

The Zenoss Enablement Series:

## About Resource Manager 5.0 Data Stores

---

Document Version 1.1

Zenoss, Inc.

[www.zenoss.com](http://www.zenoss.com)

Copyright © 2014-2015 Zenoss, Inc. 11305 Four Points Drive, Bldg. 1 - Suite 300, Austin, Texas 78726, U.S.A.  
All rights reserved.

Zenoss and the Zenoss logo are trademarks or registered trademarks of Zenoss, Inc. in the United States and other countries. All other trademarks, logos, and service marks are the property of Zenoss or other third parties. Use of these marks is prohibited without the express written consent of Zenoss, Inc. or the third-party owner.

Cisco, Cisco UCS, Cisco Unified Computing System, Cisco Catalyst, and Cisco Nexus are trademarks or registered trademarks of Cisco and/or its affiliates in the United States and certain other countries.

Flash is a registered trademark of Adobe Systems Incorporated.

Oracle, the Oracle logo, Java, and MySQL are registered trademarks of the Oracle Corporation and/or its affiliates. Linux is a registered trademark of Linus Torvalds.

SNMP Informant is a trademark of Garth K. Williams (Informant Systems, Inc.). Sybase is a registered trademark of Sybase, Inc.

Tomcat is a trademark of the Apache Software Foundation.

vSphere is a trademark of VMware, Inc. in the United States and/or other jurisdictions.

Windows is a registered trademark of Microsoft Corporation in the United States and other countries. All other companies and products mentioned are trademarks and property of their respective owners.

# Table of Contents

<b>Resource Manager Version 5.0 Data Stores.....</b>	<b>1</b>
<b>Applies To.....</b>	<b>1</b>
<b>Introduction .....</b>	<b>1</b>
Europa Data Categories .....	1
Default Storage Path for Persistent Container Data.....	1
<b>Auditing Containers for Persistent File Stores.....</b>	<b>2</b>
<b>Internal Services: Persistent Data Stores .....</b>	<b>2</b>
<b>Resource Manager Data.....</b>	<b>3</b>



# Resource Manager Version 5.0 Data Stores

## Applies To

The procedure outlined in this document applies to the following Zenoss version:

- Zenoss “Europa” Beta 3 (Control Center v1.0 and Resource Manager v5.0)

## Introduction

**Note:** For the purposes of clarity, this document will refer to the combination of Zenoss Control Center Version 1.0 and Resource Manager Version 5.0 as “Europa,” the code name for the current Beta 3 release.

In general, most data stored by Europa – both by Control Center and Resource Manager – is stored in the `/opt` and `/var` directories. Although not deliberately obfuscated, the location of Resource Manager and Control Center data on Europa hosts is not revealed by the Europa user interface. Europa uses bind mounts on the host(s) and within the Docker containers to make the replication of containers from a single image seamless and this architecture can make it difficult for administrators to locate data stores. This article explores the various Europa data stores, their locations, and the methods administrators can use to locate them on any given Europa deployment.

## Europa Data Categories

In general, the data stored by Europa falls into two categories:

- Application data such as templates, Docker images, etc.
- Data stored by Resource Manager containers pertaining to the performance of monitored infrastructure.

Although it's an over simplification, for the purposes of this document we refer to these two categories of data as “Internal Services” data and “Managed Resource” data, respectively. In at least one case, the categorization is imperfect; although the *OpenTSDB* instance is an internal service, it hosts managed resource performance metrics when Resource Manager is started. Considering the categories separately is useful because the administrators' means for auditing the data stores differs between Internal Services and Resource Manager data.

### A Note about an Important Host Bind Mount

On Europa hosts, `/opt/serviced/var` is bind mounted on `/exports/serviced_var/`. On a single host Europa deployment, you will find that the default naming convention for the host directories containing persistent container data takes the form:

```
/opt/serviced/var/volumes/[top level service ID]/
```

...where the top level service ID corresponds to a top level service such as “Resource Manager.” Methods for confirming this default location on a particular Europa instance are detailed below.

## Default Storage Path for Persistent Container Data

In the `/etc/default/serviced` file, Europa sets the environmental variable `SERVICED_VARPATH` to define where persistent data is stored by both Internal Services and Resource Manager containers. Unless edited to be otherwise, the path is:

```
/opt/serviced/var/
```

When a Resource Manager service lists only a short name for its data store in its service definition, that short name completes the path:

```
SERVICED_VARPATH/volumes/[top level service ID]/
```

### A Note about Auditing Folder Sizes on Europa Hosts

When Control Center or Resource Manager containers are running, the `/var` directory might appear to contain many gigabytes of data if a command such as `du -h -max-depth=1` is run to determine its size. Docker containers create virtual file systems within `/var`, that appear to be large when audited using such a command. Running the `du` command with the `'-xmh'` flag set provides a more accurate result of storage levels in any given directory because the `-xm` flags will result in an output that ignores files on “different” [remote] file systems.

## Auditing Containers for Persistent File Stores

Administrators who want to orient themselves to the manner in which Europa data stores are presented to the Docker containers might find themselves confused if they attach to a running container and attempt to identify its persistent data stores using standard command line tools. This is because the filesystem presented to each container includes a large number of mounts, none of which need necessarily be mounted to a persistent data store. Instead, administrators should examine a service's “service definition” to discern if (and where) the service retains persistent data. For Resource Manager services, the command for doing so is run on the host (as opposed to within a container) and takes the following form:

```
serviced service list [service name] | less
```

The command results in an output with a complete service definition containing comprehensive information about how the service operates. The “Volumes” section of this document contains information on any persistent data stores. The two pertinent lines of this section are *ResourcePath* and *ContainerPath*, that correspond respectively to the location of the data store on the host system and the location of the data store as it is presented to the processes running with the container. On a hypothetical single server Europa deployment, we might find values of “mariadb” for the *ResourcePath* and a value of “/var/lib/mysql” for *ContainerPath* in the definition of the mariadb service. In such an example, where the default `SERVICED_VARPATH` variable has been retained, the “mariadb” value for *ResourcePath* corresponds to the “mariadb” directory found in:

```
/opt/serviced/var/volumes/[top level service ID]/
```

For example:

```
/opt/serviced/var/volumes/78s20bmerf94qe3xw00pjzmq6/mariadb
```

In this example that host directory has been bind mounted on the `/var/lib/mysql/` directory of the mariadb container.

For Internal Services, a different command must be used to examine the definition:

```
docker inspect [container ID]
```

...where the container ID can be found by running the command `dockers ps` and looking up the desired container's ID.

## Internal Services: Persistent Data Stores

When `Serviced` starts, the first persistent data stores will be created and maintained in the `/opt/serviced/` directory.

Note that although the OpenTSDB datastore is set up to host performance metrics for resources managed by Resource Manager, it is listed in this section as Internal Services data. This is because (i) this container starts

and runs as part of the Internal Services application and (ii) the means of auditing the OpenTSDB location is the same as for other Internal Services containers.

Container	Purpose	Host Data Store Location
Logstash	Storage of Resource Manager logs	/opt/serviced/isvcs/resources
Elasticsearch-serviced	Index of service definitions	/opt/serviced/var/isvcs/elasticsearch-serviced/data and /opt/serviced/isvcs/resources
Elasticsearch-logstash	Index of Resource Manager logs	/opt/serviced/var/isvcs/elasticsearch-logstash/data and /opt/serviced/isvcs/resources
Docker-registry	Hosting /delivery of docker images	/opt/serviced/var/isvcs/docker-registry/registry and /opt/serviced/isvcs/resources
Opentsdb	Storage of managed resource performance metrics	/opt/serviced/var/isvcs/opentsdb/hbase and /opt/serviced/isvcs/resources
Zookeeper	Coordination of information between Resource Manager services	/opt/serviced/var/isvcs/zookeeper/data and /opt/serviced/isvcs/resources
Celery	Job scheduling	/opt/serviced/var/isvcs/celery/celery and /opt/serviced/isvcs/resources

## Resource Manager Data

When Resource Manager starts, the following services (Docker containers) start. When 'None' is listed under "Host Data Store Location," the service either produces no persistent data, or connects to other services that in turn store persistent data. For those services that directly store persistent data, the directory listed would complete the path `SERVICED_VARPATH/[top level service ID]/` on the host system.

Service	Host Data Store Location
Hbase RegionServer	hbase-master
Hmaster	hbase-master
Zookeeper	hbase-zookeeper-{{ plus 1 .InstanceID }}
mariadb	mariadb
Rabbitmq	rabbitmq
Zencatalogservice	zencatalogservice
Zeneventserver	zeneventserver
Zenjobs	zenjobs
zope	Zenjobs
zminion (that has ssh)	zenoss-custom-patches-pc, zenoss-custom-patches, zenoss-var-ext