results	37
2.5.5. Troubleshooting Enhanced Reports	42
2.6. Portal: Subscriptions Used in Both RHN Classic and the Portal	44
3. RESPONDING TO NOTIFICATIONS AND CHANGED STATUS	45
3.1. Red Hat Subscription Manager: Autoattaching and Updating Subscriptions	45
3.1.1. Setting Preferences for Systems	45
3.1.2. Autoattaching in Response to Notifications	51
3.1.3. Autoattaching at Registration	52
3.1.4. Autoattaching after Registration	52
3.2. Portal: Autoattaching and Updating Subscriptions	52
3.2.1. Setting a Preferred Service Level	52
3.2.2. Viewing the Operating System Release Preference	54
3.2.3. Autoattaching Subscriptions	54
3.3. Portal: Resolving Over-Utilizing Subscriptions	55
3.4. Subscription Asset Manager: Autoattach and Preferences	57
3.4.1. Configuring Autoattach Preferences for a System	57
3.4.2. Running Autoattach Operations Manually	58

- 4. MANAGING ERRATA NOTIFICATIONS 59
 - 4.1. Portal: Managing Errata Notifications for Registered Systems 59
 - 4.2. Red Hat Subscription Manager: Viewing Package Profiles for the Local System 62
- 5. REVISION HISTORY 63

1. UNDERSTANDING SUBSCRIPTION STATUS, NOTIFICATIONS, AND COMPLIANCE

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

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.

Subscription management is a way of identifying and creating relationships between the systems in your IT environment and the software products that you have available through Red Hat.

Subscriptions follow a set lifecycle:

1. An account buys a *subscription* to a *product*, which gives them access to Red Hat's Content Delivery Network, errata and patches, upgrades, and support.

A subscription is usable or *valid* for a set amount of time and can be used a certain number of times (the *quantity*).

2. A server is added, or *registered*, to the *inventory* for the subscription management service. Once the system is added, there is a list of facts defining attributes about the system and a profile which lists the installed products and their versions.
3. A subscription is *attached* to a system, so that the system is entitled to support services and content for that product.

Subscriptions can be manually assigned by an administrator or they can be automatically attached by the subscription process based on what subscriptions best match the system attributes and installed products.

4. As the validity period of the subscription ends or as new products are added, then new subscriptions must be attached to the system so that it maintains coverage.

1.1. About Relationships Between Subscriptions and Systems

1.1.1. 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 – 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.

1.1.2. 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.

1.2. Status: Expiration and Validity Ranges

Subscriptions are active for a certain period of time, called the *validity period*. When a subscription is purchased, the start and end dates for the contract are set.

On a system, there can be multiple subscriptions attached. Each product requires its own subscription. Additionally, some products may require multiple quantities of subscriptions for the product to be fully covered. For example, a 16 socket machine may require four 4-socket operating system subscriptions to cover the socket count.

The **My Installed Software** tab shows the subscription status for the entire system. It also shows a date; that is the first date that a product subscription goes from valid to invalid (meaning it expires).

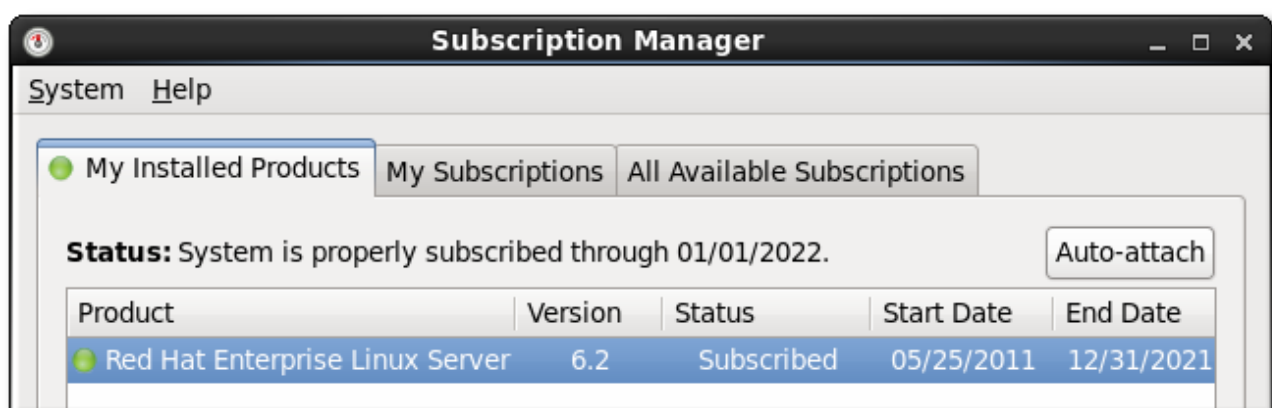


Figure 1. Valid Until...

For example, if you have a Load Balancer subscription that expires on April 17 and all other product subscriptions are valid until October 1, the **Certificate Status** summary shows that the certificates are valid until April 17, the closest expiration date.

Subscriptions can string together in a queue. For example, you have a 4-socket system that uses two 2-socket subscriptions to cover the socket count. However, the system actually has three subscriptions attached to it:

- 2-socket subscription A expires April 1, 2012.
- 2-socket subscription B expires July 31, 2013.
- 2-socket subscription C starts March 1, 2012 and expires April 1, 2014.

The system is valid through July 31, 2013, because Subscription C is already queued up to replace Subscription A when it expires.

1.3. Different Subscription States

All of the subscription management tools – Red Hat Subscription Manager on the local systems, Subscription Asset Manager for on-premise services, and Customer Portal Subscription Management for hosted services – provide a series of log and UI messages that indicate any changes to the valid certificates of any installed products for a system.

The status of the system subscriptions is color-coded, where *green* means all subscriptions are attached for all installed products, *yellow* means that some products may not have all of the required subscriptions attached but updates are still in effect, and *red* means that updates are disabled.

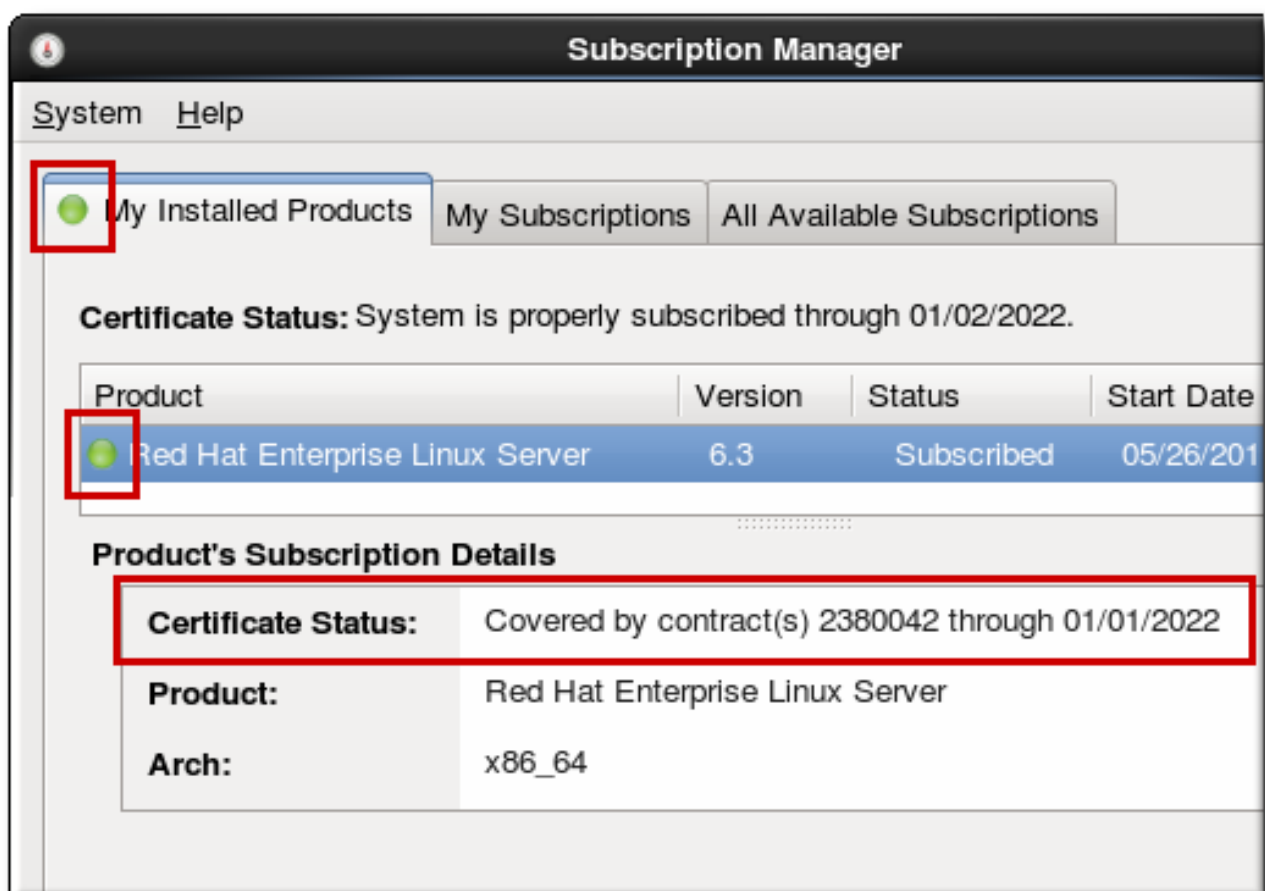





Figure 2. Color-Coded Status Views

The command-line tools also indicate that status of the machine. The green, yellow, and red codes translate to text status messages of *subscribed*, *partially subscribed*, and *expired/not subscribed*, respectively.

```
[root@server ~]# subscription-manager list
+-----+
      Installed Product Status
+-----+

ProductName:          Red Hat Enterprise Linux Server
Status: Not Subscribed
Expires:
SerialNumber:
ContractNumber:
AccountNumber:
```

Table 1. Status Labels and Icons

Icon	Message	Description
	<ul style="list-style-type: none"> Valid Subscribed Sufficient 	All products on the system have a valid subscription attached for it, and the system itself is fully covered (there are adequate subscriptions for the core, socket, or RAM counts).
	<ul style="list-style-type: none"> Insufficient Partially subscribed 	Some products do not have subscriptions or do not have an adequate number of subscriptions attached. Updates and support policies are still in effect.
	<ul style="list-style-type: none"> Invalid Not subscribed 	No subscriptions are attached to the system or to the product. Updates will be disabled and support responses may be affected.

1.4. Notifications and Messages

Part of Red Hat Subscription Manager on local systems is the `rhsmcertd` process, which interacts with the subscription service to check for new subscriptions, monitors subscription expiration for subscriptions attached to the local system, and tracks installed products for required subscriptions.

`rhsmcertd` provides a mechanism for issuing warnings for expiring or insufficient subscriptions on the local system. Because it interacts with the subscription service (such as Customer Portal Subscription Management or Subscription Asset Manager), there are also views in the subscription management application dashboards which show the overall status of individual systems and of the infrastructure as a whole.

In practice, subscription compliance is managed at the system-level, so Red Hat Subscription Manager messages and notifications are invaluable for responding to systems. For overall infrastructure

tracking, resource planning, and regulatory or standards compliance, then the high-level overviews in Customer Portal Subscription Management or Subscription Asset Manager are exceptionally useful for auditing, purchasing, and other planning activities.

1.5. Compliance: Ways to Remediate Changing Subscription States

When a system has insufficient or expired subscriptions, the way to remediate it is to update the subscriptions for the system. This can be done manually, but the most efficient management is to enable *autoattaching* on the systems so that subscriptions are automatically attached or updated when the subscription status for the system changes.

Autoattaching is part of the `rhsmcert` process. Every four hours (in the default configuration), the `rhsmcert` process checks the current installed products, the current attached and active subscriptions, and the available subscriptions in the subscription service. If autoattaching is enabled, then the `rhsmcert` process can automatically use the best-matched subscriptions.

It is also possible to run an asynchronous autoattach operation. On the local system, this can be done with the Red Hat Subscription Manager utilities. Both Subscription Asset Manager and Customer Portal Subscription Management also provide options to initiate an autoattach operation on all registered systems.

2. VIEWING STATUS AND NOTIFICATIONS

2.1. Red Hat Subscription Manager: Status and Notifications

Red Hat Subscription Manager provides system-level information about installed products, subscription status (per product), and available subscriptions.

2.1.1. Status

Red Hat Subscription Manager provides a series of log and UI messages that indicate any changes to the valid certificates of any installed products for a system. In the Subscription Manager GUI, the status of the system subscriptions is color-coded, where *green* means all subscriptions are attached for all installed products, *yellow* means that some products may not have all of the required subscriptions attached but updates are still in effect, and *red* means that updates are disabled.

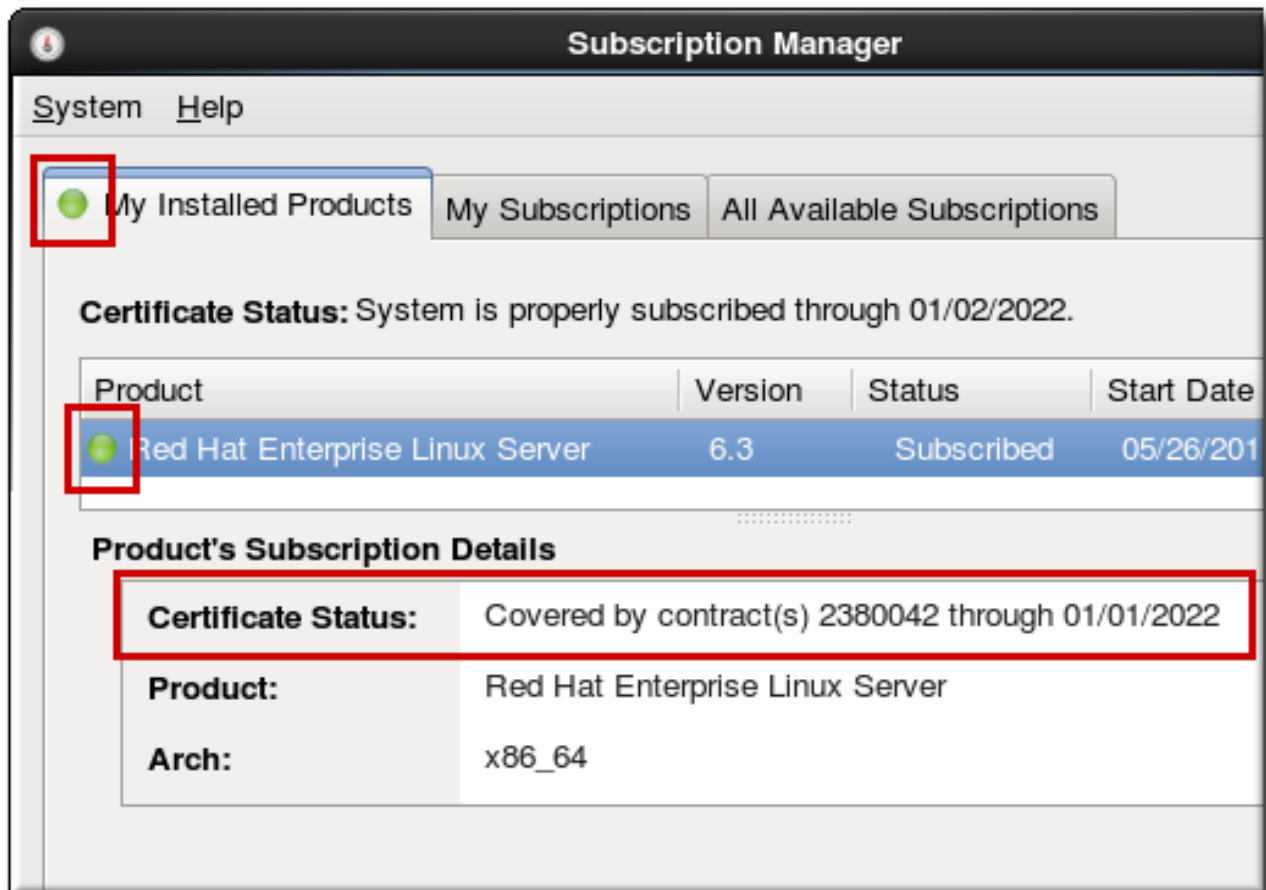


Figure 3. Color-Coded Status Views

The command-line tools also indicate that status of the machine. The green, yellow, and red codes translate to text status messages of *subscribed*, *partially subscribed*, and *expired/not subscribed*, respectively.

```
[root@server ~]# subscription-manager list
+-----+
|           Installed Product Status           |
+-----+
|
| ProductName:           Red Hat Enterprise Linux Server
| Status: Not Subscribed
| Expires:
| SerialNumber:
| ContractNumber:
| AccountNumber:
```

2.1.2. Notifications for Subscription Expiration

Whenever there is a warning about subscription changes, a small icon appears in the top menu bar, similar to a fuel gauge.

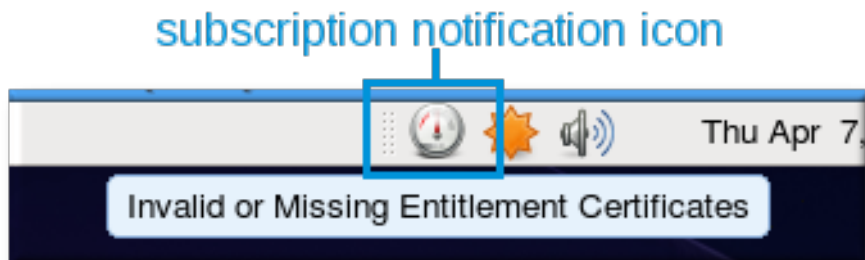


Figure 4. Subscription Notification Icon

As any installed product nears the expiration date of the subscription, the Subscription Manager daemon will issue a warning. A similar message is given when the system has products without a valid certificate, meaning either there is not subscription attached that covers that product or the product is installed past the expiration of the subscription. Clicking the **Manage My Subscriptions...** button in the subscription notification window opens the Red Hat Subscription Manager GUI to view and update subscriptions.

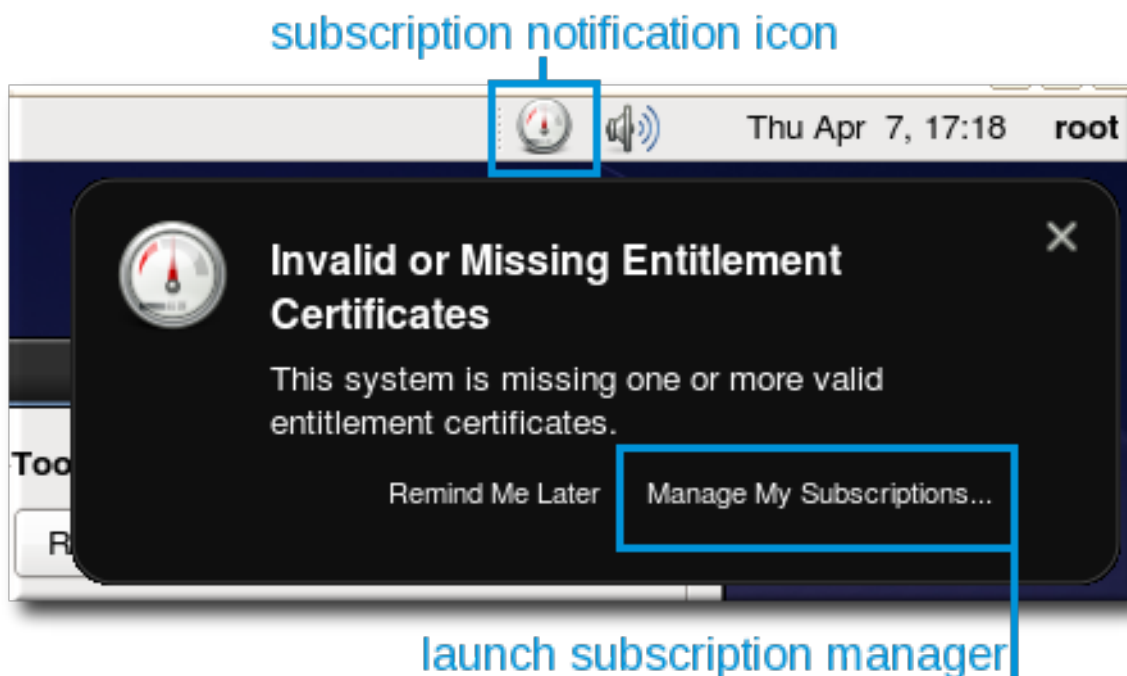


Figure 5. Subscription Warning Message

2.2. Portal: Status and Utilization

In Customer Portal Subscription Management, the subscription status shows the overall status for the system, but not the subscription status for individual installed products on that system.

The Portal provides high-level information, such as the status summary for a system or the overall status for all systems.

2.2.1. Checking Status

The portal provides a way to make sure that a system has all of its subscriptions up-to-date. The system's details page has a **Certificate Status** summary that shows whether all of the installed products on that system have the appropriate subscriptions attached.

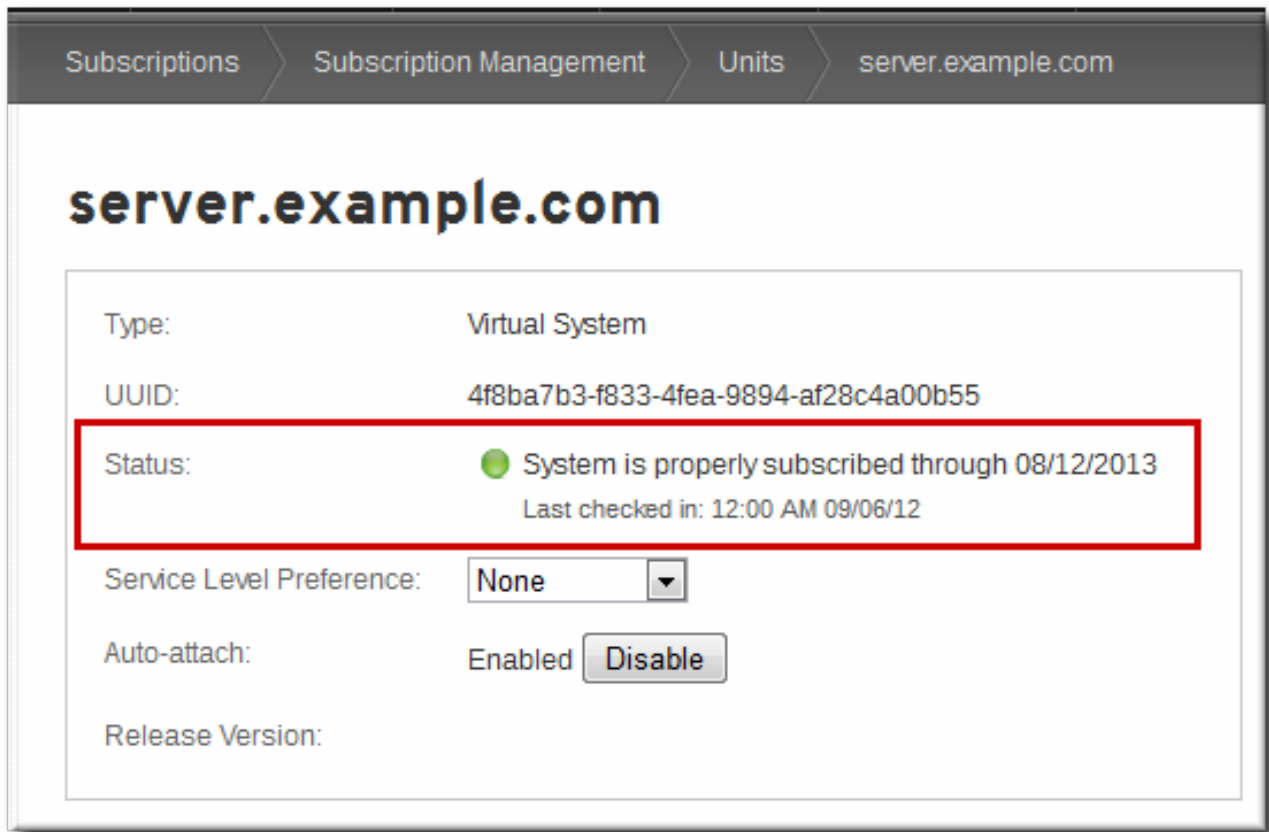


Figure 6. Subscription Status

The status shows how long the current subscriptions are valid and the last time that the subscription certificates were updated.

The status of the system subscriptions is color-coded:

- *Green* means all products have a valid subscription.
- *Yellow* means that some products may not have active subscriptions but updates are still in effect.
- *Red* means that updates are disabled.

The system status is determined by the local system itself, based on its system facts, installed products, and local subscription certificates. This information is synced with the subscription service every 24 hours (by default). The Customer Portal does not store the information about installed products; it relies on the information from the system itself. If a system is offline and cannot send its information, then the Customer Portal reflects an unknown status.

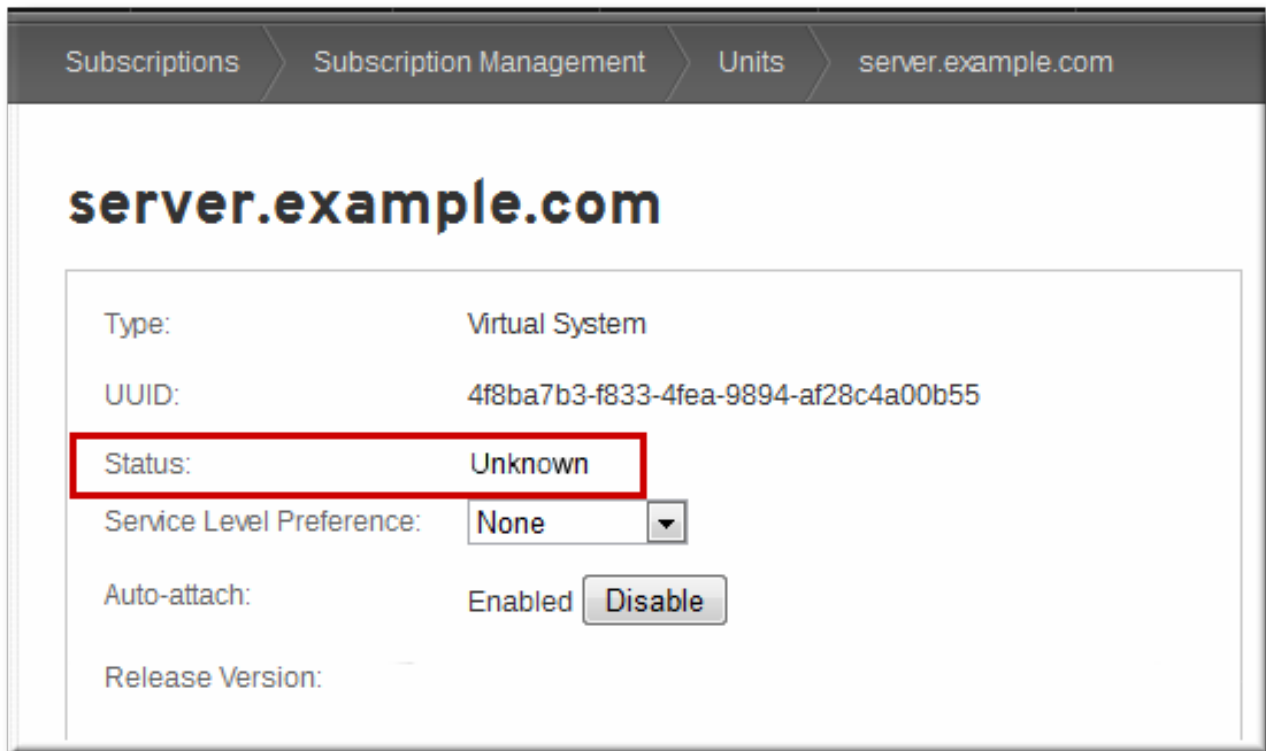


Figure 7. Unknown Subscription Status

2.2.2. Viewing Subscriptions for a System

When a subscription is attached to a system, then it is listed on the **Attached Subscriptions** tab with its contract number and expiration date.

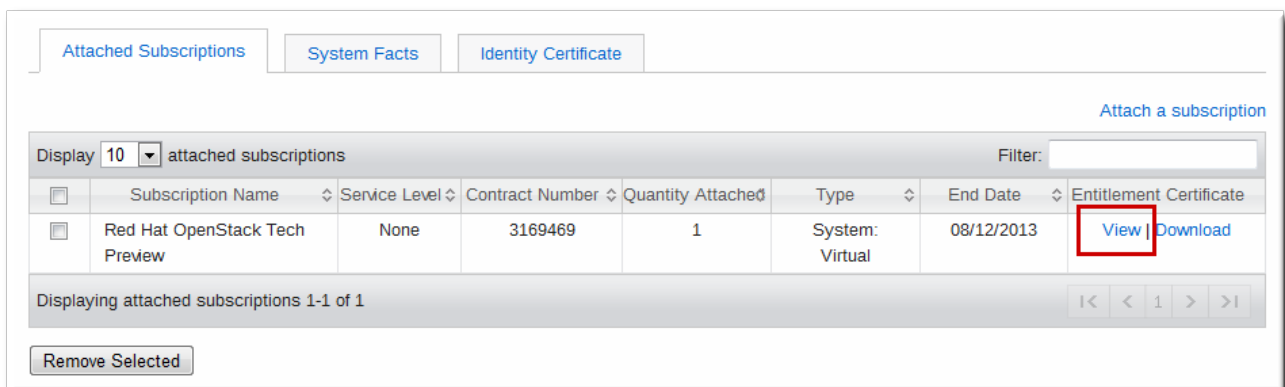


Figure 8. Subscription Details Link

Clicking the **View** link for that subscription opens up much more detail about the subscription, including a list of products that the subscription provides, its order information, the quantity and type of subscriptions available in that one subscription, and its certificate (which can be regenerated on the system or downloaded for view).

Subscriptions
Subscription Management
Units
server.example.com
Entitlement Certificate

Entitlement Certificate: 8a85f9843c01d7fa013c179c39c745c5

Download
Regenerate

Details

Name: [server.example.com](#)
Type: Virtual System
UUID: 4f8ba7b3-f833-4fea-9894-af28c4a00b55

Products
Order Info
Content Sets

Contract Number : 3169469
Order Number : 2607527
Account Number : 1460290
Start Date : 08/12/2012
End Date : 08/12/2013
SKU : SER0406
Quantity : 1
Quantity Used : 1
Warning Period : 0
Provides Management : 0

Figure 9. Subscription Details

2.2.3. Subscription Utilization

Administrators need to have a sense of all of the subscriptions, altogether, regardless of whether they match the architecture or installed products on any of the systems in inventory. The Customer Portal provides three ways of looking at subscriptions, with slightly different perspectives:

- All subscriptions that are active and attached (total counts)
- All available subscriptions that can be used by systems in Customer Portal Subscription Management
- Subscriptions in Customer Portal Subscription Management that match a specific system's architecture, socket count, installed products, or other characteristics

In the **Subscriptions Overview** page has a link to the utilization summary. The **Subscription Utilization** page gives the current count for every active subscription for the entire account, and a total count of every used subscription, regardless of whether it is used in RHN Classic or Customer Portal Subscription Management. These numbers are updated whenever the subscription count changes in the subscription management service.

The total counts are broken down by type first, and then the number of subscriptions of that type per subscription management service.

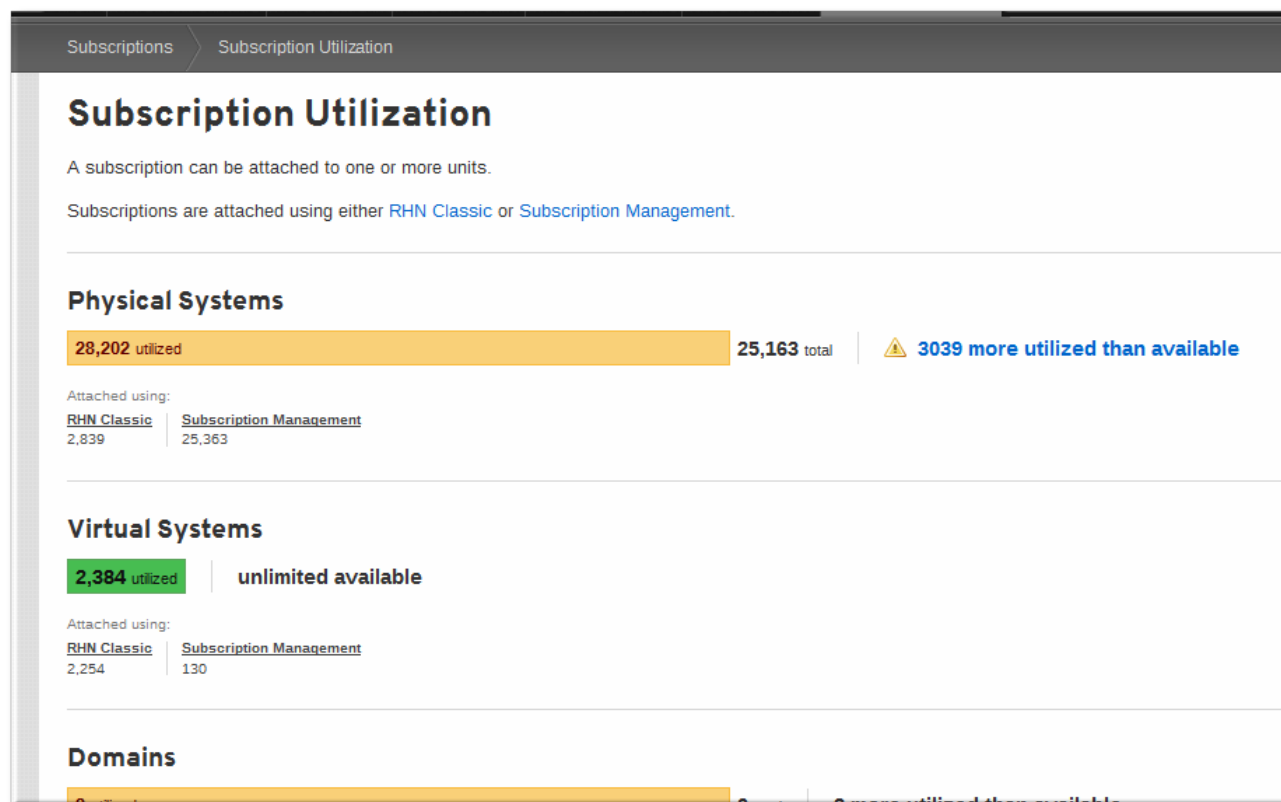


Figure 10. Total Counts of Subscriptions for All Subscription Services

2.2.4. Expired and Expiring Subscriptions

The top of the subscription overview page gives three simple, clear numbers related to subscriptions: how many are active, how many will expire within 120 days, and how many have expired in the past 30 days (and are eligible for renewal).

These are total numbers for the entire account. It does not matter how many subscriptions are attached, how many systems there are, and whether the subscription is registered with Customer Portal Subscription Management, RHN Classic, or a subscription management application.

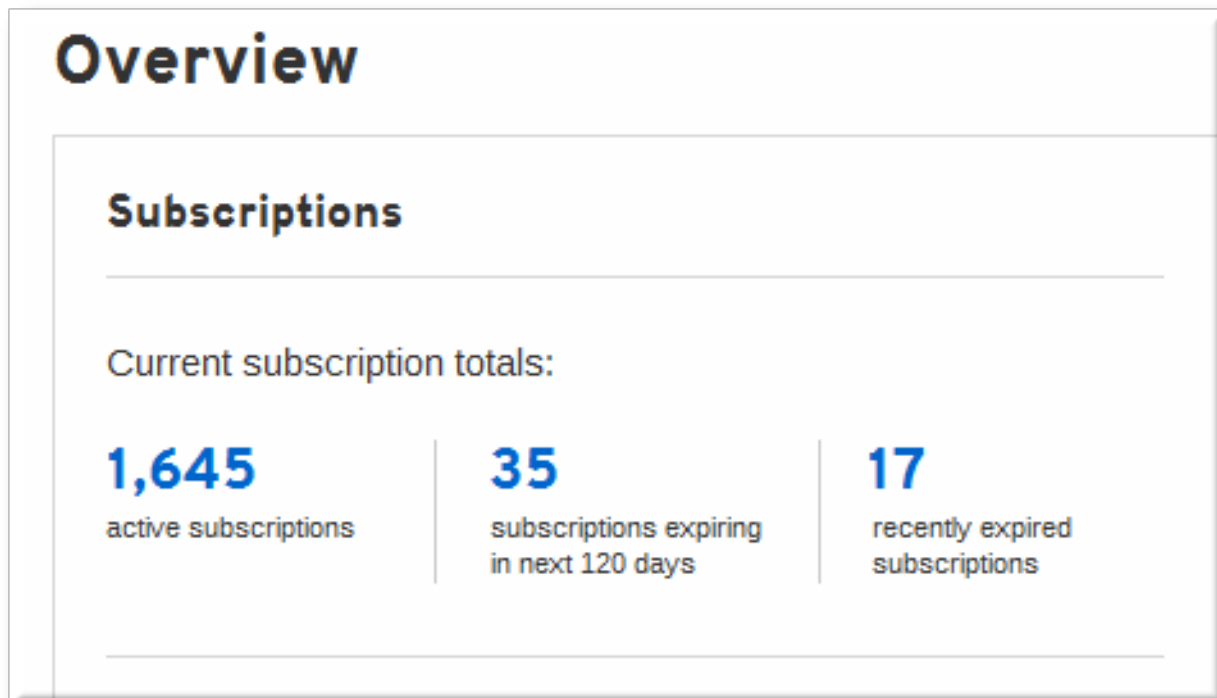


Figure 11. Subscription Overview

Clicking on any of the numbers opens the tab in the subscription inventory for that category of subscriptions.

- **active subscriptions** goes to the **Active** tab.
- **subscriptions expiring in the next 120 days** goes to the **Available for Renewal** tab.
- **recently expired subscriptions** goes to the **Recently Expired** tab.

The tab lists the subscription names, contract numbers, and start/end dates to make it easier to track and manage subscriptions.

Products ▾	Support ▾	Downloads	Security ▾	Subscriptions ▾
Subscriptions	Inventory			
Subscriptions				
<div>Active</div> <div>Expiring Soon</div> <div>Recently Expired</div> <div>All</div>				
<p>The following is a list of subscriptions that have recently expired. Click on a subscription name for additional information or to learn more about your subscription renewal options. If you have any questions, please contact Customer Service.</p>				
Display 10 ▾ available subscriptions Filter: <input type="text"/>				
Subscription Name ▲	Contract Number ⇅	Start Date ⇅	End Date ⇅	
Self-Supported Red Hat Enterprise Linux Server, (2 sockets) (Up to 1 guest)	10241790	2013/07/29	2013/08/27	
OpenShift Enterprise, 2 Core, 8 GB	10236688	2013/07/23	2013/08/21	
OpenShift Enterprise, 2 Core, 8 GB	10237259	2013/07/23	2013/08/21	

Figure 12. Recently Expired Tab

Clicking the name of any subscription in the **Available for Renewal** or **Recently Expired** tabs opens the details page for that contract, with renewal information. If the subscription was purchased directly from Red Hat, then it can be renewed through that page; if it was purchased from a vendor, then there is contact and ordering information supplied.

The screenshot shows the 'Fuse' page for a subscription. The top navigation bar includes links for Home, Knowledge, Groups, Support, Downloads, Security, and Subscriptions. The main content area is divided into several sections:

- Status:** Contract Number: 3211498, Start Date: 2012-09-14, End Date: 2012-12-12, Subscription Status: **expired** (highlighted with a red box).
- Installation:** Installation Number: ac01747388b83574, Subscription Number: 2645507.
- Support:** Phone Support: None, Web Support: None, RHN Support: Yes.
- Paper Certificate:** A dropdown menu set to 'English' and a 'Download' button.
- Utilization:** Total Entitlements: 1.
- Renewal Information:** Information about renewing a product purchased through a partner or Sales Associate. It includes text about renewing through a partner and a link to 'Find a reseller in EMEA'.

Figure 13. (Expired) Product Renewal Information

2.3. Subscription Asset Manager: System-Level Subscription Information

The ultimate goal of subscription 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 (top-level account), looking down at the total infrastructure of systems and all subscriptions.

Subscription Asset Manager has several different ways of conveying subscription and system information. This includes information about insufficient or expiring subscriptions, which is invaluable to administrators for maintaining current subscriptions.

2.3.1. High-Level Information in the Dashboard

The Subscription Asset Manager dashboard shows a count of all *systems* registered with that organization and their overall subscription status.

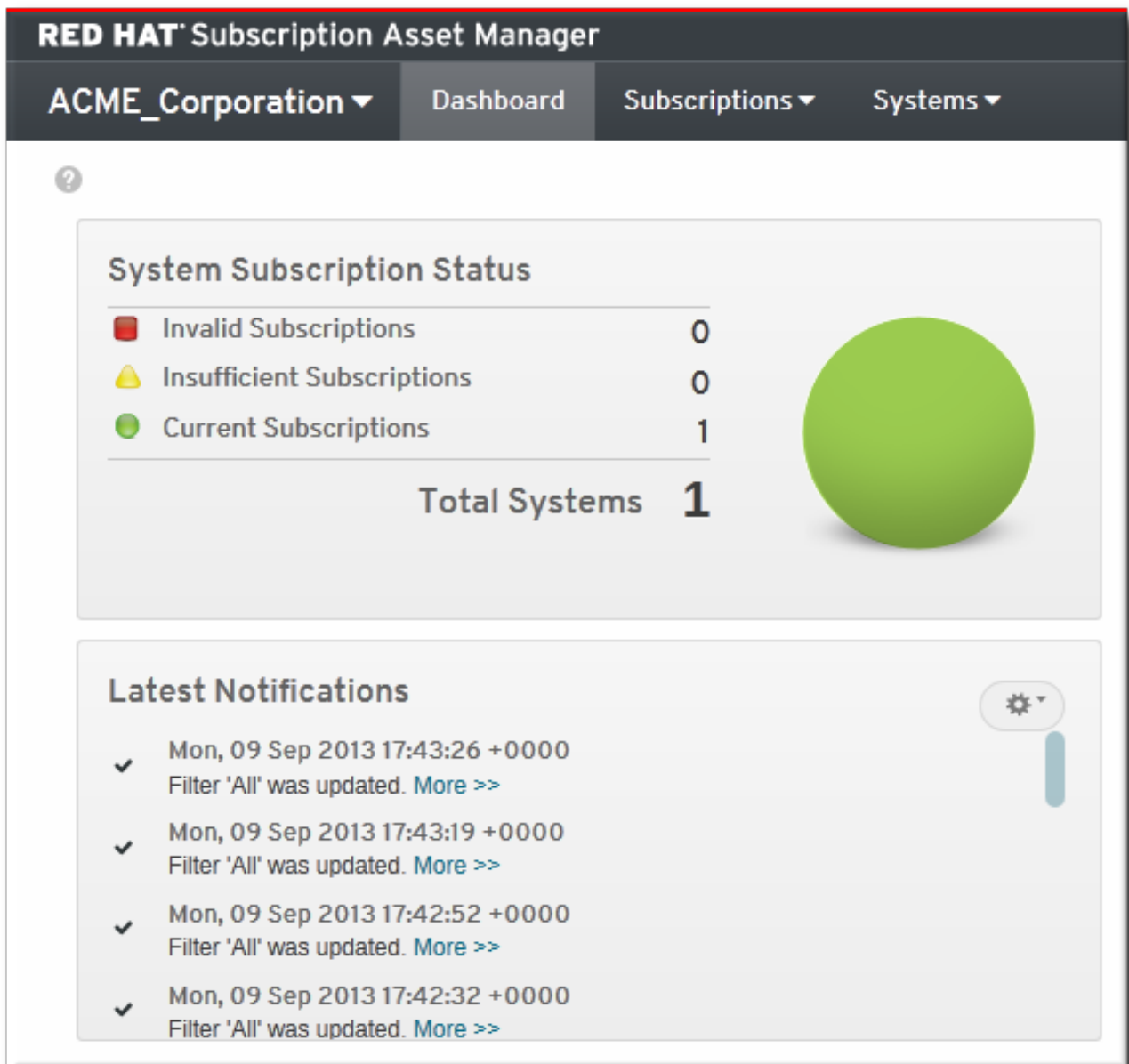


Figure 14. The Subscription Asset Manager Dashboard

Subscription status refers to the status of all subscriptions for all products which are installed on a system. For example, if a system has Red Hat Enterprise Linux, OpenShift, and Directory Server all installed, then that system must have subscriptions for Red Hat Enterprise Linux, OpenShift, and Directory Server attached to it so that it is current.

There are three categories of subscription status:

- *Current subscriptions* mean that a system has a subscription for every install product, in the appropriate quantity.
- *Invalid subscriptions* mean that a system has installed products but at least one of those products has no corresponding subscription for it.
- *Insufficient subscriptions* is a slightly more complex state. It means that at least one installed product has some subscriptions for it, but not enough. Each subscription states some attribute that applies to it. For example, an operating system subscription may specify a certain number of cores or a certain amount of RAM. If a system has four cores and the subscription specifies that it covers two cores, then the system requires two subscriptions. If only one subscription is attached, then the system is in an *insufficient* state.

2.3.2. Viewing System Notifications

Every time any action is taken in Subscription Asset Manager – such as adding a new user, editing an organization's configuration, importing a manifest, or running a report – a system notification is recorded. Notifications include both success and error messages.

The most recent notifications are shown in the **Latest Notifications** area of the **Dashboard** tab.

The screenshot shows the Red Hat Subscription Asset Manager interface. At the top, there is a header bar with the title "RED HAT Subscription Asset Manager" and a navigation menu with "ACME_Corporation", "Dashboard", "Subscriptions", and "Systems". The "Dashboard" tab is selected. Below the header, there is a "System Subscription Status" section with a table showing the status of subscriptions and a large green circle representing the total systems.

System Subscription Status		
Invalid Subscriptions	0	
Insufficient Subscriptions	0	
Current Subscriptions	1	
Total Systems		1

Below the status section, there is a "Latest Notifications" section with a list of notifications and a gear icon to adjust the number of notifications displayed.

- ✓ Mon, 09 Sep 2013 17:43:26 +0000
Filter 'All' was updated. [More >>](#)
- ✓ Mon, 09 Sep 2013 17:43:19 +0000
Filter 'All' was updated. [More >>](#)
- ✓ Mon, 09 Sep 2013 17:42:52 +0000
Filter 'All' was updated. [More >>](#)
- ✓ Mon, 09 Sep 2013 17:42:32 +0000
Filter 'All' was updated. [More >>](#)

Figure 15. Notifications in the Dashboard

By default, the most recent five notifications are displayed, but that can be changed to 15 or 30 notifications by clicking the gear icon.

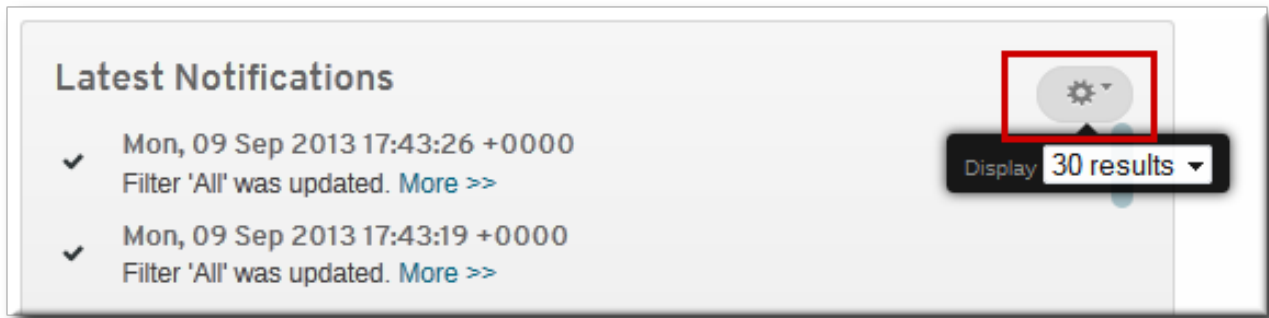


Figure 16. Changing the Number of Notifications Displayed

The complete list of notifications can be opened by clicking the **More** link by any notification in the **Dashboard** or by clicking the notification icon in the upper right of the Subscription Asset Manager UI.

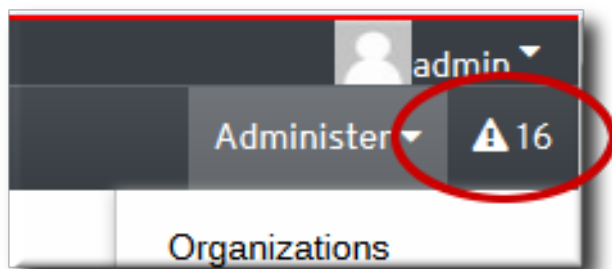


Figure 17. Notification Link in the Admin Menu

The **User Notifications** page lists the time and date of each notification, the type of notification (either **success** or **error**), and a description of the notification.

User Notifications			
Delete All		Filter...	
CREATED (UTC)	LEVEL	ORGANIZATION	DESCRIPTION
Tue, 06 Aug 2013 15:31:15 +0000	success	Red Hat SatTeam	Successfully added system[s] to group 'Example Group B'.
Tue, 06 Aug 2013 15:30:52 +0000	success	Red Hat SatTeam	Successfully added system[s] to group 'Example Group B'.
Tue, 06 Aug 2013 15:30:38 +0000	success	Red Hat SatTeam	Successfully added system[s] to group 'Example Group B'.
Tue, 06 Aug 2013 15:30:30 +0000	success	Red Hat SatTeam	System Group Example Group B created successfully.
Tue, 06 Aug 2013 15:30:05 +0000	success	Red Hat SatTeam	Successfully added system[s] to group 'Example Group A'.
Tue, 06 Aug 2013 15:29:51 +0000	success	Red Hat SatTeam	System Group Example Group A created successfully.
Fri, 02 Aug 2013 12:29:07 +0000	success		User updated successfully.

Figure 18. List of Notifications

Individual notifications cannot be deleted, but it is possible to clear the entire notifications queue by clicking the **Delete All** link.

2.3.3. Checking Individual System Status

The **Dashboard** shows the cumulative subscription status counts for all systems within the Subscription Asset Manager organization. However, it is possible to view the information for each individual system, including subscription expiration dates and installed products.

First, open the system entry:

1. Hover over the **Systems** tab in the top menu and select the **All** item.

2. In the search box on the left of the systems column, search for the specific system.
3. Click the name of the system in the column on the left.

The server list entry itself shows the overall subscription status for the system by using a red (invalid), yellow (insufficient), or green (current) icon.

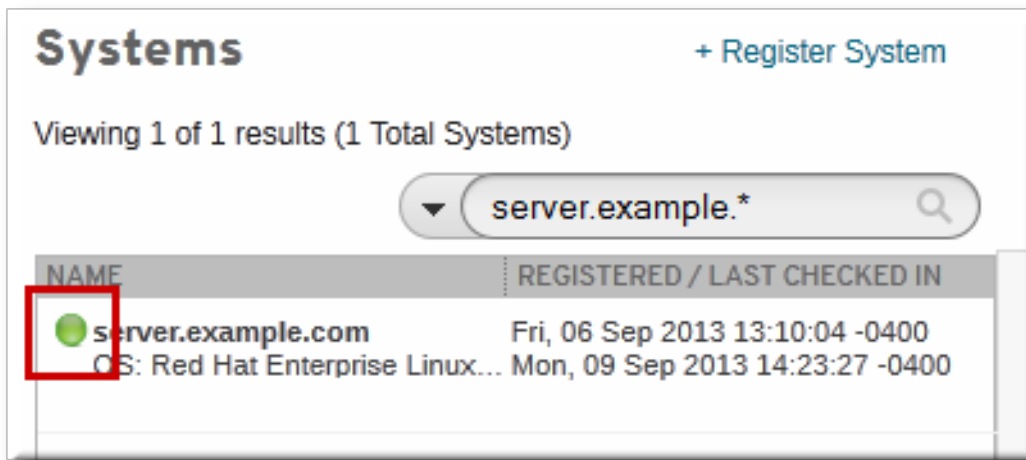


Figure 19. Status Icon in the System List

There are a couple of different places in the system page that show the details of the subscription status.

- The **Details** tab shows the status and (if current) the expiration date for the subscriptions.

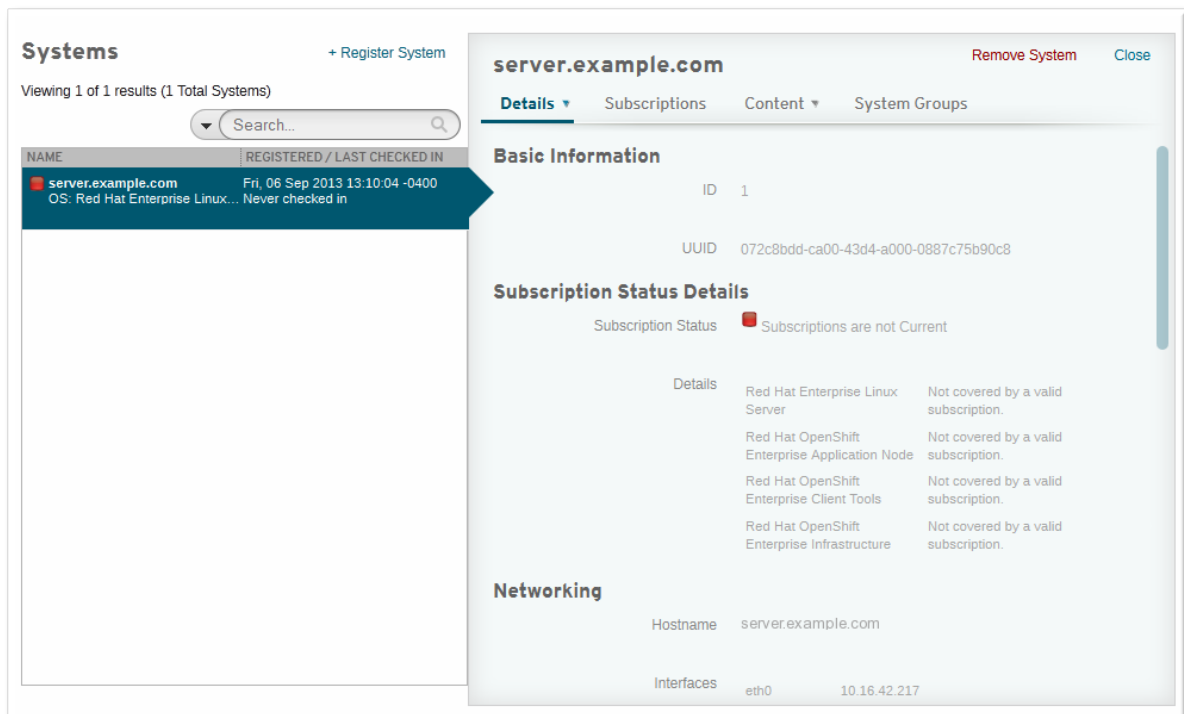


Figure 20. Status Details

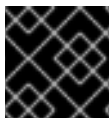
- The **Subscriptions** tab also shows the status and the expiration date (if subscriptions are current). Additionally, the **Subscriptions** tab has a list of available and attached subscriptions, so that the subscriptions for the system can be reassigned as necessary.

The screenshot displays the 'Systems' page on the left, showing a search for 'server.example.*' and a list of systems. The right pane shows the details for 'server.example.com', including its registered/last checked in times and a list of current subscriptions. The subscriptions table is as follows:

SUBSCRIPTION	SLA	CONTRACT	QUANTITY	START	END
OpenShift Subscription	None	10046117	1	2012-11-26	2022-01-01
OpenShift Subscription	None	10046117	1	2012-11-26	2022-01-01
Red Hat Subscription	None	2596950	1	2011-10-08	2022-01-01

Figure 21. Status and Subscription Lists

2.4. Subscription Asset Manager: Organizational Usage Reports (TECH PREVIEW)



IMPORTANT

Running reports for Subscription Asset Manager organizations is tech preview.

Organizations in Subscription Asset Manager are discrete and separate. Cumulative information in areas such as the **Dashboard** are for the organization being viewed only. Even if multiple organizations are configured on a single Subscription Asset Manager server, only one organization is viewable at a time.

It can be useful or even necessary to have a cross-organization or cross-distributor view of subscription allocation and status in order to maintain regulatory and policy compliance. Subscription Asset Manager reports can be configured to return information for multiple organizations, which allows that kind of cross-organizational data to be compiled.

2.4.1. Prerequisites

When using enhanced reporting, there are some additional system requirements:

- All of the other Subscription Asset Manager installation prerequisites
- The **cron** service must be running.
- An additional 4 GB of disk space for the reporting database.
- Additional packages for the reporting server
 - `splice`
 - `ruby193-rubygem-splice_reports`
 - `spacewalk-splice-tool`

2.4.2. Setting up Reporting

Subscription Asset Manager reporting is an additional module that requires additional packages to be installed.

If a Subscription Asset Manager instance is already configured, then the additional packages can be pulled in. For example, using `yum`:

```
[root@server ~]# yum install splice ruby193-rubygem-splice_reports  
spacewalk-splice-tool
```

When the reporting packages are installed, there is an additional item in the administration menu to create and run reports.

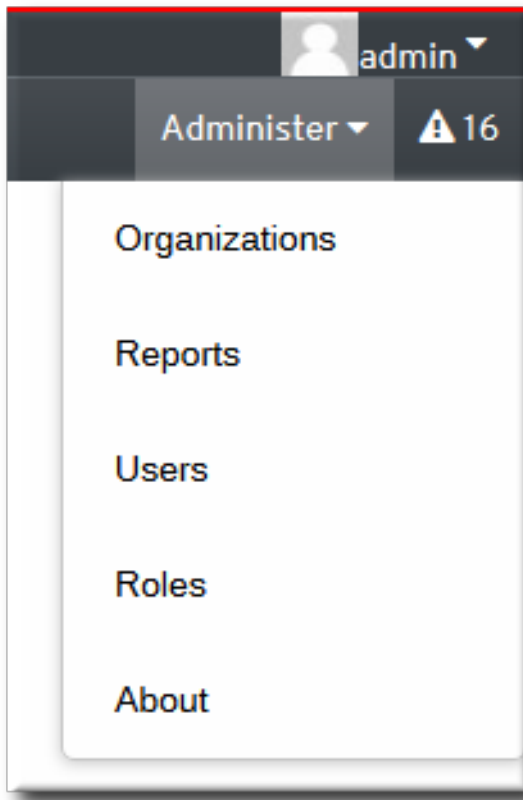


Figure 22. The Reports Menu Item

2.4.3. Creating Report Filters

The Subscription Asset Manager report is actually a collection of filters that collect and structure data for different organizations.

There is one default report that checks for subscriptions for all statuses, in all organizations, in all Satellite servers (if configured), over the past 24 hours.

There is a lot more flexibility possible with the report form, however. In particular, there are three versatile settings:

- The organizations to check for the report
- The subscription statuses to include
- The date range to check; this looks for systems which had the status *within the given range* which may not necessarily be the current status for the system



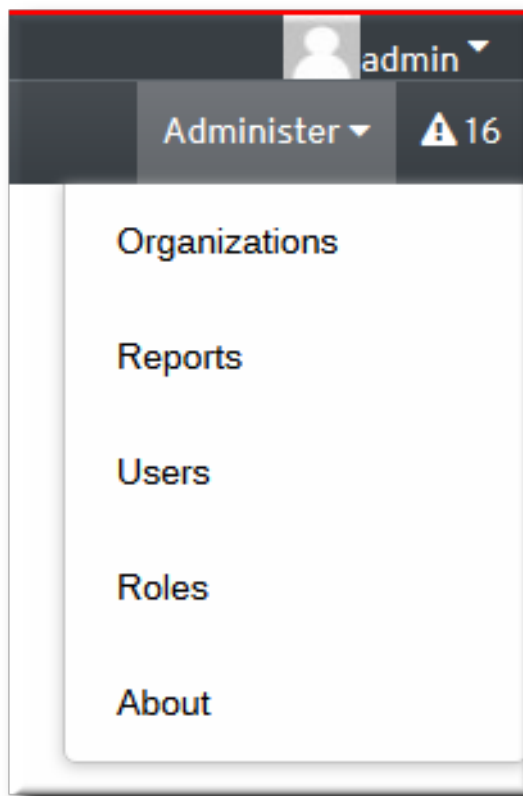
NOTE

Here are some good reports for tracking infrastructure status and for compliance audits:

- All systems that have changed to invalid or insufficient (status) in the past 24 hours.
- All systems that will have invalid or insufficient subscriptions (meaning, the existing subscriptions will expire) within the next three months.

To create a new report filter:

1. Click the **Reports** item in the administration menu.



2. In the left column, click the **New Filter** link.
3. Fill in the required information for the report, including the organizations, statuses, date range, and active states.

Report Filters + New Filter

Viewing 2 of 2 results (2 Total Report Filters)

Search...

NAME

Red Hat Default Report

All

New Filter

Name: Example report

Description: for 2013

Subscription Status: Current Invalid

Satellite Server: sam13-dlackey-demo

Organizations: ACME_Corporation

LifeCycle State: Active Inactive

Search Previous Number of Hours: Search by Date Range:

Start Date: 01/01/2013

End Date: 12/31/2013

Save Filter

4. Click the **Save Filter** button.

NOTE

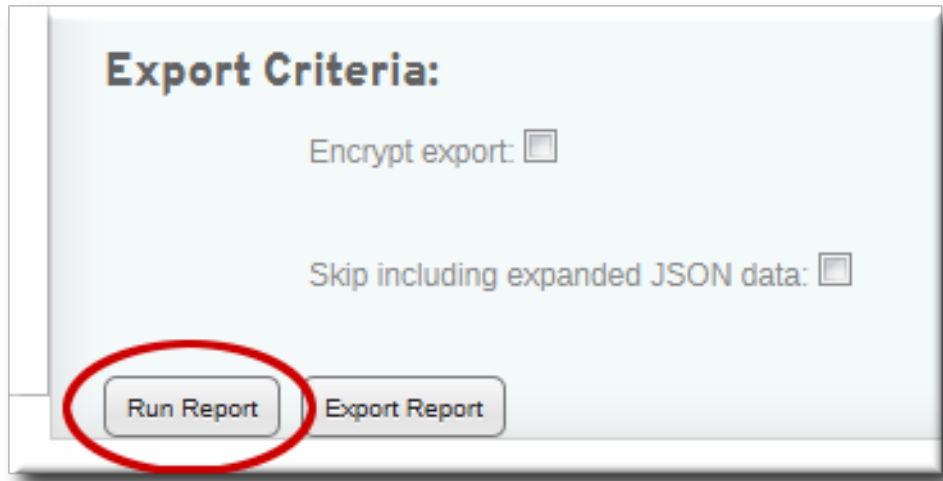
The data within the Subscription Asset Manager database includes historic subscription statuses. This allows reports to be generated to track subscriptions at a given point in time, not just the current date.

For example, if purchasing occurs in July, then a report can be configured to search for insufficient or invalid systems from April through June, to influence purchasing decisions.

The time filter also allows very short windows of time – the previous 24 or 48 hours – to be able to identify and remediate suscription issues immediately.

2.4.4. Running Reports

1. Click the **Reports** item in the administration menu.
2. In the left column, click the name of the report filter to run.
3. Scroll to the bottom of the report page, and click the **Run Report** button.



Alternatively, the report results can be exported to a CSV file instead of being rendered in the Subscription Asset Manager UI. To export the data, click the **Export Report** button.

The data are exported to a CSV file and, optionally, a JSON file which contains the system details. These files are contained in a ZIP archive named **report - YEAR-MONTH-DAY-TIMESTAMPZ.zip**.



NOTE

Selecting the **Encrypt export** checkbox means that the exported CSV and JSON files are encrypted and can only be accessed by a private key used by Red Hat support.

2.4.5. Subscription Asset Manager Reports Results and Data

The Subscription Asset Manager report returns a chart of all registered systems for all selected organizations, similar to the **Dashboard**.

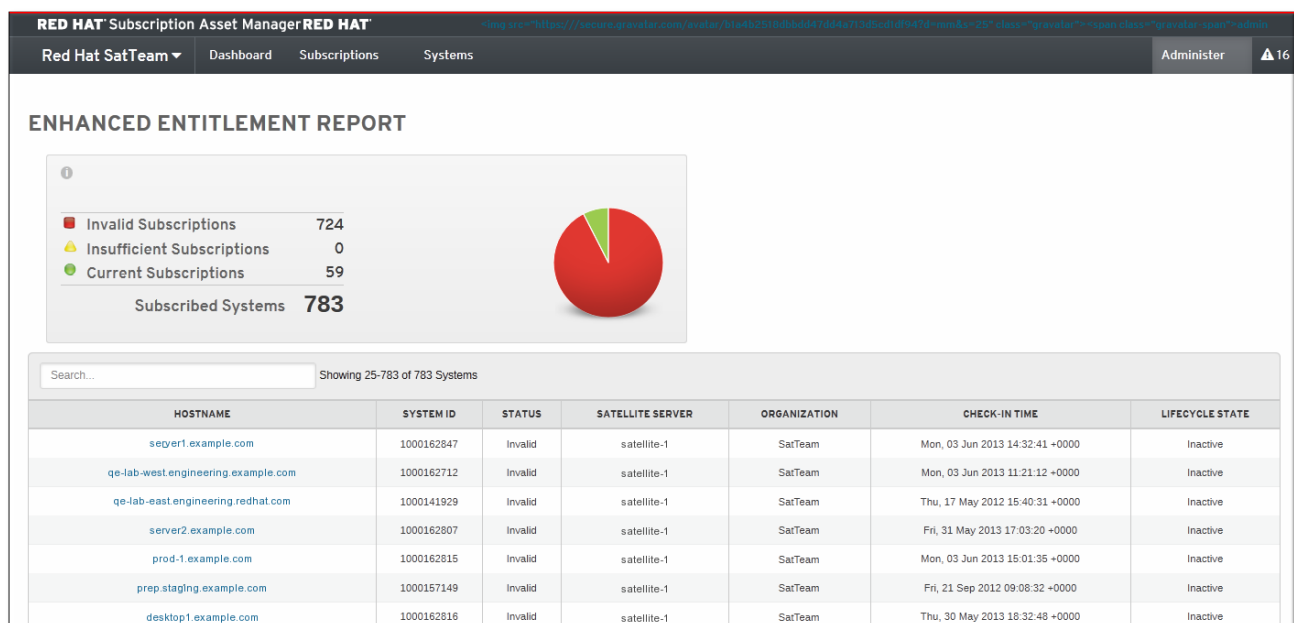


Figure 23. The Reports Results

There is also a list of all included systems. The list itself contains summary information such as the Subscription Asset Manager or Satellite server to which the system is registered, its status, its organization, and its most recent checkin time.

Clicking the name of any system pulls up more details subscription data for the system. The details page includes a history of subscription status changes for the given report period, the list of installed products and subscription information for each, and system facts (attributes about the physical machine and operating system, such as CPU, socket count, RAM, and cores).

The screenshot displays the Red Hat Subscription Management web interface. At the top, a navigation bar shows 'ACME_Corporation' as the selected organization, with tabs for 'Dashboard', 'Subscriptions', 'Systems', and 'Administer'. Below the navigation bar, a summary card shows subscription status: 0 Invalid Subscriptions, 0 Insufficient Subscriptions, and 1 Current Subscriptions, totaling 1 Subscribed System. A large green circle represents the system's status. Below this, a search bar and a table header are visible. The table has columns for 'HOSTNAME' and 'SYSTEM ID'. The first row shows 'server.example.com'. To the right of the table, a detailed view for the selected system is shown, including tabs for 'Subscription Details', 'Facts', 'Products', 'System Check-Ins', and a 'Close' button. The 'Subscription Details' tab is active, displaying three 'OpenShift Subscription' entries with their respective start and end dates and consumed counts.

HOSTNAME	SYSTEM ID
server.example.com	

Subscription Details

Compliant Product Info:

OpenShift Subscription
 Start Date: 2012-11-26T05:00:00.000+0000
 End Date: 2022-01-01T04:59:59.000+0000
 Consumed: 5 out of 10

OpenShift Subscription
 Start Date: 2012-11-26T05:00:00.000+0000
 End Date: 2022-01-01T04:59:59.000+0000
 Consumed: 5 out of 10

OpenShift Subscription
 Start Date: 2012-11-26T05:00:00.000+0000
 End Date: 2022-01-01T04:59:59.000+0000

Figure 24. The Reports Results: System Details

When the report results are exported, the same information is included in the export files.

The CSV report contains the summary information in the initial reports page: organization, registered subscription server, hostname, and subscription state, among others.

```
_id, record, CHECK-IN TIME, STATUS, DB ID, SATELLITE SERVER, HOSTNAME,
ORGANIZATION, LIFECYCLE STATE,
{"ident"=>"072c8bdd-ca00-43d4-a000-0887c75b90c8"},
522e0970af5d242094000002, 2013-09-09T14:23:27Z, "Current", "072c8bdd-ca00-
43d4-a000-0887c75b90c8", "sam-server.example.com", "server.example.com",
"ACME_Corporation", "Active",
```

The (optional) JSON file contains the same summary information, as well as the complete list of system facts and product information that is available on the details page.

```
[{"_id":{"_id":"522e0970af5d242094000002"},"_types":
["MarketingProductUsage"],"instance_identifier":"072c8bdd-ca00-43d4-a000-
0887c75b90c8","updated":"2013-09-09T17:46:24Z","splice_server":"sam13-
dlackey-demo","name":"server.example.com","facts":
{"memory_dot_memtotal":"3780964", ...
```

2.4.6. Enhanced Reporting Logs

Reporting Log Sizes

By default, enhanced reporting takes up to 200 MB of additional log space on a system. Logs grow at roughly 750 KB per system per month.

If this log size needs to be changed, it can be edited in the log configuration file in `/etc/splice/logging/basic.log`.

Sync Log

All of the errors, messages, and operations for the sync tool are recorded in a specific tool log at `/var/log/splice/spacewalk_splice_tool.log`

2.5. Subscription Asset Manager: Satellite 5.6 Usage Reports

Red Hat Satellite 5.6 has a utility (`spacewalk-reports`) to export information on the system inventory, organizations and associated subscriptions, errata, and users.

Subscription management establishes relationships between systems, organizations, and subscriptions. The types of relationships, and the ways those relationships are described, are slightly different in Satellite when compared to the new subscription services at Red Hat. (This is described more in the *Subscriptions Concepts and Workflows* document.) The new subscription services create a direct relationship between a subscription and the system to which it is applied. In Satellite 5.x, the concept of channels meant that a system was granted access to a content stream and the overall subscription allowed a certain number of systems to have access – but there was no direct association between a subscription and a system.

Satellite enhanced reporting allows Satellite 5.6 servers to sync their system, subscription, and channel data over to a Subscription Asset Manager server. That Subscription Asset Manager server can then take the underlying subscription information and generate a report using Red Hat Subscription Management rules, revealing the relationships between systems and subscriptions in Satellite.

This gives Satellite administrators a greater level of detail and control over the systems in their Satellite inventory.

2.5.1. About Satellite Consolidated Reports

2.5.1.1. The Advantages of Enhanced Reporting

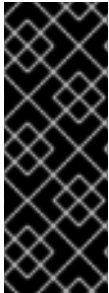
Subscriptions are frequently predicated on attributes of the underlying physical system, such as socket count, RAM, CPU, and cores. For virtual systems, subscriptions can be based on the host/guest relationship and inherited or restricted. Subscriptions can also be related to *other* products installed on

the system.

The reporting in Satellite is more limited; it measures the overall counts of systems and subscriptions, but it does not associate subscriptions and systems or identify required subscriptions based on system attributes.

The enhanced reporting provides more detailed reports in two ways:

- Determining actual subscription usage based on system attributes, host/guest relationships, and installed products.
- Tracking *historical* subscription usage based on subscription statuses at different points in time.



IMPORTANT

The enhanced reporting in Subscription Asset Manager only displays data about the Satellite 5.6 systems and organizations. It does not alter, update, or manage any systems, subscription assignments, or organizations in Satellite.

Both system management and subscription management for Satellite 5.6 must be performed in Satellite.

Like the Satellite reports, all of the Subscription Asset Manager report data can be exported to a CSV, so any additional data analysis can be performed. Additionally, a Subscription Asset Manager report can be exported to a JSON file and can be rendered visually in the Subscription Asset Manager UI (including system-level details) so it can be quickly read and interpreted.

2.5.1.2. Differences in Subscription Statuses from Satellite

Because enhanced reporting uses a different set of criteria – system attributes instead of channel access – to calculate the required subscriptions for a system, there are potentially differences between how enhanced reporting reports subscription status for a system and how the same information is reported in Satellite.

For example, many Red Hat Enterprise Linux subscriptions are for two sockets. A four-socket system, then, requires two subscriptions to cover all its socket pairs. If only one subscription is attached to that system, then the Subscription Asset Manager report shows that system as having an insufficient status. However, in Satellite, the same system only consumes a single channel entitlement, regardless of the number of sockets.

Additionally, Satellite has two different types of subscriptions: *system entitlements* (which are required to register a system) and *software channel entitlements* (which is what actually provides access to content, updates, and support). In the new subscription structure (and, thus, in Subscription Asset Manager), the system and channel entitlements are merged into a single product subscription which most closely correlates to channel entitlements.

Lastly, Satellite allows channels to be cloned. If a system is registered with a cloned channel, then no channel entitlements are used, so no entitlements are decremented from the available entitlements pool. However, when the channel information is synchronized, the cloned channel is associated with its original Red Hat channel, and subscriptions are then properly attached to the system (or its status is shown to be invalid or insufficient).

2.5.1.3. Syncing Data from Satellite 5.6 to Subscription Asset Manager

A script is run periodically to sync data from the Satellite 5.6 server to the Subscription Asset Manager database, so the data are accessible for Subscription Asset Manager reports. Only certain Satellite information is synced:

- System information (called system facts in Subscription Asset Manager) including the hostname, socket count, any host/guest relationships, and other relevant attributes
- Satellite organizations and associated subscriptions
- User information, including roles and administrator accounts such as Satellite Administrator and Organization Administrator
- Satellite cloned channels and their associated, originating channel.

Synchronization occurs in two phases. First, the inventory information is pulled out of Satellite as a **spacewalk-reports** report, using the **spacewalk-splice-checkin** process. The information then is sent to the Subscription Asset Manager server. This synchronization step is run every four hours, by default.

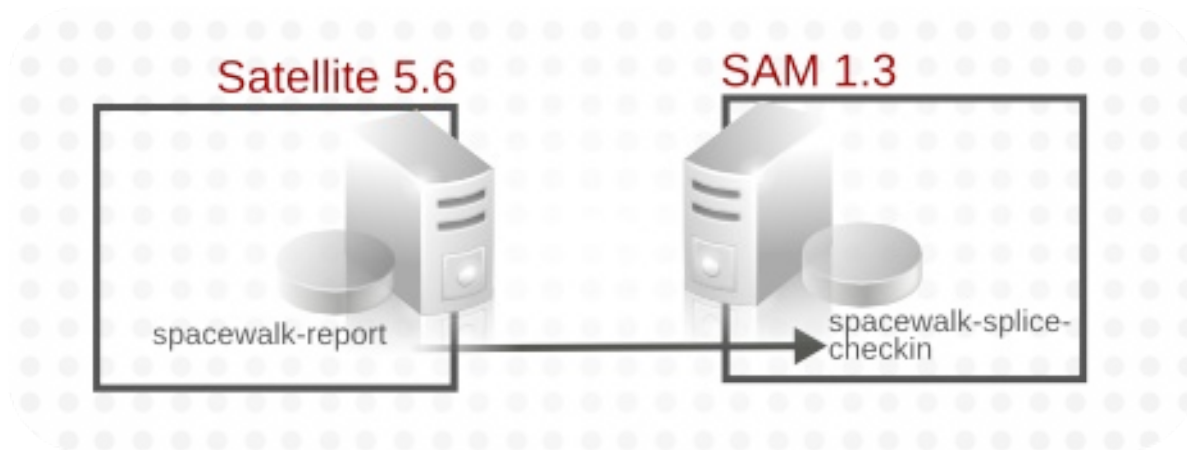


Figure 25. Satellite 5.6 to Subscription Asset Manager Sync

From there, the information is collected from Subscription Asset Manager and sent to a dedicated backend database (separate from the normal Subscription Asset Manager database) and stored in the separate reporting server. This step is performed every ten minutes.

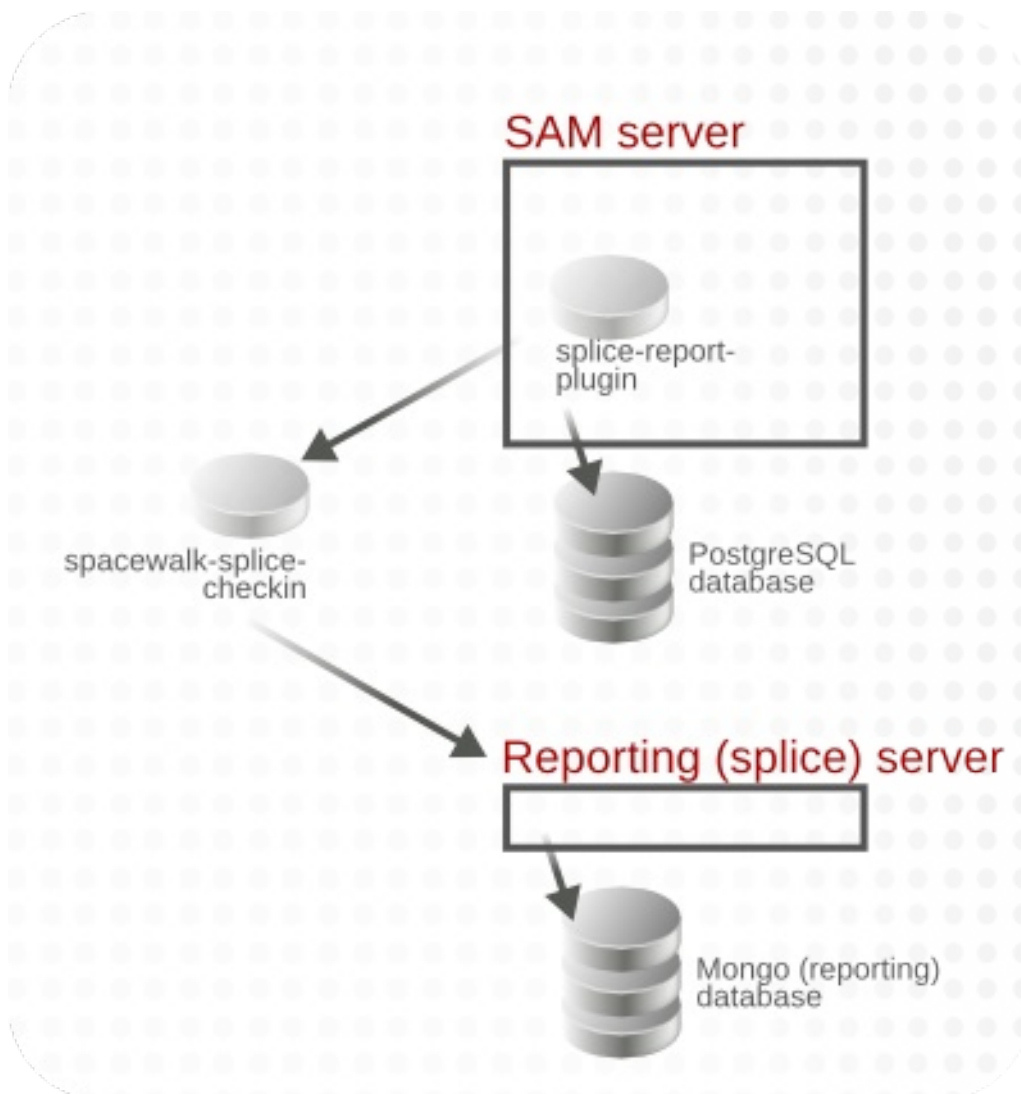


Figure 26. Subscription Asset Manager to Reporting Server Sync

After the second phase of synchronization, once the data are stored in the reporting database, then enhanced reports can be run, using the populated system data.

2.5.1.4. Users in Satellite 5.6 and Subscription Asset Manager

As mentioned in [Section 2.5.1.3, “Syncing Data from Satellite 5.6 to Subscription Asset Manager”](#), Satellite are synced over to Subscription Asset Manager and added as Subscription Asset Manager users. All of their organization and role assignments are preserved.

Satellite 5.6 passwords are not synced over to Subscription Asset Manager. Satellite users must log in with their Satellite username and a default password of *CHANGEME*.



NOTE

Satellite users are added to Subscription Asset Manager if they are added to the Satellite server, but they are not deleted in Subscription Asset Manager if they are deleted on the Satellite server. If a Satellite user is deleted, then that account must be manually deleted on the Subscription Asset Manager side.

2.5.2. Prerequisites

When using enhanced reporting, there are some additional system requirements:

- A dedicated Subscription Asset Manager instance specifically for Satellite reporting.



WARNING

A Subscription Asset Manager instance used for enhanced reporting can *only* be used as a reporting server for Satellite. It cannot be used as a regular Subscription Asset Manager instance to manage systems or data could be lost.

- The **crond** service must be running.
- An additional 4 GB of disk space must be available for the reporting database journal.
- Additional packages for the reporting server
 - `splice`
 - `ruby193-rubygem-splice_reports`
 - `spacewalk-splice-tool`

2.5.3. Configuring Reporting

1. Install Subscription Asset Manager, including the additional packages.

1. Register the host system. Use the **--autoattach** option to attach the required subscriptions for the operating system immediately.

```
[root@server ~]# subscription-manager register --autoattach
Username: jsmith@example.com
Password:
```

2. Wait several minutes for the updated content repositories to be added to the system configuration.

3. Enable the **[rhel-6-server-sam-rpms]** repository.

```
[root@server ~]# yum-config-manager --enable rhel-6-server-sam-rpms
Loaded plugins: product-id, refresh-packagekit
===== repo: rhel-6-server-sam-rpms
=====
[rhel-6-server-sam-rpms]
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/6Server
baseurl =
https://cdn.redhat.com/content/dist/rhel/server/6/6Server/x86_64/
subscription-asset-manager/1/os
cache = 0
cachedir = /var/cache/yum/x86_64/6Server/rhel-6-server-sam-rpms
```

```
cost = 1000
enabled = 1
enablegroups = True
exclude =
failovermethod = priority
...
```

4. Install the **katello-headpin-all** package using **yum install**:

```
[root@server ~]# yum install -y katello-headpin-all splice
ruby193-rubygem-splice_reports spacewalk-splice-tool
```

This can also be done when installing from an ISO image by using the **--enhanced_reporting** option.

```
[root@server cdrom]# ./install_packages --enhanced_reporting
```

2. The reporting database is a MongoDB database. Set up the Mongo service on the system to start automatically, and then start the service.

```
[root@sam-server ~]# chkconfig mongod on
[root@sam-server ~]# service mongod start
```

3. Run the configuration script to set up the Subscription Asset Manager server, the default admin user, and the initial organization.

```
[root@server ~]# katello-configure --deployment=sam --
org=Example_Org --user-name=samadmin --user-pass=secret
```

The Subscription Asset Manager admin user is not the same as the Satellite 5.6 admin user.

4. Still on the Subscription Asset Manager machine, create an SSH key to use to authenticate to the Satellite 5.6 machine.

```
[root@sam-server ~]# su - splice -s /bin/sh -c 'ssh-keygen -t rsa -f
/var/lib/splice/id_rsa-sat -N ""'
```

```
Generating public/private rsa key pair.
Your identification has been saved in /var/lib/splice/id_rsa-sat.
Your public key has been saved in /var/lib/splice/id_rsa-sat.pub.
The key fingerprint is:
78:fa:c9:68:71:a2:a7:c1:ec:35:e3:43:ce:27:b7:d8 splice@dhcp129-
162.rdu.redhat.com
```

The key's randomart image is:

```
++--[ RSA 1024]-----+
|
|
|
|      .
|      . S
|    o  +o.
|    +==+
|    ..+B0o.
|
```

```
|      o++=E.      |
+-----+
```

5. Switch to the Satellite 5.6 machine.
6. Create a new user which can run the required Satellite reports to sent to the Subscription Asset Manager server.

```
[root@sat-server ~]# useradd swreport
```

7. Add the key file that was created on the Subscription Asset Manager machine to the **authorized_keys** file for the **swreport** user on the Satellite 5.6 machine. Include the **command=** option to restrict the **swreport** user to only running Satellite reports on the system.

```
[root@sat-server ~]# vim /home/swreport/.ssh/authorized_keys

command="/usr/bin/spacewalk-report $SSH_ORIGINAL_COMMAND" \
ssh-rsa key_hash swreport@sat-server
```

The **command** directive should be all on one line in the keys file.

8. Set the proper permissions on the **.ssh** directory and the **authorized_keys** file:

```
[root@sat-server ~]# chown -R swreport:swreport /home/swreport/.ssh
[root@sat-server ~]# chmod 700 /home/swreport/.ssh
[root@sat-server ~]# chmod 600 /home/swreport/.ssh/authorized_keys
```

9. Add the **swreports** user to the **apache** system group so that it can connect to the database.

```
[root@sat-server ~]# gpasswd -a swreport apache
```

10. Switch back to the Subscription Asset Manager machine.
11. Switch to the reporting service user (**splice**), and test that the user can SSH into the Satellite machine using the **swreport** key.

```
[root@sam-server ~]# su - splice -s /bin/bash
[splice@sam-server ~]$ ssh -i /var/lib/splice/id_rsa-sat
swreport@sat-server.example.com splice-export
```

Accept the key fingerprint if prompted.

12. Edit the reporting configuration to recognize the Satellite 5.6 server.

```
[root@sam-server ~]# vim /etc/splice/checkin.conf

[spacewalk]
host=sat-server.example.com
ssh_key_path=/var/lib/splice/id_rsa-sat
login=swreport
```

13. Edit the reporting configuration to use the Subscription Asset Manager administrator

password that was set during the Subscription Asset Manager setup.

```
admin-pass=secret
```

14. On the Subscription Asset Manager server, run the sync utility to populate the Subscription Asset Manager database with the Satellite 5.6 data.

```
[root@sam-server ~]# su - splice -s /bin/bash
[splice@sam-server ~]$ spacewalk-splice-checkin
```



NOTE

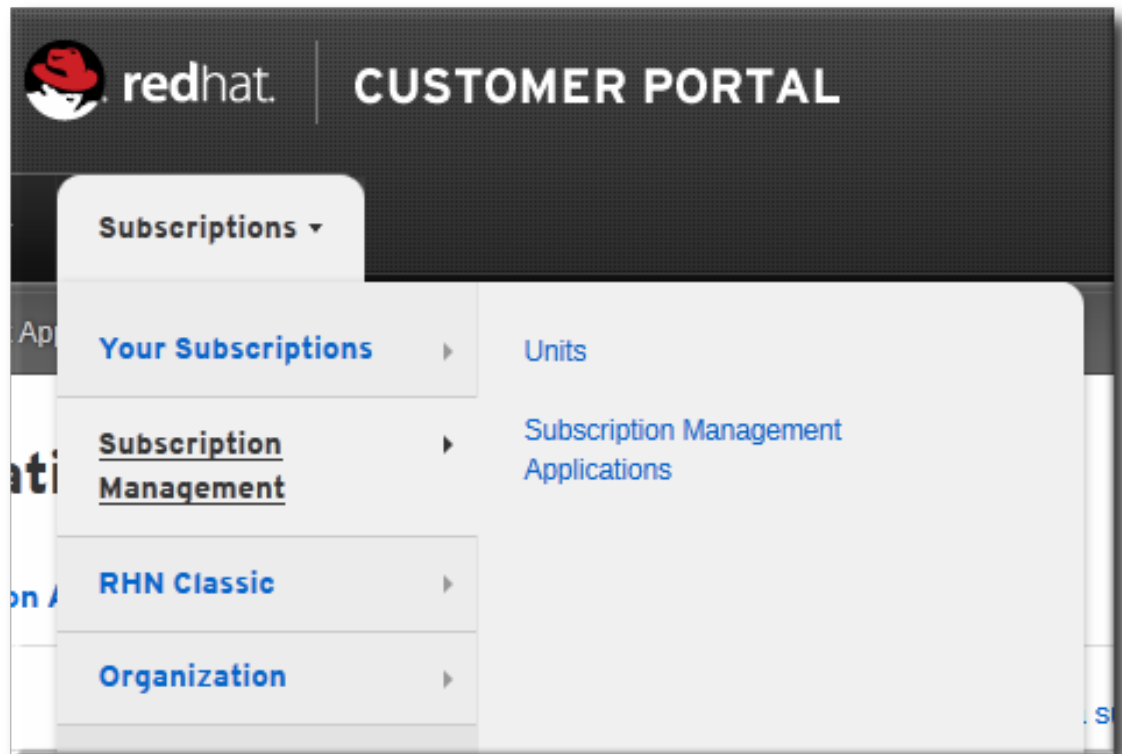
This can a long time to run on the initial sync operation.

To improve the tool performance, set the number of threads for the **spacewalk-splice-tool** process to use. This should be one thread for every two cores on a low-utilization system or one thread for every three cores on a high-utilization system.

For example:

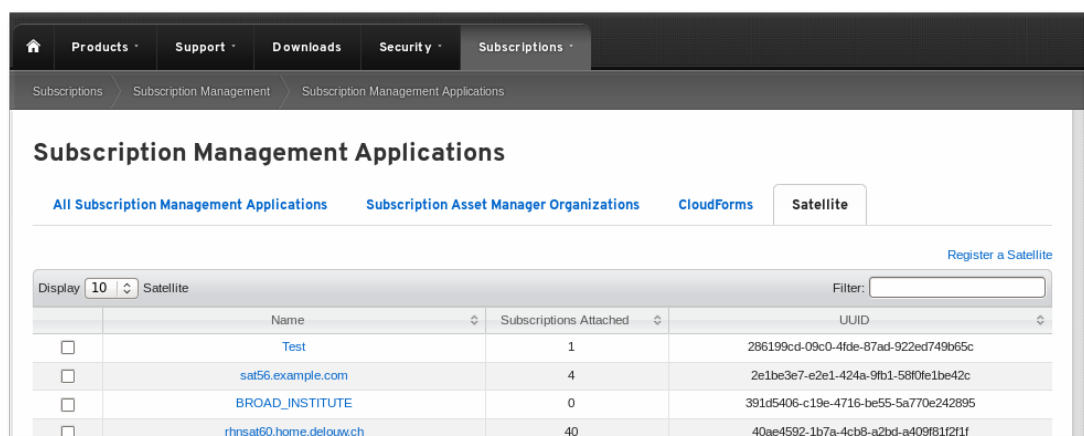
```
[root@sam-server]# /etc/splice/checkin.conf
num-threads=3
```

15. Get the Satellite 5.6 manifest from the Customer Portal.
 1. Log into the Customer Portal.
 2. Expand the **Subscriptions** tab, and select the **Subscription Management > Subscription Management Applications** item.



3. Open the **Satellite** tab.
4. If the Portal entry does not already exist, create the Satellite 5.6 entry and attach the required subscriptions.

1. In the **Satellite** tab, click the **Register a Satellite** link.



2. Fill in the required information for the Satellite 5.6 instance:

- The name for the Satellite server entry.
- The version of the Satellite instance; this should be 5.6.

The screenshot shows the 'Register a New Satellite' page. At the top, there is a navigation bar with links: Home, Products, Support, Downloads, Security, and Subscriptions. Below this, a breadcrumb trail shows 'Subscriptions' > 'Subscription Management' > 'Register a New Satellite'. The main heading is 'Register a New Satellite'. The form contains two input fields: 'Name:' with the value 'Sat56.example.com' and 'Satellite version:' with a dropdown menu set to 'Satellite 5.6'. At the bottom of the form are two buttons: 'Register' and 'or Cancel'.

3. Click the **Register** button.

4. In the Satellite 5.6 server's **Subscriptions** tab, select the subscriptions to add in the **Available Subscriptions** area.

The screenshot shows the 'sat56.example.com' page. At the top, there is a navigation bar with links: Subscriptions, Subscription Management, Subscription Management Applications, and sat56.example.com. Below this, the page title is 'sat56.example.com'. There are three buttons: 'Download Satellite Certificate', 'Download manifest', and 'Delete this Satellite'. A note below these buttons says 'Note: download both the manifest and satellite certificate'. On the left, there is a box containing the following information: UUID: 2e1be3e7-e2e1-424a-9fb1-58f0e1be42c, Created: 04/23/2014, Type: Satellite, and Version: Satellite 5.6 (with an 'Update' button). Below this box, there are two tabs: 'Attached Subscriptions' (selected) and 'Identity Certificate'. The 'Attached Subscriptions' section shows a table of subscriptions. The table has columns: Subscription Name, Contract Number, Quantity Attached, and End Date. There are four rows of subscriptions, each with a checkbox in the first column. Below the table, there is a 'Remove Selected' button. At the bottom right, there is a link 'Attach a subscription'.

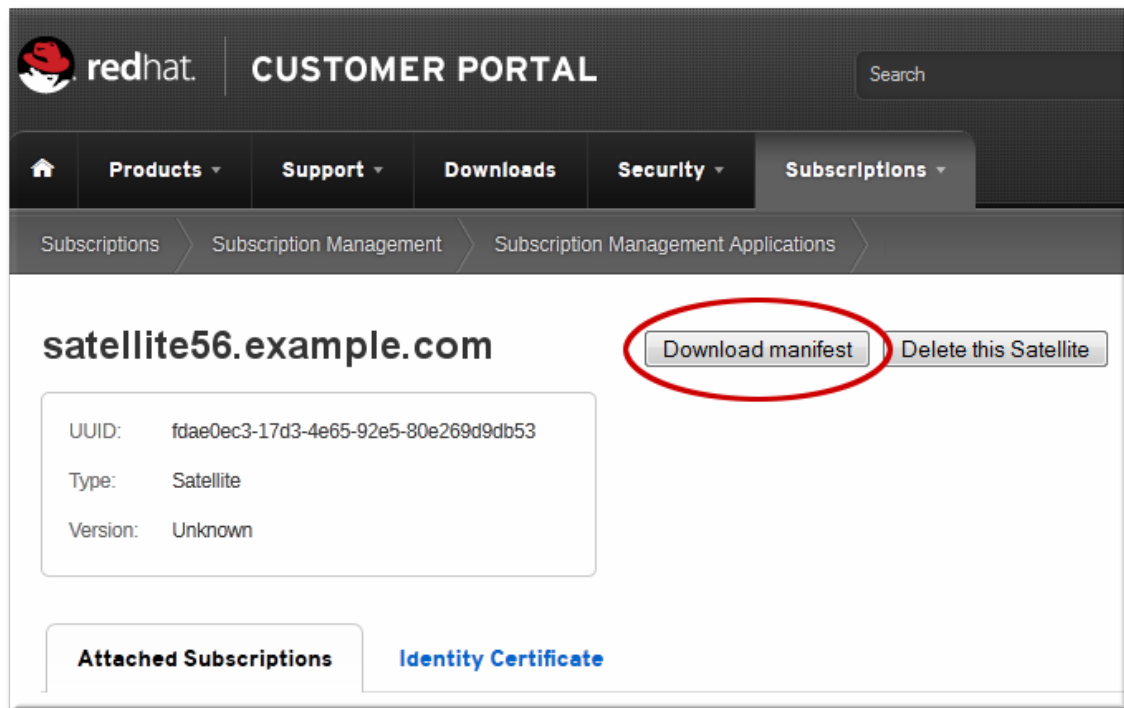
	Subscription Name	Contract Number	Quantity Attached	End Date
<input type="checkbox"/>	90 Day Supported Red Hat JBoss Data Grid Evaluation	10375937	1	05/03/2014
<input type="checkbox"/>	60 Day Self-Supported Red Hat Enterprise Linux OpenStack Platform Preview	10407707	1	05/16/2014
<input type="checkbox"/>	90-day Unsupported Red Hat JBoss BPM Suite Evaluation	10375941	1	05/03/2014
<input type="checkbox"/>	JBoss Business Partner Self-Supported NFR	10283972	1	09/25/2014

Be sure to set the appropriate quantity of subscriptions for each product selected. The quantity is the total number of subscriptions of that type available to the child organization.

5. Scroll down, and click the **Attach** at the bottom of the window.

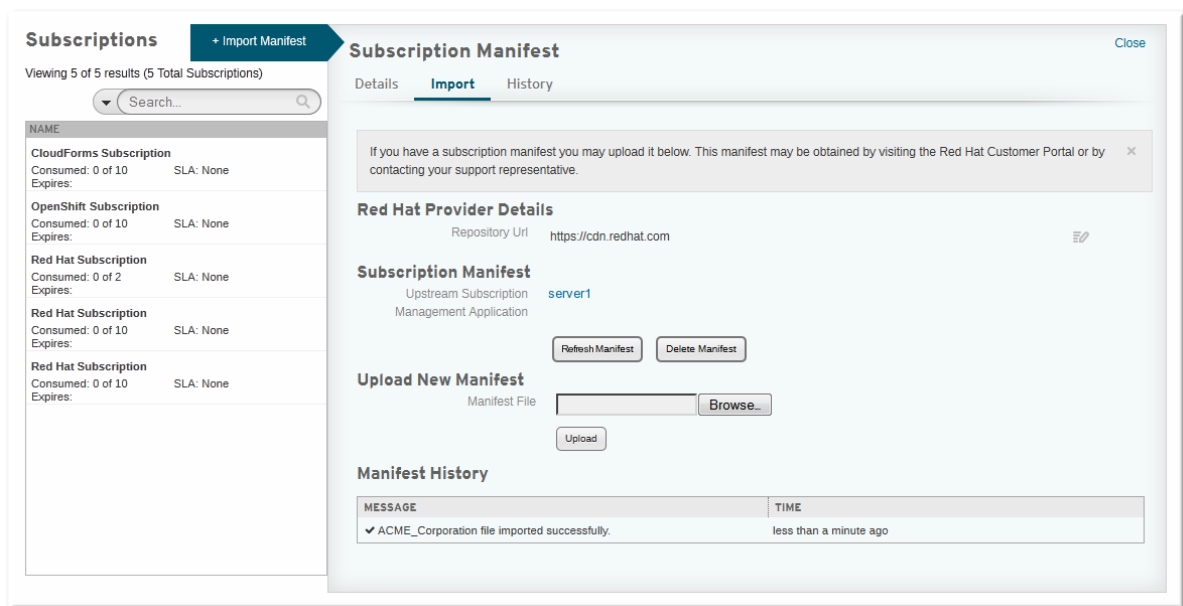
Attaching subscriptions automatically updates the child organization's manifest.

5. In the Satellite 5.6 server's entry page, click the **Download manifest** button, and save the archive file.



16. Log into the Subscription Asset Manager UI (<https://sam-hostname/sam>) as a Satellite administrator, and switch to the appropriate Satellite 5.6 organization.

17. Open the **Subscriptions** > **Subscriptions** tab, and click the **Import Manifest** link.



18. In the middle of the import tab, click browse to navigate to the saved manifest file.

19. Click the **Upload** button.

2.5.4. Running Reports and Getting Results

There is a default report configured which returns information for every organization and every system in every state that is being managed by Subscription Asset Manager.

It is possible to create additional report filters that return specific subsets of information or information for a given period of time. These custom reports are very useful for analyzing usage and compliance trends.

There is a lot more flexibility possible with the report form, however. In particular, there are three versatile settings:

- The organizations to check for the report
- The subscription statuses to include
- The date range to check; this looks for systems which had the status *within the given range* which may not necessarily be the current status for the system



NOTE

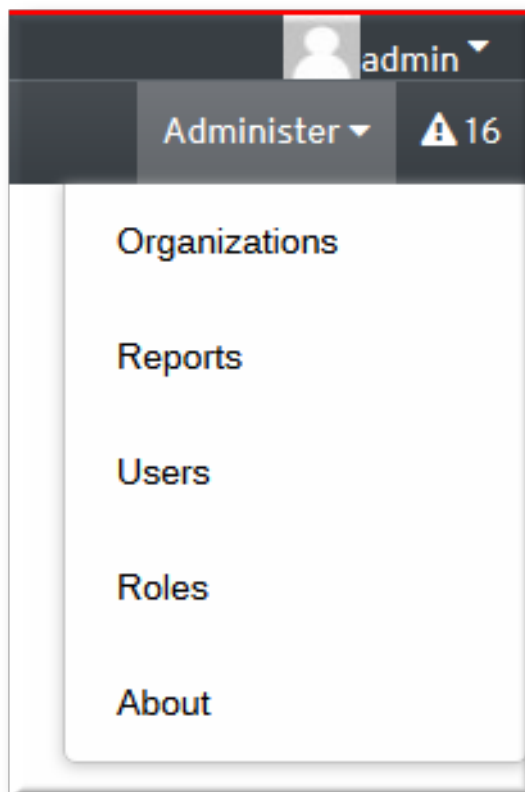
The data within the Subscription Asset Manager database includes historic subscription statuses. This allows reports to be generated to track subscriptions at a given point in time, not just the current date.

For example, if purchasing occurs in July, then a report can be configured to search for insufficient or invalid systems from April through June, to influence purchasing decisions.

The time filter also allows very short windows of time – the previous 24 or 48 hours – to be able to identify and remediate suscription issues immediately.

2.5.4.1. Creating Report Filters

1. Click the **Reports** item in the administration menu.



2. In the left column, click the **New Filter** link.
3. Fill in the required information for the report, including the organizations, statuses, date range, and active states.

Report Filters + New Filter

Viewing 2 of 2 results (2 Total Report Filters)

NAME

Red Hat Default Report

All

New Filter

Name: Example report

Description: for 2013

Subscription Status: Current Invalid

Satellite Server: sam13-dlackey-demo

Organizations: ACME_Corporation

LifeCycle State: Active Inactive

Search Previous Number of Hours: ☐ ☒

Search by Date Range: ☒

Start Date: 01/01/2013

End Date: 12/31/2013

Save Filter

4. Click the **Save Filter** button.

2.5.4.2. Running Reports

1. Click the **Reports** item in the administration menu.
2. In the left column, click the name of the report filter to run.
3. Scroll to the bottom of the report page, and click the **Run Report** button.

Export Criteria:

Encrypt export: ☐

Skip including expanded JSON data: ☐

Run Report Export Report

Alternatively, the report results can be exported to a CSV file. To export the data, click the **Export Report** button.

The data are exported to a CSV file and, optionally, a JSON file which contains the system details. These files are contained in a ZIP archive named **report - YEAR-MONTH-DAY-TIMESTAMPZ.zip**.



NOTE

Selecting the **Encrypt export** checkbox means that the exported CSV and JSON files are encrypted and can only be accessed by a private key used by Red Hat support.

2.5.4.3. Subscription Asset Manager Reports Results and Data

The Subscription Asset Manager report returns a chart of all registered systems for all selected organizations, similar to the **Dashboard**.

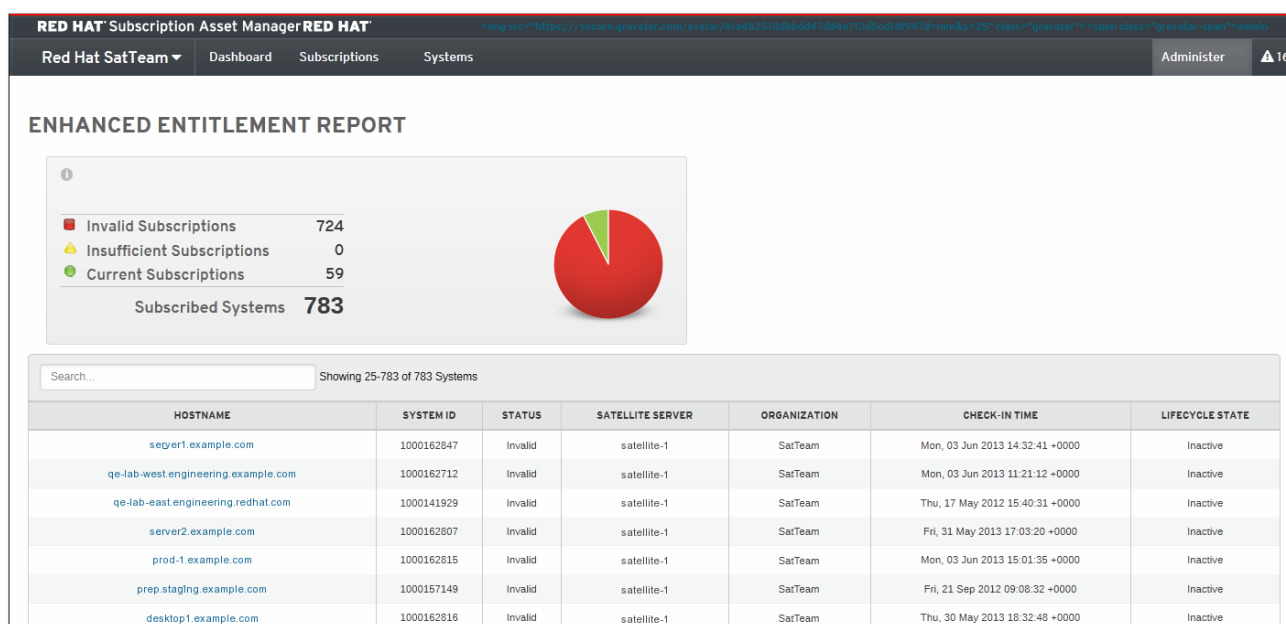


Figure 27. The Reports Results

There is also a list of all included systems. The list itself contains summary information such as the Subscription Asset Manager or Satellite server to which the system is registered, its status, its organization, and its most recent checkin time.

Clicking the name of any system pulls up more details subscription data for the system. The details page includes a history of subscription status changes for the given report period, the list of installed products and subscription information for each, and system facts (attributes about the physical machine and operating system, such as CPU, socket count, RAM, and cores).

The screenshot shows the ACME_Corporation dashboard with the following elements:

- Navigation Bar:** ACME_Corporation, Dashboard, Subscriptions, Systems, and Administer.
- Subscription Summary:**
 - Invalid Subscriptions: 0
 - Insufficient Subscriptions: 0
 - Current Subscriptions: 1
 - Subscribed Systems: 1**
- System Details View:**
 - Search bar and "Showing 1-1 of 1 Systems" indicator.
 - Table with columns: HOSTNAME, SYSTEM ID.
 - System: server.example.com
 - Subscription Details tab selected, showing:
 - Compliant Product Info:**
 - OpenShift Subscription** (repeated three times):
 - Start Date: 2012-11-26T05:00:00.000+0000
 - End Date: 2022-01-01T04:59:59.000+0000
 - Consumed: 5 out of 10

Figure 28. The Reports Results: System Details



NOTE

Only the last 250 check-ins are shown on the system's details page when viewing the enhanced report in the Subscription Asset Manager UI.

When the report results are exported, the same information is included in the export files.

The CSV report contains the summary information in the initial reports page: organization, registered subscription server, hostname, and subscription state, among others.

```
_id, record, CHECK-IN TIME, STATUS, DB ID, SATELLITE SERVER, HOSTNAME,
ORGANIZATION, LIFECYCLE STATE,
{"ident"=>"072c8bdd-ca00-43d4-a000-0887c75b90c8"},
522e0970af5d242094000002, 2013-09-09T14:23:27Z, "Current", "072c8bdd-ca00-
43d4-a000-0887c75b90c8", "sam-server.example.com", "server.example.com",
"ACME_Corporation", "Active",
```

The (optional) JSON file contains the same summary information, as well as the complete list of system facts and product information that is available on the details page.

```
[{"_id":{"_id":"522e0970af5d242094000002"},"_types":
["MarketingProductUsage"],"instance_identifier":"072c8bdd-ca00-43d4-a000-
0887c75b90c8","updated":"2013-09-09T17:46:24Z","splice_server":"sam13-
dlackey-demo","name":"server.example.com","facts":
{"memory_dot_memtotal":"3780964", ...
```

2.5.5. Troubleshooting Enhanced Reports

2.5.5.1. Enhanced Reporting Logs

Reporting Log Sizes

By default, enhanced reporting takes up to 200 MB of additional log space on a system. Logs grow at roughly 750 KB per system per month.

If the logging level needs to be changed, it can be edited in the log configuration file in `/etc/splice/logging/basic.cfg`.

Sync Log

All of the errors, messages, and operations for the sync tool are recorded in a specific tool log at `/var/log/splice/spacewalk_splice_tool.log`

2.5.5.2. Common Problems

Q: Why are no systems displayed in the report?

A: First, make sure that there are systems which match the given report filters.

If the filters should return some systems and there are still no systems displayed, this means that the information is not being pulled into the reporting database. There are several potential points of failure:

The information isn't being pulled from the Satellite server.

The information is not being properly transmitted from Subscription Asset Manager into the reporting database.

The information is not being properly stored in the database.

The information stored in Subscription Asset Manager is outdated.

First, make sure that the sync script is running by checking the history in the sync tool log, `/var/log/splice/spacewalk_splice_tool.log`.

Then, make sure that the Mongo service is running and listening on port 27017. If the Mongo service is not running, then the Subscription Asset Manager services cannot start.

```
[root@sam-server ~]# service mongod status
[root@sam-server ~]# telnet localhost 27017
```

If the service is running, check the Mongo database to look for sync entries. For example:

■

```
[root@sam-server ~]# mongo checkin_service --eval
"printjson(db.marketing_product_usage.count())"
```

If neither of those reveal a problem, or if they do not have relevant entries, then run the reporting debug script:

```
[root@sam-server ~]# /usr/bin/splice-debug
```

This collects all relevant configuration and log files for the reporting server and exports the data to a file in the /tmp directory name `splice-debug-YYYY-MM-DD-TIME`. For example, `/tmp/splice-debug-2013-06-14-T15-22-19`.

That directory can be zipped and sent to support if necessary.

Q: Why are all systems marked as invalid?

A: Check that a manifest has been imported into Subscription Asset Manager for the Satellite server. The manifest tells Subscription Asset Manager what subscriptions the Satellite server has attached to it; without the manifest, reporting assumes that no subscriptions are available.

Q: I updated subscriptions for a system or my Satellite server in Subscription Asset Manager, but those changes are not being reflected in the report.

A: The sync script runs every four hours, so it may not have synchronized the changes yet. Either wait for the next scheduled run or run the script by hand (which may take several minutes to finish):

```
[root@sam-server ~]# su - splice -s /bin/bash
[splice@sam-server ~]$ spacewalk-splice-checkin
```

Q: The link to the Satellite 5.6 UI in the report results is returning an HTTP 404 error.

A: Verify that the `rhn-search` process is running on the Satellite 5.6 machine.

2.5.5.3. Other Known Issues

These are some other issues that are recognized with the enhanced reporting and Satellite, but do not have a workaround.

Having Organizations Not Related to Satellite Reporting

If a Subscription Asset Manager instance used in enhanced reporting has non-Satellite organizations added to it, those organizations may be overwritten and removed in the Subscription Asset Manager database as part of the sync process.

**WARNING**

A Subscription Asset Manager instance used for enhanced reporting can *only* be used as a reporting server for Satellite. It cannot be used as a regular Subscription Asset Manager instance to manage systems or data could be lost.

2.6. Portal: Subscriptions Used in Both RHN Classic and the Portal

Administrators need to have a sense of all of the subscriptions attached for their account, altogether, regardless of whether the system is managed in RHN Classic or Customer Portal Subscription Management. The Customer Portal provides a way of looking at the total attached subscriptions.

In the **Subscriptions Overview** page, the **Subscription Utilization** area at the top gives the current count for every active subscription for the entire account, and a total count of every used subscription, regardless of whether it is used in RHN Classic or Customer Portal Subscription Management. These numbers are updated whenever the subscription count changes in the subscription server.

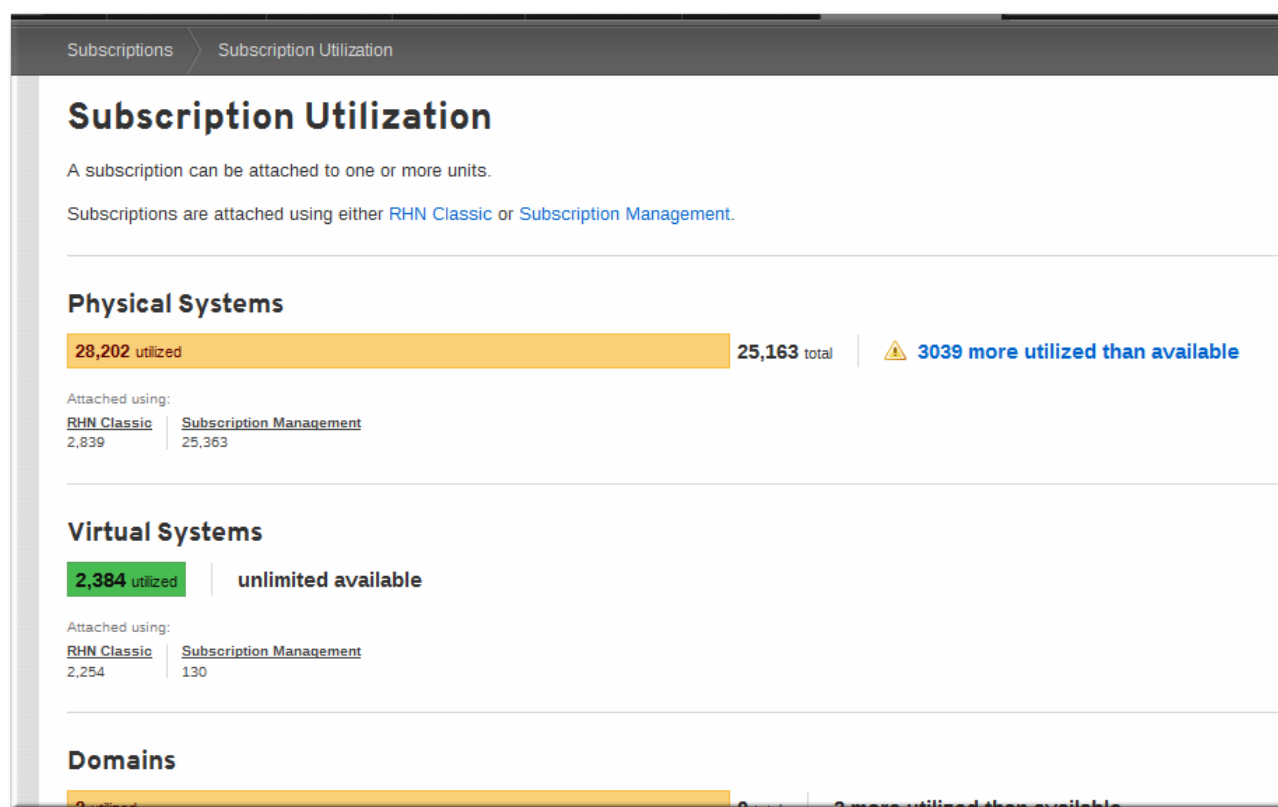


Figure 29. Total Counts of Subscriptions for All Subscription Services

**NOTE**

RHN Classic is intended to be used with legacy systems (Red Hat Enterprise Linux 6.0 or Red Hat Enterprise Linux 5.6 and earlier releases). It is strongly recommended that Red Hat Enterprise Linux 6.1/5.7 and later systems use Customer Portal Subscription Management, Subscription Asset Manager, or other certificate-based subscription management service.

3. RESPONDING TO NOTIFICATIONS AND CHANGED STATUS

3.1. Red Hat Subscription Manager: Autoattaching and Updating Subscriptions

The subscription daemon, `rhsmcertd`, monitors the subscriptions that are attached to a system and tracks when they near their expiration dates or if they are removed.

When a system has installed products without valid subscriptions, Red Hat Subscription Manager automatically attaches the best-matched subscriptions to cover the installed products on the system. This is the same process, conceptually, as autoattaching the system when registering a system, but this is all done without the administrator having to run a script. This process keeps the subscriptions updated even in a dynamic environment.

There are a couple of common scenarios where autoattaching is useful:

- As new products are installed.
- When subscriptions expire.

Within 24 hours of when the subscription expires, the Subscription Manager automatically re-attaches subscriptions to the system.

- When subscriptions are renewed.
- When a subscription management application replaces its manifest.

If a new manifest is uploaded (instead of refreshing the original manifest), the old manifest is deleted. Any subscriptions attached to a system from the old manifest are automatically removed – which means that the system could have products that no longer have a subscription for them. Enabling autoattaching on a system means that the system can automatically request and apply subscriptions from the new manifest.

System autoattaching prevents a system from having products without a subscription as long as any active, compatible subscription is available for it.

Auto-attaching is enabled by default on systems to ensure that they maintain their subscription status. Auto-attaching can be disabled and re-enabled through the given subscription service (such as the Customer Portal or Subscription Asset Manager) and the local autoattaching check for `rhsmcertd` can be changed.

3.1.1. Setting Preferences for Systems

Autoattaching and updating subscriptions selects what subscriptions to attach to a system based on a variety of criteria, including current installed products, hardware, and architecture. It is possible to set two additional preferences for Subscription Manager to use:

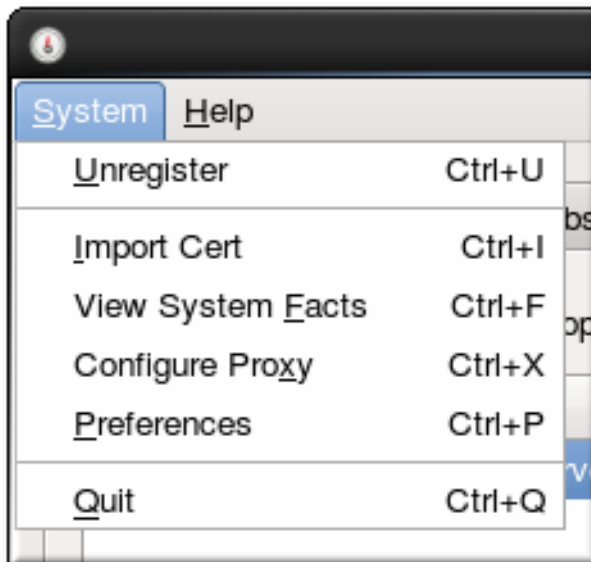
- Service levels for subscriptions
- The operating system minor version (X.Y) to use

This is especially useful for healing, which runs daily to ensure that all installed products and current subscriptions remain active.

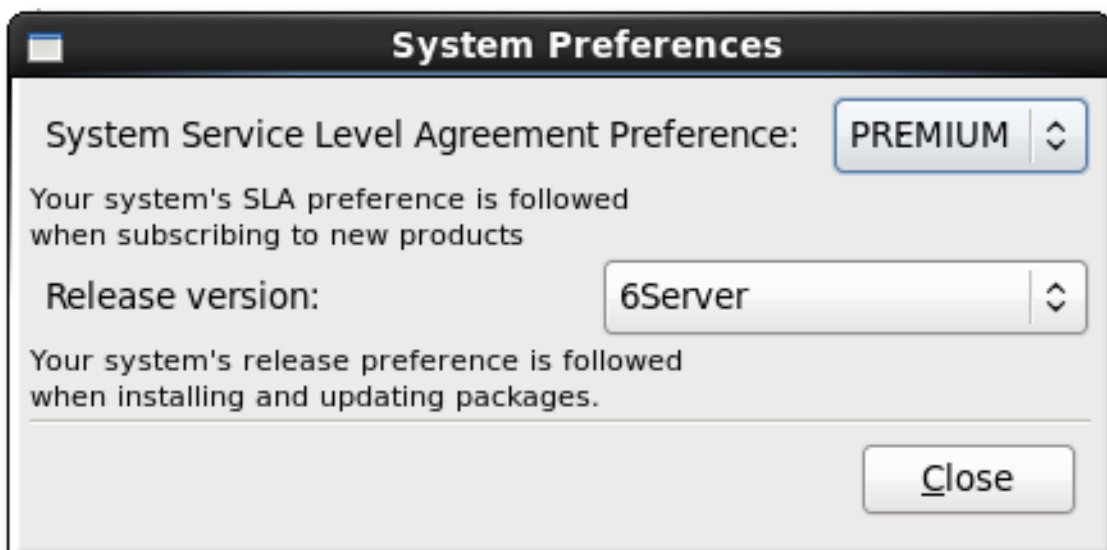
3.1.1.1. Setting Preferences in the UI

Both a service level preference and an operating system release version preference are set in the **System Preferences** dialog box in Subscription Manager.

1. Open the Subscription Manager.
2. Open the **System** menu.
3. Select the **System Preferences** menu item.



4. Select the desired service level agreement preference from the drop-down menu. Only service levels available to the Red Hat account, based on all of its active subscriptions, are listed.
5. Select the operating system release preference in the **Release version** drop-down menu. The only versions listed are Red Hat Enterprise Linux versions for which the account has an active subscription.



6. The preferences are saved and applied to future subscription operations when they are set. To close the dialog, click **Close**.

3.1.1.2. About Service-Level Preferences

Part of a subscription is recognizing the service level for a product on a given system. Setting a preferred service level for a system in Red Hat Subscription Manager means that Subscription Manager uses that preference as part of the criteria for automatically attaching subscriptions to the system the system, either when initially applying subscriptions or when healing subscriptions.

Red Hat service levels are defined in the contract; a summary of production support levels is available at <https://access.redhat.com/support/offerings/production/sla.html>.

The support, or service, information for each subscription is part of the subscription details. Two attributes are displayed, per subscription: its service level and its service type. The *level* defines how quickly support is guaranteed to respond to a case. The *type* defines the methods of communication; for production level support, this is always web and phone.

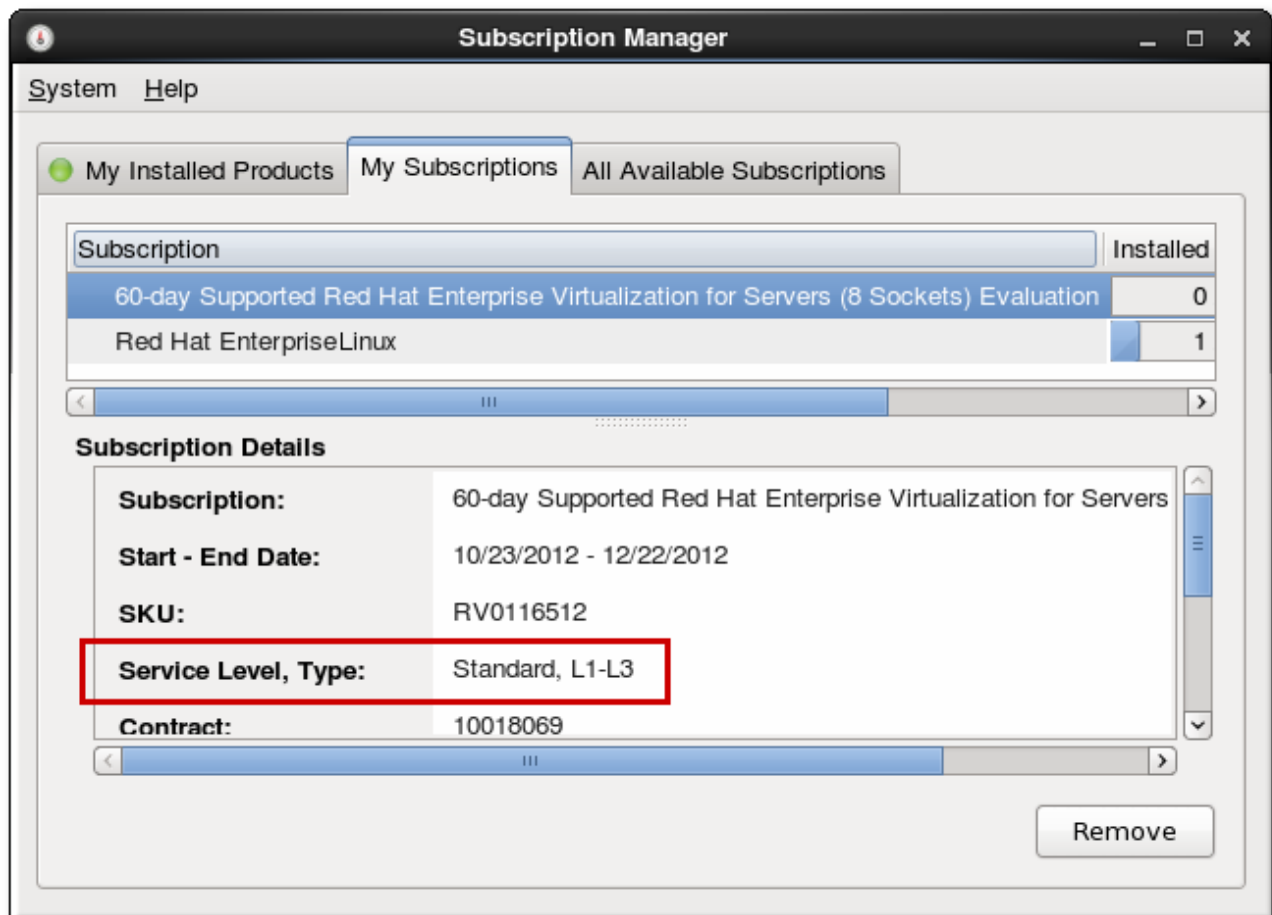


Figure 30. Service Level Information for a Subscription

An account can have multiple levels of support available, even for the same product. The support level for a given system can be configured so that the appropriate level of support is available. A production system usually has a premium support level since it is a business critical system, while a development system may have standard support or be self-supported.



NOTE

By default, the highest available level of support is selected for the subscription and system.

3.1.1.3. Setting Service Levels Through the Command Line

Autoattaching a system sets the best match to the system based on its hardware and architecture,

operating system, and installed products. There is an option, when autoattaching a system, to include service level in the process for selecting subscriptions. For example, a production system may prefer to have premium subscriptions over standard.

Setting a service level during registration or subscription does two things:

- It selects subscriptions on a best match with the desired service level (and other criteria).
- It sets the system preference to the specified service level.



NOTE

When a preference is set during registration, during subscription, or in the UI, this preference is applied both to current subscriptions and to renewals and healed subscriptions.

A general service level preference can be set using the **service-level --set** command.

Example 1. Setting a Service Level Preference

First, list the available service levels for the system, using the **--list** option with the **service-level** command.

```
[root@server ~]# subscription-manager service-level --list
+-----+
          Available Service Levels
+-----+
Standard
None
Premium
Self-Support
```

Then, set the desired level for the system.

```
[root@server ~]# subscription-manager service-level --set=self-support
Service level set to: self-support
```

The current setting for the local system is shown with the **--show** option:

```
[root@server ~]# subscription-manager service-level --show
Current service level: self-support
```

A service level preference can be defined when a subscription operation is being run (such as registering a system or attaching subscriptions after registration). This can be used to override a system preference. Both the **register** and **attach** commands have the **--servicelevel** option to set a preference for that action.

Example 2. Autoattaching Subscriptions with a Premium Service Level

```
[root@server ~]# subscription-manager attach --auto --servicelevel
Premium
Service level set to: Premium
```

```

Installed Product Current Status:
ProductName:           RHEL 6 for Workstations
Status:               Subscribed

```



NOTE

The `--servicelevel` option requires the `--auto-attach` option (for register) or `--auto` option (for attach). It cannot be used when attaching a specified pool or when importing a subscription.

3.1.1.4. Viewing Service Level Settings in the Command Line

A system can have multiple service levels available to it. An available level may not be the one that is actually in effect for the system or for a product on the system; it depends on the subscriptions themselves.

Service level information is viewed using the `service-level` command. This is an information command; it returns the current information about service levels for the system or the account, but it does not make any changes to the assigned service level.

To view available service levels for the system, use the `--list` option. Simply knowing what levels are available is helpful for setting preferences, autoattaching, or even purchasing subscriptions.

```

[root@server ~]# subscription-manager service-level --list
+-----+
          Available Service Levels
+-----+
Standard
None
Premium

```

Preferences for default service levels can be specified for an account and for a local system, and those two settings do not have to be the same.

The current setting for the local system is shown with the `--show` option:

```

[root@server ~]# subscription-manager service-level --show
Current service level: Premium

```

3.1.1.5. Setting a Preferred Operating System Release Version in the Command Line

Many IT environments have to be certified to meet a certain level of security or other criteria. In that case, major upgrades must be carefully planned and controlled – so administrators cannot simply run `yum update` and move from version to version.

Setting a release version preference limits the system access to content repositories associated with that operating system version instead of automatically using the newest or latest version repositories.

For example, if the preferred operating system version is 6.3, then 6.3 content repositories will be preferred for all installed products and attached subscriptions for the system, even as other repositories become available.

Only packages, updates, and errata for that specific version will be used for the system.

Example 3. Setting an Operating System Release During Registration

A preference for a release version can be set when the system is registered by using `--release` option with the `register`. This applies the release preference to any subscriptions selected and auto-attached to the system at registration time.

Setting a preference requires the `--auto-attach` option, because it is one of the criteria used to select subscriptions to auto-attach.

```
[root#server ~]# subscription-manager register --auto-attach --
release=6.4 --username=admin@example.com...
```



NOTE

Unlike setting a service level preference, a release preference can only be used during registration or set as a preference. It cannot be specified with the `attach` command.

Example 4. Setting an Operating System Release Preference

The `release` command can display the available operating system releases, based on the available, purchased (not only attached) subscriptions for the account.

```
[root#server ~]# subscription-manager release --list
+-----+
          Available Releases
+-----+
6.2
6.3
```

The `--set` then sets the preference to one of the available release versions:

```
[root#server ~]# subscription-manager release --set=6.3
Release version set to: 6.3
```

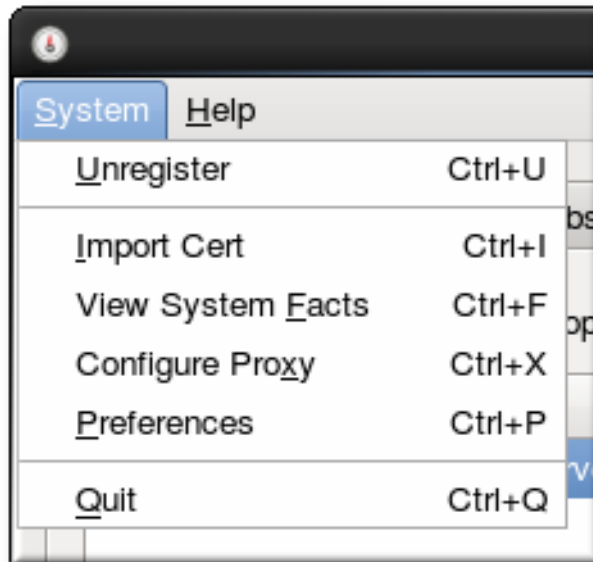
3.1.1.6. Removing a Preference

To remove a preference through the command line, use the `--unset` with the appropriate command. For example, to unset a release version preference:

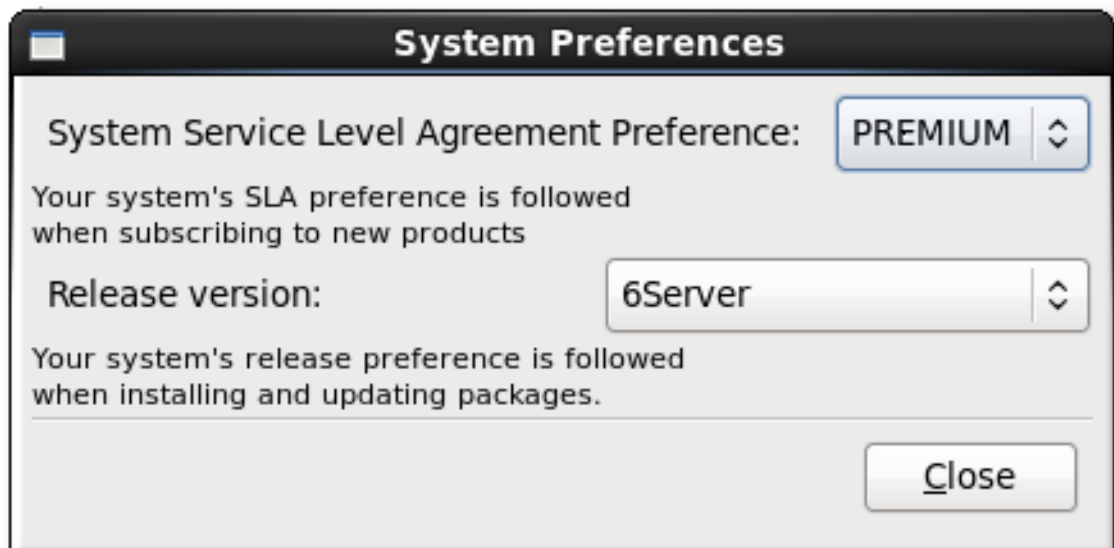
```
[root#server ~]# subscription-manager release --unset
Release version set to:
```

To remove or unset a preference in the UI:

1. Open the Subscription Manager.
2. Open the **System** menu.
3. Select the **System Preferences** menu item.



4. Set the service level or release version value to the blank line in the corresponding drop-down menu.



5. Click **C**lose.

3.1.2. Autoattaching in Response to Notifications

When the Subscription Manager UI opens, whether it was opened through a notification or just opened normally, there is an icon in the upper left corner that shows whether products lack a valid certificate. The easiest way to attach subscriptions which match invalidated products is to click the **Auto-attach** button.

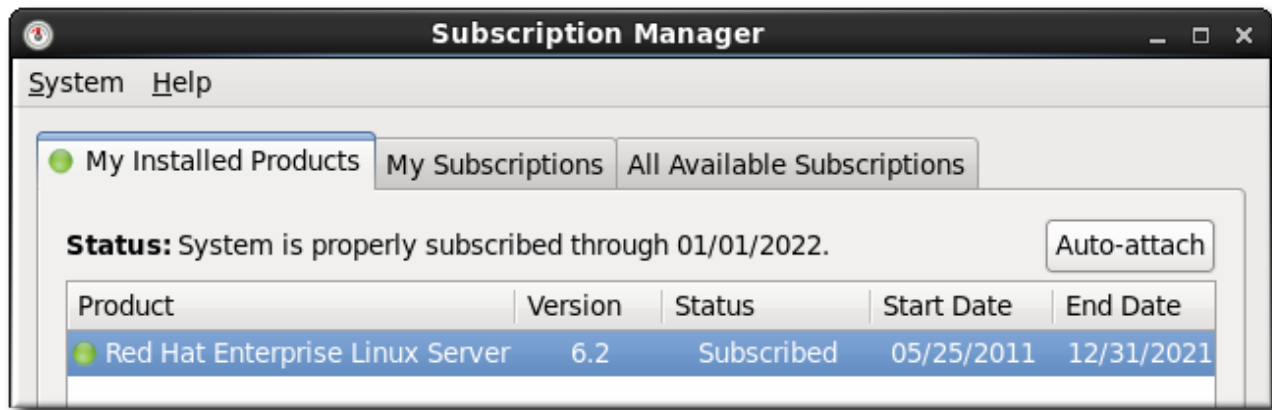


Figure 31. Auto-attach Button

3.1.3. Autoattaching at Registration

The `register` command has an option, `--auto-attach`, which allows the system to be registered to the subscription service and immediately attaches the subscriptions which best match the system's architecture, in a single step.

```
[root@server1 ~]# subscription-manager register --username admin-example --password secret --auto-attach
```

3.1.4. Autoattaching after Registration

The `--auto` option with the `attach` (which is analogous to the `--auto-attach` option with the `register` command) attaches the best-fitting subscriptions to the system based on its currently installed products and other criteria.

```
[root@server1 ~]# subscription-manager attach --auto
```

Running this after registration may be useful if the system has add-on products like Cluster Suite or layered products like Red Hat Directory Server installed. Since these products are not installed by default, autoattaching at registration time may not apply the appropriate subscriptions to the machine. Autoattaching after they are installed can make it easier to attach the proper subscriptions, and it is a little easier since it is not necessary to find and select a pool ID.

3.2. Portal: Autoattaching and Updating Subscriptions

3.2.1. Setting a Preferred Service Level

Part of a subscription is a defined service level for that product on a given system. Red Hat service levels are defined in the contract; a summary of production support levels is available at <https://access.redhat.com/support/offerings/production/sla.html>.

There are three basic support levels:

- Premium
- Standard
- None (self-supported)

An account can have multiple levels of support available, even for the same product, and, obviously, not every system within an IT environment demands the same response times and support as other systems. For example, a production system usually has a premium support level since it is a business critical system, while a development system may have standard support or be self-supported.



NOTE

By default, the highest available level of support is selected for the subscription and system.

When a system is configured, it can be assigned a preferred service level. When subscriptions are autoattached to the system and the preferred service level is available, then the subscription matching that preference is used. (Service-level preferences are not evaluated or enforced for manually selecting and attaching subscriptions.)



NOTE

Service-level preferences must first be set locally on the client when it is registered, by autoattaching, or when editing the configuration later. For example:

```
[root#server ~]# subscription-manager attach --auto --
servicelevel Premium
```

After the service-level preference is set for the system, then that preference can be viewed and edited through the Portal.

The service-level preference is set in the system details page.

The screenshot shows the 'server.example.com' system details page. The breadcrumb navigation at the top includes 'Subscriptions', 'Subscription Management', 'Units', and 'server.example.com'. The main title is 'server.example.com'. Below this, a box contains system details: 'Type: Virtual System', 'UUID: 4f8ba7b3-f833-4fea-9894-af28c4a00b55', and 'Status: System is properly subscribed through 08/12/2013' with a green dot icon. Below the status, it says 'Last checked in: 12:00 AM 09/06/12'. A red rectangle highlights the 'Service Level Preference' section, which shows a dropdown menu currently set to 'None'. Below this, there is an 'Auto-attach' section with 'Enabled' and 'Disable' buttons. At the bottom, 'Release Version:' is listed.

Figure 32. Service-Level Preference

3.2.2. Viewing the Operating System Release Preference

Many IT environments have to be certified to meet a certain level of security or other criteria. In that case, major upgrades must be carefully planned and controlled – so administrators cannot simply run `yum update` and move from version to version.

Setting a release version preference limits the system access to content repositories associated with that operating system version instead of automatically using the newest or latest version repositories.

For example, if the preferred operating system version is 6.3, then 6.3 content repositories will be preferred for all installed products and attached subscriptions for the system, even as other repositories become available.

Only packages, updates, and errata for that specific version will be used for the system.

A release version preference can only be set using the local Red Hat Subscription Manager tools. However, if a release preference is set for the local system, that preference is viewable for that system in the portal.

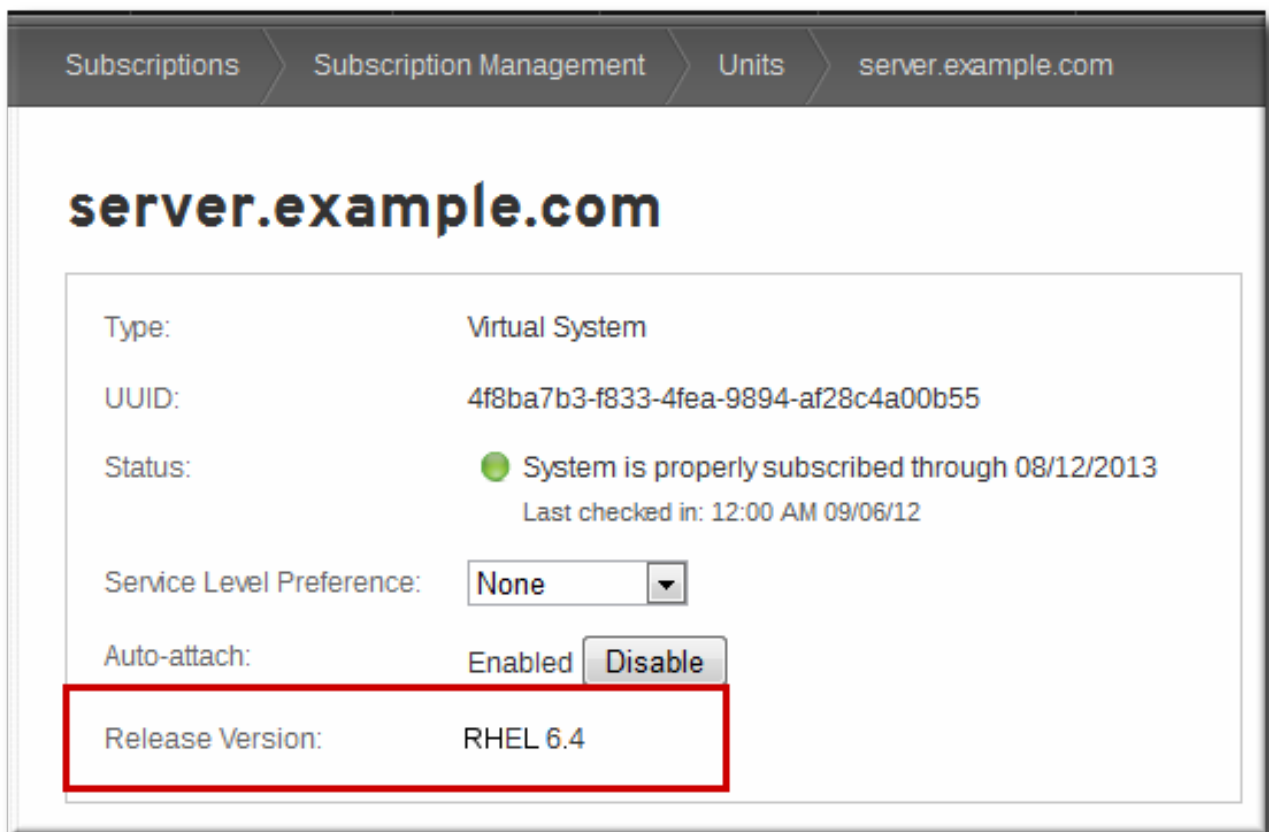


Figure 33. Operating System Release Version Preference Setting

3.2.3. Autoattaching Subscriptions

The subscription service can monitor the subscriptions that are attached to a system and track when they near their expiration dates. Within 24 hours of when the subscription expires, the Subscription Manager automatically re-attaches the system to a matching new subscription so that the subscription status remains green.

Autoattaching prevents a system from having uncovered products as long as any active, compatible subscription is available for it.

3.2.3.1. Enabling Autoattach on a System

Autoattaching is enabled by default on systems to ensure that they maintain their subscription status. Autoattaching can be disabled and re-enabled by toggling the **Disable/Enable** buttons on the system's details page.

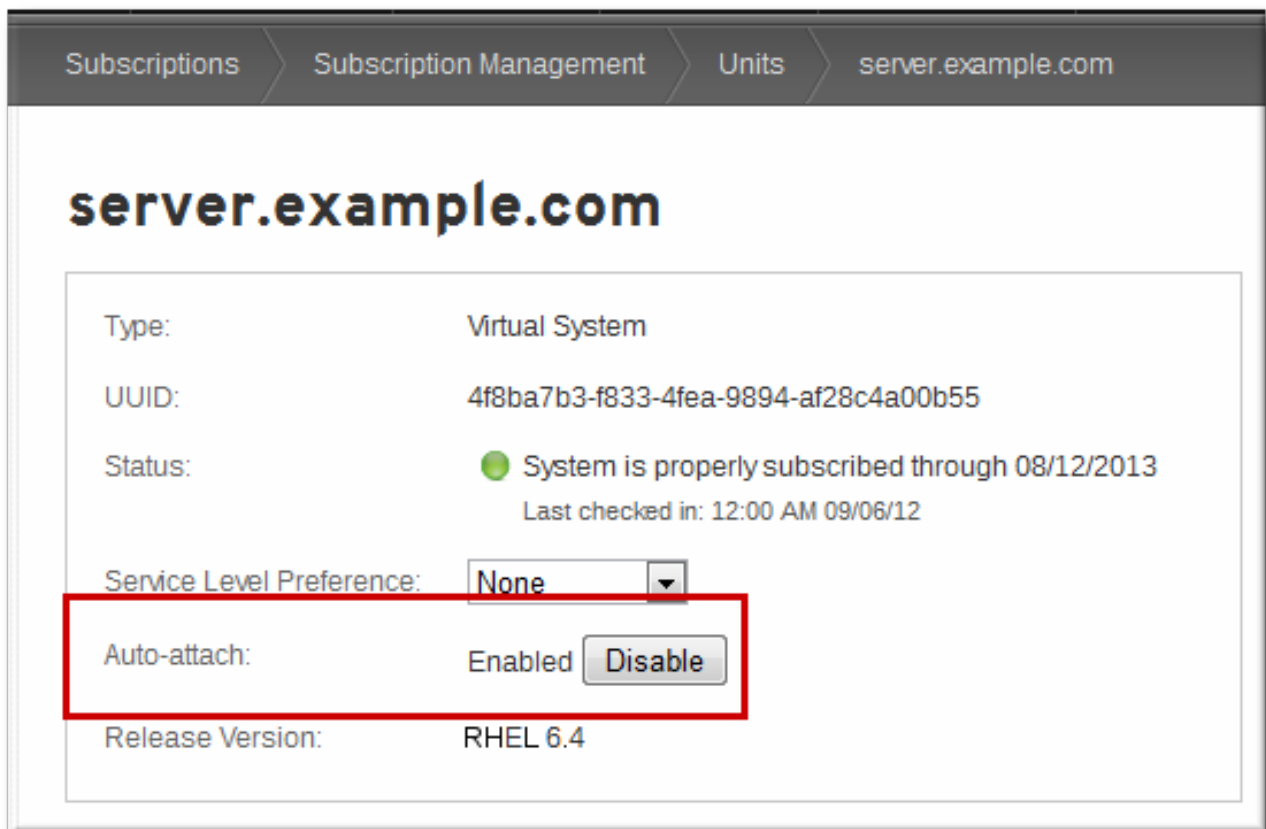


Figure 34. Toggling Autoattaching

3.2.3.2. Initiating an Autoattach Operation on All Systems

Autoattach is configured on individual systems and is generally a local operation. However, there can be times when it is easier or more beneficial to initiate an asynchronous, mass-update of systems, such as when a new set of subscriptions are purchased or a large number of new systems are provisioned.

At the bottom of the **Units** area is a button to run an autoattach operation on all registered systems. This applies any available subscriptions to any system as required to bring them all into a proper (green) status.

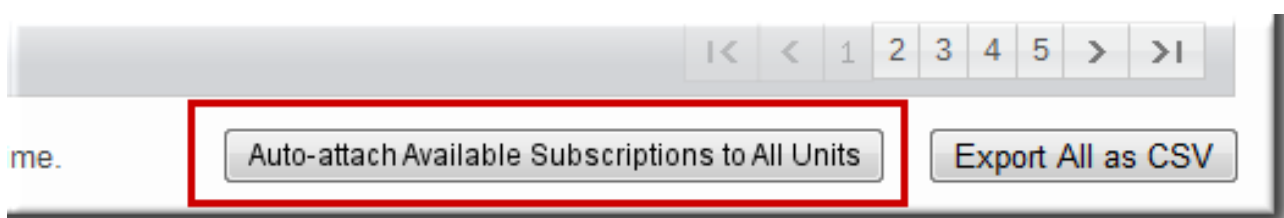


Figure 35. Attaching to All Systems

3.3. Portal: Resolving Over-Utilizing Subscriptions

Red Hat does not restrict how you attach subscriptions – which means that you run the risk of attaching more subscriptions than you actually have purchased.



WARNING

Attaching more subscriptions than you have is the same as running systems without subscriptions.

Along with potentially violating your service contract, this situation can also run afoul of regulations and industry standards – including Sarbanes-Oxley, PCI-DSS, and SAS-70 – that require appropriate licenses for all software in an IT infrastructure.

If you have over-used subscriptions, then the **Utilization** area in the **Overview** page shows a bright yellow warning. The **Utilization** page then shows the subscription counts per type, with a bright yellow bar and a negative number indicating how many subscriptions are over-used for any given type.

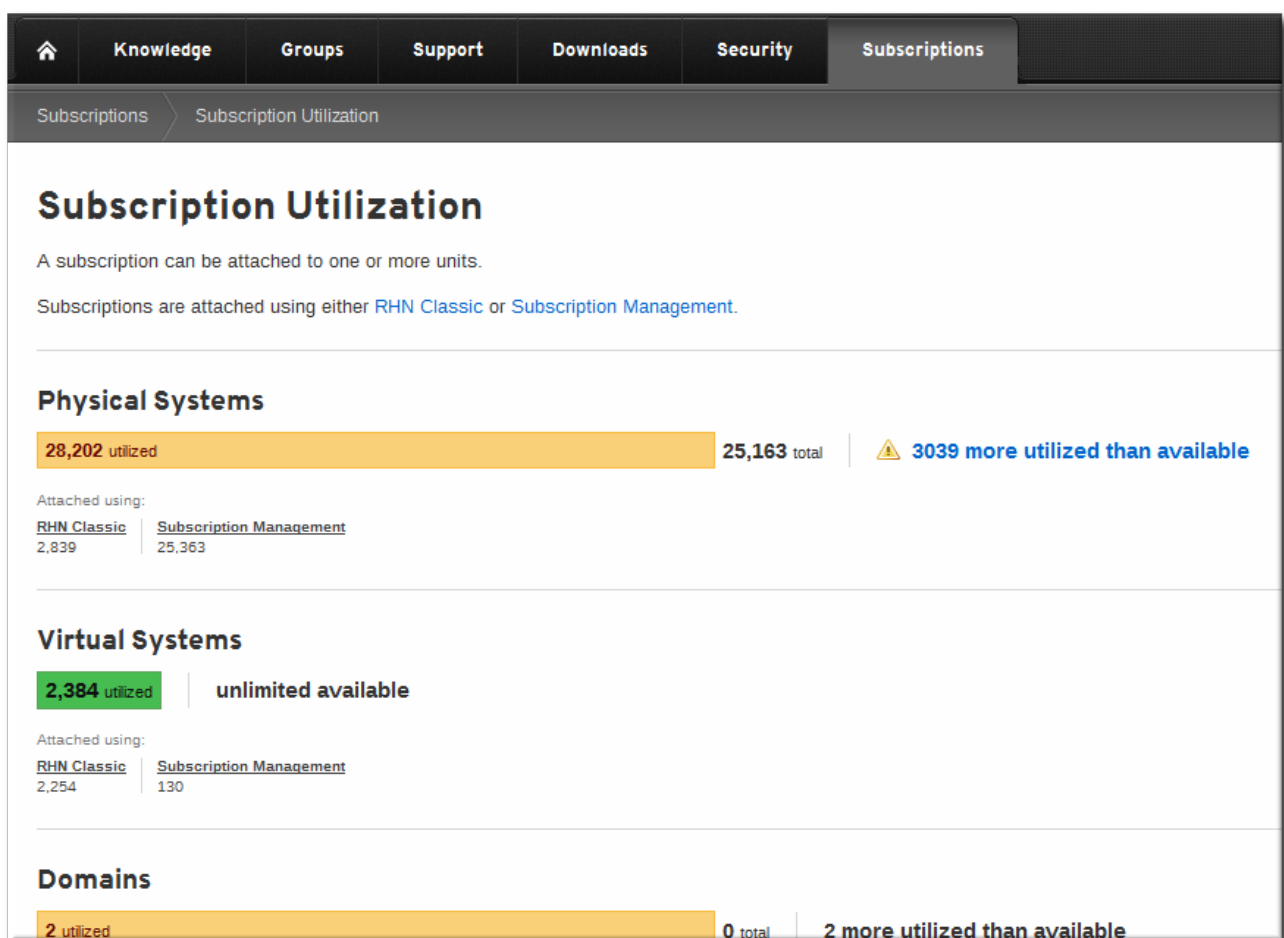
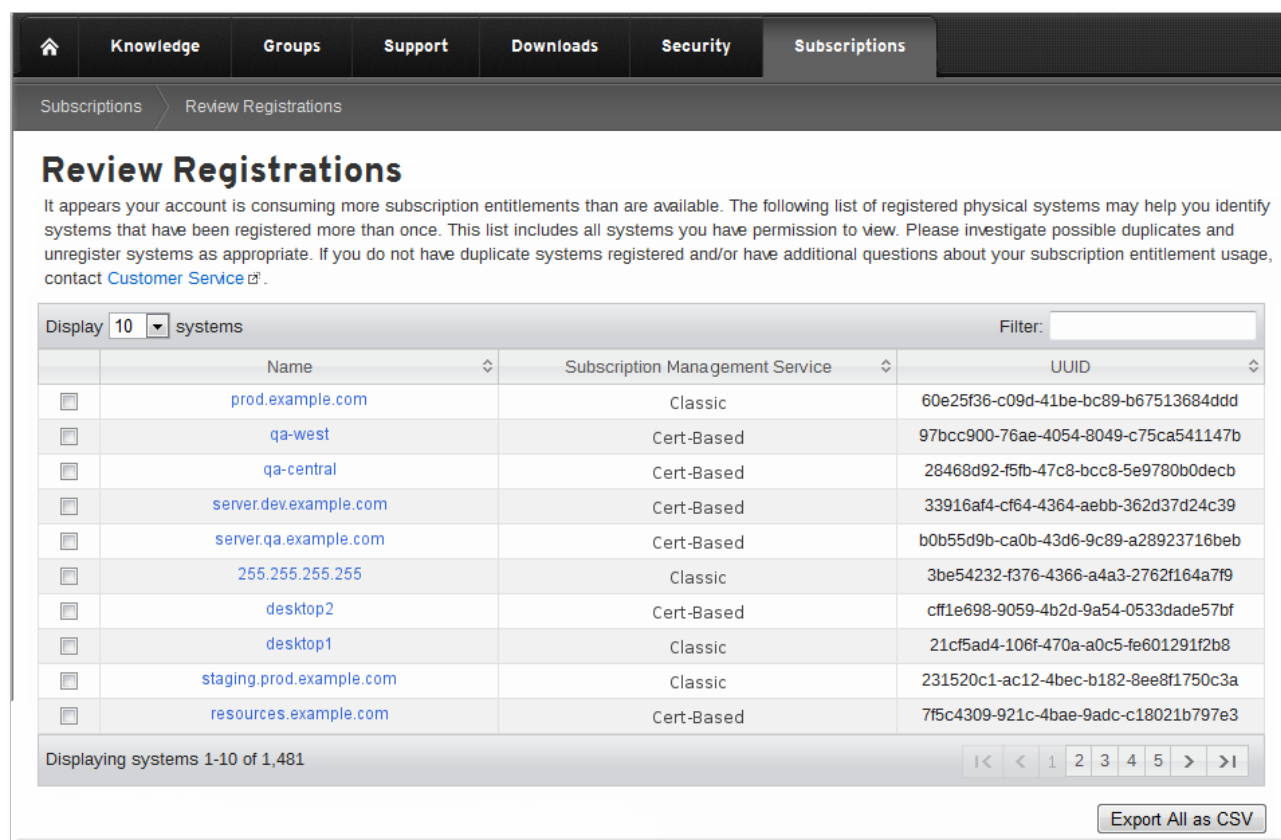


Figure 36. Overutilizing Subscriptions

There is no automatic remediation and Red Hat does not make assumptions or rules about what systems or subscriptions should be changed. That is entirely at the discretion of the administrator. Clicking the **Total** value opens up a list of registered systems or units. Administrators can then edit

system entries and attach and remove subscriptions manually.



Review Registrations

It appears your account is consuming more subscription entitlements than are available. The following list of registered physical systems may help you identify systems that have been registered more than once. This list includes all systems you have permission to view. Please investigate possible duplicates and unregister systems as appropriate. If you do not have duplicate systems registered and/or have additional questions about your subscription entitlement usage, contact [Customer Service](#).

Display **10** systems Filter:

	Name	Subscription Management Service	UUID
<input type="checkbox"/>	prod.example.com	Classic	60e25f36-c09d-41be-bc89-b67513684ddd
<input type="checkbox"/>	qa-west	Cert-Based	97bcc900-76ae-4054-8049-c75ca541147b
<input type="checkbox"/>	qa-central	Cert-Based	28468d92-f5fb-47c8-bcc8-5e9780b0decb
<input type="checkbox"/>	server.dev.example.com	Cert-Based	33916af4-cf64-4364-aebb-362d37d24c39
<input type="checkbox"/>	server.qa.example.com	Cert-Based	b0b55d9b-ca0b-43d6-9c89-a28923716beb
<input type="checkbox"/>	255.255.255.255	Classic	3be54232-f376-4366-a4a3-2762f164a7f9
<input type="checkbox"/>	desktop2	Cert-Based	cff1e698-9059-4b2d-9a54-0533dade57bf
<input type="checkbox"/>	desktop1	Classic	21cf5ad4-106f-470a-a0c5-fe601291f2b8
<input type="checkbox"/>	staging.prod.example.com	Classic	231520c1-ac12-4bec-b182-8ee8f1750c3a
<input type="checkbox"/>	resources.example.com	Cert-Based	7f5c4309-921c-4bae-9adc-c18021b797e3

Displaying systems 1-10 of 1,481 1 < > 1 2 3 4 5 > >1

[Export All as CSV](#)

Figure 37. Reviewing Systems

Clicking the name of any system opens up its details page, so that you can change the subscriptions for it.



NOTE

You must have org admin permissions to be able to see the **Review Registrations** page. Even then, you will only be able to view systems to which you have access – not necessarily every system within the account.

3.4. Subscription Asset Manager: Autoattach and Preferences

3.4.1. Configuring Autoattach Preferences for a System

Autoattaching and updating subscriptions selects what subscriptions to attach to a system based on a variety of criteria, including current installed products, hardware, and architecture. Most of the factors used for selecting the best-matched subscriptions are based on the characteristics of the system, but it is possible to include characteristics of the subscriptions.

Part of a subscription is recognizing the service level for a product on a given system. That service level can be used as a criterion for selecting a subscription to attach to a system.

Red Hat service levels are defined in the contract; a summary of production support levels is available at <https://access.redhat.com/support/offerings/production/sla.html>.

An account can have multiple levels of support available, even for the same product. The support level for a given system can be configured so that the appropriate level of support is available. A production

system usually has a premium support level since it is a business critical system, while a development system may have standard support or be self-supported.

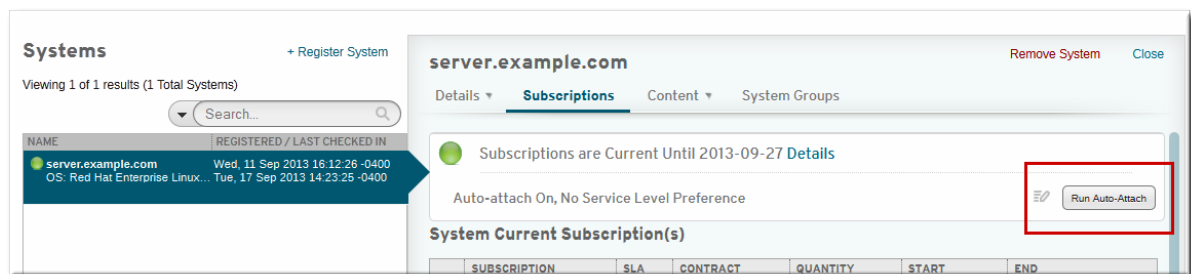
An organization can set a default service level which all systems use when autoattaching subscriptions. A local, system-level preference can be set to override that organizational default, if desired.



NOTE

By default, the highest available level of support is selected for the subscription and system.

1. Hover over the **Systems** tab in the top menu and select the **All** item.
2. Select the name of the system from the column on the left.
3. Open the **Subscriptions** tab.
4. Click the edit icon in the top box to change the autoattach settings.



5. Select the appropriate autoattach setting.

The options in the list depend on the available support levels in the subscriptions for the organization. From a high level, the options are:

- Enable autoattach and use a specific system-level preference for the support level.
- Enable autoattach and use the default support level preference for the organization.
- Disable autoattach and set the support level preference to either a system-level setting or the organizational default. (In either case, the preference is not used since autoattach is disabled.)

6. Click the **Save** button.

3.4.2. Running Autoattach Operations Manually

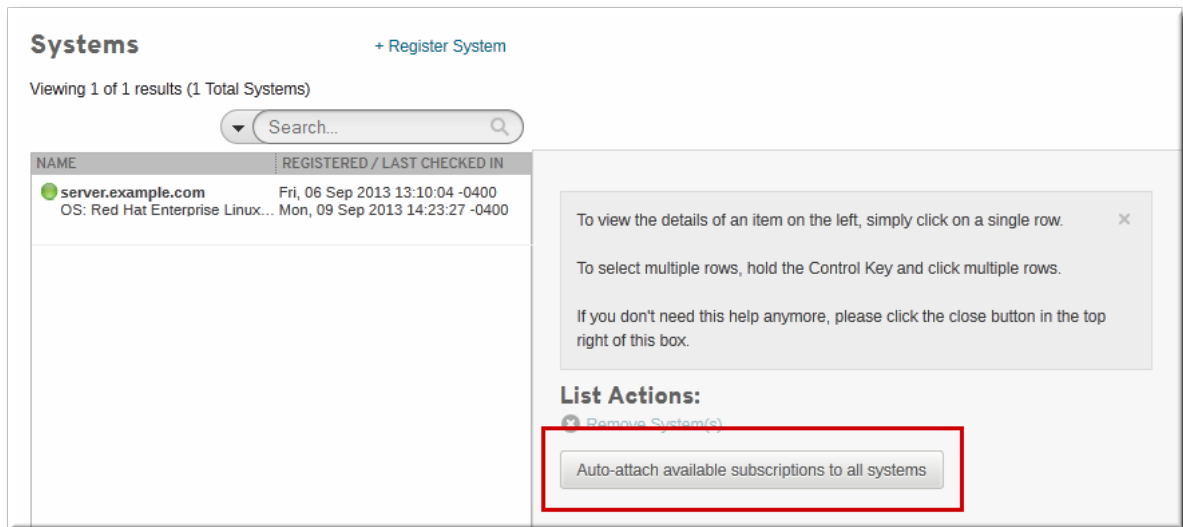
The local system can run a job every four hours to update subscriptions automatically, and attach or remove any subscriptions depending on the installed products and subscription status.

It is possible to run an asynchronous autoattach operation on a single system or for every system, to update all subscriptions on the system immediately.

3.4.2.1. Running Autoattach on All Systems

To update subscriptions for all systems:

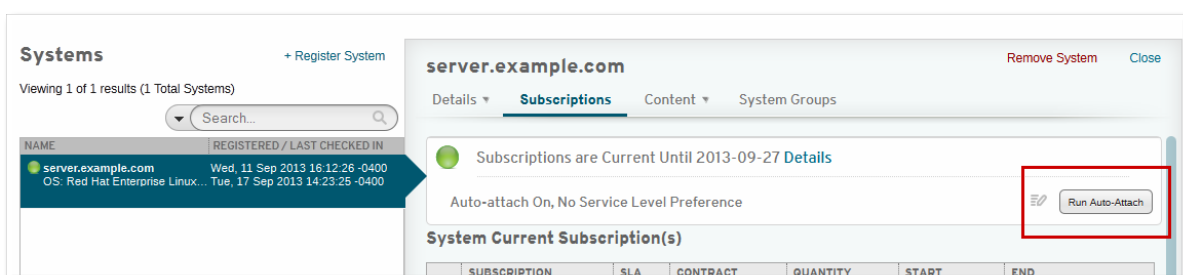
1. Hover over the **Systems** tab in the top menu and select the **All** item.
2. In the main page for systems, click the **Auto-attach available subscriptions to all systems** button.



3.4.2.2. Running Autoattach on a Single System

To update subscriptions for all systems:

1. Hover over the **Systems** tab in the top menu and select the **All** item.
2. In the search box on the left of the systems column, search for the specific system.
3. Click the name of the system in the column on the left.
4. Click the **Subscriptions** tab for the system.
5. In the upper right of the subscriptions area, click the **Run Auto-Attach** button.



4. MANAGING ERRATA NOTIFICATIONS

4.1. Portal: Managing Errata Notifications for Registered Systems

Part of subscription management is tracking updates and new releases of software. Whenever an update is available – from a bug fix to a new release – a notification email can be sent to an administrator.

The notifications are smart, so they are only sent for 1) registered systems which 2) have subscriptions for that product attached to them. If there are no systems with attached subscriptions for that product, even if the account does have subscriptions for it, then no notification is sent.

Errata notifications are set as a preference for the user account, not for an individual system. So, when checking for potential errata updates, Subscription Management checks the entire inventory, not specific systems.

An errata notification is sent if *any* registered system is affected, but the email does not list what systems are actually affected.

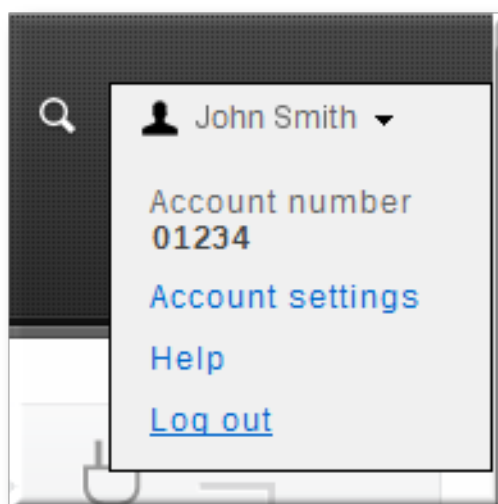


NOTE

Because Customer Portal Subscription Management Subscription Management and RHN Classic have separate inventories, they each have their own errata notification setting. Even if errata notifications are already enabled for RHN Classic, they must still be enabled for Customer Portal Subscription Management Subscription Management, or no errata notifications will be sent for systems managed in the Customer Portal Subscription Management Subscription Management inventory.

To configure errata notifications for a user account:

1. In the upper right corner of the Customer Portal, expand the details for the logged in user.



2. Click the **Account settings** link.
3. In the settings main page, click the **Account Details** link in the middle of the **Your Red Hat Account** box.



4. In the **Your Preferences** menu on the left, click the **Errata Notifications** link.
5. Select the checkboxes for each type of errata for which to receive an update. Security errata relate to critical security issues. Bug fixes and enhancement notifications relate to incremental updates to the product.

redhat.com • Customer Portal • Account Details • User Management • Certifications Navigate

Your Information

[Personal](#)

[Login & Password](#)

[Postal Address](#)

Your Preferences

[Language & Location](#)

[Email Preferences](#)

[Errata Notifications](#)

Errata Notifications

Red Hat customers can receive notifications of security and enhancement updates. Please select in which cases you'd like to be notified.

After making your choices, you must select the "Save" button to complete your action(s).

Save
Cancel

SUBSCRIPTION MANAGEMENT:

Errata notifications: Receive an email for each erratum that affects a subscription assigned to a registered consumer. ([More Info](#))

- Choose advisory type(s) to receive notifications: (select any or all)

☒ Security

☒ Bug Fix

☒ Enhancement

- Choose frequency to receive notifications: (select one)

☐ Instant

☐ Daily

☐ Weekly

CLASSIC MANAGEMENT:

To opt-in to errata and system notifications, visit [Classic Management notifications](#).

6. Set the frequency to receive errata notifications. This applies to all selected types of errata notifications.

7. Click the **Save** button.

4.2. Red Hat Subscription Manager: Viewing Package Profiles for the Local System

A *package profile* is the list of installed packages on a system (regardless of its subscription status). Once a system is registered, then the `rhsmc` process polls the system to determine what products are installed and forwards that information to the subscription service. The package list is an integral part of managing updates, system notifications, and errata notifications.

Red Hat Subscription Manager maintains a local list of installed packages to track the subscription status of the system. The package profile contains some general information about each package in the list:

- Package name
- Package version
- Epoch
- Publisher

All of that information about currently installed packages is collected in a regular job by the `rhsmc` process and sent to the registering subscription service, along with the user login information.

The package list itself is handled slightly differently depending on how the system is registered.

- For systems registered with Customer Portal Subscription Management through the local Subscription Manager, the package list is sent periodically to the Customer Portal Subscription Management hosted services to check for updates.

The package list is viewable in the **Installed Products** tab or by using the **list --installed** command.

- For systems where their inventory entry was created in the Customer Portal (rather than using Subscription Manager), the package list is generated by the **rhsmcertd** process, sent to the subscription service along with the user login, and then stored.

The package list is displayed on the system entry and used to generate errata notifications (although it is possible to opt out of the notifications themselves).

5. REVISION HISTORY

Revision 1.3-7	April 13, 2014	Ella Deon Ballard
Updating instance-based and virtual setup sections.		
Revision 1.3-4	October 1, 2013	Deon Ballard
New content and reorganization for the SAM 1.3 release.		