



Red Hat Subscription Management 2022

Configuring Virtual Machine Subscriptions in Red Hat Subscription Management

Using virt-who to manage host-based subscriptions

Red Hat Subscription Management 2022 Configuring Virtual Machine Subscriptions in Red Hat Subscription Management

Using virt-who to manage host-based subscriptions

Red Hat Subscription Management Documentation Team
rhsm-docs@redhat.com

Legal Notice

Copyright © 2022 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

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, the Red Hat 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 is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

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

All other trademarks are the property of their respective owners.

Abstract

This guide provides information about preparing subscriptions in Red Hat Subscription Management, configuring virt-who, and registering virtual machines so that they inherit a subscription from their hypervisor.

Table of Contents

CHAPTER 1. INTRODUCTION	3
1.1. HOST-BASED SUBSCRIPTIONS	3
1.2. CONFIGURATION OVERVIEW	5
1.3. VIRT-WHO CONFIGURATION FOR EACH VIRTUALIZATION PLATFORM	5
1.4. VIRT-WHO GENERAL CONFIGURATION	6
CHAPTER 2. ATTACHING A HOST-BASED SUBSCRIPTION TO HYPERVISORS	8
CHAPTER 3. PREPARING A VIRT-WHO HOST	9
CHAPTER 4. CONFIGURING VIRT-WHO	10
4.1. INSTALLING AND CONFIGURING VIRT-WHO ON RED HAT HYPERVISORS	10
4.2. CONFIGURING VIRT-WHO TO CONNECT TO VMWARE VCENTER	11
4.3. CONFIGURING VIRT-WHO TO CONNECT TO MICROSOFT HYPER-V	13
4.4. CONFIGURING VIRT-WHO TO CONNECT TO OPENSIFT VIRTUALIZATION	14
CHAPTER 5. REGISTERING VIRTUAL MACHINES TO USE A HOST-BASED SUBSCRIPTION	17
CHAPTER 6. VIRT-WHO TROUBLESHOOTING METHODS	18
CHAPTER 7. VIRT-WHO TROUBLESHOOTING SCENARIOS	22

CHAPTER 1. INTRODUCTION

You can use host-based subscriptions for Red Hat Enterprise Linux virtual machines in the following virtualization platforms:

- Red Hat Virtualization
- Red Hat Enterprise Linux Virtualization (KVM)
- Red Hat OpenStack Platform
- VMware vSphere
- Microsoft Hyper-V
- OpenShift Virtualization

1.1. HOST-BASED SUBSCRIPTIONS

Virtual machines can use host-based subscriptions instead of consuming entitlements from physical subscriptions. A host-based subscription is attached to a hypervisor and entitles it to provide subscriptions to its virtual machines. Many host-based subscriptions provide entitlements for unlimited virtual machines.

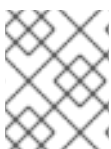
To allow virtual machines to inherit subscriptions from their hypervisors, you must install and configure `virt-who`. `Virt-who` queries the virtualization platform and reports hypervisor and virtual machine information to Red Hat Subscription Management.

When a virtual machine is registered with auto-attach enabled, and sufficient host-based subscriptions are available, one of the following behaviors occurs:

- If the virtual machine has been reported by `virt-who` and a host-based subscription is attached to the hypervisor, the virtual machine inherits a subscription from the hypervisor.
- If the virtual machine has been reported by `virt-who`, and the hypervisor is registered to Subscription Management but does not have a host-based subscription attached, a host-based subscription is attached to the hypervisor and inherited by the virtual machine.
- If the virtual machine, or its hypervisor, has not been reported by `virt-who`, Subscription Management grants the virtual machine a temporary subscription, valid for up to seven days. After `virt-who` reports updated information, Subscription Management can determine which hypervisor the virtual machine is running on and attach a permanent subscription to the virtual machine.

If auto-attach is enabled, but `virt-who` is not running or there are no host-based subscriptions available, Subscription Management attaches physical subscriptions to the virtual machines instead, which might consume more entitlements than intended.

If auto-attach is not enabled, virtual machines cannot use host-based subscriptions.



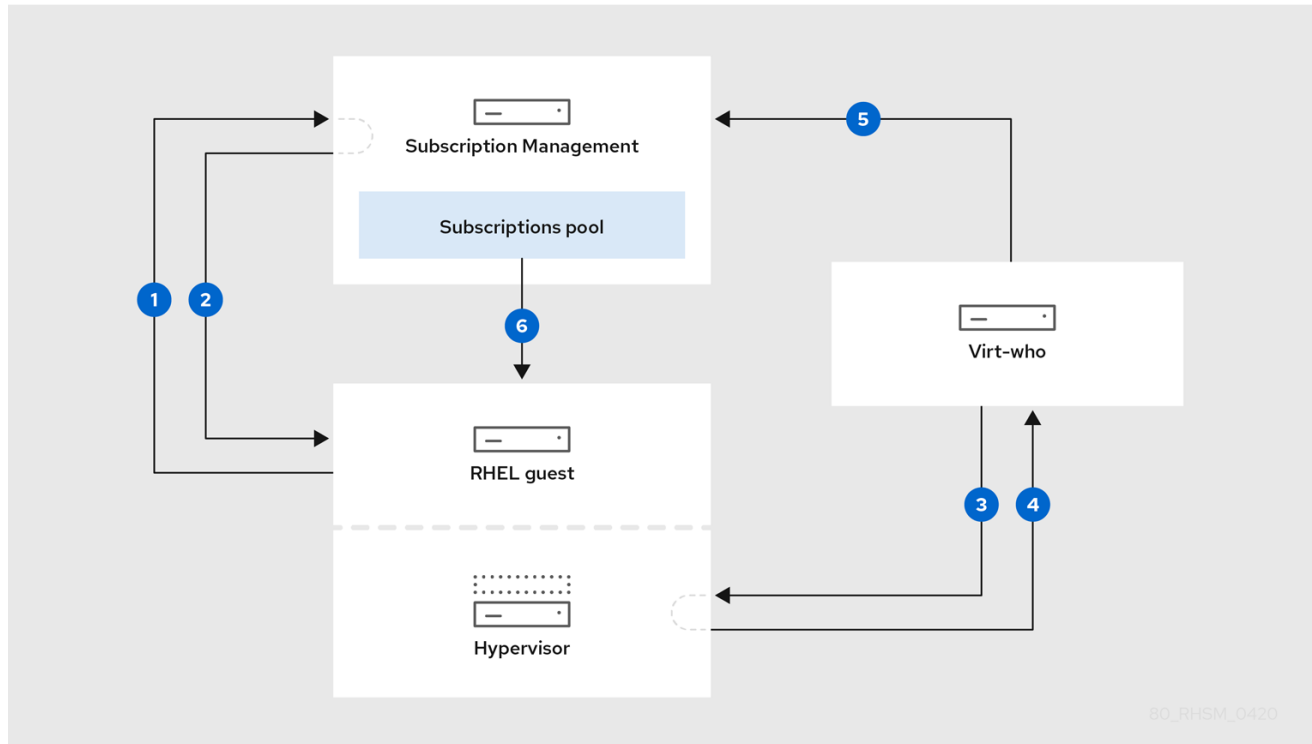
NOTE

System Purpose [add-ons have no effect on the auto-attach feature](#) in Red Hat Enterprise Linux 8.0, 8.1, and 8.2.

To see if a subscription requires virt-who, log in to the Customer Portal at <https://access.redhat.com>, navigate to **Subscriptions > Subscription Utilization**, and select a subscription. If "**Virt-Who: Required**" appears in the **SKU Details**, you must configure virt-who in order to use that subscription.

Virtual machine subscription process

This diagram shows the subscription workflow when a virtual machine has not yet been reported by virt-who:



- 1 A virtual machine requests a subscription from Subscription Management.
- 2 Subscription Management grants the virtual machine a temporary subscription, valid for a maximum of seven days, while it determines which hypervisor the virtual machine belongs to.
- 3 Virt-who connects to the hypervisor or virtualization manager and requests information about its virtual machines.
- 4 The hypervisor or virtualization manager returns a list of its virtual machines to virt-who, including each UUID.
- 5 Virt-who reports the list of virtual machines and their hypervisors to Subscription Management.
- 6 Subscription Management attaches a permanent subscription to the virtual machine, if sufficient entitlements are available.

Additional resources

For more information about the Red Hat subscription model, see [Introduction to Red Hat Subscription Management Workflows](#).

1.2. CONFIGURATION OVERVIEW

To allow virtual machines to inherit subscriptions from their hypervisors, complete the following steps:

Prerequisites

- Ensure you have sufficient entitlements for the host-based subscription to cover all of the hypervisors you plan to use.
- For Microsoft Hyper-V, create a read-only virt-who user with a non-expiring password on each hypervisor that runs Red Hat Enterprise Linux virtual machines.
- For VMware vSphere, create a read-only virt-who user with a non-expiring password on the vCenter Server. The virt-who user requires at least read-only access to all objects in the vCenter Data Center.
- For OpenShift Virtualization, create a Service Account and grant it with an admin role in the OpenShift cluster master, virt-who need the Service Account Token to connect the OpenShift cluster.

Procedure overview

1. [Section 1.3, “Virt-who configuration for each virtualization platform”](#) . Use the table in this section to plan how to install and configure virt-who for your virtualization platform.
2. [Chapter 2, Attaching a host-based subscription to hypervisors](#) . Attach a host-based subscription to all of the hypervisors you plan to use.
3. [Chapter 4, Configuring virt-who](#) . Create individual configuration files to connect virt-who to your hypervisors.
4. [Chapter 5, Registering virtual machines to use a host-based subscription](#) . Register the virtual machines and enable auto-attach.

1.3. VIRT-WHO CONFIGURATION FOR EACH VIRTUALIZATION PLATFORM

Virt-who is configured using files that specify details such as the virtualization type and the hypervisor or virtualization manager to query. The supported configuration is different for each virtualization platform.

- Individual configuration files are stored in the `/etc/virt-who.d/` directory. You must create an individual configuration file for each hypervisor or virtualization manager.

Example virt-who configuration file

This example shows an individual virt-who configuration file for a Microsoft Hyper-V hypervisor:

```
[hypervisor1]
type=hyperv
server=hypervisor1.example.com
username=virt_who_user
encrypted_password=bd257f93d@482B76e6390cc54aec1a4d
hypervisor_id=hostname
owner=1234567
```

The **type** and **server** values depend on the virtualization platform. The following table provides more detail.

The **username** refers to a read-only user on Microsoft Hyper-V or VMware vCenter, which you must create before configuring virt-who. Virt-who uses this account to retrieve the list of virtual machines. You do not need a dedicated virt-who user for Red Hat hypervisors.

Required configuration for each virtualization platform

Use this table to plan your virt-who configuration:

Supported virtualization platform	Type specified in the configuration file	Server specified in the configuration file	Server where virt-who is installed
Red Hat Virtualization RHEL Virtualization (KVM) Red Hat OpenStack Platform	libvirt	Not required	Each hypervisor
VMware vSphere	esx	vCenter Server	A dedicated RHEL server
Microsoft Hyper-V	hyperv	Hypervisor	A dedicated RHEL server
OpenShift Virtualization	kubevirt	OpenShift Cluster Master	A dedicated RHEL server



IMPORTANT

The **rhev** and **xen** hypervisor types are not supported.

1.4. VIRT-WHO GENERAL CONFIGURATION



NOTE

'/etc/sysconfig/virt-who' will not be supported in the next major release, the global configuration file will be replaced by '/etc/virt-who.conf'. (i.e. 'VIRTWHO_DEBUG', 'VIRTWHO_ONE_SHOT', 'VIRTWHO_INTERVAL', 'HTTPS_PROXY, NO_PROXY').

The general configuration file (located at '/etc/virt-who.conf') is created automatically when you install virt-who. You can use the default values or edit this file if required. It has three special sections: '[global]', '[defaults]', and '[system_environment]'.

The settings in the global section affect the overall operation of the application.

Example: Global section

```
[global]
interval=3600 1
debug=True 2
```

- 1 How often to check connected hypervisors for changes (seconds). Also affects how often a mapping is reported. Because the virtual machines are granted temporary subscriptions for up to seven days, frequent queries are not required; you can select an interval that suits the size of your environment.
- 2 Enable debugging output

The settings in the defaults that can be applied as defaults to the configurations found in `"/etc/virt-who.d/.conf"`. **If you enable the options in this section, you don't need to set them in `"/etc/virt-who.d/.conf"` again.**

Example: Defaults section

```
[defaults]
owner=1234567 1
hypervisor_id=hostname 2
```

- 1 The organization the hypervisor belongs to. You can find the organization by running `subscription-manager orgs` on the hypervisor.
- 2 How will be the hypervisor identified, one of: `uuid`, `hostname`, `hwuuid`

The settings in the `system_environment` are written to the system's environment and are available for the duration of the process execution, it will be used whether `virt-who` was started as a service or from the command line.

Example: system_environment section

```
[system_environment]
http_proxy= https://proxy.example.com:443 1
no_proxy=* 2
```

- 1 Use an HTTP proxy for `virt-who` communication
- 2 If you do not want to use an HTTP proxy for any `virt-who` communication from this server, you can set `no_proxy` to `*`.



NOTE

The section `[system_environment]` is supported from `virt-who-0.30.x-1.el8` (RHEL 8.4). If you are using the old `virt-who` version, please set `'HTTP_PROXY'`, `'NO_PROXY'` by `"/etc/sysconfig/virt-who"`.

CHAPTER 2. ATTACHING A HOST-BASED SUBSCRIPTION TO HYPERVISORS

Use this procedure to attach a host-based subscription, such as **Red Hat Enterprise Linux for Virtual Datacenters**, to hypervisors that are already registered.

To register a new hypervisor, see [Using and Configuring Red Hat Subscription Manager](#). You must register a hypervisor before configuring virt-who to query it.

Prerequisites

- Ensure you have sufficient entitlements for the host-based subscription to cover all of the hypervisors you plan to use.

Web UI procedure

1. Log in to the Customer Portal at <https://access.redhat.com>.
2. Navigate to **Subscriptions > Systems** and click the name of the hypervisor.
3. Click the **Subscriptions** tab.
4. Click **Attach Subscriptions**.
5. Select the host-based subscription, then click **Attach Subscriptions**.

Repeat these steps for each hypervisor.

CLI procedure

1. On the hypervisor, identify and make a note of your host-based subscription's Pool ID:

```
# subscription-manager list --all --available --matches 'Host-based Subscription Name'
```

2. Attach the host-based subscription to the hypervisor:

```
# subscription-manager attach --pool=Pool_ID
```

3. Verify that the host-based subscription is attached:

```
# subscription-manager list --consumed
```

Repeat these steps for each hypervisor.

CHAPTER 3. PREPARING A VIRT-WHO HOST

Use this procedure to configure a Red Hat Enterprise Linux 7 server to run the virt-who service for VMware vCenter and Microsoft Hyper-V. The server can be physical or virtual.

You do not need a separate virt-who host for Red Hat hypervisors.

Procedure

1. Install a Red Hat Enterprise Linux 7 server. Only a CLI environment is required. For more information, see the [Red Hat Enterprise Linux 7 Installation Guide](#).

2. Register the server:

```
# subscription-manager register --auto-attach
```

3. Open a network port for communication between virt-who and the subscription service:

```
# firewall-cmd --add-port="443/tcp"  
# firewall-cmd --add-port="443/tcp" --permanent
```

4. Open a network port for communication between virt-who and each hypervisor or virtualization manager:

- VMware vCenter: TCP port 443
- Microsoft Hyper-V: TCP port 5985

5. Install virt-who:

```
# yum install virt-who
```

6. Optional: Edit the `/etc/virt-who.conf` file to change or add global settings. These settings apply to all virt-who connections from this server.

- Change the value of **VIRTWHO_INTERVAL** to specify how often, in minutes, virt-who queries the virtualization platform. Because the virtual machines are granted temporary subscriptions for up to seven days, frequent queries are not required; you can select an interval that suits the size of your environment. Once a day (**1440**) is suitable for most environments.
- If you want to use an HTTP proxy for virt-who communication, add a line specifying the proxy:

```
http_proxy=https://proxy.example.com:443
```

- If you do not want to use an HTTP proxy for any virt-who communication from this server, add the following line:

```
NO_PROXY=*
```

7. Start and enable the virt-who service:

```
# systemctl enable --now virt-who
```

CHAPTER 4. CONFIGURING VIRT-WHO



IMPORTANT

The use of environment variables and the use of the `sysconfig` file to configure `virt-who` are deprecated. Their use will be ignored in the next major release.

The supported `virt-who` configuration is different for each virtualization platform:

- To configure `virt-who` for Red Hat products, see [Section 4.1, “Installing and configuring `virt-who` on Red Hat hypervisors”](#).
- To configure `virt-who` for VMware vCenter, see [Section 4.2, “Configuring `virt-who` to connect to VMware vCenter”](#).
- To configure `virt-who` for Microsoft Hyper-V, see [Section 4.3, “Configuring `virt-who` to connect to Microsoft Hyper-V”](#).
- To configure `virt-who` for OpenShift Virtualization, see [Section 4.4, “Configuring `virt-who` to connect to OpenShift Virtualization”](#).

4.1. INSTALLING AND CONFIGURING VIRT-WHO ON RED HAT HYPERVISORS

Use this procedure to install and configure `virt-who` on each hypervisor in Red Hat Enterprise Linux Virtualization (KVM), Red Hat Virtualization, or Red Hat OpenStack Platform.

Prerequisites

- Register the hypervisor to Red Hat Subscription Management.
- If you are using Red Hat Virtualization Host (RHVH), update it to the latest version so that the minimum `virt-who` version is available. `virt-who` is available by default on RHVH, but cannot be updated individually from the `rhel-7-server-rhvh-4-rpms` repository.

Procedure

1. Install `virt-who` on the hypervisor:

```
# yum install virt-who
```

2. Optional: Edit the `/etc/virt-who.conf` file to change or add global settings. Because `virt-who` is installed locally, these settings apply only to this hypervisor.
 - Change the value of `VIRTWHO_INTERVAL` to specify how often, in minutes, `virt-who` queries the hypervisor. Because the virtual machines are granted temporary subscriptions for up to seven days, frequent queries are not required; you can select an interval that suits the size of your environment. Once a day (**1440**) is suitable for most environments.
 - If you want to use an HTTP proxy for `virt-who` communication, add a line specifying the proxy:

```
http_proxy=https://proxy.example.com:443
```

- If you do not want to use an HTTP proxy for any virt-who communication from this server, add the following line:

```
NO_PROXY=*
```



NOTE

NO_PROXY=* can be used but only in `/etc/sysconfig/virt-who`.

NO_PROXY is not a valid configuration in `/etc/virt-who.conf`.

3. Copy the template configuration file to a new individual configuration file:

```
# cp /etc/virt-who.d/template.conf /etc/virt-who.d/local.conf
```

4. Edit the configuration file you just created, changing the example values to those specific to your configuration:

```
[local] 1
type=libvirt 2
owner=1234567 3
hypervisor_id=hostname 4
```

- 1 The name does not need to be unique, because this configuration file is the only one managed by this instance of virt-who.
- 2 Specifies that this virt-who connection is to a Red Hat hypervisor.
- 3 The organization the hypervisor belongs to. You can find the organization by running **subscription-manager orgs** on the hypervisor.
- 4 Specifies how to identify the hypervisor. Use **hostname** to provide meaningful host names to Subscription Management. Alternatively, you can use **uuid** to avoid duplication if a hypervisor is renamed. Do not use **hwuuid** for an individual hypervisor.

5. Start and enable the virt-who service:

```
# systemctl enable --now virt-who
```

Repeat these steps for each hypervisor.

4.2. CONFIGURING VIRT-WHO TO CONNECT TO VMWARE VCENTER

Use this procedure to configure virt-who to connect to a VMware vCenter Server.

Prerequisites

- Create a read-only virt-who user on the vCenter Server. The virt-who user requires at least read-only access to all objects in the vCenter Data Center.
- Prepare a virt-who host on a Red Hat Enterprise Linux server.

Procedure

1. On the virt-who host, encrypt the virt-who user's password with the **virt-who-password** utility:

```
# virt-who-password
```

When prompted, enter the password of the virt-who user, then make a note of the encrypted form of the password.

2. Copy the template configuration file to a new individual configuration file:

```
# cp /etc/virt-who.d/template.conf /etc/virt-who.d/vcenter1.conf
```

To make it easy to identify the configuration file when troubleshooting, use the VMware vCenter host name as the new file's name. In this example, the host name is *vcenter1*.

3. Edit the configuration file you just created, changing the example values with those specific to your configuration:

```
[vcenter1] 1
type=esx 2
server=vcenter1.example.com 3
username=virt_who_user 4
encrypted_password=bd257f93d@482B76e6390cc54aec1a4d 5
owner=1234567 6
hypervisor_id=hostname 7
filter_hosts=esx1.example.com, esx2.example.com 8
```

- 1 The name must be unique for each individual configuration file. Use the vCenter Server host name to make it easy to identify the configuration file for each hypervisor.
- 2 Specifies that this virt-who connection is to a VMware vCenter Server.
- 3 The FQDN of the vCenter Server.
- 4 The name of the virt-who user on the vCenter Server.
- 5 The encrypted password of the virt-who user.
- 6 The organization the hypervisors belong to. You can find the organization by running **subscription-manager orgs** on a hypervisor.
- 7 Specifies how to identify the hypervisors. Use **hostname** to provide meaningful host names to Subscription Management. Alternatively, you can use **uuid** or **hwuuid** to avoid duplication if a hypervisor is renamed.
- 8 If some hypervisors never run Red Hat Enterprise Linux virtual machines, those hypervisors do not need to be reported by virt-who. You can filter hypervisors using one of the following options. Wildcards and regular expressions are supported. If a name contains special characters, enclose it in quotation marks.
 - **filter_hosts** or **exclude_hosts**: Provide a comma-separated list of hypervisors according to the specified **hypervisor_id**. For example, if hypervisors are identified by their host name, they must be included or excluded by their host name.

- **filter_host_parents** or **exclude_host_parents**: Provide a comma-separated list of clusters. Hypervisors in a filtered cluster are reported by virt-who. Hypervisors in an excluded cluster are not reported by virt-who.

4. Restart the virt-who service:

```
# systemctl restart virt-who
```

Repeat these steps for each vCenter Server.

4.3. CONFIGURING VIRT-WHO TO CONNECT TO MICROSOFT HYPER-V

Use this procedure to configure virt-who to connect to a Microsoft Hyper-V hypervisor.

Prerequisites

- Red Hat Enterprise Linux 9 or later.
- Prepare a virt-who host on a Red Hat Enterprise Linux server.
- Enable basic authentication mode for the hypervisor.
- Enable remote management on the hypervisor.
- Create a read-only virt-who user on the hypervisor.

Procedure

1. On the virt-who host, encrypt the password of the hypervisor's virt-who user with the **virt-who-password** utility:

```
# virt-who-password
```

When prompted, enter the password of the virt-who user, then make a note of the encrypted form of the password.

2. Copy the template configuration file to a new individual configuration file:

```
# cp /etc/virt-who.d/template.conf /etc/virt-who.d/hyperv1.conf
```

To make it easy to identify the configuration file when troubleshooting, use the hypervisor's host name as the new file's name. In this example, the host name is *hyperv1*.

3. Edit the configuration file you just created, changing the example values with those specific to your configuration:

```
[hyperv1] 1
type=hyperv 2
server=hyperv1.example.com 3
username=virt_who_user 4
```

```
encrypted_password=bd257f93d@482B76e6390cc54aec1a4d 5
owner=1234567 6
hypervisor_id=hostname 7
```

- 1 The name must be unique for each individual configuration file. Use the hypervisor's host name to make it easy to identify the configuration file for each hypervisor.
- 2 Specifies that this virt-who connection is to a Microsoft Hyper-V hypervisor.
- 3 The FQDN of the Hyper-V hypervisor.
- 4 The name of the virt-who user on the hypervisor.
- 5 The encrypted password of the virt-who user.
- 6 The organization this hypervisor belongs to. You can find the organization by running **subscription-manager orgs** on the hypervisor.
- 7 Specifies how to identify the hypervisor. Use **hostname** to provide meaningful host names to Subscription Management. Alternatively, you can use **uuid** to avoid duplication if a hypervisor is renamed. Do not use **hwuuid** for an individual hypervisor.

4. Restart the virt-who service:

```
# systemctl restart virt-who
```

Repeat these steps for each hypervisor.

4.4. CONFIGURING VIRT-WHO TO CONNECT TO OPENSIFT VIRTUALIZATION

Supported Platforms

OpenShift Virtualization supported status by virt-who:

- virt-who-0.28.x-1.el7 (RHEL 7.9)
- virt-who-0.29.x-1.el8 (RHEL 8.3)

Procedure

1. In the cluster you want to subscribe, create a project and a service account named virt-who:

```
$ oc new-project virt-who
$ oc create serviceaccount virt-who
```

2. Create cluster roles to list nodes and virtual machine Instances.

```
$ oc create clusterrole lsnodes --verb=list --resource=nodes
$ oc create clusterrole lsvmis --verb=list --resource=vmis
```

3. Create cluster role bindings.

```
$ oc adm policy add-cluster-role-to-user lsnodes system:serviceaccount:virt-who:virt-who
$ oc adm policy add-cluster-role-to-user lsvmis system:serviceaccount:virt-who:virt-who
```

- Verify that the virt-who system account has the permissions to list all running VMs:

```
$ oc get vmis -A --as=system:serviceaccount:virt-who:virt-who
```

- Install virt-who on a host, which can be a VM running on OpenShift Virtualization itself:

```
[virtwho-host]$ yum install virt-who
```

- Find your owner number on a subscribed host:

```
$ subscription-manager orgs
```

- Copy the template configuration file to a new individual configuration file. To make it easy to identify the configuration file when troubleshooting, use the hostname of the cluster API. In this example, the host name is **openshift-cluster-1**.

```
[virtwho-host]# cp /etc/virt-who.d/template.conf /etc/virt-who.d/openshift-cluster-1.conf
[cnv]
type=kubevirt
kubeconfig=/root/.kube/config
hypervisor_id=hostname
owner=<owner_number>
```

- Get the token of the virt-who service account:

```
# oc serviceaccounts get-token virt-who
```

- If **/usr/bin/oc** is not available, install **/usr/bin/oc** and use the token to log in and to create a valid kubeconfig file. You must specify the cluster api by including the url. For example:

```
[virtwho-host]# oc login https://api.testcluster-1.example.org:6443 --token=<token>
```

- To use the OpenShift Virtualization certificate-authority (CA) certificate in the kubeconfig file, extract it from the cluster and save it to a file on the system running virt-who as the controller daemon:

```
oc get secret -n openshift-kube-apiserver-operator loadbalancer-serving-signer -o
jsonpath='{.data.tls.crt}' | base64 -d > $cluster-ca.pem
```

- Change the kubeconfig file to include the extracted CA certificate. For example:

```
[virtwho-host]$ cat /root/.kube/config
apiVersion: v1
clusters:
- cluster:
  server: https://api.testcluster.example.org:6443
  certificate-authority: /root/testcluster-ca.pem
  name: api-testcluster-example-org:6443
contexts:
```

```
- context:  
  cluster: api-test-cluster-example-org:6443  
  namespace: default
```

10. Before starting the service, you can test the configuration manually:

```
[virtwho-host]# virt-who --print
```



NOTE

If the **jq** program is installed, you can use it to make the output easier to read: **# virt-who --print | jq**

1. Enable the virt-who service:

```
[virtwho-host]# systemctl enable virt-who
```

2. Restart the virt-who service to use the new configuration.

```
[virtwho-host]# systemctl restart virt-who
```

Virt-who logs are available in **/var/log/rhsm/rhsm.log**. In this file, you can view configuration or connectivity errors.

CHAPTER 5. REGISTERING VIRTUAL MACHINES TO USE A HOST-BASED SUBSCRIPTION

Register virtual machines with auto-attach so that they inherit a subscription from their hypervisor.

Prerequisites

- Attach a host-based subscription to the virtual machine's hypervisor.
- Configure virt-who to query the virtual machine's hypervisor.
- Ensure that all hypervisors the virtual machine can migrate to have host-based subscriptions attached and report to virt-who, or limit the virtual machine's migration to specific hypervisors.

Web UI procedure

1. Log in to the Customer Portal at <https://access.redhat.com>.
2. Navigate to **Subscriptions** > **Systems** and click the name of the virtual machine.
3. Click the **Subscriptions** tab.
4. Click **Run Auto-Attach**.

Repeat these steps for each virtual machine.

CLI procedure

1. Register the virtual machine using the auto-attach option:

```
# subscription-manager register --auto-attach
```

2. When prompted, enter your user name and password.

Repeat these steps for each virtual machine.

If the virtual machine has already been reported by virt-who, the virtual machine inherits a subscription from its hypervisor.

If the virtual machine has not been reported by virt who, the virtual machine receives a temporary subscription while Subscription Management waits for virt-who to provide information about which hypervisor the virtual machine is running on. After virt-who provides this information, the virtual machine inherits a subscription from its hypervisor.

CHAPTER 6. VIRT-WHO TROUBLESHOOTING METHODS

Verifying virt-who status

Verify the status of the virt-who service:

```
# systemctl status virt-who.service
```

Debug logging

Check the `/var/log/rhsm/rhsm.log` file, where virt-who logs all its activity by default.

For more detailed logging, enable the debugging option in the `/etc/virt-who.conf` file:

```
[global]
debug=True
```

Restart the virt-who service for the change to take effect.

When the underlying issue is resolved, modify the `/etc/virt-who.conf` file to disable debugging, then restart the virt-who service.

Testing configuration options

Make a change and test the result, repeating as needed. Virt-who provides three options to help test the configuration files, credentials, and connectivity to the virtualization platform:

- The **virt-who --one-shot** command reads the configuration files, retrieves the list of virtual machines and sends it to the subscription management system, such as Red Hat Satellite or Red Hat Subscription Management, then exits immediately.
- The **virt-who --print** command reads the configuration files and prints the list of virtual machines, but does not send it to the subscription management system.
- Starting with RHEL 9 Beta, the **virt-who --status** command reads the configuration files and outputs a summary of the connection status for both the source and destination systems.
 - The **virt-who --status** command with the **--json** option provides additional connectivity data, in JSON format, for each configuration.

The expected output of the **virt-who --one-shot** and **virt-who --print** commands is a list of hypervisors and their virtual machines, in JSON format. The following is an extract from a VMware vSphere instance. The output from all hypervisors follows the same structure.

```
{
  "guestId": "422f24ed-71f1-8ddf-de53-86da7900df12",
  "state": 5,
  "attributes": {
    "active": 0,
    "virtWhoType": "esx",
    "hypervisorType": "vmware"
  }
},
```

The expected output for the **virt-who --status** command is a plain-text summary of the connection status for each configuration in virt-who.

```

+-----+
      Configuration Status
+-----+
Configuration Name: esx_config1
Source Status: success
Destination Status: success

Configuration Name: hyperv-55
Source Status: failure
Destination Status: failure

```

The expected output for the **virt-who --status** command with the **--json** option provides additional information about each configuration, including its last successful run, in JSON format. This output also includes details about the success or failure status of each configuration.

- When the status report indicates a configuration success, the JSON output includes the number of hypervisors and guests that virt-who reported during its last successful run cycle.
- When the status report indicates a configuration failure, the JSON output includes the associated error message.

```

"configurations": [
  {
    "name":"esx-conf1",
    "source":{
      "connection":"https://esx_system.example.com",
      "status":"success",
      "last_successful_retrieve":"2020-02-28 07:25:25 UTC",
      "hypervisors":20,
      "guests":37
    },
    "destination":{
      "connection":"candlepin.example.com",
      "status":"success",
      "last_successful_send":"2020-02-28 07:25:27 UTC",
      "last_successful_send_job_status":"FINISHED"
    }
  },
  {
    "name":"hyperv-55",
    "source":{
      "connection":"windows10-3.company.com",
      "status":"failure",
      "message":"Unable to connect to server: invalid credentials",
      "last_successful_retrieve":null
    },
    "destination":{
      "connection":"candlepin.company.com",
      "status":"failure",
      "message":"ConnectionRefusedError: [Errno 111] Connection refused",
      "last_successful_send":null,
      "last_successful_send_job_status":null
    }
  }
]
}

```

The **virt-who --status** command can also be used with the **--debug** and **--config** options to provide additional information about the configuration files.

Identifying issues when using multiple virt-who configuration files

If you have multiple virt-who configuration files on one server, move one file at a time to a different directory while testing after each file move. If the issue no longer occurs, the cause is associated with the most recently moved file. After you have resolved the issue, return the virt-who configuration files to their original location.

Alternatively, you can test an individual file after moving it by using the **--config** option to specify its location. For example:

```
# virt-who --debug --one-shot --config /tmp/conf_name.conf
```

Starting with RHEL 9 Beta, you can enter **virt-who --status** with the **--debug** and **--config** options to identify the configuration file causing the issue without removing any other files from the directory. For example:

```
#virt-who --debug --status --config /tmp/conf_name.conf
```

You can also enter the command with the **--json** option to view more detailed information about each configuration in JSON format. For example:

```
#virt-who --debug --status --json --config /tmp/conf_name.conf
```

Identifying duplicate hypervisors

Duplicate hypervisors can cause subscription and entitlement errors. Enter the following commands to check for duplicate hypervisors:

```
# systemctl stop virt-who
# virt-who -op >/tmp/virt-who.json
# systemctl start virt-who
# cat /tmp/virt-who.json | json_reformat | grep name | sort | uniq -c | sort -nr | head -n10
3  "name": "localhost"
1  "name": "rhel1.example.com"
1  "name": "rhel2.example.com"
1  "name": "rhel3.example.com"
1  "name": "rhel4.example.com"
1  "name": "rhvh1.example.com"
1  "name": "rhvh2.example.com"
1  "name": "rhvh3.example.com"
1  "name": "rhvh4.example.com"
1  "name": "rhvh5.example.com"
```

In this example, three hypervisors have the same FQDN (**localhost**), and must be corrected to use unique FQDNs.

Identifying duplicate virtual machines

Enter the following commands to check for duplicate virtual machines:

```
# systemctl stop virt-who
```



```
# virt-who -op >/tmp/virt-who.json  
# systemctl start virt-who  
# cat /tmp/virt-who.json | json_reformat | grep "guestId" | sort | uniq -c | sort -nr | head -n10
```

Checking the number of hypervisors

Enter the following commands to check the number of hypervisors virt-who currently reports:

```
# systemctl stop virt-who  
# virt-who -op >/tmp/virt-who.json  
# systemctl start virt-who  
# cat /tmp/virt-who.json | json_reformat | grep name | sort | uniq -c | wc -l
```

Starting with RHEL 9 Beta, enter the following command to check the number of hypervisors that virt-who reported during its last successful run cycle:

```
# virt-who --status --json
```

Checking the number of virtual machines

Enter the following commands to check the number of virtual machines that virt-who currently reports:

```
# systemctl stop virt-who  
# virt-who -op >/tmp/virt-who.json  
# systemctl start virt-who  
# cat /tmp/virt-who.json | json_reformat | grep "guestId" | sort | uniq -c | wc -l
```

Starting with RHEL 9 Beta, enter the following command to check the number of guests that virt-who reported during its last successful run cycle:

```
# virt-who --status --json
```

CHAPTER 7. VIRT-WHO TROUBLESHOOTING SCENARIOS

Virt-who fails to connect to the virtualization platform

If virt-who fails to connect to the hypervisor or virtualization manager, check the Red Hat Subscription Manager log file `/var/log/rhsm/rhsm.log`. If you find the message **No route to host**, the hypervisor might be listening on the wrong port. In this case, modify the virt-who configuration file for that hypervisor and append the correct port number to the **server** value.

You must restart the virt-who service after modifying a configuration file.

Virt-who fails to connect to the virtualization platform through an HTTP proxy on the local network

If virt-who cannot connect to the hypervisor or virtualization manager through an HTTP proxy, either configure the proxy to allow local traffic to pass through, or modify the virt-who service to use no proxy by adding the following line to `/etc/virt-who.conf`:

```
[system_environment]
no_proxy=*
```

You must restart the virt-who service after modifying a configuration file.



NOTE

The section `[system_environment]` is only supported from virt-who-0.30.x-1.el8 (RHEL 8.4), if you are using the old virt-who version, please set `NO_PROXY` by `/etc/sysconfig/virt-who`.