

Moab HPC Suite

Installation and Configuration Guide 9.0.1

Released: March 2016; Revised: June 13, 2016



© 2016 Adaptive Computing Enterprises, Inc. All rights reserved.

Distribution of this document for commercial purposes in either hard or soft copy form is strictly prohibited without prior written consent from Adaptive Computing Enterprises, Inc.

Adaptive Computing, Cluster Resources, Moab, Moab Workload Manager, Moab Viewpoint, Moab Cluster Manager, Moab Cluster Suite, Moab Grid Scheduler, Moab Grid Suite, Moab Access Portal, and other Adaptive Computing products are either registered trademarks or trademarks of Adaptive Computing Enterprises, Inc. The Adaptive Computing logo and the Cluster Resources logo are trademarks of Adaptive Computing Enterprises, Inc. All other company and product names may be trademarks of their respective companies.

Adaptive Computing Enterprises, Inc.

1712 S. East Bay Blvd., Suite 300

Provo, UT 84606

+1 (801) 717-3700

www.adaptivecomputing.com



Scan to open online help

| | |
|---|------------|
| Welcome | 1 |
| Chapter 1 Planning Your Installation | 3 |
| Server Hardware Requirements | 4 |
| Component Requirements | 9 |
| Identify The Manual Or RPM Installation Methods | 20 |
| Chapter 2 Manual Installation | 21 |
| Manual Installation | 22 |
| Preparing For Manual Installation | 22 |
| Installing Torque Resource Manager | 24 |
| Installing Moab Workload Manager | 33 |
| Installing Moab Accounting Manager | 39 |
| Installing Moab Web Services | 54 |
| Installing Moab Insight | 67 |
| Installing Moab Viewpoint | 68 |
| Installing RLM Server | 68 |
| Nitro Integration | 72 |
| Preparing For Nitro Manual Installation | 72 |
| Installing Nitro | 73 |
| Installing Nitro Web Services | 77 |
| Additional Configuration | 86 |
| Configuring SSL In Tomcat | 86 |
| Setting Up OpenLDAP On CentOS 6 | 86 |
| Moab Workload Manager Configuration Options | 93 |
| Moab Accounting Manager Configuration Options | 94 |
| Trusting Servers In Java | 96 |
| Manual Upgrade | 98 |
| Preparing For Upgrade | 98 |
| Upgrading Torque | 99 |
| Upgrading Moab Workload Manager | 105 |
| Upgrading Moab Accounting Manager | 108 |
| Upgrading Moab Web Services | 112 |
| Migrating The MAM Database From MySQL To PostgreSQL | 119 |
| Chapter 3 RPM Installation Method | 122 |
| About RPM Installations And Upgrades | 123 |
| Preparing The Host – Typical Method | 125 |
| Creating The Moab-offline Tarball | 127 |
| Preparing The Host – Offline Method | 130 |
| RPM Installations | 132 |
| Installing Torque Resource Manager | 132 |
| Installing Moab Workload Manager | 137 |
| Installing Moab Accounting Manager | 142 |

| | |
|---|------------|
| Installing Moab Web Services | 154 |
| Installing Moab Insight | 165 |
| Installing Moab Viewpoint | 178 |
| Installing RLM Server | 195 |
| Installing Remote Visualization | 198 |
| Nitro Integration | 217 |
| Installing Nitro | 218 |
| Installing Nitro Web Services | 222 |
| Additional Configuration | 231 |
| Configuring SSL In Tomcat | 231 |
| Setting Up OpenLDAP On CentOS 6 | 231 |
| Trusting Servers In Java | 238 |
| RPM Upgrades | 240 |
| Upgrading Torque Resource Manager (RPM) | 240 |
| Upgrading Moab Workload Manager (RPM) | 243 |
| Upgrading Moab Accounting Manager (RPM) | 247 |
| Upgrading Moab Web Services (RPM) | 250 |
| Upgrading Moab Insight (RPM) | 253 |
| Upgrading Moab Viewpoint (RPM) | 256 |
| Migrating The MAM Database From MySQL To PostgreSQL | 263 |
| Chapter 4 Troubleshooting | 266 |
| General Issues | 266 |
| Moab Web Services Issues | 269 |
| Moab Viewpoint Issues | 273 |

Welcome

Welcome to the Moab HPC Suite 9.0.1 Installation and Configuration Guide, which will help you install or upgrade and configure your Moab HPC Suite. This guide includes detailed instructions for installing each component of the suite so that you can quickly get up and running.

This guide is intended for system administrators who are responsible for installing the Moab HPC Suite components.

i Depending on your system configuration and license, not all of the HPC Suite components may be available.

The Moab HPC Suite 9.0.1 contains the following components:

- Torque Resource Manager 6.0.1
- Moab Workload Manager 9.0.1
- Moab Accounting Manager 9.0.1
- Moab Web Services 9.0.1
- Moab Insight 9.0.1
- Moab Viewpoint 9.0.1
- Remote Visualization 9.0.1
- Nitro and Nitro Web Services 2.0
- Reprise License Manager 12

Before commencing the installation or upgrade, please see [Chapter 1 Planning your Installation on page 3](#) to verify your system conforms to minimum prerequisites.

Chapter 1 Planning your Installation

 It is highly recommended that you *first* perform installations and upgrades in a *test environment*. Standard installation and upgrade procedures and use cases are tested prior to release. However, due to the wide range of possible configurations and customizations, it is important to exercise caution when deploying new versions of software into your production environments. This is especially true when the workload has vital bearing on your organization's day-to-day operations. We recommend that you test in an environment that mirrors your production environment's configuration, workflow and load as closely as possible. Please contact your Adaptive Computing account manager for suggestions and options for installing/upgrading to newer versions.

There are many different ways to install and configure the Moab HPC Suite. Each environment has its own set of requirements and preferences. This chapter is intended to help an administrator understand how each of the Moab HPC Suite components interact, basic requirements and configuration information to prepare for the installation.

 Code samples have been provided for convenience. Some code samples provide sample passwords (i.e. "changeme!"). We strongly recommend that you do not use these passwords during installation, as using the documented passwords could introduce unnecessary security vulnerabilities into your system.

In this chapter:

- [Installation Terminology on page 3](#)
- [Where to Start on page 4](#)
- [Server Hardware Requirements on page 4](#)
- [Component Requirements on page 9](#)
- [Identify the Manual or RPM Installation Methods on page 20](#)

Installation Terminology

To aid in documentation clarity, Adaptive Computing uses the following terms in this Installation and Configuration Guide:

- **Components** – The different "products" included in the Moab HPC Suite. For example, Moab Workload Manager, Moab Web Services.

- Servers – Also known as components, but specifically relating to the actual services. For example, the Moab Workload Manager component is referred to as the Moab Server for non-client services.
- Host – The actual box where an Moab HPC Suite component (server or client) is installed.

i Previous documentation typically used Head Node to designate a host or a Server.

Where to Start

You will need to plan your environment and determine how many hosts you will need and for which you components you will install using the Manual Installation or the RPM Installation method. The following are suggested steps to help you in your planning and installing process.

1. Determine whether you have a small, medium, High-Throughput or large environment; including an example, and required and recommended hardware requirements. See [Server Hardware Requirements on page 4](#).
2. Decide whether you will perform a Manual Installation or an RPM Installation for the various components. See [Identify the Manual or RPM Installation Methods on page 20](#).

i The Manual Installation and the RPM Installation chapters each have an "Additional Configuration" section that provides additional information and instructions for optional, but recommended configurations (for example, Configuring SSL in Tomcat).

3. Review the software requirements for your components and set up your hosts accordingly. See [Component Requirements on page 9](#).
4. Install the individual components on their respective host(s). See [Preparing for Manual Installation on page 22](#) or [Preparing the Host – Typical Method on page 125](#) as applicable.
5. Refer to [Chapter 4 Troubleshooting on page 266](#) for assistance in addressing common problems during installation and configuration.

Server Hardware Requirements

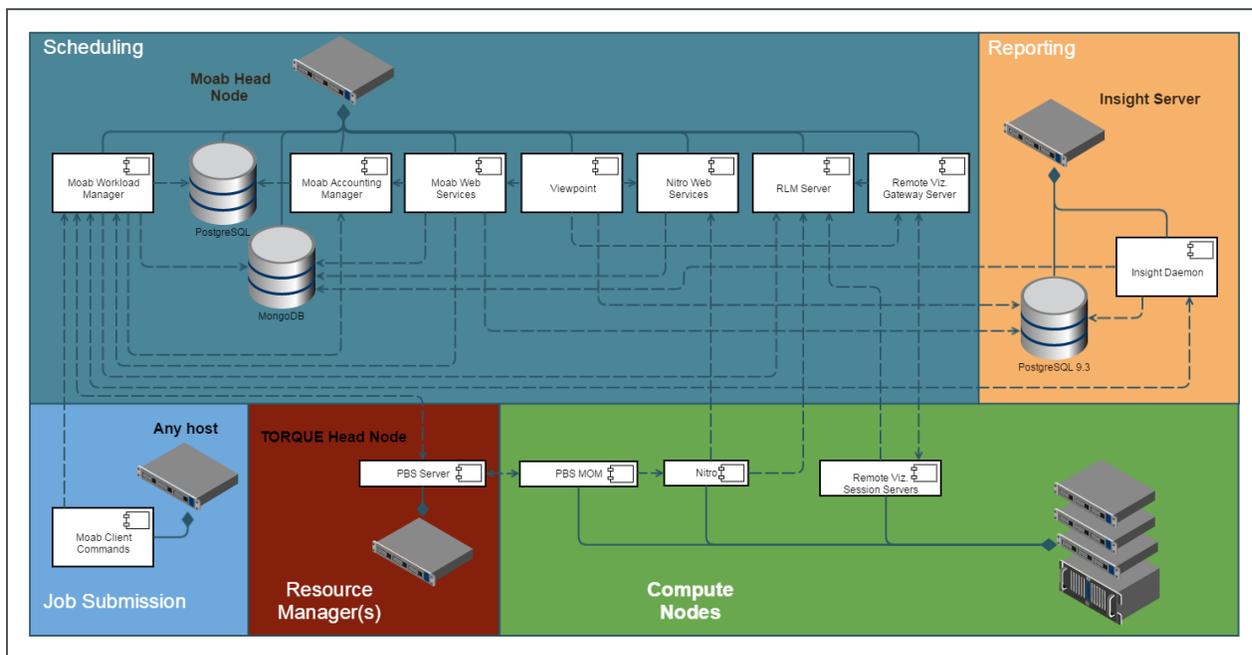
The Moab HPC Suite is installed and configured differently for small, medium or large environment types. This topic provides a general topology of the Moab HPC Suite and the server hardware requirements depending on your environment size.

In this topic:

- [Topology on page 5](#)
- [Hardware Requirements on page 5](#)

Topology

The following diagram provides a general topology of the Moab HPC Suite for a medium (with high throughput) or a large environment.



Please note the following:

- Smaller environments may elect to consolidate the Torque Server with the Moab Server on the same host, including PBS Server in the list of components installed on the same host.
- Although Moab Workload Manager and Moab Accounting Manager may share the same database instance, it is not a requirement. Two database instances may be used, one for each component.
- Larger systems will require more dedicated resources for each component, in which case it may be necessary to move individual components from the Moab Server Host (i.e. databases, Moab Accounting Manager, and/or Viewpoint) to their own respective servers.

Hardware Requirements

The following table identifies the minimum and recommended hardware requirements for the different environment types. Use this table as a guide when planing out your suite topology.

i Software requirements are listed per-component rather than suite-wide as the suite components reside on different hosts. See [Component Requirements on page 9](#).

| Environment Type | # of Compute Nodes | Jobs/Week | Minimum Requirements (per Host Distribution) | Recommended Requirements (targeting minimum number of hosts) |
|-------------------------------|--------------------|-----------|--|--|
| Proof of Concept / Small Demo | 50 | <1k | Moab Server+Torque Server Host <ul style="list-style-type: none"> • 4 Intel/AMD x86-64 cores • At least 8 GB RAM • At least 100 GB dedicated disk space Insight Server Host <ul style="list-style-type: none"> • 4 Intel/AMD x86-64 cores • At least 8 GB RAM • At least 256 GB dedicated disk space | Same as minimum |

| Environment Type | # of Compute Nodes | Jobs/Week | Minimum Requirements (per Host Distribution) | Recommended Requirements (targeting minimum number of hosts) |
|------------------|--------------------|-----------|--|---|
| Medium | 500 | <100k | <p>Moab Server+Torque Server Host</p> <ul style="list-style-type: none"> • 8 Intel/AMD x86-64 cores • At least 16 GB RAM • At least 512 GB dedicated disk space <p>Insight Server Host</p> <ul style="list-style-type: none"> • 8 Intel/AMD x86-64 cores • At least 8 GB of RAM; a dedicated 1 Gbit channel between Insight and Moab • 128 GB local SSD for swapAt least 512 GB disk | <p>Moab Server+Torque Server Host</p> <ul style="list-style-type: none"> • 16 Intel/AMD x86-64 cores • At least 32 GB RAM • At least 1 TB dedicated disk space <p>Insight Server Host</p> <ul style="list-style-type: none"> • 8 Intel/AMD x86-64 cores • At least 8 GB of RAMA dedicated 1 Gbit channel between Insight and Moab • 128 GB local SSD for swapAt least 512 GB disk |

| Environment Type | # of Compute Nodes | Jobs/Week | Minimum Requirements (per Host Distribution) | Recommended Requirements (targeting minimum number of hosts) |
|---------------------------------------|--------------------|-----------|---|---|
| Medium with High Throughput or Larger | >500 | >100k | <p>Moab Server Host</p> <ul style="list-style-type: none"> • 8 Intel/AMD x86-64 cores • At least 16 GB RAM • At least 512 GB dedicated disk space <p>Torque Server Host</p> <ul style="list-style-type: none"> • 8 Intel/AMD x86-64 cores • At least 16 GB RAM • At least 512 GB dedicated disk space <p>Insight Server Host</p> <ul style="list-style-type: none"> • 8 Intel/AMD x86-64 cores • At least 16 GB of RAM; a dedicated 1 Gbit channel between Insight and Moab • 128 GB local SSD for swap • At least 512 GB disk | <p>The Moab Server should <i>not</i> reside on the same host as the Torque Server.</p> <p>MWS Server <i>must</i> reside on the same host as the Moab Server (Moab Server Host).</p> <p>The MAM Server may reside on its own host, on the Moab Host (preferred), or another server's host (except for the Insight Host).</p> <p>The Viewpoint Server may reside on its own host, on the Moab Server Host (preferred), or another server's host (except for the Insight Server Host).</p> <p>Databases may also reside on the same or a different host from its server component.</p> |

Please note the following:

- All requirements above (minimum and recommended) target a minimum number of management servers. Administrators are encouraged to separate the Torque Server and the Moab Server onto different hosts where possible for better results; especially when High Throughput is enabled.
- Although many factors may have an impact on performance (network bandwidth, intended use and configuration, etc.), we consider High

Throughput as something that makes a significant enough difference between minimum and recommended hardware requirements to merit mention in the table above.

- Moab and Torque are both multi-threaded and perform better with more processors.
- Due to the large amount of data Moab must send to Insight, Moab performs better without Insight enabled (for environments that do not require Viewpoint, or use Crystal Reporting).
- Regarding disk space, consideration should be given to requirements related to log files, log depth, number of jobs/nodes/reservations (more objects impact database journal size), average number of events generated (more events take more space), etc.

Component Requirements

This topic provides the various software requirements and dependencies for the suite components (servers).

i On RHEL systems, you must be registered for a Red Hat subscription in order to have access to required RPM package dependencies.

i On SLES systems, you must be registered for a SUSE Linux Enterprise subscription in order to have access to required RPM package dependencies.

In this topic:

- [Torque on page 10](#)
- [Moab Workload Manager on page 11](#)
- [Moab Accounting Manager on page 12](#)
- [Moab Web Services on page 13](#)
- [Moab Insight on page 14](#)
- [Moab Viewpoint on page 16](#)
- [RLM Server on page 16](#)
- [Remote Visualization on page 17](#)
- [Nitro on page 18](#)
- [Nitro Web Services on page 19](#)

Torque

⚠ If you intend to use Torque 6.0 with Moab Workload Manager, you must run Moab version 9.0 or 8.0 or later. Torque 6.0 will not work with versions earlier than Moab 8.0.

In this section:

- [Supported Operating Systems on page 10](#)
- [Software Requirements on page 10](#)

Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11, 12

Software Requirements

- libxml2-devel package (package name may vary)
- openssl-devel package (package name may vary)
- Tcl/Tk version 8 or later if you plan to build the GUI portion of Torque or use a Tcl-based scheduler
- cpusets and cgroups

- NUMA-awareness uses cgroups, which include cpusets. Red Hat systems must use libcgroup version 0.40.rc1-16.el6 or later; SUSE systems need to use a comparative libcgroup version.

i NUMA-awareness is *not* available for Cray systems.

- cpusets: libhwloc 1.9.1 is the minimum supported, however NVIDIA K80 requires libhwloc 1.11.0. If you need to install libhwloc and the corresponding hwloc-devel package, see [Linux Cpuset Support](#) in the *Torque Resource Manager Administrator Guide*.

i If `--enable-cgroups` is specified, `--enable-cpuset` is ignored.

i If you are building with cgroups enabled, you must have boost version 1.41 or later.

- if you build Torque from source (i.e. clone from github), the following additional software is required:

- gcc
- gcc-c++
- posix-compatible version of make
- libtool 1.5.22 or later
- boost-devel 1.36.0 or later

 Red Hat 6-based systems come packaged with 1.41.0 and Red Hat 7-based systems come packaged with 1.53.0. If needed, use the `--with-boost-path=DIR` option to change the packaged boost version. See [Customizing the Install](#) in the *Torque Resource Manager Administrator Guide*.

Moab Workload Manager

In this section:

- [Supported Operating Systems on page 11](#)
- [Software Requirements on page 11](#)
- [Supported Resource Managers on page 12](#)

Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11, 12

 A SUSE 11- based OS is *only* supported for Moab Server if your configuration does *not* include MWS.

Software Requirements

- libcurl (<http://curl.haxx.se/libcurl/>)
- Perl 5.8.8 or later
- perl-CPAN (package name may vary)
- libxml2-devel (package name may vary)
- *(Optional)* Moab Accounting Manager 8.1
- *(Optional)* MySQL, PostgreSQL, or Oracle with ODBC driver (see [Database Configuration](#) in the *Moab Workload Manager Administrator Guide* for details)

Supported Resource Managers

- Torque 5.1
- SLURM

Moab Accounting Manager

i MAM is commonly installed on the same host as Moab Workload Manager; however, in some cases you might obtain better performance by installing them on different hosts.

In this topic:

- [Supported Operating Systems on page 12](#)
- [Software Requirements on page 12](#)
- [Depends On \(not necessarily on the same host\) on page 12](#)

Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11, 12

Software Requirements

- gcc
- perl-suidperl
- httpd
- mod_ssl
- rrdtool
- Moab Workload Manager 9.0.1
- Perl modules; see [Installing Moab Accounting Manager on page 39 \(Manual Installation\)](#) [Installing Moab Accounting Manager on page 142 \(RPM Installation\)](#) for more details

Depends On (not necessarily on the same host)

MAM uses an RDBMS as a back end.

- PostgreSQL 7.2 or later

Adaptive Computing recommends that the database used by MAM does *not* reside on the same host as the database used by Insight. However, if you

choose to install the MAM PostgreSQL database on the *same* host where the Insight PostgreSQL database, then the MAM PostgreSQL database *must* be version 9.3.

Moab Web Services



MWS Server *must* reside same host as Moab Server (Moab Server Host).

In this topic:

- [Supported Operating Systems on page 13](#)
- [Software Requirements on page 13](#)
- [Depends On \(not necessarily on the same host\) on page 13](#)

Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 12

Software Requirements

- Moab Workload Manager 9.0.1
- Oracle® Java® 8 Runtime Environment



Oracle Java 8 Runtime Environment is the recommended Java environment, but Oracle Java 7 is also supported. All other versions of Java, including OpenJDK/IcedTea, GNU Compiler for Java, and so on cannot run Moab Web Services.

- Apache Tomcat™ 7

Depends On (not necessarily on the same host)

- OpenLDAP *or* PAM; see [Installing Moab Web Services on page 54](#) (Manual Installation) [Installing Moab Web Services on page 154](#) (RPM Installation) for more details
- MongoDB® 2.4.x

Moab Insight

 Moab Workload Manager and Insight both tend to heavily consume system resources. The Insight Server and the Moab Server *must* run on different hosts.

 Only an RPM-based installation is supported for installing Moab Insight.

In this section:

- [Supported Operating Systems on page 14](#)
- [Software Requirements on page 14](#)
- [Depends On \(not on the same host\) on page 14](#)
- [Performance Benchmarks on page 14](#)

Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 12

Software Requirements

- Oracle® Java® 8 Runtime Environment

 Oracle Java 8 Runtime Environment is the recommended Java environment, but Oracle Java 7 is also supported. All other versions of Java, including OpenJDK/IcedTea, GNU Compiler for Java, and so on cannot run Moab Web Services.

Depends On (not on the same host)

- Moab Workload Manager 9.0.1
- MongoDB 2.4.x
- PostgreSQL 9.3 or later

Performance Benchmarks

Adaptive Computing has tested and certified Insight's scale and performance under the following server configuration and load scenarios.

Server Configuration

Host hardware: 8 core AMD Opteron 6320 2.8 GHz servers, with 32GB of ram and a 500GB WD Blue hard drive

Installed services: Moab Workload Manager, Moab Web Services, Moab Insight, Moab Viewpoint (all at version 9.0.0 and running on the same host)

i The benchmarks were ran with multiple services on a single host to benchmark Insight under very aggressive working conditions. Moab Insight must be installed on its own host.

Load Scenarios

| Jobs in queue | Avg Job Duration | Avg job Size (ppn) | Number of Nodes | Procs per Node | Avg Jobs per Week |
|---------------|------------------|--------------------|-----------------|----------------|-------------------|
| 1000 | 200 | 32 | 500 | 32 | 25200 |
| 1000 | 60 | 32 | 500 | 32 | 84000 |
| 1000 | 10 | 32 | 500 | 32 | 504000 |
| 1000 | 200 | 16 | 6384 | 16 | 321754 |
| 1000 | 60 | 16 | 6384 | 16 | 1072512 |
| 1000 | 10 | 16 | 6384 | 16 | 6435072 |
| 10000 | 200 | 32 | 500 | 32 | 25200 |
| 10000 | 60 | 32 | 500 | 32 | 84000 |
| 10000 | 10 | 32 | 500 | 32 | 504000 |
| 10000 | 200 | 16 | 6384 | 16 | 321754 |
| 10000 | 60 | 16 | 6384 | 16 | 1072512 |
| 25000 | 200 | 32 | 500 | 32 | 25200 |
| 25000 | 60 | 32 | 500 | 32 | 84000 |
| 25000 | 10 | 32 | 500 | 32 | 504000 |

Moab Viewpoint

i Only an RPM-based installation is supported for installing Moab Viewpoint.

In this section:

- [Supported Operating Systems on page 16](#)
- [Depends On \(not necessarily on the same host\) on page 16](#)
- [Supported Browsers on page 16](#)

Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 12

i The Linux kernel version must be at least 2.6.13 and the glibc version must be at least 2.5.

Depends On (not necessarily on the same host)

- Moab Web Services 9.0.1
- Moab Insight 9.0.1

Supported Browsers

- Mozilla Firefox 25+
- Internet Explorer 10+
- Chrome 35+

RLM Server

Remote Visualization and Nitro require access to a centralized Reprise License Manager (RLM) server.

This server is not load-extensive so it may be installed on any host within your Moab HPC Suite environment. It may also be installed on its own host.

i If your company already utilizes an RLM Server, you do not have to install another as long as the Moab HPC Suite components can access it.

 The host on which you install RLM Server must always be on and should have High Availability (uptime).

Remote Visualization

 Remote Visualization (also referred to as FastX) resides on the Torque compute nodes.

 Only an RPM-based installation is supported for installing Remote Visualization.

In this section:

- [Supported Operating Systems on page 17](#)
- [License Requirements on page 17](#)
- [Software Requirements on page 17](#)
- [Depends On \(not necessarily on the same host\) on page 18](#)

Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 12

License Requirements

Remote Visualization requires access to a centralized Reprise License Manager (RLM) server. See [RLM Server on page 16](#) for more information.

Software Requirements

Gateway Server Host (Remote Visualization Server Host) and Session Servers (Torque MOM Hosts):

- ImageMagick
- ImageMagick-perl
- perl-Crypt-SSLeay
- perl-X11-Protocol

 The installation of these packages is included in the Install Remote Visualization procedure.

In addition, *each* Session Server must include the graphical applications (resources) you will have Moab schedule. For example, desktop (gnome-session), xterm, firefox, chrome.

Depends On (not necessarily on the same host)

- Torque Resource Manager 6.0.1
- Moab Workload Manager 9.0.1
- Moab Web Services 9.0.1
- Moab Insight 9.0.1
- Moab Viewpoint 9.0.1

Nitro

i When integrated with the Moab HPC Suite, Nitro resides on the Torque compute nodes.

In this section:

- [Hardware Requirements on page 18](#)
- [Supported Operating Systems on page 19](#)
- [License Requirements on page 19](#)
- [Software Requirements on page 19](#)

Hardware Requirements

- Nitro requires one or more multi-core processors per host. Generally the more processors (sockets) and/or OS cores a host has, the more tasks Nitro can execute simultaneously on each host; although this will be application-dependent.
- It is recommended that hosts should have sufficient memory to execute as many applications as possible so that Nitro can run them at a rate of one application instance per OS core (especially if they are not multi-threaded). This eliminates the need for users to have to request memory in their Nitro task definitions.

i See the *Nitro Administrator Guide* for information on specifying memory requirements.

Supported Operating Systems

- CentOS 6.x, 7.x
- Red Hat 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11, 12

License Requirements

Nitro requires access to a centralized Reprise License Manager (RLM) server. See [RLM Server on page 16](#) for more information.

Software Requirements

Nitro is built with all needed libraries statically linked. This provides for a quick and simple installation and helps avoid troublesome library mismatches. No additional packages need to be installed on the compute nodes.

However, users running the nitrostat utility require Python 2.6.6 or later on the system from which they are running it.

Nitro Web Services

i Nitro Web Services is commonly installed on the Moab Server Host.

In this section:

- [Supported Operating Systems on page 19](#)
- [Depends On \(not necessarily on the same host\) on page 19](#)

Supported Operating Systems

- CentOS 6.x, 7.x
- Red Hat 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11

Depends On (not necessarily on the same host)

- Nitro 2.0 – Installed on Torque compute nodes
- Viewpoint 9.0.1
- MongoDB 2.4.x

Identify the Manual or RPM Installation Methods

Adaptive Computing provides two methods for installing the Moab HPC Suite components, Manual Installation and RPM Installation.

Depending on your environment and which components you are installing (and on which host), you may need to use a combination of Manual Installation and RPM Installation.

i Most components can be installed using either method. Please choose one method for each component.

Manual Installation

This method provides advantages for administrators who want non-standard configure options.

- This method has more supported operating systems than the RPM Installation method.
- Insight or Viewpoint can not be installed using the Manual Installation method.

RPM Installation

This method provides advantages for administrator who want a standard installation, with little customization.

- This method only supports CentOS, RHEL, or Scientific Linux operating systems.
- Whether you are installing RPMs on one host or on several hosts, each host must have the Adaptive Computing Package Repository enabled. See [Preparing the Host – Typical Method on page 125](#)

Chapter 2 Manual Installation

This chapter provides installation, configuration, and upgrading information using the Manual Installation method.

Be aware of the following:

- On RHEL systems, you must be registered for a Red Hat subscription in order to have access to required rpm package dependencies.
- On SLES systems, you must be registered for a SUSE Linux Enterprise subscription in order to have access to required rpm package dependencies.
- Manual Installation is not available for Insight, Viewpoint, or Remote Visualization.
- Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges. You will see that the instructions execute commands as the root user. Also be aware that the same commands will work for a non-root user with the `sudo` command.

Related Topics

- [Chapter 1 Planning your Installation on page 3](#)
- [Preparing for Manual Installation on page 22](#)

Manual Installation

This section provides instructions and other information for installing your Moab HPC Suite components using the Manual installation method.

In this section:

- [Preparing for Manual Installation on page 22](#)
- [Installing Torque Resource Manager on page 24](#)
- [Installing Moab Workload Manager on page 33](#)
- [Installing Moab Accounting Manager on page 39](#)
- [Installing Moab Web Services on page 54](#)
- [Installing Moab Insight on page 67](#)
- [Installing Moab Viewpoint on page 68](#)
- [Installing RLM Server on page 68](#)
- [1.1 Installing Remote Visualization](#)
- [Nitro Integration on page 72](#)

Preparing for Manual Installation

The manual installation process of the Moab HPC Suite includes installing the different components in the suite. This guide contains detailed instructions for installing each component.

i Many individual components have dependencies on other components (see [Chapter 1 Planning your Installation on page 3](#)). However, if you do not require a certain component (Moab Web Services, for example), you do not have to install it.

The install instructions for each component include information about system requirements and dependencies. Some include prerequisite instructions that you will need to complete before you begin the install. Please read this information carefully, and make sure you have installed all the dependencies and packages that are necessary in order to avoid errors during the Moab HPC Suite install process.

i Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

Set Up Proxies

If your site uses a proxy to connect to the internet, configure yum to use a proxy by editing the `/etc/yum.conf` file as follows:

```
proxy=http://<proxy_server_id>:<port>
```

If your site uses an external repository to install python dependencies (for example, the host where you install Viewpoint might need to download extra packages), you will need to set up pip to use a proxy. Do the following:

```
export http_proxy=http://<proxy_server_id>:<port>
export https_proxy=http://<proxy_server_id>:<port>
```

Enable Extra Packages for the Repository

Many individual components have dependencies that are found in the optional add-on repositories for the distribution. You must enable the respective repository for your distribution on all hosts upon which you install Adaptive Computing software components.

- Red Hat-based systems
 - On non-RHEL systems (e.g. CentOS and Scientific Linux), you will need to install the `epel-release` package in order to have access to required rpm package dependencies.


```
[root]# yum install epel-release
```
 - On RHEL systems you must be registered for a Red Hat subscription in order to have access to required rpm package dependencies.
- SUSE-based systems
 1. Verify that you have a licensed installation of SLES 11 or 12 and that you are registered for a SUSE Linux Enterprise subscription.
 2. Download the SUSE Linux Enterprise 11 or 12 Software Development Kit e-Media Kit and add the ISO to the repository.
 3. Add the `devel:languages:perl` and the `devel:languages:python` repos for your SLES version to the your software repositories. For example, if you are using SLES 12, you might use the following

commands:

```
[root]# zypper addrepo
http://download.opensuse.org/repositories/devel:languages:perl/SLE_
12/devel:languages:perl.repo
[root]# zypper addrepo
http://download.opensuse.org/repositories/devel:languages:python/SLE_
12/devel:languages:python.repo
[root]# zypper refresh
```

Install the Moab HPC Suite Software Components

To install the Moab HPC Suite, install the packages in the following order:

1. Torque. See [Installing Torque Resource Manager on page 24](#).
2. Moab Workload Manager. See [Installing Moab Workload Manager on page 33](#).
3. Moab Accounting Manager. See [Installing Moab Accounting Manager on page 39](#).
4. Moab Web Services. See [Installing Moab Web Services on page 54](#).
5. Moab Insight (RPM install method only). See [Installing Moab Insight on page 165](#).
6. Moab Viewpoint (RPM install method only). See [Installing Moab Viewpoint on page 178](#).
7. RLM Server. See [Installing RLM Server on page 68](#).
8. Remote Visualization (RPM install method only). See [Installing Remote Visualization on page 198](#).
9. Integrate Nitro with your Moab HPC Suite. See [Nitro Integration on page 72](#).

Installing Torque Resource Manager



If you intend to use Torque Resource Manager 6.0.1 with Moab Workload Manager, you must run Moab version 8.0 or later. However, some Torque 6.0 functionality requires Moab 9.0 or later.

This topic contains instructions on how to install and start Torque Resource Manager (Torque).

i For Cray systems, Adaptive Computing recommends that you install Moab and Torque Servers (head nodes) on commodity hardware (*not* on Cray compute/service/login nodes).

However, you must install the Torque pbs_mom daemon and Torque client commands on Cray login and "mom" service nodes since the pbs_mom must run on a Cray service node within the Cray system so it has access to the Cray ALPS subsystem.

See [Installation Notes for Moab and Torque for Cray](#) in the *Moab Workload Manager Administrator Guide* for instructions on installing Moab and Torque on a non-Cray server.

In this topic:

- [Prerequisites on page 25](#)
- [Install Dependencies, Packages, or Clients on page 27](#)
- [Install Torque Server on page 27](#)
- [Install Torque MOMs on page 30](#)
- [Install Torque Clients on page 31](#)
- [Configure Data Management on page 32](#)

Prerequisites

In this section:

- [Open Necessary Ports on page 25](#)
- [Verify the hostname on page 27](#)

Open Necessary Ports

Torque requires certain ports to be open for essential communication.

- For client and pbs_mom communication to pbs_server, the default port is 15001.
- For pbs_server communication to pbs_mom, the default port is 15002.
- For pbs_mom communication to pbs_mom, the default port is 15003.

For more information on how to configure the ports that Torque uses for communication, see [Configuring Ports](#) in the *Torque Resource Manager Administrator Guide* for more information.

If you have a firewall enabled, do the following:

1. On the Torque Server Host:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following line immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 15001 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=15001/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following port to the FW_SERVICES_EXT_TCP parameter
FW_SERVICES_EXT_TCP="15001"

[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following port to the FW_SERVICES_EXT_TCP parameter
FW_SERVICES_EXT_TCP="15001"

[root]# service SuSEfirewall2 restart
```

2. On the Torque MOM Hosts (compute nodes):

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 15002 -j ACCEPT
-A INPUT -p tcp --dport 15003 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=15002-15003/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following ports to the FW_SERVICES_EXT_TCP parameter
FW_SERVICES_EXT_TCP="15002 15003"

[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following ports to the FW_SERVICES_EXT_TCP parameter
FW_SERVICES_EXT_TCP="15002 15003"

[root]# service SuSEfirewall2 restart
```

Verify the hostname

On the Torque Server Host, confirm your host (with the correct IP address) is in your `/etc/hosts` file. To verify that the hostname resolves correctly, make sure that `hostname` and `hostname -f` report the correct name for the host.

Install Dependencies, Packages, or Clients

Install Packages

On the Torque Server Host, use the following commands to install the `libxml2-devel`, `openssl-devel`, and `boost-devel` packages.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install libtool openssl-devel libxml2-devel boost-devel gcc gcc-c++
```

- SUSE 11-based or SUSE 12-based systems

```
[root]# zypper install libopenssl-devel libtool libxml2-devel boost-devel gcc
gcc-c++ make gmake
```

Install Torque Server

i You *must* complete the prerequisite tasks and the tasks to install the dependencies, packages, or clients before installing Torque Server. See [Prerequisites on page 25](#) and [Install Dependencies, Packages, or Clients on page 27](#).

On the Torque Server Host, do the following:

1. Download the latest 6.0.1 build from the [Adaptive Computing](#) website. It can also be downloaded via command line (github method or the tarball

distribution).

- Clone the source from github.

i If git is not installed:

```
# Red Hat-based systems
[root]# yum install git

# SUSE-based systems
[root]# zypper install git
```

```
[root]# git clone https://github.com/adaptivecomputing/torque.git -b 6.0.1 6.0.1
[root]# cd 6.0.1
[root]# ./autogen.sh
```

- Get the tarball source distribution.

- Red Hat-based systems

```
[root]# yum install wget
[root]# wget http://www.adaptivecomputing.com/download/torque/torque-6.0.1-
<filename>.tar.gz -O torque-6.0.1.tar.gz
[root]# tar -xzvf torque-6.0.1.tar.gz
[root]# cd torque-6.0.1/
```

- SUSE-based systems

```
[root]# zypper install wget
[root]# wget http://www.adaptivecomputing.com/download/torque/torque-6.0.1-
<filename>.tar.gz -O torque-6.0.1.tar.gz
[root]# tar -xzvf torque-6.0.1.tar.gz
[root]# cd torque-6.0.1/
```

2. Run each of the following commands in order.

```
[root]# ./configure
[root]# make
[root]# make install
```

See [Customizing the Install](#) in the *Torque Resource Manager Administrator Guide* for information on which options are available to customize the `./configure` command.

3. Verify that the `/var/spool/torque/server_name` file exists and contains the correct name of the server.

```
[root]# echo <torque_server_hostname> > /var/spool/torque/server_name
```

4. Configure the `trqauthd` daemon to start automatically at system boot.

- Red Hat 6-based systems

```
[root]# cp contrib/init.d/trqauthd /etc/init.d/
[root]# chkconfig --add trqauthd
[root]# echo /usr/local/lib > /etc/ld.so.conf.d/torque.conf
[root]# ldconfig
[root]# service trqauthd start
```

- SUSE 11-based systems

```
[root]# cp contrib/init.d/suse.trqauthd /etc/init.d/trqauthd
[root]# chkconfig --add trqauthd
[root]# echo /usr/local/lib > /etc/ld.so.conf.d/torque.conf
[root]# ldconfig
[root]# service trqauthd start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# cp contrib/systemd/trqauthd.service /usr/lib/systemd/system/
[root]# systemctl enable trqauthd.service
[root]# echo /usr/local/lib > /etc/ld.so.conf.d/torque.conf
[root]# ldconfig
[root]# systemctl start trqauthd.service
```

5. By default, Torque installs all binary files to `/usr/local/bin` and `/usr/local/sbin`. Make sure the path environment variable includes these directories for both the installation user and the root user.

```
[root]# export PATH=/usr/local/bin/:/usr/local/sbin/:$PATH
```

6. Initialize `serverdb` by executing the `torque.setup` script.

```
[root]# ./torque.setup root
```

7. Add nodes to the `/var/spool/torque/server_priv/nodes` file. See [Specifying Compute Nodes](#) in the *Torque Resource Manager Administrator Guide* for information on syntax and options for specifying compute nodes.
8. Configure `pbs_server` to start automatically at system boot, and then start the daemon.

- Red Hat 6-based systems

```
[root]# cp contrib/init.d/pbs_server /etc/init.d
[root]# chkconfig --add pbs_server
[root]# service pbs_server restart
```

- SUSE 11-based systems

```
[root]# cp contrib/init.d/suse.pbs_server /etc/init.d/pbs_server
[root]# chkconfig --add pbs_server
[root]# service pbs_server restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# qterm
[root]# cp contrib/systemd/pbs_server.service /usr/lib/systemd/system/
[root]# systemctl enable pbs_server.service
[root]# systemctl start pbs_server.service
```

Install Torque MOMs

In most installations, you will install a Torque MOM on each of your compute nodes.

i See [Specifying Compute Nodes](#) or [Configuring on Compute Nodes](#) in the Torque Resource Manager Administrator Guide for more information.

Do the following:

1. On the Torque Server Host, do the following:

- a. Create the self-extracting packages that are copied and executed on your nodes.

```
[root]# make packages
Building ./torque-package-clients-linux-x86_64.sh ...
Building ./torque-package-mom-linux-x86_64.sh ...
Building ./torque-package-server-linux-x86_64.sh ...
Building ./torque-package-gui-linux-x86_64.sh ...
Building ./torque-package-devel-linux-x86_64.sh ...
Done.
```

The package files are self-extracting packages that can be copied and executed on your production machines. Use --help for options.

- b. Copy the self-extracting packages to each Torque MOM Host.

Adaptive Computing recommends that you use a remote shell, such as SSH, to install packages on remote systems. Set up shared SSH keys if you do not want to supply a password for each Torque MOM Host.

i The only required package for the compute node is mom-linux. Additional packages are recommended so you can use client commands and submit jobs from compute nodes.

```
[root]# scp torque-package-mom-linux-x86_64.sh <mom-node>:
[root]# scp torque-package-clients-linux-x86_64.sh <mom-node>:
```

- c. Copy the pbs_mom startup script to each Torque MOM Host.

- Red Hat 6-based systems

```
[root]# scp contrib/init.d/pbs_mom <mom-node>:/etc/init.d
```

- SUSE 11-based systems

```
[root]# scp contrib/init.d/suse.pbs_mom <mom-node>:/etc/init.d/pbs_mom
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# scp contrib/systemd/pbs_mom.service <mom-node>:/usr/lib/systemd/system/
```

i Not all sites see an inherited ulimit but those that do can change the ulimit in the pbs_mom init script. The pbs_mom init script is responsible for starting and stopping the pbs_mom process.

2. On each Torque MOM Host, do the following:

- a. Install the self-extracting packages and run ldconfig.

```
[root]# ssh root@<mom-node>
[root]# ./torque-package-mom-linux-x86_64.sh --install
[root]# ./torque-package-clients-linux-x86_64.sh --install
[root]# ldconfig
```

- b. Configure pbs_mom to start at system boot, and then start the daemon.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# chkconfig --add pbs_mom
[root]# service pbs_mom start
```

- Red Hat 7-based or SUSE-12 based systems

```
[root]# systemctl enable pbs_mom.service
[root]# systemctl start pbs_mom.service
```

Install Torque Clients

If you want to have the Torque client commands installed on hosts other than the Torque Server Host (such as the compute nodes or separate login nodes), do the following:

1. On the Torque Server Host, do the following:

- a. Copy the self-extracting client package to each Torque Client Host.

i Adaptive Computing recommends that you use a remote shell, such as SSH, to install packages on remote systems. Set up shared SSH keys if you do not want to supply a password for each Torque MOM Host.

```
[root]# scp torque-package-clients-linux-x86_64.sh <torque-client-host>:
```

b. Copy the trqauthd startup script to each Torque Client Host.

- Red Hat 6-based systems

```
[root]# scp contrib/init.d/trqauthd <torque-client-host>:/etc/init.d
```

- SUSE 11-based systems

```
[root]# scp contrib/init.d/suse.trqauthd <torque-client-host>:/etc/init.d/trqauthd
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# scp contrib/systemd/trqauthd.service <torque-client-host>:/usr/lib/systemd/system/
```

2. On each Torque Client Host, do the following:

i Many of these steps can be done from the Torque server from a remote shell, such as SSH. Set up shared SSH keys if you do not want to supply a password for each Torque Client Host.

a. Install the self-extracting client package.

```
[root]# ./torque-package-clients-linux-x86_64.sh --install
[root]# echo /usr/local/lib > /etc/ld.so.conf.d/torque.conf
[root]# ldconfig
```

b. Enable and start the trqauthd service.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# chkconfig --add trqauthd
[root]# service trqauthd start
```

- Red Hat 7-based or SUSE-12 based systems

```
[root]# systemctl enable trqauthd.service
[root]# systemctl start trqauthd.service
```

Configure Data Management

When a batch job completes, stdout and stderr files are generated and placed in the spool directory on the master Torque MOM Host for the job instead of the submit host. You can configure the Torque batch environment to copy the stdout and stderr files back to the submit host. See [Configuring Data Management](#) in the *Torque Resource Manager Administrator Guide* for more information.

Related Topics

[Preparing for Manual Installation on page 22](#)

Installing Moab Workload Manager

This topic contains instructions on how to install and start Moab Workload Manager (Moab).

i For Cray systems, Adaptive Computing recommends that you install Moab and Torque Servers (head nodes) on commodity hardware (*not* on Cray compute/service/login nodes).

However, you must install the Torque pbs_mom daemon and Torque client commands on Cray login and "mom" service nodes since the pbs_mom must run on a Cray service node within the Cray system so it has access to the Cray ALPS subsystem.

See [Installation Notes for Moab and Torque for Cray](#) in the *Moab Workload Manager Administrator Guide* for instructions on installing Moab and Torque on a non-Cray server.

In this topic:

- [Open Necessary Ports on page 33](#)
- [Install Dependencies, Packages, or Clients on page 34](#)
- [\(Optional\) Build a Custom RPM on page 35](#)
- [Install Moab Server on page 36](#)
- [Configure Torque to Trust Moab on page 38](#)
- [Verify the Installation on page 38](#)
- [\(Optional\) Install Moab Client on page 38](#)

Open Necessary Ports

Moab uses a configurable server port (default 42559) for client-server communication. If you intend to run client commands on a different host from the Moab Server Host, or if you will be using Moab in a grid, and if you have a firewall enabled, you will need to configure the firewall to allow the server port.

On the Moab Server Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

# Needed on the Moab server for off-host client communication
-A INPUT -p tcp --dport 42559 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=42559/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following ports to the FW_SERVICES_EXT_TCP parameter as required

# Needed on the Moab server for off-host client communication
FW_SERVICES_EXT_TCP="42559"

[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following ports to the FW_SERVICES_EXT_TCP parameter as required

# Needed on the Moab server for off-host client communication
FW_SERVICES_EXT_TCP="42559"

[root]# service SuSEfirewall2 restart
```

Install Dependencies, Packages, or Clients

In this section:

- [Dependencies and Packages on page 34](#)
- [Torque Client on page 35](#)
- [Installing Moab Workload Manager on page 33](#)

Dependencies and Packages

On the Moab Server Host, use the following commands to install the required Moab dependencies and packages.

- Red Hat-based systems

```
[root]# yum update
[root]# yum install make libcurl perl-CPAN libxml2-devel gcc
```

- SUSE-based systems

```
[root]# zypper update
[root]# zypper install make curl libxml2-devel
```

Torque Client

If you are using Torque and are installing the Torque Server on a different host (Torque Server Host) from the Moab Server (Moab Server Host), you will need to install the Torque client on the Moab Server Host in order for Moab to interact with Torque.

Follow the instructions in [Install Torque Clients on page 31](#) using the Moab Server Host as the Torque Client Host; with the exception that you must copy and install the `torque-package-devel-linux-<arch>.sh` self-extracting package in addition to the `torque-package-client-linux-<arch>.sh` package.

(Optional) Build a Custom RPM

If you want to build a custom RPM, do the following:

1. Install `rpm-build`.

```
[root]# yum install rpm-build
```

2. Download the latest Moab build (`moab-<version>-<OS>.tar.gz`) from the [Adaptive Computing](#) website.

i The variable marked `<version>` is the desired version of the suite; for example, `8.1.0-2015010514-ed0c40a` would be Moab 8.1.0 revision 2015010514 at changeset `ed0c40a`. The variable marked `<OS>` indicates the OS for which the build was designed.

3. Untar the downloaded package.
4. Change directories into the untarred directory.
5. Edit the `./moab.spec` file for RPM customization.
6. Run `./rpm-build`.
7. Locate the custom RPM in `rpm/RPMS/x86_64`.

Install Moab Server

i You *must* complete the tasks to install the dependencies, packages, or clients before installing Moab Server. See [Install Dependencies, Packages, or Clients on page 34](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing the Moab Server. See [Open Necessary Ports on page 33](#).

On the Moab Server Host, do the following:

1. Download the latest Moab build (`moab-<version>-<OS>.tar.gz`) from the [Adaptive Computing](#) website.

i The variable marked `<version>` indicates the build's version, revision, and changeset information. The variable marked `<OS>` indicates the OS for which the build was designed.

2. As the root user, run each of the following commands in order.

```
[root]# tar xzvf moab-<version>-<OS>.tar.gz
[root]# cd moab-<version>-<OS>
```

If Elastic Computing is part of your Moab Workload Manager configuration, install `deps/acpython-base*`.

```
# Red Hat-based systems
[root]# yum install deps/acpython-base*

# SUSE-based systems
[root]# zypper install deps/acpython-base*
```

3. Configure Moab. If you are installing Moab Accounting Manager, configure Moab with the `--with-am` option.

```
[root]# ./configure <options>
```

i See [Moab Workload Manager Configuration Options on page 93](#) for a list of commonly used options or use `./configure --help` for a complete list of available options.

4. *ONLY* if you are using green computing, or if you are using a resource manager other than Torque.

Run the `make perldeps` command to install the necessary perl modules using CPAN. When first running CPAN, you will be asked for configuration information. It is recommended that you choose an automatic configuration. You will be prompted to provide input during module installation; running the `make perldeps` command with a script is not recommended.

```
[root]# make perldeps
```

5. Install Moab.

```
[root]# make install
```

6. Modify the Moab configuration file.

```
[root]# vi /opt/moab/etc/moab.cfg
```

Do the following:

- a. Verify that **SUBMITCMD** is set up for your Torque resource manager and that it points to a valid `qsub` executable. For example:

```
RMCFG[torque] SUBMITCMD=/usr/local/bin/qsub
```

If you use a SLURM resource manager, see [Moab-SLURM Integration Guide](#) in the *Moab Workload Manager Administrator Guide* for configuration information. If you use a NATIVE resource manager, see [Managing Resources Directly with the Native Interface](#) in the *Moab Workload Manager Administrator Guide* for configuration information.

- b. If you are using Torque as a resource manager and you installed the Torque Server on a different host (Torque Server Host), configure the RMCFG HOST parameter to tell Moab the host on which Torque Server is running.

```
RMCFG[torque] HOST=<torque_server_hostname>
```

7. Source the appropriate profile script to add the Moab executable directories to your current shell `$PATH` environment.

```
[root]# . /etc/profile.d/moab.sh
```

8. Copy your license file into the same directory as `moab.cfg` (`/opt/moab/etc/` by default). For example:

```
[root]# cp moab.lic $MOABHOMEDIR/etc/moab.lic
```

To verify the current status of your license, use `moab --about`.

Moab checks the status of the license every day just after midnight. At 60 and 45 days before, and daily from 30 days before license expiration to and including the license expiration date, Moab sends an e-mail to all level 1 administrators informing them of the pending Moab license expiration. A log record is also made of the upcoming expiration event. For the notifications to occur correctly, you must enable administrator email notification (see [Notifying Administrators of Failures](#) in the *Moab Workload Manager Administrator Guide*) and `moab.cfg` must contain email addresses for level 1 administrators. For example:

```
ADMINCFG [1] USERS=u1,u2,u3[,...]

USERCFG [u1] EMAILADDRESS=u1@company.com
USERCFG [u2] EMAILADDRESS=u2@company.com
USERCFG [u3] EMAILADDRESS=u3@company.com

MAILPROGRAM DEFAULT
```

i Moab will not run without a license. For information about obtaining a trial license, please contact [Adaptive Computing](#).

9. Start Moab.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# chkconfig moab on
[root]# service moab start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start moab.service
```

Configure Torque to Trust Moab

If you are using Torque as a resource manager and you installed the Torque Server on a different host (Torque Server Host); recommended, do the following:

- On the Torque Server Host, add the name of the Moab Server Host (where Moab Server is installed) as a manager and as a submit host.

```
[root]# qmgr
Qmgr: set server managers += root@<moab_server_hostname>
Qmgr: set server submit_hosts += <moab_server_hostname>
Qmgr: exit
```

Verify the Installation

If you have a resource manager configured, verify that the scheduler is able to schedule a job.

- Submit a sleep job as a non-root user (adaptive is used in this example) and verify the job is running.

```
[root]# su - adaptive
[adaptive]$ echo sleep 150 | msub
[adaptive]$ showq
[adaptive]$ exit
```

(Optional) Install Moab Client

After you have installed Moab Server, you can create a client tarball to install just the Moab client commands on a login/client host. This tarball uses a single `tar` command to install the binary Moab client command files and their man pages. The tarball also contains a `moab.cfg` file configured with the Moab

Server host name and port number so you do not have to manually configure this information on the login/client node.

i If your site needs secure communication and authentication between Moab Client Host and the Moab Server Host, create a site-specific key and place it in the same directory as your `moab.cfg` file. By default, this would be `$MOABHOMEDIR/etc/.moab.key`. When the Moab server and client commands detect the presence of those two files they will use the key in those files to authenticate and communicate, instead of the default key. See [Mauth Authentication](#) in the *Moab Workload Manager Administrator Guide* for more information.

Do the following:

1. On the Moab Server Host, create the client tarball.

```
[root]# make client-pkg
```

2. Copy the tarball to the root directory of the Moab Client Host.
3. On the Moab Client Host, run the tarball to install the Moab client commands.

```
[root]# tar xvf client.tgz
```

Related Topics

[Preparing for Manual Installation on page 22](#)

Installing Moab Accounting Manager

This topic contains instructions on how to install and start Moab Accounting Manager (MAM).

Perform the following in order:

- [Plan Your Installation](#)
- [Open Necessary Ports](#)
- [Install and Initialize the MAM PostgreSQL Server](#)
- [Install Dependencies, Packages, or Clients](#)
- [\(Optional\) Build a Custom RPM](#)
- [Install MAM Server](#)
- [Configure the MAM GUI](#)
- [Access the MAM GUI](#)

- [Configure Moab Workload Manager to Use Moab Accounting Manager](#)
- [Initialize Moab Accounting Manager](#)

Plan Your Installation

The first step is determining the number of different hosts (physical machines) required for your MAM installation.

Your MAM installation includes:

- MAM Server
- MAM Database
- MAM GUI (optional)
- MAM Clients (possibly several hosts)

Each of these components can be installed on their own hosts (meaning the actual physical machine) or can be combined on same hosts. For example, the MAM Database can be installed on the same *host* as the MAM Server. Or the MAM Server may be installed on the same host you installed the Moab Server.



If Moab Insight is part of your configuration, Insight requires a PostgreSQL 9.3. database. If your configuration will have the MAM PostgreSQL database on the *same* host as the Insight PostgreSQL database, you must install version 9.3 for the MAM PostgreSQL database.

Once you have determined which components are installed on which hosts, complete the rest of the instructions for the MAM installation.



The instructions that follow in this topic will use the term Host after each component to reflect installing on a host (again, meaning the physical machine). For example, MAM Server Host and MAM Database Host. Depending on your configuration, Host may refer to as installed on its own machine or installed on the same machine as another component.

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

Do the following as needed:

1. If you will be installing the MAM Server on a different host from where you installed the Moab Server *or* you will be installing the MAM Clients on other hosts, then on the MAM Server Host, open the MAM Server port (7112) in the firewall.

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod

[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 7112 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod

[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=7112/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="7112"

[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="7112"

[root]# service SuSEfirewall2 restart
```

2. If using the MAM GUI, then on the MAM GUI Host, open the https port (443) in the firewall for secure browser communication.

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod

[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 443 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod

[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=https/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="443"
[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="443"
[root]# service SuSEfirewall2 restart
```

3. If you will be installing the MAM PostgreSQL Database on a different host from the MAM Server, then on the host where the MAM PostgreSQL Database Host, open the postgres port (5432) in the firewall.

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod
# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 5432 -j ACCEPT
[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=postgres/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5432"
[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5432"
[root]# service SuSEfirewall2 restart
```

Install and Initialize the MAM PostgreSQL Server

Moab Accounting Manager uses a database for transactions and data persistence.

The MAM PostgreSQL database may be installed on:

- the same host as the MAM Server.
- a separate PostgreSQL database host.
- a separate *shared* PostgreSQL database host. If this shared database host *will* include the Insight PostgreSQL database, then the MAM PostgreSQL database *must* be version 9.3.

On the host where the MAM PostgreSQL database will reside, do the following:



These instructions assume you will be installing the MAM PostgreSQL database on a *different host* from where the Insight PostgreSQL database will reside.

If you wish to install *both* the MAM and the Insight PostgreSQL databases on the same host, different instructions are required. For example, you will need to enable the postgresql-9.3 RPM repo by following the RPM instructions to prepare the host (see [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#)) and you will need to modify the MAM PostgreSQL install instructions to reflect the 9.3 naming (see [Install PostgreSQL](#) for an example of how to install PostgreSQL 9.3 for Insight).

1. Install and initialize the PostgreSQL Server.

- Red Hat 6-based systems

```
[root]# yum install postgresql-server
[root]# service postgresql initdb
```

- Red Hat 7-based systems

```
[root]# yum install postgresql-server
[root]# postgresql-setup initdb
```

- SUSE 11-based systems

```
[root]# zypper install postgresql-server
[root]# service postgresql start
```

- SUSE 12-based systems

```
[root]# zypper install postgresql-server
[root]# systemctl start postgresql.service
```

2. Configure trusted connections.

Edit or add a "host" line in the `pg_hba.conf` file for the interface from which the MAM Server will be connecting to the database and ensure that it specifies a secure password-based authentication method (for example, `md5`).

```
[root]# vi /var/lib/pgsql/data/pg_hba.conf

# Replace 127.0.0.1 with the IP address of the MAM Server Host if the
# MAM PostgreSQL server is on a separate host from the MAM server.
host    all             all             127.0.0.1/32         md5
host    all             all             ::1/128              md5
---
```

3. If the MAM Database Host is installed on a *different* host from where you will install the MAM Server, configure PostgreSQL to accept connections from the MAM Server Host.

```
[root]# vi /var/lib/pgsql/data/postgresql.conf

# Replace <mam-server-host> with the interface name from which the MAM server
# will be connecting to the database.
listen_addresses = '<mam-server-host>'
---
```

4. Start or restart the database.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# chkconfig postgresql on
[root]# service postgresql restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable postgresql.service
[root]# systemctl restart postgresql.service
```

Install Dependencies, Packages, or Clients

Use the following instructions to install the required Moab Accounting Manager dependencies, packages, or clients.

i Depending on your configuration, the MAM Server Host and the MAM GUI Host may be installed on the same host. The MAM Client Host is automatically installed on the same host as the MAM Server Host; however, you can also install the MAM Client Host on any other hosts on which you want to have the MAM client commands available to users or administrators.

1. On the MAM Server Host, the MAM GUI Host, and the MAM Client Hosts, do the following:

- Red Hat-based systems

```
[root]# yum install gcc redhat-lsb-core perl rrdtool perl-Config-Tiny perl-
Crypt-CBC perl-Crypt-DES perl-Crypt-DES_EDE3 perl-Digest-HMAC perl-Error perl-
Log-Dispatch-FileRotate perl-Log-Log4perl perl-XML-LibXML
```

i If installing on RHEL, some packages may not be found in the standard RHEL distribution repositories.

- One way to overcome this problem is to install the missing dependencies from reputable CentOS and EPEL repositories. For example (for the current RHEL 7 repositories):

```
[root]# cat > /etc/yum.repos.d/centos.repo <<'EOF'
[centos]
name=CentOS $releasever - $basearch
baseurl=http://mirror.centos.org/7/os/$basearch/
enabled=0
gpgcheck=0
EOF
[root]# cat > /etc/yum.repos.d/epel.repo <<'EOF'
[epel]
name=Extra Packages for Enterprise Linux 7 - $basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-
7&arch=$basearch
failovermethod=priority
enabled=0
gpgcheck=0
EOF
[root]# yum install --enablerepo=centos,epel gcc redhat-lsb-core perl
rrdtool perl-Config-Tiny perl-Crypt-CBC perl-Crypt-DES perl-Crypt-DES_
EDE3 perl-Digest-HMAC perl-Error perl-Log-Dispatch-FileRotate perl-Log-
Log4perl perl-XML-LibXML
```

- *Alternatively*, you can install the available packages in the RHEL repository and then install the missing modules from CPAN.

```
[root]# yum install --skip-broken gcc redhat-lsb-core perl rrdtool perl-
Config-Tiny perl-Crypt-CBC perl-Crypt-DES perl-Crypt-DES_EDE3 perl-
Digest-HMAC perl-Error perl-Log-Dispatch-FileRotate perl-Log-Log4perl
perl-XML-LibXML perl-CPAN
[root]# cpan YAML Config::Tiny Log::Log4perl Log::Dispatch::FileRotate
Compress::Zlib
```

You may need to run the cpan command more than once for it to complete successfully.

- SUSE-based systems

```
[root]# zypper install gcc lsb-release perl-Config-Tiny perl-Crypt-CBC perl-
Crypt-DES perl-Crypt-DES_EDE3 perl-Digest-HMAC perl-Error perl-Log-Log4perl
perl-XML-LibXML perl-Params-Validate perl-YAML perl-Log-Dispatch perl-Log-
Dispatch-FileRotate
```

2. On the MAM Server Host, do the following:

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install postgresql postgresql-libs perl-DBD-Pg perl-Date-Manip perl-
Time-HiRes perl-DBI
```

- SUSE 11-based systems

```
[root]# zypper install postgresql postgresql-libs perl-DBD-Pg perl-Date-Manip
perl-DBI
```

- SUSE 12-based systems

```
[root]# zypper install postgresql postgresql-devel libpq5 perl-DBD-Pg perl-Date-
Manip perl-DBI
```

3. On the MAM GUI Host, do the following:

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install httpd mod_ssl perl-CGI perl-CGI-Session
```

- SUSE 11-based or SUSE 12-based systems

```
[root]# zypper install apache2 perl-CGI perl-CGI-Session
```

4. On each of the MAM Client Hosts (including the MAM Server Host), do the following:

- Red Hat 6-based systems

```
[root]# yum install perl-suidperl perl-Term-ReadLine-Gnu perl-TermReadKey
```

- Red Hat 7-based systems

```
[root]# yum install perl-CPAN openssl-devel readline-devel ncurses-devel perl-
TermReadKey perl-Term-ReadLine-Gnu
```

- SUSE 11-based systems

```
[root]# zypper install libopenssl-devel perl-TermReadLine-Gnu perl-TermReadKey
[root]# chmod 4755 /usr/bin/sperl*
```

- SUSE 12-based systems

```
[root]# zypper install libopenssl-devel perl-TermReadLine-Gnu perl-Term-ReadKey
```

i If any of the Perl module packages fail to install or are unavailable for your system, you can install it from CPAN by running `cpan MODULENAME` where *MODULENAME* is the respective perl module name.

(Optional) Build a Custom RPM

If you want to build a custom RPM, do the following:

1. Install `rpm-build`.

```
[root]# yum install rpm-build
```

2. Download the latest MAM build (`mam-<version>.tar.gz`) from the [Adaptive Computing](#) website.

i The variable marked *<version>* is the desired version of the suite; for example, 9.0.1.

3. Untar the downloaded package.
4. Change directories into the untarred directory.
5. Edit the `./mam.spec` file for RPM customization.
6. Run `./rpm-build`.
7. Locate the custom RPM in `rpm/RPMS/x86_64`.

Install MAM Server

On the MAM Server Host, do the following:

1. Create a user called `mam` and switch to that user.

```
[root]# useradd -m mam
[root]# su - mam
[mam]$ mkdir src
[mam]$ cd src
```

2. Download the latest MAM build (`mam-<version>.tar.gz`) from the [Adaptive Computing](#) website.

i The variable marked *<version>* indicates the build's version.

3. Untar the MAM tarball.

```
[mam]$ tar -zxvf mam-9.0.1.tar.gz
```

4. Navigate to `mam-9.0.1`.

```
[mam]$ cd mam-9.0.1
```

5. Configure the software. For a list of all the configuration options, see [Moab Accounting Manager Configuration Options on page 94](#).

```
[mam]$ ./configure
```

6. Compile the software.

```
[mam]$ make
```

i If you only need to install the clients on a particular system, replace `make` with `make clients-only`. If you only need to install the web GUI on a particular system, replace `make` with `make gui-only`.

7. Install the software.

```
[mam]$ exit
[root]# cd ~mam/src/mam-9.0.1
[root]# make install
```

i If you only need to install the clients on a particular system, replace `make install` with `make install-clients-only`. If you only need to install the web GUI on a particular system, replace `make install` with `make install-gui-only`.

8. As the database user, create a database called `mam` and grant database privileges to the `mam` user.

i PostgreSQL should have previously been installed using the instructions in [Preparing for Manual Installation on page 22](#).

```
[root]# su - postgres
[postgres]$ psql

create database mam;
create user mam with password 'changeme!';
\q

[postgres]$ exit
```

The password you define must be synchronized with the `database.password` value in `/opt/mam/etc/mam-server.conf`

```
[root]# vi /opt/mam/etc/mam-server.conf

database.password = changeme!
```

9. Run the `hpc.sql` script to populate the Moab Accounting Manager database with objects, actions, and attributes necessary to function as an Accounting Manager.

```
[root]# su - mam
[mam]$ cd src/mam-9.0.1
[mam]$ psql mam < hpc.sql
[mam]$ exit
```

10. Configure MAM to automatically start up at system boot; start the `mam` service.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# chkconfig --add mam
[root]# service mam start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable mam.service
[root]# systemctl start mam.service
```

Configure the MAM GUI

If you plan to use the web GUI, then on the MAM GUI Host, do the following:

1. As `root`, add or edit the SSL virtual host definition as appropriate for your environment. To do so, configure the `cgi-bin` directory in `ssl.conf`. Below the `cgi-bin` directory element, create an alias for `/cgi-bin` pointing to your `cgi-bin` directory. If you chose to install to a `cgi-bin` sub-directory, you might want to create an alias for that as well. Also, add `index.cgi` to the `DirectoryIndex` so you can use the shorter sub-directory name.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# vi /etc/httpd/conf.d/ssl.conf

<Directory "/var/www/cgi-bin">
## Add these lines
  Options ExecCGI
  AddHandler cgi-script .cgi
  AllowOverride All
  Order allow,deny
  Allow from all
</Directory>

# Aliases for /cgi-bin
Alias /cgi-bin/ /var/www/cgi-bin/
Alias /mam /var/www/cgi-bin/mam/

# Make shorter sub-dir name available
DirectoryIndex index.cgi
```

- SUSE 11-based systems

```
[root]# a2enflag SSL
[root]# cp /etc/apache2/vhosts.d/vhost-ssl.template /etc/apache2/vhosts.d/mam-ssl.conf
[root]# vi /etc/apache2/vhosts.d/mam-ssl.conf

<Directory "/srv/www/cgi-bin">
## Add these lines
  Options ExecCGI
  AddHandler cgi-script .cgi
  AllowOverride All
  Order allow,deny
  Allow from all
</Directory>

# Aliases for /cgi-bin
Alias /cgi-bin/ /srv/www/cgi-bin/
Alias /mam /srv/www/cgi-bin/mam/

# Make shorter sub-dir name available
DirectoryIndex index.cgi
```

- SUSE 12-based systems

```
[root]# a2enflag SSL
[root]# cp /etc/apache2/vhosts.d/vhost-ssl.template /etc/apache2/vhosts.d/mam-ssl.conf
[root]# vi /etc/apache2/vhosts.d/mam-ssl.conf

<Directory "/srv/www/cgi-bin">
  Options ExecCGI
  AddHandler cgi-script .cgi
  AllowOverride All
  Require all granted
</Directory>

# Aliases for /cgi-bin
Alias /cgi-bin/ /srv/www/cgi-bin/
Alias /mam /srv/www/cgi-bin/mam/

# Make shorter sub-dir name available
DirectoryIndex index.cgi
```

2. For Red Hat-based systems where Security Enhanced Linux (SELinux) is enforced, you may need to customize SELinux to allow the web server to make network connections, use `setuid` for authentication, and write to the log file.
 - a. Determine the current mode of SELinux.

```
[root]# getenforce

Enforcing
```

- If the command returns a mode of **Disabled** or **Permissive**, or if the `getenforce` command is not found, you can skip the rest of this step.

- If the command returns a mode of **Enforcing**, you can choose between options of customizing SELinux to allow the web GUI to perform its required functions or disabling SELinux on your system.
- b. If you choose to customize SELinux, do the following:

i SELinux can vary by version and architecture and that these instructions may not work in all possible environments.

i If you used the `--prefix=<prefix>` configuration option when you configured Moab Accounting Manager, you must replace references to `/opt/mam` in the example below with the `<prefix>` you specified. See [Moab Accounting Manager Configuration Options on page 94](#).

- Red Hat 6-based systems

```
[root]# cat > mamgui.te <<EOF
module mamgui 1.0;
require {
    type httpd_sys_script_t;
    type port_t;
    class capability setuid;
    class tcp_socket name_connect;
}
allow httpd_sys_script_t port_t:tcp_socket name_connect;
allow httpd_sys_script_t self:capability setuid;
EOF
[root]# checkmodule -M -m -o mamgui.mod mamgui.te
[root]# semodule_package -m mamgui.mod -o mamgui.pp
[root]# semodule -i mamgui.pp
[root]# setenforce 0
[root]# chcon -v -t httpd_sys_content_t /opt/mam/log
[root]# setenforce 1
```

- Red Hat 7-based systems

```
[root]# yum install checkpolicy policycoreutils-python
[root]# cat > mamgui.te <<EOF
module mamgui 1.0;
require {
    type httpd_sys_script_t;
    type unreserved_port_t;
    class tcp_socket name_connect;
}
allow httpd_sys_script_t unreserved_port_t:tcp_socket name_connect;
EOF
[root]# checkmodule -M -m -o mamgui.mod mamgui.te
[root]# semodule_package -m mamgui.mod -o mamgui.pp
[root]# semodule -i mamgui.pp
[root]# setenforce 0
[root]# chcon -v -t httpd_sys_rw_content_t /opt/mam/log
[root]# setenforce 1
```

3. For the highest security, it is recommended that you install a public key certificate that has been signed by a certificate authority. The exact steps to

do this are specific to your distribution and the chosen certificate authority. An overview of this process for CentOS 7 is documented at https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/ch-Web_Servers.html#s2-apache-mod_ssl.

Alternatively, if your network domain can be secured from man-in-the-middle attacks, you could use a self-signed certificate. Often this does not require any additional steps since in many distributions, such as Red Hat, the Apache SSL configuration provides self-signed certificates by default.

If your configuration uses self-signed certificates, do the following:

- Red Hat 6-based or Red Hat 7-based systems
No action required. RedHat ships with ready-made certificates.
- SUSE 11-based or SUSE 12-based systems

```
[root]# cd /etc/apache2
[root]# openssl genrsa -out ssl.key/server.key 1024
[root]# openssl req -new -key ssl.key/server.key -x509 -out ssl.crt/server.crt
```

4. Start or restart the HTTP server daemon.

- Red Hat 6-based systems

```
[root]# chkconfig httpd on
[root]# service httpd restart
```

- Red Hat 7-based systems

```
[root]# systemctl enable httpd.service
[root]# systemctl restart httpd.service
```

- SUSE 11-based systems

```
[root]# chkconfig apache2 on
[root]# service apache2 restart
```

- SUSE 12-based systems

```
[root]# systemctl enable apache2.service
[root]# systemctl restart apache2.service
```

Access the MAM GUI

If you plan to use the web GUI, then on the MAM Server Host, do the following:

1. Create a password for the `mam` user to be used with the MAM Web GUI.

```
[root]# su - mam
[mam]$ mam-set-password
```

2. Verify the connection.

- a. Open a web browser and navigate to `https://<mam-server-host>/cgi-bin/mam`.
- b. Log in as the `mam` user with the password you set in step 1.

Configure Moab Workload Manager to Use Moab Accounting Manager

Do the following:

1. Configure Moab to talk to MAM

Do *one* of the following:

- **MAM Option.** If you are will be using the MAM (direct network) accounting manager interface with Moab Workload Manager (this is the default), do the following:
 - a. On the Moab Server Host, edit the Moab configuration file, uncomment the AMCFG lines and set the TYPE to MAM and set the HOST. If the Moab Server and the MAM Server are on the same host, set HOST to 'localhost'; otherwise, set HOST to the host name for the MAM Server (MAM Server Host).

```
[root]# vi /opt/moab/etc/moab.cfg
AMCFG[mam] TYPE=MAM HOST=<mam_server_host>
```

Customize additionally as needed. See [Accounting, Charging, and Allocation Management](#) in the *Moab Workload Manager Administrator Guide*

- b. Configure Moab to authenticate with MAM using the MAM secret key.
 - a. On the MAM Server Host, copy the auto-generated secret key from the token.value value in the `/opt/mam/etc/mam-site.conf` file.
 - b. On the Moab Server Host, add the secret key to the `moab-private.cfg` file as the value of the CLIENTCFG KEY attribute.

```
[root]# vi /opt/moab/etc/moab-private.cfg
CLIENTCFG[AM:mam] KEY=<MAMSecretKey>
```

- **Native Option.** If you are will be using the Native (custom script) accounting manager interface with Moab Workload Manager, do the following:
 - a. On the Moab Server Host, edit the Moab configuration file, uncomment the AMCFG lines and set the TYPE to NATIVE.

```
[root]# vi /opt/moab/etc/moab.cfg
AMCFG[mam] TYPE=NATIVE
```

- b. If you are installing Moab Accounting Manager on a different host (MAM Server Host) from the Moab Server (Moab Server Host), you will need to install the Moab Accounting Manager client on the Moab Server Host in order for the custom scripts to use the MAM API.

On the *Moab* Server Host, follow the instructions in [Install Dependencies, Packages, or Clients on page 44](#) and [Install MAM Server on page 47](#); with the following exceptions:

- Install only the dependent packages applicable to MAM Client Hosts
- Use the configure option `--without-init`
- Instead of running `make`, use `make clients-only`
- Instead of running `make install`, use `make install-clients-only`
- Omit the step to create the database and all of the steps thereafter

2. On the Moab Server Host, restart Moab.

- Red Hat 6-based or SUSE 11-based systems

```
service moab restart
```

- Red Hat 7-based or SUSE 12-based systems

```
systemctl restart moab.service
```

Initialize Moab Accounting Manager

You will need to initialize Moab Accounting Manager to function in the way that is most applicable to the needs of your site. See [Initial Setup](#) in the *Moab Accounting Manager Administrator Guide* to set up Moab Accounting Manager for your desired accounting mode.

Related Topics

[Preparing for Manual Installation on page 22](#)

Installing Moab Web Services

 You must deploy Moab Web Services on the *same* host as Moab Server (Moab Server Host). If using Viewpoint, this shared host must have a Red Hat-based OS; regardless of whether Viewpoint is also installed on that host. For documentation clarity, these instructions refer to the shared host for Moab Server and MWS as the MWS Server Host.

This topic contains instructions on how to install Moab Web Services (MWS).

In this topic:

- [Open Necessary Ports on page 55](#)
- [Install Dependencies, Packages, or Clients on page 56](#)
- [Install MWS Server on page 60](#)

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

In this section:

- [Open the Tomcat Port \(8080\) on page 55](#)
- [Open the MWS MongoDB Database Port \(27017\) on page 56](#)

Open the Tomcat Port (8080)

On the MWS Server Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 8080 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=8080/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="8080"

[root]# service SuSEfirewall2 restart
```

Open the MWS MongoDB Database Port (27017)

i Depending on your system configuration, your MongoDB databases may not be installed on the same host as their corresponding component servers. For example, you may choose to install the MWS MongoDB database on the same host where you have installed other MongoDB databases instead of on the MWS Server Host.

Do the following, as needed:

- If you have chosen to install the MWS MongoDB database on the *same* host you installed other MongoDB databases (for example, the same host you installed the Moab MongoDB database), confirm the firewall port (27017) is already opened on that host.
- If you have chosen to install the MWS MongoDB database on a *different* host from other MongoDB databases, you will need to open the MWS MongoDB database port in firewall for that host. To open the port in the firewall, do the following:
 - Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

Add the following line immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 27017 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=27017/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="27017"

[root]# service SuSEfirewall2 restart
```

Install Dependencies, Packages, or Clients

In this section:

- [Install Java on page 57](#)
- [Install Tomcat on page 57](#)
- [Install MongoDB on page 57](#)

Install Java

Install the Linux x64 RPM version of Oracle® Java® 8 Runtime Environment.

i Oracle Java 8 Runtime Environment is the recommended Java environment, but Oracle Java 7 is also supported. All other versions of Java, including OpenJDK/IcedTea, GNU Compiler for Java, and so on cannot run MWS.

On the MWS Server Host, do the following:

1. Install the Linux x64 RPM version of Oracle Java SE 8 JRE.
 - a. Go to the [Oracle Java download page](http://java.com/en/download/linux_manual.jsp) (http://java.com/en/download/linux_manual.jsp).
 - b. Copy the URL for the Linux x64 RPM version, and run the following command:
 - Red Hat 6-based or Red Hat 7-based systems

```
[root]# rpm -Uh <URL>
```

- SUSE 12-based systems

```
[root]# ln -s /usr/sbin/update-alternatives /usr/sbin/alternatives
[root]# rpm -Uh <URL>
```

Install Tomcat

Install Tomcat 7.

⚠ Tomcat 7 is required to run MWS 9.0 and after. MWS 9.0 will not run on Tomcat 6.

On the MWS Server Host, do the following:

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install tomcat
```

- SUSE 12-based systems

```
[root]# zypper install tomcat
```

Install MongoDB

To install and enable MongoDB, do the following on the MWS Server Host:

1. Install MongoDB.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# cat > /etc/yum.repos.d/mongodb.repo <<End-of-file
[mongodb]
name=MongoDB Repository
baseurl=http://downloads-distro.mongodb.org/repo/redhat/os/x86_64
gpgcheck=0
enabled=1
exclude=mongodb-org mongodb-org-server
End-of-file
[root]# yum install mongo-10gen-server
```

- SUSE-12 based systems

```
[root]# zypper ar --refresh -r
http://download.opensuse.org/repositories/server:/database/SLE_
12/server:database.repo
[root]# zypper install mongodb
```

2. Start MongoDB.

i There may be a short delay (approximately three minutes) for Mongo to start the first time.

- Red Hat 6-based systems

```
[root]# chkconfig mongod on
[root]# service mongod start
```

- Red Hat 7-based systems

```
[root]# cat > /usr/lib/systemd/system/mongodb.service <<End-of-file
[Unit]
Description=High-performance, schema-free document-oriented database
After=syslog.target network.target

[Service]
Type=forking
User=mongod
Group=mongod
Environment=CONFIG=/etc/mongod.conf
Environment=OPTIONS=
EnvironmentFile=-/etc/sysconfig/mongod
ExecStart=/usr/bin/mongod -f \${CONFIG} \${OPTIONS}
PrivateTmp=true
LimitNOFILE=65536
TimeoutStartSec=180
StandardOutput=syslog
StandardError=syslog

[Install]
WantedBy=multi-user.target
End-of-file
[root]# rm -f /etc/init.d/mongod
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
```

- SUSE 12-based systems

```
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
```

3. Prepare the MongoDB database by doing the following:

- a. Add the required MongoDB users.

i The passwords used below (`secret1`, `secret2`, and `secret3`) are examples. Choose your own passwords for these users.

```
[root]# mongo
> use admin;
> db.addUser("admin_user", "secret1");
> db.auth("admin_user", "secret1");

> use moab;
> db.addUser("moab_user", "secret2");
> db.addUser("mws_user", "secret3", true);

> use mws;
> db.addUser("mws_user", "secret3");
> exit
```

i Because the `admin_user` has read and write rights to the `admin` database, it also has read and write rights to all other databases. See [Control Access to MongoDB Instances with Authentication](http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication) (at <http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication>) for more information.

d. Enable authentication in MongoDB.

- Red Hat 6-based systems

```
[root]# vi /etc/mongod.conf
auth = true
[root]# service mongod restart
```

- Red Hat 7-based systems

```
[root]# vi /etc/mongod.conf
auth = true
[root]# systemctl restart mongod.service
```

- SUSE 12-based systems

MongoDB authentication is enabled (`auth = true`) by default. No further action is needed.

Install MWS Server

i You *must* complete the tasks to install the dependencies, packages, or clients before installing MWS Server. See [Install Dependencies, Packages, or Clients on page 56](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing the MWS Server. See [Open Necessary Ports on page 55](#).

On the MWS Server Host, do the following:

1. Verify Moab Server is installed and configured as desired (for details, see [Installing Moab Workload Manager on page 33](#)).
2. Start Moab.
 - Red Hat 6-based systems


```
[root]# service moab start
```
 - Red Hat 7-based or SUSE 12-based systems


```
[root]# systemctl start moab.service
```
3. Create the MWS home directory and subdirectories.

See [Configuration](#) in the *Moab Web Services Administrator Guide* for more information.

i The default location for the MWS home directory is `/opt/mws`. These instructions assume the default location.

Do the following:

```
[root]# mkdir -p \
/opt/mws/etc/mws.d \
/opt/mws/hooks \
/opt/mws/log \
/opt/mws/plugins \
/opt/mws/spool/hooks \
/opt/mws/utils
[root]# chown -R tomcat:tomcat /opt/mws
[root]# chmod -R 555 /opt/mws
[root]# chmod u+w \
/opt/mws/log \
/opt/mws/plugins \
/opt/mws/spool \
/opt/mws/spool/hooks \
/opt/mws/utils
```

4. Download the latest MWS build (`mws-<version>.tar.gz`) from the [Adaptive Computing](#) website.

i The variable marked `<version>` is the desired version of the suite; for example, `9.0.1`.

5. Extract the contents of the MWS download tarball into a temporary directory. For example:

```
[root]# mkdir /tmp/mws-install
[root]# cd /tmp/mws-install
[root]# tar xvzf $HOME/Downloads/mws-9.0.1.tar.gz
```

6. Copy the extracted utility files to the utility directory created in the previous step and give the tomcat user ownership of the directory.

```
[root]# cd /tmp/mws-install/mws-9.0.1/utils
[root]# cp * /opt/mws/utils
[root]# chown tomcat:tomcat /opt/mws/utils/*
```

7. Connect Moab to MongoDB.

i The `USEDATABASE` parameter is unrelated to the MongoDB configuration.

- a. Set the `MONGOSERVER` parameter in `/opt/moab/etc/moab.cfg` to the MongoDB server hostname. Use `localhost` as the hostname if Moab and MongoDB are hosted on the same server.

```
MONGOSERVER <host>[:<port>]
```

If your **MONGOSERVER** host is set to anything other than localhost, edit the `/etc/mongod.conf` file on the MongoDB server host and either comment out any `bind_ip` parameter or set it to the correct IP address.

```
# Listen to local interface only. Comment out to listen on all interfaces.
#bind_ip=127.0.0.1
```

- b. In the `/opt/moab/etc/moab-private.cfg` file, set the **MONGOUSER** and **MONGOPASSWORD** parameters to the MongoDB `moab_user` credentials you set. See [Install MongoDB on page 57](#).

```
MONGOUSER      moab_user
MONGOPASSWORD  secret2
```

- c. Verify that Moab is able to connect to MongoDB.

- Red Hat 6-based systems

```
[root]# service moab restart
[root]# mdiag -S

Mongo connection (localhost) is up (credentials are set)
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl restart moab.service
[root]# mdiag -S

Mongo connection (localhost) is up (credentials are set)
```

8. Secure communication using secret keys.

- a. (Required) Moab and MWS use Message Authentication Codes (MAC) to ensure messages have not been altered or corrupted in transit. Generate a key and store the result in `/opt/moab/etc/.moab.key`.

- Red Hat 6-based systems

```
[root]# service moab stop
[root]# dd if=/dev/urandom count=24 bs=1 2>/dev/null | base64 >
/opt/moab/etc/.moab.key
[root]# chown root:root /opt/moab/etc/.moab.key
[root]# chmod 400 /opt/moab/etc/.moab.key
[root]# service moab start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl stop moab.service
[root]# dd if=/dev/urandom count=24 bs=1 2>/dev/null | base64 >
/opt/moab/etc/.moab.key
[root]# chown root:root /opt/moab/etc/.moab.key
[root]# chmod 400 /opt/moab/etc/.moab.key
[root]# systemctl start moab.service
```

b. (Optional) Moab supports message queue security using AES. This feature requires a Base64-encoded 16-byte (128-bit) shared secret. Do the following:

i. Generate a key and append the result to `/opt/moab/etc/moab-private.cfg`

- Red Hat 6-based systems

```
[root]# service moab stop
[root]# echo "MESSAGEQUEUESECRETKEY $(dd if=/dev/urandom count=16 bs=1
2>/dev/null | base64)" >> /opt/moab/etc/moab-private.cfg
[root]# service moab start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl stop moab.service
[root]# echo "MESSAGEQUEUESECRETKEY $(dd if=/dev/urandom count=16 bs=1
2>/dev/null | base64)" >> /opt/moab/etc/moab-private.cfg
[root]# systemctl start moab.service
```

i If MWS is configured to encrypt the message queue and Moab is not (or vice versa), then MWS will ignore the messages from Moab. Furthermore, all attempts to access the MWS service resource will fail.

ii. Verify that encryption is on for the ZeroMQ connection.

```
[root]# mdiag -S|grep 'ZeroMQ MWS'
ZeroMQ MWS connection is bound on port 5570 (encryption is on)
```

9. Set up the MWS configuration files. In the extracted directory are several configuration files.

a. Copy the configuration files into place and grant the tomcat user ownership.

```
[root]# cd /tmp/mws-install/mws-9.0.1
[root]# cp mws-config.groovy /opt/mws/etc
[root]# cp mws-config-hpc.groovy /opt/mws/etc/mws.d
[root]# chown tomcat:tomcat /opt/mws/etc/mws-config.groovy
/opt/mws/etc/mws.d/mws-config-hpc.groovy
```

- b. In the `/opt/mws/etc/mws-config.groovy` file, change these settings:
- **moab.secretKey:** Must match the Moab secret key you generated earlier (contained in `/opt/moab/etc/.moab.key`).
 - **auth.defaultUser.username:** Any value you like, or leave as is.
 - **auth.defaultUser.password:** Any value you like, but choose a strong password.
 - **moab.messageQueue.secretKey:** If you opted to configure a message queue security key in MWS, this parameter value should match exactly that key specified in `/opt/moab/etc/moab-private.cfg` for the `MESSAGEQUEUESECRETKEY` Moab configuration parameter you generated earlier.

 If MWS is configured to encrypt the message queue and Moab is not (or vice versa), then the messages from Moab will be ignored. Furthermore, all attempts to access the MWS service resource will fail.

```
[root]# vi /opt/mws/etc/mws-config.groovy

// Replace <ENTER-KEY-HERE> with the contents of /opt/moab/etc/.moab.key.

moab.secretKey = "<ENTER-KEY-HERE>"
moab.server = "localhost"
moab.port = 42559

// Replace <ENTER-KEY-HERE> with the value of MESSAGEQUEUESECRETKEY in
/opt/moab/etc/moab-private.cfg.

moab.messageQueue.secretKey = "<ENTER-KEY-HERE>"

// Change these to be whatever you like.

auth.defaultUser.username = "moab-admin"
auth.defaultUser.password = "changeme!"
```

 If you do not change **auth.defaultUser.password**, your MWS will not be secure (because anyone reading these instructions would be able to log into your MWS). Here are some [tips](http://www.us-cert.gov/cas/tips/ST04-002.html) (<http://www.us-cert.gov/cas/tips/ST04-002.html>) for choosing a good password.

- c. If you are using Moab Accounting Manager, change these settings in `/opt/mws/etc/mws.d/mws-config-hpc.groovy`:
- **mam.secretKey:** needs to match the MAM secret key in `/opt/mam/etc/mam-site.conf` on the MAM Server (as **token.value**)
 - **mam.server:** set to the hostname of the MAM Server
 - **mam.port:** set to the port of the MAM Server

```
[root]# vi /opt/mws/etc/mws.d/mws-config-hpc.groovy
mam.secretKey = "<ENTER-KEY-HERE>"
mam.server = "localhost"
mam.port = 7112
```

d. Do *one* of the following:

- i** You can configure only one authentication method in `mws-config.groovy`—LDAP or PAM, but not both. If you have configured both LDAP and PAM, MWS defaults to using LDAP.

If you need multiple authentication methods, you must add them to your local PAM configuration. See your distribution documentation for details.

- If you are configuring an MWS connection to your LDAP server, add the following parameters to the `mws-config.groovy` file:

```
ldap.server = "192.168.0.5"
ldap.port = 389
ldap.baseDNs = ["dc=acme,dc=com"]
ldap.bindUser = "cn=Manager,dc=acme,dc=com"
ldap.password = "*****"
ldap.directory.type = "OpenLDAP Using InetOrgPerson Schema"
```

This is just an example LDAP connection. Be sure to use the appropriate domain controllers (dc) and common names (cn) for your environment.

- i** If you followed the Adaptive Computing tutorial, [Setting Up OpenLDAP on CentOS 6](#), your `ldap.directory.type` should be set to "OpenLDAP Using InetOrgPerson Schema." However, the use of other schemas is supported. For more information see [LDAP Configuration Using mws-config.groovy](#).

- i** To see how to configure a secure connection to the LDAP server, see [Securing the LDAP Connection](#).

- If you are configuring MWS to use PAM, add the `pam.configuration.service` parameter to the `mws-config.groovy` file. For example:

```
pam.configuration.service = "login"
```

This is just an example PAM configuration file name. Make sure you specify the name of the configuration file you want MWS to use.

⚠ Configuring MWS to authenticate via PAM using local `passwd` and `shadow` files presents a significant security risk. To make local authentication work, you would need to run Tomcat as root or give Tomcat read access to `/etc/shadow`. This configuration is highly discouraged and is not supported by Adaptive Computing.

The recommended approach is to configure PAM and NSS to authenticate against NIS or LDAP. For example, to make sure users with both local and NIS accounts are authenticating against NIS, configure the `nsswitch.conf` file as shown below.

```
passwd: nis files
shadow: nis files
group:  nis files
```

i For more information about PAM configuration with MWS, see [PAM \(Pluggable Authentication Module\) Configuration Using mws-config.groovy](#).

- e. Add the `grails.mongo.username` and `grails.mongo.password` parameters to the `mws-config.groovy` file. Use the MWS credentials you added to MongoDB in the [Preparing for Manual Installation](#) section.

```
...
grails.mongo.username = "mws_user"
grails.mongo.password = "secret3"
```

- f. Make the MWS configuration files read-only.

```
[root]# chmod 400 /opt/mws/etc/mws-config.groovy /opt/mws/etc/mws.d/mws-config-hpc.groovy
```

10. Configure Tomcat

Add the following lines to the end of `/etc/tomcat/tomcat.conf`.

```
CATALINA_OPTS="-DMWS_HOME=/opt/mws -Xms256m -Xmx3g -XX:MaxPermSize=384m -Dfile.encoding=UTF8"
JAVA_HOME="/usr/java/latest"
```

i `MaxPermSize` is ignored using Java 8; and therefore can be omitted.

11. Deploy the `mws.war` file and start Tomcat.

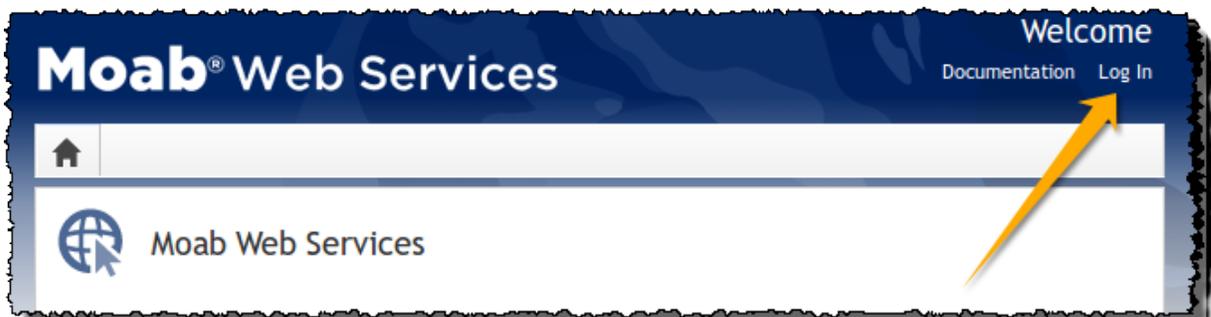
- Red Hat 6-based systems

```
[root]# chkconfig tomcat on
[root]# service tomcat stop
[root]# cp /tmp/mws-install/mws-9.0.1/mws.war /usr/share/tomcat/webapps
[root]# service tomcat start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable tomcat.service
[root]# systemctl stop tomcat.service
[root]# cp /tmp/mws-install/mws-9.0.1/mws.war /usr/share/tomcat/webapps
[root]# systemctl start tomcat.service
```

12. Navigate to `http://<server>:8080/mws/` in a web browser to verify that MWS is running (you will see some sample queries and a few other actions).
13. Log in to MWS to verify that your credentials are working. (Your login credentials are the `auth.defaultUser.username` and `auth.defaultUser.password` values you set in the `/opt/mws/etc/mws-config.groovy` file.)



i If you encounter problems, or if the application does not seem to be running, see the steps in [Moab Web Services Issues on page 269](#).

Related Topics

[Preparing for Manual Installation on page 22](#)

Installing Moab Insight

Manual installation is not available for Insight.

See [Installing Moab Insight on page 165](#) for detailed instructions on installing the Insight RPM.

i Adaptive Computing *requires* the Insight Server and the Moab Workload Manager Server to run on different hosts.

Installing Moab Viewpoint

Manual installation is not available for Viewpoint.

See [Installing Moab Viewpoint on page 178](#) for detailed instructions on installing the Viewpoint RPM.

Installing RLM Server

Access to a Reprise License Manager (RLM) server is required when using Remote Visualization and/or Nitro.

 The RLM Server can run multiple licenses. If your company already uses an RLM Server, you do not need to install a new one for Remote Visualization or Nitro. However, Remote Visualization and Nitro will use a different port than the default RLM Server port (5053). Skip this topic and follow the instructions in [Installing Remote Visualization on page 198](#) or [Installing Nitro on page 73](#) as applicable.

This topic contains instructions on how to install an RLM Server.

In this topic:

- [Open Necessary Ports on page 68](#)
- [Install the RLM Server on page 70](#)
- [Change the Default Passwords on page 71](#)

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

 These instructions assume you are using the default ports. If your configuration will use other ports, then substitute your port numbers when opening the ports.

On the RLM Server do the following:

1. Open the RLM Server port (5053) and the RLM Web Interface port (5054).
 - Red Hat 6-based systems using iptables

```
[root]# iptables -A INPUT -p tcp --dport 5053 -j ACCEPT
[root]# iptables -A INPUT -p tcp --dport 5054 -j ACCEPT
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=5053-5054/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5053 5054"
[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5053 5054"
[root]# service SuSEfirewall2 restart
```

2. If Remote Visualization is part of your configuration, open the Remote Visualization port (57889).

- Red Hat 6-based systems using iptables

```
[root]# iptables -A INPUT -p tcp --dport 57889 -j ACCEPT
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=57889/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="57889"
[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="57889"
[root]# service SuSEfirewall2 restart
```

3. If Nitro is part of your configuration, open the ISV adaptiveco port for the Adaptive license-enabled products (for example: 5135).

- Red Hat 6-based systems using iptables

```
[root]# iptables -A INPUT -p tcp --dport 5135 -j ACCEPT
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=5135/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5135"
[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5135"
[root]# service SuSEfirewall2 restart
```

Install the RLM Server

On the host where the RLM Server will reside, do the following:

1. If you have not already done so, extract the RLM tarball from the Nitro Tarball Bundle.

i Nitro is not required to install the RLM Server. For example, you need the RLM Server if using Remote Visualization, which also does not require Nitro. However, since Nitro cannot run without access to an RLM Server, the RLM tarball has been bundled with the other tarballs needed when using Nitro.

- a. Download and unpack the Nitro Tarball Bundle. See [Preparing for Nitro Manual Installation on page 72](#)
 - b. Identify the RLM tarball (`ac-rlm-<version>.tar.gz`).
2. Create a non-root user and group (rlm is used in the example).

```
[root]# groupadd -r rlm
[root]# useradd -r -g rlm -d /opt/rlm -c "A non-root user under which to run Reprise License Manager" rlm
```

3. Create a directory and install the tarball files in that location (we are using `/opt/rlm` as the install location in the example).

```
[root]# mkdir -p -m 0744 /opt/rlm
[root]# cd /opt/rlm
[root]# tar -xzf /tmp/ac-rlm-<version>.tar.gz --strip-components=1
[root]# chown -R rlm:rlm /opt/rlm
```

i The `--strip-components=1` removes the `"ac-rlm-<version>/"` from the relative path so that they are extracted into the current directory.

4. Install the startup scripts.

i If you are using a user:group other than `rlm:rlm` or a location other than `/opt/rlm`, then edit the following files to reflect those changes after copying them.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# cp init.d/rlm /etc/init.d
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# cp systemd/rlm.service /etc/systemd/system
```

5. Start the services and configure the RLM Server to start automatically at system reboot.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# chkconfig --add rlm
[root]# chkconfig rlm on
[root]# service rlm start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable rlm.service
[root]# systemctl start rlm.service
```

Change the Default Passwords

The RLM Web interface includes two usernames (admin and user) by default. These usernames have the default password "changeme!".

⚠ If you do not change this password, RLM, and Remote Visualization, will not be secure. For tips on choosing a good password, see <https://www.us-cert.gov/ncas/tips/ST04-002>.

Do the following for *both* the user and the admin usernames:

1. Using a web browser, navigate to your RLM instance (`http://<RLM_host>:5054`; where `<RLM_host>` is the IP address or name of the RLM Server Host).
2. Log in.
3. Select **Change Password** and change the password according to your password security process.

i The password for "user" will be needed as part of the Remote Visualization installation.

Nitro Integration

This section provides instructions on integrating Nitro as part of your Moab HPC Suite configuration.

- [Preparing for Nitro Manual Installation on page 72](#)
- [Installing Nitro on page 73](#)
- [Installing Nitro Web Services on page 77](#)

Preparing for Nitro Manual Installation

This topic contains instructions on how to download and unpack the Nitro Tarball Bundle for all the hosts in your configuration.

i Whether you are installing tarballs on one host or on several hosts, each host (physical machine) on which a server is installed (RLM Server, Nitro, Nitro Web Services) *must* have the Nitro Tarball Bundle.

⚠ Nitro Web Services is currently not available for SUSE 11-based systems; it is not in the Nitro Tarball Bundle for that OS.

Set Up Proxies

If your site uses a proxy to connect to the Internet, do the following:

```
export http_proxy=http://<proxy_server_id>:<port>
export https_proxy=http://<proxy_server_id>:<port>
```

Download and Unpack the Nitro Tarball Bundle

The Nitro Tarball Bundle contains all the tarballs available for Nitro. However, not every tarball may be installed on the same host.

On each host (physical machine), do the following:

1. Using a web browser, navigate to the [Adaptive Computing Nitro Download](http://www.adaptivecomputing.com/support/download-center/nitro/) website (<http://www.adaptivecomputing.com/support/download-center/nitro/>).
2. Download the Nitro Tarball Bundle (`nitro-tarball-bundle-<version>-<OS>.tar.gz`).

i The variable marked <version> indicates the build's version, revision, and changeset information. The variable marked <OS> indicates the OS for which the build was designed.

3. Unpack the Nitro Tarball Bundle.

```
[root]# tar xzvf nitro-tarball-bundle-<version>-<OS>.tar.gz
```

Related Topics

- [Nitro Integration on page 72](#)

Installing Nitro

This topic contains instructions on how to install Nitro, including licensing and configuration information.

Nitro

- needs to be available to all of the nodes that will be used as part of the Nitro job.
- can be installed either to each node individually *or* to a shared file system that each node can access.
- can be installed to integrate with a scheduler, such as Moab, or without (Nitro standalone). The instructions are the same.

In this topic:

- [Obtain a Nitro License on page 73](#)
- [Open Necessary Ports on page 75](#)
- [Install Nitro on page 76](#)
- [Verify Network Communication on page 77](#)

Obtain a Nitro License

The Nitro license file is installed on an RLM Server.

⚠ These instructions assume you already have access to an RLM Server. See [Installing RLM Server on page 68](#) for instructions on how to set up a new RLM Server.

Do the following:

1. On the RLM server, obtain the hostid and hostname.

- hostid

```
[root]# /opt/rlm/rlmhostid
```

You should see output similar to the following.

```
rlmhostid v12.0
Copyright (C) 2006-2015, Reprise Software, Inc. All rights reserved.

Hostid of this machine: 00259096f004
```

- hostname

```
[root]# /opt/rlm/rlmhostid host
```

You should see output similar to the following.

```
rlmhostid v12.0
Copyright (C) 2006-2015, Reprise Software, Inc. All rights reserved.

Hostid of this machine: host=<your-host-name>
```

2. Email licenses@adaptivecomputing.com for a license and include the hostid and hostname you just obtained.
3. Adaptive Computing will generate the license and send you the Nitro license file (.lic) file in a return email.
4. On the RLM server, do the following:

- a. Download and install the license file.

```
[root]# cd /opt/rlm
[root]# chown rlm:rlm <licenseFileName>.lic
```

- b. If the RLM Server in your configuration uses a firewall, edit the license file to reference the ISV adaptiveco port for the Adaptive license-enabled products. This is the same port number you opened during the RLM Server installation. See the instructions to open necessary ports in the [Installing RLM Server on page 68](#) (manual installation method) or [Installing RLM Server on page 195](#) (RPM installation method) for more information.

```
[root]# vi /opt/rlm/nitro.lic

ISV adaptiveco port=5135
```

The license file already references the RLM Server port (5053 by default).

i If the RLM Server in your configuration uses different ports, you will need to modify the license file to reflect the actual ports. See the instructions to open necessary ports in the [Installing RLM Server on page 68](#) (manual installation method) or [Installing RLM Server on page 195](#) (RPM installation method) for more information.

- c. If you did *not* install an RLM Server using the file available from Adaptive Computing (for example, because your system configuration already uses one), do the following:
 - i. If you have not already done so, download and unpack the Nitro - Tarball Bundle (`nitro-tarball-bundle-<version>-<OS>.tar.gz`).
 - ii. Identify and unpack the RLM tarball (`ac-rlm-<version>.tar.gz`).
 - iii. Locate the 'adaptiveco.set' file and copy it into the same directory where the Nitro license resides (`/opt/rlm`).
- d. Perform a reread to update the RLM Server with you license.

```
[root]# /opt/rlm/rlmreread
```

Open Necessary Ports

Nitro uses several ports for communication between the workers and the coordinator.

The default port is 47000, and up to four ports are used in running Nitro (ports 47000-47003).

On each compute node (coordinator), open the necessary ports.

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 47000 -j ACCEPT
-A INPUT -p tcp --dport 47001 -j ACCEPT
-A INPUT -p tcp --dport 47002 -j ACCEPT
-A INPUT -p tcp --dport 47003 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=47000-47003/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="47000 47001 47002 47003"
[root]# service SuSEfirewall2_setup restart
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="47000 47001 47002 47003"
[root]# service SuSEfirewall2 restart
```

Install Nitro

i You *must* complete the tasks to obtain a Nitro license before installing Nitro. See [Obtain a Nitro License on page 73](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing Nitro. See [Open Necessary Ports on page 75](#).

On the host where Nitro will reside, do the following:

1. If you have not already done so, complete the steps to prepare the host. See [Preparing for Nitro Manual Installation on page 72](#).
2. Identify the Nitro product tarball (nitro-*<version>*-*<OS>*.tar.gz).
3. Run each of the following commands in order.

```
[root]# mkdir /opt/nitro
[root]# tar xzvpf nitro-<version>-<OS>.tar.gz -C /opt/nitro --strip-components=1
```

4. Copy the license file you generated earlier in this topic to each compute node (coordinator). On each compute node, *or* on the shared file system, do the following:

```
[root]# cp <licenseFileName>.lic /opt/nitro/bin/
```

5. Identify the `launch.nitro.sh` script version for your resource manager. This script will be copied to the bin directory from where user job scripts will execute Nitro. See the *Nitro Administrator Guide* for more information.

Reference scripts are provided in `/opt/nitro/scripts`.

```
[root]# find /opt/nitro -name launch_nitro.sh
./scripts/lsf/launch_nitro.sh
./scripts/torque/launch_nitro.sh
./scripts/slurm/launch_nitro.sh
./scripts/alps/torque/launch_nitro.sh
./scripts/alps/slurm/launch_nitro.sh
```

6. Copy the launch script to the bin directory. (This example uses the Torque-based launch script.)

```
[root]# cp /opt/nitro/scripts/torque/launch_nitro.sh /opt/nitro/bin/
```

i This is a "copy" file operation and not a "move" operation. This allows you to customize your version of the script and always have the factory version available for consultation and/or comparison.

7. Customize the bin/launch.nitro.sh script as needed for your site's administrative policies. For example, to enable the Nitro coordinator's host to always execute a local Nitro worker, modify the bin/launch.nitro.sh script version to always pass the `--run-local-worker` command line option to the coordinator. See the *Nitro Administrator Guide* for more information on editing the launch script.
8. If you are *not* using a shared file system, copy the Nitro installation directory to *all* hosts.

```
[root]# scp -r /opt/nitro root@host002:/opt
```

Verify Network Communication

Verify that the nodes that will be running Nitro are able to communicate with the Nitro ports *and* that the nodes are able to communicate with one another.

Related Topics

- [Nitro Integration on page 72](#)

Installing Nitro Web Services

This topic contains instructions on how to install Nitro Web Services.

Do the following in the order presented:

1. [Open Necessary Ports](#)
2. [Install MongoDB](#)
3. [Install and Configure Nitro Web Services](#)
4. [Configure Viewpoint for Nitro Web Services](#)
5. [Configure Nitro Compute Nodes](#)

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

In this section:

- [Open the Tornado Web Port \(9443\) and the ZMQ Port \(47100\) on page 78](#)
- [Open the MongoDB Database Port \(27017\) on page 78](#)

Open the Tornado Web Port (9443) and the ZMQ Port (47100)

On the Nitro Web Services Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 9443 -j ACCEPT
-A INPUT -p tcp --dport 47100 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=9443/tcp --permanent
[root]# firewall-cmd --add-port=47100/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="9443 47100"

[root]# service SuSEfirewall2 restart
```

Open the MongoDB Database Port (27017)

i Nitro Web Services requires access to a MongoDB database. Depending on your system configuration, your MongoDB databases may not be installed on the same host as their corresponding component servers. For example, you may choose to install the Nitro Web Services MongoDB on the same host where you have installed other MongoDB databases.

Do the following, as needed:

- If you have chosen to install the Nitro Web Services MongoDB database on the *same* host you installed other MongoDB databases, confirm the firewall port (27017) is already opened on that host.
- If you have chosen to install the Nitro Web Services MongoDB database on a *different* host from other MongoDB databases, you will need to open the Nitro Web Services MongoDB database port in the firewall for that host. To open the port in the firewall, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 27017 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=27017/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="27017"

[root]# service SuSEfirewall2 restart
```

Install MongoDB

To install and enable MongoDB on the Nitro Web Services Host, do the following:

1. Install MongoDB.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# cat > /etc/yum.repos.d/mongodb.repo <<End-of-file
[mongodb]
name=MongoDB Repository
baseurl=http://downloads-distro.mongodb.org/repo/redhat/os/x86_64
gpgcheck=0
enabled=1
exclude=mongodb-org mongodb-org-server
End-of-file
[root]# yum install mongo-10gen-server
```

- SUSE 12-based systems

```
[root]# zypper ar --refresh -r
http://download.opensuse.org/repositories/server:/database/SLE_
12/server:database.repo
[root]# zypper install mongodb
```

2. Start MongoDB

i There may be a short delay (approximately 3 minutes) for Mongo to start the first time.

- Red Hat 6-based systems

```
[root]# chkconfig mongod on
[root]# service mongod start
```

- Red Hat 7-based systems

```
[root]# cat > /usr/lib/systemd/system/mongodb.service <<End-of-file
[Unit]
Description=High-performance, schema-free document-oriented database
After=syslog.target network.target

[Service]
Type=forking
User=mongod
Group=mongod
Environment=CONFIG=/etc/mongod.conf
Environment=OPTIONS=
EnvironmentFile=-/etc/sysconfig/mongod
ExecStart=/usr/bin/mongod -f \${CONFIG} \${OPTIONS}
PrivateTmp=true
LimitNOFILE=65536
TimeoutStartSec=180
StandardOutput=syslog
StandardError=syslog

[Install]
WantedBy=multi-user.target
End-of-file
[root]# rm -f /etc/init.d/mongod
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
[root]# systemctl daemon-reload
```

- SUSE 12-based systems

```
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
[root]# systemctl daemon-reload
```

3. Prepare the MongoDB database by doing the following:

a. Add the required MongoDB users.

i The password used below (secret1) is an example. Choose your own password for this user.

```
[root]# mongo
> use admin;
> db.addUser("admin_user", "secret1");
> db.auth("admin_user", "secret1");
> exit
```

i Because the `admin_user` has read and write rights to the `admin` database, it also has read and write rights to all other databases. See [Control Access to MongoDB Instances with Authentication](http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication/) (<http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication/>) for more information.

b. Enable authentication in MongoDB.

- Red Hat 6-based systems

```
[root]# vi /etc/mongod.conf
auth = true
[root]# service mongod restart
```

- Red Hat 7-based systems

```
[root]# vi /etc/mongod.conf
auth = true
[root]# systemctl restart mongod.service
```

- SUSE 12-based systems

MongoDB authentication is already enabled. No further action is needed.

Install and Configure Nitro Web Services

i You *must* complete the tasks earlier in this topic before installing Nitro Web Services.

On the host where Nitro Web Services will reside, do the following:

1. If you have not already done so, complete the steps to prepare the host. See [Preparing for Nitro Manual Installation on page 72](#).
2. Identify and unpack the Nitro Web Services tarball (`nitro-web-services-<version>-<OS>.tar.gz`). This step assumes the Nitro Web Services tarball was downloaded to `/tmp` and the installation directory is `/opt/nitro-web-services`.

```
[root] cd /tmp
# Create the directory where the tarball will be extracted/installed
[root] mkdir -p /opt/nitro-web-services
# Extract the tarball contents into the directory
[root] tar -xvzpf nitro-web-services-<version>-<OS>.tar.gz -C /opt/nitro-web-
services --strip-components=1
```

3. Install Nitro Web Services. This step assumes the installation directory is `/opt/nitro-web-services`.

```
[root] cd /opt/nitro-web-services
[root] ./install.sh
```

4. Understand and edit the configuration files, including clarify what each configuration file is for and what to expect the first time the NWS service is started vs. each subsequent start.

MongoDB user, table, and index creation is performed at initial startup. Many of the options defined in the Nitro Web Service configuration files influence Mongo user/password and index creation.

 **Warning:** Usernames and passwords are created *only* if they do not yet exist. Changing a password in the configuration file after initial startup will not update the password in Mongo.

The installation provides two configuration files.

- `/opt/nitro-web-services/etc/nitro.cfg`

This is the Nitro Web Services web application configuration file.

Before initial startup, set "admin_username" and "admin_password" to the MongoDB admin username and password you used when setting up MongoDB. It is also recommended that you change all other default passwords before starting Nitro Web Services. If you do not change the passwords at this point, it will be more difficult to change them later.

By default, NWS uses an auto-generated self-signed SSL certificate. However, you can use your own certfile, keyfile, and ca_certs files if you wish.

- `/opt/nitro-web-services/etc/zmq_job_status_adapter.cfg`

This is the Nitro ZMQ Job Status Adapter configuration file.

The Nitro ZMQ Job Status Adapter listens to job status updates on the ZQM bus and publishes them to MongoDB using the Nitro Web Services REST API. The username and password must be set to a MongoDB user with write permissions. At minimum, set the password for nitro-writeonly-user to the password defined in `/opt/nitro-web-services/etc/nitro.cfg` and make sure the SSL options are set correctly based on SSL settings in `/opt/nitro-web-services/etc/nitro.cfg`.

5. Start the services and configure Nitro Web Services to start automatically at system boot.

- Red Hat 6-based systems

```
[root] chkconfig --add nitro-web-services
[root] chkconfig --add nitro-zmq-job-status-adapter
[root] service nitro-web-services start
[root] service nitro-zmq-job-status-adapter start
```

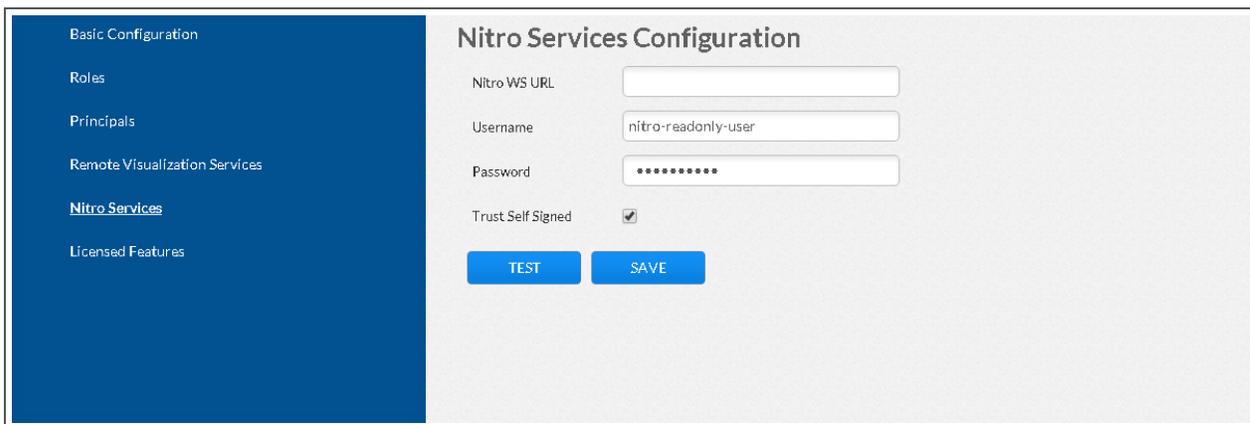
- Red Hat 7-based or SUSE 12-based systems

```
[root] systemctl enable nitro-web-services.service
[root] systemctl enable nitro-zmq-job-status-adapter.service
[root] systemctl daemon-reload
[root] systemctl start nitro-web-services.service
[root] systemctl start nitro-zmq-job-status-adapter.service
```

Configure Viewpoint for Nitro Web Services

Do the following:

1. Using a web browser, navigate to your Viewpoint instance (`http://<server>:8081`) and then log in as the MWS administrative user (`moab-admin`, by default).
2. Click **Configuration** from the menu and then click **Nitro Services** from the left pane. The following is an example of the Nitro Services Configuration page.



3. Enter the configuration information. The following table describes the required information.

| Field | Description |
|-------------------|--|
| Nitro WS URL | Hostname (or IP address) and port number for the host on which you installed Nitro Web Services. For example, <code>https://<hostname>:9443</code> |
| Username | Name of the user. This typically <code>nitro-readonly-user</code> . |
| Password | The user's password. |
| Trust Self Signed | Indicates whether Nitro Web Services was set up using self-signed certificates. |

4. Click **TEST** to confirm the settings are correct. This confirms whether Nitro Web Services is up and receiving connections.
5. Click **SAVE**.
6. (Recommended) Use curl to test Nitro Web Services connectivity.

```
curl --insecure --data '{"username": "nitro-admin", "password": "ChangeMe2!"}' \
https://<hostname>:9443/auth
```

You should see output similar to the following.

```
{
  "status": 200,
  "data": {
    "nitro-key": "3e0fb95e9a0e44ae91daef4deb500dcc67a3714880e851d781512a49",
    "user": {
      "username": "nitro-admin",
      "last_updated": "2016-02-26 23:34:55.604000",
      "name": "Nitro Admin",
      "created": "2016-02-26 23:34:55.604000",
      "auth": {
        "job": [
          "read",
          "write",
          "delete"
        ],
        "user": [
          "read",
          "write",
          "delete"
        ]
      }
    }
  }
}
```

Configure Nitro Compute Nodes

You need to configure the Nitro coordinators to send job status updates to the Nitro Web Services's ZMQ Job Status Adapter. The ZMQ Job Status Adapter is responsible for reading job status updates off of the ZMQ bus and persisting them to Mongo. Nitro Web Services can then be used to access Nitro job status.

Each Nitro job has a Nitro Coordinator. Nitro Coordinators can be configured to publish job status updates to ZMQ by setting the "nws-connector-address" configuration option in Nitro's nitro.cfg file. Each compute node allocated/scheduled to a Nitro Job can play the role of a Nitro coordinator. Therefore, you must update the "nws-connector-address" in each compute node's nitro.cfg file.

i Configuring nws-connector-address is simplified if each node is sharing Nitro's configuration over a shared filesystem. If you are *not* using a shared filesystem, update the Nitro configuration on *each* compute node.

Do the following:

1. If you have not already done so, on the Nitro Web Services host, locate the `msg_port` number in the `/opt/nitro-web-services/etc/zmq_job_status_adapter.cfg` file. This is the port number you need to specify for the `nws-connector-address`.
2. On *each* Nitro compute node (Torque MOM Host), specify the `nws-connector-address` in the `/opt/nitro/etc/nitro.cfg` file.

```
...
# Viewpoint connection allows Nitro to communicate job status information
# to viewpoint. This option indicates name and port of the remote server
# in the form: <host>:<port>
nws-connector-address <nitro-web-services-hostname>:47100
...
```

Related Topics

- [Nitro Integration on page 72](#)

Additional Configuration

In this section:

- [Configuring SSL in Tomcat on page 231](#)
- [Setting Up OpenLDAP on CentOS 6 on page 231](#)
- [Moab Workload Manager Configuration Options on page 93](#)
- [Moab Accounting Manager Configuration Options on page 94](#)
- [Trusting Servers in Java on page 238](#)

Configuring SSL in Tomcat

To configure SSL in Tomcat, please refer to the Apache Tomcat [documentation](http://tomcat.apache.org/tomcat-7.0-doc/ssl-howto.html) (<http://tomcat.apache.org/tomcat-7.0-doc/ssl-howto.html>).

Setting Up OpenLDAP on CentOS 6

These instructions are intended to help first-time LDAP administrators get up and running. The following procedures contain instructions for getting started using OpenLDAP on a CentOS 6 system. For more complete information on how to set up OpenLDAP see the [OpenLDAP documentation](http://www.openldap.org/doc/admin24/) (<http://www.openldap.org/doc/admin24/>).

In this topic:

- [Installing and Configuring OpenLDAP on Centos 6 on page 86](#)
- [Adding an Organizational Unit \(OU\) on page 91](#)
- [Adding a User on page 91](#)
- [Adding a Group on page 92](#)
- [Adding a User to a Group on page 92](#)

i Adaptive Computing is not responsible for creating, maintaining, or supporting customer LDAP or Active Directory configurations.

Installing and Configuring OpenLDAP on Centos 6

First, you will need to install OpenLDAP. These instructions explain how you can do this on a CentOS 6 system.

To install and configure OpenLDAP on Centos 6

1. Run the following command:

```
[root]# yum -y install openldap openldap-clients openldap-servers
```

2. Generate a password hash to be used as the admin password. This password hash will be used when you create the root user for your LDAP installation. For example:

```
[root]# slappasswd
New password : p@ssw0rd
Re-enter new password : p@ssw0rd
{SSHA}51PFVw19zeh7LT53hQH69znzj8TuBrLv
```

3. Add the root user and the root user's password hash to the OpenLDAP configuration in the `olcDatabase={2}bdb.ldif` file. The root user will have permissions to add other users, groups, organizational units, etc. Do the following:

- a. Run this command:

```
[root]# cd /etc/openldap/slapd.d/cn\=config
[root]# vi olcDatabase=\{2\}bdb.ldif
```

- b. If the **olcRootPW** attribute does not already exist, create it. Then set the value to be the hash you created from `slappasswd`. For example:

```
olcRootPW: {SSHA}51PFVw19zeh7LT53hQH69znzj8TuBrLv
...
```

4. While editing this file, change the distinguished name (DN) of the **olcSuffix** to something appropriate. The suffix typically corresponds to your DNS domain name, and it will be appended to the DN of every other LDAP entry in your LDAP tree.

For example, let's say your company is called Acme Corporation, and that your domain name is "acme.com". You might make the following changes to the `olcDatabase={2}bdb.ldif` file:

```
olcSuffix: dc=acme,dc=com
...
olcRootDN: cn=Manager,dc=acme,dc=com
...
olcRootPW: {SSHA}51PFVw19zeh7LT53hQH69znzj8TuBrLv
...
```



Do not set the cn of your root user to "root" (`cn=root,dc=acme,dc=com`), or OpenLDAP will have problems.

i Throughout the following examples in this topic, you will see `dc=acme,dc=com`. "acme" is only used as an example to illustrate what you would use as your own domain controller if your domain name was "acme.com". You should replace any references to "acme" with your own organization's domain name.

5. Modify the DN of the root user in the `olcDatabase={1}monitor.ldif` file to match the **olcRootDN** line in the `olcDatabase={2}bdb.ldif` file. Do the following:

- a. Run this command to edit the `olcDatabase={2}bdb.ldif` file:

```
[root]# vi olcDatabase=\{1\}monitor.ldif
```

- b. Modify the **olcAccess** line so that the **dn.base** matches the **olcRootDN** from the `olcDatabase={2}bdb.ldif` file. (In this example, **dn.base** should be `"cn=Manager,dc=acme,dc=com"`.)

```
olcAccess: {0}to * by
dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth" read by
dn.base="cn=Manager,dc=acme,dc=com" read by * none
```

- c. Now the root user for your LDAP is `cn=Manager,dc=acme,dc=com`. The root user's password is the password that you entered using `slappasswd` earlier in this procedure, which, in this example, is **p@ssw0rd**
6. Hide the password hashes from users who should not have permission to view them.

i A full discussion on configuring access control in OpenLDAP is beyond the scope of this tutorial. For help, see [the OpenLDAP Access Control documentation](http://www.openldap.org/doc/admin24/access-control.html) (<http://www.openldap.org/doc/admin24/access-control.html>).

- a. Run this command to edit the `olcDatabase=\{2\}bdb.ldif` file:

```
[root]# vi olcDatabase=\{2\}bdb.ldif
```

- b. Add the following two lines to the end of the file to restrict users from viewing other users' password hashes.

```
olcAccess: {0}to attrs=userPassword by self write by
dn.base="cn=Manager,dc=acme,dc=com" write by anonymous auth by * none
olcAccess: {1}to * by dn.base="cn=Manager,dc=acme,dc=com" write by self write by
* read
```

These lines allow a user to read and write his or her own password. It also allows a manager to read and write anyone's password. Anyone, including anonymous users, is allowed to view non-password attributes of other users.

7. Make sure that OpenLDAP is configured to start when the machine starts up, and start the OpenLDAP service.

```
[root]# chkconfig slapd on
[root]# service slapd start
```

8. Now, you must manually create the "dc=acme,dc=com" LDAP entry in your LDAP tree.

An LDAP directory is analogous to a tree. Nodes in this tree are called LDAP "entries" and may represent users, groups, organizational units, domain controllers, or other objects. The attributes in each entry are determined by the LDAP schema. In this tutorial we will build entries based on the InetOrgPerson schema (which ships with OpenLDAP by default).

In order to build our LDAP tree we must first create the root entry. Root entries are usually a special type of entry called a domain controller (DC). Because we are assuming that the organization is called Acme Corporation, and that the domain is "acme.com," we will create a domain controller LDAP entry called `dc=acme,dc=com`. Again, you will need to replace "acme" with your organization's domain name. Also note that `dc=acme,dc=com` is what is called an LDAP distinguished name (DN). An LDAP distinguished name uniquely identifies an LDAP entry.

Do the following:

- a. Create a file called `acme.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi acme.ldif
```

- b. Add the following lines in `acme.ldif`:

```
dn: dc=acme,dc=com
objectClass: dcObject
objectClass: organization
dc: acme
o : acme
```

- c. Now add the contents of this file to LDAP. Run this command:

```
[root]# ldapadd -f acme.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

- d. Verify that your entry was added correctly.

```
[root]# ldapsearch -x -LLL -b dc=acme,dc=com
dn: dc=acme,dc=com
objectClass: dcObject
objectClass: organization
dc: acme
o: acme
```

9. Run the following:

```
[root]# sudo iptables -L
[root]# sudo service iptables save
```

10. By default, the CentOS 6 firewall will block external requests to OpenLDAP. In order to allow MWS to access LDAP, you will have to configure your firewall to allow connections on port 389. (Port 389 is the default LDAP port.)

Configuring your firewall is beyond the scope of this tutorial; however, it may be helpful to know that the default firewall on CentOS is a service called `iptables`. For more information, see the documentation on [iptables](http://wiki.centos.org/HowTos/Network/IPTables) (<http://wiki.centos.org/HowTos/Network/IPTables>). In the most basic case, you may be able to add a rule to your firewall that accepts connections to port 389 by doing the following:

- a. Edit your `iptables` file:

```
[root]# vi /etc/sysconfig/iptables
```

- b. Add the following line *after* all the **ACCEPT** lines but *before* any of the **REJECT** lines in your `iptables` file:

```
# ... lines with ACCEPT should be above
-A INPUT -p tcp --dport 389 -j ACCEPT
# .. lines with REJECT should be below
```

For example, here is a sample `iptables` file with this line added:

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -p tcp --dport 389 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited

COMMIT
```

- c. Now reload `iptables`.

```
[root]# service iptables reload
```

i Although providing instructions is beyond the scope of this tutorial, it is also highly recommended that you set up OpenLDAP to use SSL or TLS security to prevent passwords and other sensitive data from being sent in plain text. For information on how to do this, see the [OpenLDAP TLS documentation](http://www.openldap.org/doc/admin24/tls.html) (<http://www.openldap.org/doc/admin24/tls.html>).

Now that you have installed and set up Open LDAP, you are ready to add organizational units. See [Adding an Organizational Unit \(OU\) on page 91](#).

Adding an Organizational Unit (OU)

These instructions will describe how to populate the LDAP tree with organizational units (OUs), groups, and users, all of which are different types of LDAP entries. The examples that follow also presume an InetOrgPerson schema, because the InetOrgPerson schema is delivered with OpenLDAP by default.

To add an organizational unit (OU) entry to the LDAP tree

In this example, we are going to add an OU called "Users".

1. Create a temporary file called `users.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi users.ldif
```

2. Add these lines to `users.ldif`:

```
dn: ou=Users,dc=acme,dc=com
objectClass: organizationalUnit
ou: Users
```

3. Add the contents of `users.ldif` file to LDAP.

```
[root]# ldapadd -f users.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

Adding a User

To add a user to LDAP

In this example, we will add a user named "Bob Jones" to LDAP inside the "Users" OU.

1. Create a temporary file called `bob.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi bob.ldif
```

2. Add these lines to `bob.ldif`:

```
dn: cn=Bob Jones,ou=Users,dc=acme,dc=com
cn: Bob Jones
sn: Jones
objectClass: inetOrgPerson
userPassword: p@ssw0rd
uid: bjones
```

3. Add the contents of `bob.ldif` file to LDAP.

```
[root]# ldapadd -f bob.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

Adding a Group

To add a group to LDAP

In this example, we will add a group called "Engineering" to LDAP inside the "Users" OU.

1. Create a temporary file called `engineering.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi engineering.ldif
```

2. Add these lines to `engineering.ldif`:

```
dn: cn=Engineering,ou=Users,dc=acme,dc=com
cn: Engineering
objectClass: groupOfNames
member: cn=Bob Jones,ou=Users,dc=acme,dc=com
```

3. Add the contents of `engineering.ldif` file to LDAP.

```
[root]# ldapadd -f engineering.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

Adding a User to a Group

To add a user to an LDAP group

In this example, we will add an LDAP member named "Al Smith" to the "Engineering" LDAP group. This example assumes that user, Al Smith, has already been added to LDAP.

i Before you add a user to an LDAP group, the user must first be added to LDAP. For more information, see [Adding a User on page 91](#).

1. Create a temporary file called `addUserToGroup.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi addUserToGroup.ldif
```

2. Add these lines to `addUserToGroup.ldif`:

```
dn: cn=Engineering,ou=Users,dc=acme,dc=com
changetype: modify
add: member
member: cn=Al Smith,ou=Users,dc=acme,dc=com
```

3. Now add the contents of `addUserToGroup.ldif` file to LDAP.

```
[root]# ldapadd -f addUserToGroup.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

Moab Workload Manager Configuration Options

The following is a list of commonly used configure options. For a complete list, use `./configure --help` when configuring Moab.

| Option | Description | Example |
|----------------------|---|--|
| --prefix | Specifies the location of the binaries and libraries of the Moab install. The default location is <code>/opt/moab</code> . | <pre>[root]# ./configure --prefix=/usr/local</pre> |
| --with-am | Specifies that you want to configure Moab with Moab Accounting Manager. <div style="border: 1px solid #0070c0; padding: 5px; margin-top: 10px;"> <p>i There is a similar <code>--with-torque</code> option that configures Moab with Torque, but you do not need to specify this option if you install the "torque" tarball version.</p> </div> | <pre>[root]# ./configure --with-am</pre> |
| --with-flexlm | Causes Moab to install the <code>license.-mon.flexLM.pl</code> script in the <code>/opt/moab/tools</code> directory. For more information about this script, see the Interfacing to FLEXlm section in the Moab Administrator Guide. | <pre>[root]# ./configure --with-flexlm</pre> |

| Option | Description | Example |
|--------------------------|---|---|
| --with-homedir | <p>Specifies the location of the Moab configuration directory and the MOABHOMEDIR environment variable. The default location is <code>/opt/moab</code>.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>i MOABHOMEDIR is automatically set on some distributions during installation, when the <code>--with-profile</code> option is enabled.</p> </div> | <pre>[root]# ./configure --with-homedir=/var/moab</pre> <p><i>The Moab HPC Suite home directory will be <code>/var/moab</code> instead of the default <code>/opt/moab</code>.</i></p> |
| --without-init | <p>Disables the installation of a distribution-specific, Moab service startup file. By default, <code>make install</code> will install an <code>init.d</code> or <code>systemd</code> service startup file as appropriate for your distribution. The installed file (<code>/etc/init.d/moab</code> or <code>/usr/lib/systemd/system/moab.service</code>) may be customized to your needs. If you do not want this file to be installed, use this option to exclude it.</p> | <pre>[root]# ./configure --without-init</pre> |
| --without-profile | <p>Disables the installation of a distribution-specific shell profile for <code>bash</code> and <code>C</code> shell. By default, <code>make install</code> will install the Moab shell initialization scripts as appropriate for your operating system. These scripts help to establish the <code>MOABHOMEDIR</code>, <code>PERL5LIB</code>, <code>PATH</code> and <code>MANPATH</code> environment variables to specify where the new moab configuration, scripts, binaries and man pages reside. The installed scripts (<code>/etc/profile.d/moab.{csh,sh}</code>) may be customized to your needs. If you do not want these scripts to be installed, use this option to exclude them.</p> | <pre>[root]# ./configure --without-profile</pre> |

Moab Accounting Manager Configuration Options

The following table comprises commonly-used configure options.

| Option | Description |
|------------------|---|
| -h,--help | Run <code>./configure --help</code> to see the list of configure options. |

| Option | Description |
|--|---|
| --localstatedir=DIR | Home directory where per-configuration subdirectories (such as <code>etc</code> , <code>log</code> , <code>data</code>) will be installed (defaults to <code>PREFIX</code>). |
| --prefix=PREFIX | Base installation directory where all subdirectories will be installed unless otherwise designated (defaults to <code>/opt/mam</code>). |
| --with-cgi-bin=DIR | If you intend to use the web GUI, use <code>--with-cgi-bin</code> to specify the directory where you want the Moab Accounting Manager CGI files to reside (defaults to <code>/var/www/cgi-bin/mam</code>). |
| --with-db-name=NAME | Name of the SQL database that the server will sync with (defaults to <code>mam</code>). |
| --with-legacy-links | Creates symbolic links allowing the use of the old client and server command names (for example, <code>mam-list-users</code> would be created as symbolic link to <code>mam-list-users</code>). When running a command under its old name, the command will issue a deprecation warning. This warning can be disabled by setting <code>client.deprecationwarning = false</code> in the <code>mam-client.conf</code> file. The default is not to install the legacy links. |
| --with-mam-libs=local site | Use <code>--with-mam-libs</code> to indicate whether you want to install the Perl MAM modules in a local directory (<code>\${exec_prefix}/lib</code>) or in the default system site-perl directory (defaults to <code>local</code>). |
| --with-promotion=mamauth suidperl | Command-line clients and scripts using the API need to use a security promotion method to authenticate and encrypt the communication using the symmetric key. The default is <code>suidperl</code> if it is installed on the system, otherwise the default is <code>mamauth</code> . See the description for the security.promotion configuration parameter in the Client Configuration section for more information about the two security promotion methods. |
| --with-sha=SHA SHA1 | Allows you to override the auto-detected SHA Digest Perl (whether <code>Digest::SHA1</code> or <code>Digest::SHA</code>) that should be used for your system. |
| --with-user=USER | Use <code>--with-user</code> to specify the accounting admin userid that the server will run under and who will have full administrative privileges (defaults to the user running the <code>configure</code> command). It is recommended that this be a non-privileged user for the highest security. |
| --without-gui | Specifies whether to install the CGI web GUI. If you do not intend to use the CGI web GUI, you can specify <code>--without-gui</code> to not install the CGI scripts. Otherwise, the default is to install the GUI CGI scripts. |

| Option | Description |
|--------------------------|---|
| --without-init | If you do not intend to use the <code>mam init.d</code> service, you can use <code>--without-init</code> to specify that Moab HPC Suite should not install the <code>mam init.d</code> script. Otherwise, the script is installed by default. |
| --without-profile | If you do not intend to use the <code>mam profile.d</code> environment scripts, you can use <code>--without-profile</code> to specify that Moab HPC Suite should not install the <code>mam profile.d</code> scripts. Otherwise, the scripts are installed by default. |

Trusting Servers in Java

In this topic:

[Prerequisites on page 96](#)

[Retrieve the Server's X.509 Public Certificate on page 96](#)

[Add the Server's Certificate to Java's Keystore on page 97](#)

Prerequisites

Some of these instructions refer to `JAVA_HOME`, which must point to the same directory that Tomcat uses. To set `JAVA_HOME`, do this:

```
[root]# source /etc/tomcat/tomcat.conf
```

Your system administrator might have defined Tomcat's `JAVA_HOME` in a different file.

Retrieve the Server's X.509 Public Certificate

To retrieve the server's certificate, use the following command:

```
[root]# $JAVA_HOME/bin/keytool -printcert -rfc -sslserver <servername>:<port> > /tmp/public.cert.pem
```

Replace `<servername>` with the server's host name and `<port>` with the secure port number. The default port for https is 443. The default port for ldaps is 636. If successful, `/tmp/public.cert.pem` contains the server's public certificate. Otherwise, `/tmp/public.cert.pem` contains an error message. This message is typical: `keytool error: java.lang.Exception: No certificate from the SSL server.` This message suggests that the server name or port is incorrect. Consult your IT department to determine the correct server name and port.

Add the Server's Certificate to Java's Keystore

Java stores trusted certificates in a database known as the keystore. Because each new version of Java has its own keystore, you need to add the server certificate to the Java keystore (using the steps below) every time you install a new version of Java.

Java's keystore is located at `$JAVA_HOME/lib/security/cacerts`. If Tomcat's `JAVA_HOME` points to a JDK, then the keystore is located at `$JAVA_HOME/jre/lib/security/cacerts`. To add the server certificate to the keystore, run the following command:

```
[root]# $JAVA_HOME/bin/keytool -import -trustcacerts -file /tmp/public.cert.pem -alias
<servername> -keystore $JAVA_HOME/lib/security/cacerts
```

You will be prompted for the keystore password, which is "changeit" by default.

i Your system administrator might have changed this password.

After you've entered the keystore password, you'll see the description of the server's certificate. At the end of the description it prompts you to trust the certificate.

```
Trust this certificate? [no]:
```

Type `yes` and press **Enter** to add the certificate to the keystore.

Manual Upgrade

This section provides instructions and other information when upgrading your Moab HPC Suite components using the Manual upgrade method.

In this section:

- [Preparing for Upgrade on page 98](#)
- [Upgrading Torque on page 99](#)
- [Upgrading Moab Workload Manager on page 105](#)
- [Upgrading Moab Accounting Manager on page 108](#)
- [Upgrading Moab Web Services on page 112](#)
- [Migrating the MAM Database from MySQL to PostgreSQL on page 263](#)

Preparing for Upgrade

The upgrade process of the Moab HPC Suite includes upgrading the database and different components in the suite. This guide contains detailed instructions for upgrading each component.

 It is highly recommended that you *first* perform upgrades in a *test environment*. Installation and upgrade procedures are tested prior to release; however, due to customizable variations that may be utilized by your configuration, it is not recommended to drop new versions of software directly into production environments. This is especially true when the workload has vital bearing. Contact Adaptive Computing Professional Services for more information.

 Because many system-level files and directories are accessed during the upgrade, the upgrade instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

Upgrade the Moab HPC Suite in the following order:

1. Mongo database. See [Upgrading the MongoDB Database](#).
2. Torque. See [Upgrading Torque](#).
3. Moab Workload Manager. See [Upgrading Moab Workload Manager](#).

4. Moab Accounting Manager. See [Upgrading Moab Accounting Manager](#).
5. Moab Web Services. See [Upgrading Moab Web Services](#).

Related Topics

[Chapter 1 Planning your Installation on page 3](#)

Upgrading Torque

Torque 6.0 binaries are backward compatible with Torque 5.0 or later. However they are not backward compatible with Torque versions prior to 5.0. When you upgrade to Torque 6.0.1 from versions prior to 5.0, all MOM and server daemons must be upgraded at the same time.

The job format is compatible between 6.0 and previous versions of Torque and any queued jobs will upgrade to the new version. It is not recommended to upgrade Torque while jobs are in a running state.

This topic contains instructions on how to upgrade and start Torque Resource Manager (Torque).

i If you need to upgrade a Torque version prior to 4.0, contact Adaptive Computing.

i See [Considerations Before Upgrading](#) in the *Torque Resource Manager Administrator Guide* for additional important information including about how to handle running jobs during an upgrade, mixed server/MOM versions, and the possibility of upgrading the MOMs without having to take compute nodes offline.

In this topic:

- [Before You Upgrade on page 99](#)
- [Stop Torque Services on page 100](#)
- [Upgrade the Torque Server on page 101](#)
- [Update the Torque MOMs on page 102](#)
- [Update the Torque Clients on page 103](#)
- [Start Torque Services on page 104](#)
- [Perform Status and Error Checks on page 105](#)

Before You Upgrade

This section contains information of which you should be aware before upgrading.

In this section:

- [serverdb on page 100](#)
- [Running Jobs on page 100](#)
- [Cray Systems on page 100](#)

serverdb

The `pbs_server` configuration is saved in the file `TORQUE_HOME/server_priv/serverdb`. When running Torque 4.1 or later for the first time, this file converts from a binary file to an XML-like format.

Running Jobs

Before upgrading the system, all running jobs must complete. To prevent queued jobs from starting, nodes can be set to offline or all queues can be disabled (using the "started" queue attribute). See [pbsnodes](#) or [Queue Attributes](#) in the *Torque Resource Manager Administrator Guide* for more information.

Cray Systems

For upgrading Torque to 6.0.1 on a Cray system, refer to the [Installation Notes for Moab and Torque for Cray](#) in Appendix G of the *Moab Workload Manager Administrator Guide*.

Stop Torque Services

Do the following:

1. On the Torque Server Host, shut down the Torque server.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# service pbs_server stop
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl stop pbs_server.service
```

2. On each Torque MOM Host, shut down the Torque MOM service.



Confirm all jobs have completed before stopping `pbs_mom`. You can do this by typing "`momctl -d3`". If there are no jobs running, you will see the message "NOTE: no local jobs detected" towards the bottom of the output. If jobs are still running and the MOM is shutdown, you will only be able to track when the job completes and you will not be able to get completion codes or statistics.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# service pbs_mom stop
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl stop pbs_mom.service
```

3. On each Torque Client Host (including the Moab Server Host, the Torque Server Host, and the Torque MOM Hosts, if applicable), shut down the `trqauthd` service.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# service trqauthd stop
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl stop trqauthd.service
```

Upgrade the Torque Server

i You *must* complete all the previous upgrade steps in this topic before upgrading Torque server. See the list of steps at the beginning of this topic.

On the Torque Server Host, do the following:

1. Back up your `server_priv` directory.

```
[root]# tar -cvf backup.tar.gz $TORQUE_HOME/server_priv
```

2. If not already installed, install the Boost C++ headers.

- Red Hat-based systems

```
[root]# yum install boost-devel
```

- SUSE-based systems

```
[root]# zypper install boost-devel
```

3. Download the latest 6.0.1 build from the [Adaptive Computing](#) website.
4. Install the latest Torque tarball.

```
[root]# cd /tmp
[root]# tar xzvf torque-<version>-<build number>.tar.gz
[root]# cd torque-<version>-<build number>
[root]# ./configure
[root]# make
[root]# make install
```

5. Update the `pbs_server` service startup script.
 - a. Make a backup of your current service startup script.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# cp /etc/init.d/pbs_server pbs_server.bak
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# cp /usr/lib/systemd/system/pbs_server.service pbs_server.service.bak
```

- b. Copy in the new stock service startup script.

- Red Hat 6-based systems

```
[root]# cp contrib/init.d/pbs_server /etc/init.d
```

- SUSE 11-based systems

```
[root]# cp contrib/init.d/suse.pbs_server /etc/init.d/pbs_server
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# cp contrib/systemd/pbs_server.service /usr/lib/systemd/system/
```

- c. Merge in any customizations.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# vi /etc/init.d/pbs_server
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# vi /usr/lib/systemd/system/pbs_server.service
```

Update the Torque MOMs

Do the following:

1. On the Torque Server Host, do the following:
 - a. Create the self-extracting packages that are copied and executed on your nodes.

```
[root]# make packages
Building ./torque-package-clients-linux-x86_64.sh ...
Building ./torque-package-mom-linux-x86_64.sh ...
Building ./torque-package-server-linux-x86_64.sh ...
Building ./torque-package-gui-linux-x86_64.sh ...
Building ./torque-package-devel-linux-x86_64.sh ...
Done.
```

The package files are self-extracting packages that can be copied and executed on your production machines. Use `--help` for options.

- b. Copy the self-extracting packages to each Torque MOM Host.

Adaptive Computing recommends that you use a remote shell, such as SSH, to install packages on remote systems. Set up shared SSH keys if you do not want to supply a password for each Torque MOM Host.

```
[root]# scp torque-package-mom-linux-x86_64.sh <torque-mom-host>:
```

- c. Copy the pbs_mom startup script to each Torque MOM Host.

- Red Hat 6-based systems

```
[root]# scp contrib/init.d/pbs_mom <torque-mom-host>:/etc/init.d
```

- SUSE 11-based systems

```
[root]# scp contrib/init.d/suse.pbs_mom <torque-mom-host>:/etc/init.d/pbs_mom
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# scp contrib/systemd/pbs_mom.service <torque-mom-host>:/usr/lib/systemd/system/
```

2. On each Torque MOM Host, do the following:

i Many of these steps can be done from the Torque server from a remote shell, such as SSH. Set up shared SSH keys if you do not want to supply a password for each Torque MOM Host.

```
[root]# ./torque-package-mom-linux-x86_64.sh --install
[root]# echo /usr/local/lib > /etc/ld.so.conf.d/torque.conf
[root]# ldconfig
```

Update the Torque Clients

This section contains instructions on updating the Torque clients on the Torque Client Hosts (including the Moab Server Host and Torque MOM Hosts, if applicable).

1. On the Torque Server Host, do the following:

- a. Copy the self-extracting packages to each Torque Client Host.

Adaptive Computing recommends that you use a remote shell, such as SSH, to install packages on remote systems. Set up shared SSH keys if you do not want to supply a password for each Torque MOM Host.

```
[root]# scp torque-package-clients-linux-x86_64.sh <torque-client-host>:
```

b. Copy the trqauthd startup script to each Torque Client Host.

- Red Hat 6-based systems

```
[root]# scp contrib/init.d/trqauthd <torque-client-host>:/etc/init.d
```

- SUSE 11-based systems

```
[root]# scp contrib/init.d/suse.trqauthd <torque-client-host>:/etc/init.d/trqauthd
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# scp contrib/systemd/trqauthd.service <torque-client-host>:/usr/lib/systemd/system/
```

2. On each Torque Client Host, do the following:

i Many of these steps can be done from the Torque server from a remote shell, such as SSH. Set up shared SSH keys if you do not want to supply a password for each Torque Client Host.

```
[root]# ./torque-package-clients-linux-x86_64.sh --install
[root]# echo /usr/local/lib > /etc/ld.so.conf.d/torque.conf
[root]# ldconfig
```

Start Torque Services

Do the following:

1. On each Torque Client Host (including the Moab Server Host, Torque Server Host and Torque MOM Hosts, if applicable), start up the trqauthd service.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# service trqauthd start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl daemon-reload
[root]# systemctl start trqauthd.service
```

2. On each Torque MOM Host, start up the Torque MOM service.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# service pbs_mom start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl daemon-reload
[root]# systemctl start pbs_mom.service
```

3. On the Torque Server Host, start up the Torque server.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# service pbs_server start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl daemon-reload
[root]# systemctl start pbs_server.service
```

Perform Status and Error Checks

On the Torque Server Host, do the following:

1. Check the status of the jobs in the queue.

```
[root]# qstat
```

2. Check for errors.

```
[root]# grep -i error /var/spool/torque/server_logs/*
[root]# grep -i error /var/spool/torque/mom_logs/*
```

Upgrading Moab Workload Manager

This topic provides instructions to upgrade Moab Workload Manager to the latest release version. Depending on which version of Moab you are presently running, upgrade instructions may vary.

Moab Workload Manager uses the standard `configure`, `make`, and `make install` steps for upgrades. This topic provides a number of sample steps referenced to a particular installation on a Linux platform using the bash shell. These steps indicate the user ID in brackets performing the step. The exact commands to be performed and the user that issues them will vary based on the platform, shell, installation preferences, and other factors.

It is highly recommended that you *first* perform upgrades in a *test environment*. See the warning in [Preparing for Upgrade on page 98](#). It is also recommend that you verify the policies, scripts, and queues work the way you want them to in this test environment. See [Testing New Releases and Policies](#) in the *Moab Workload Manager Administrator Guide* for more information.

If you are also upgrading Torque from an older version (pre-4.0), contact Adaptive Computing.

i Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

Upgrade Moab Workload Manager

On the Moab Server Host, do the following:

1. If you have not already done so, install extra packages from the add-on repositories. See [Enable Extra Packages for the Repository on page 23](#)
2. Download the latest Moab build (`moab-<version>-<OS>.tar.gz`) from the [Adaptive Computing](#) website.

i The variable marked `<version>` indicates the build's version, revision, and changeset information. The variable marked `<OS>` indicates the OS for which the build was designed.

3. Untar the distribution file. For example:

```
[root]# tar -xzvf moab-<version>-<OS>.tar.gz
```

4. Change directory into the extracted directory.
5. Configure the installation package.

Use the same configure options as when Moab was installed previously. If you cannot remember which options were used previously, check the `config.log` file in the directory where the previous version of Moab was installed from.

For a complete list of configure options, use `./configure --help`.

6. Stop Moab.

```
[root]# mschedctl -k
moab will be shutdown immediately
```

i While Moab is down, all currently running jobs continue to run on the nodes, the job queue remains intact, and new jobs cannot be submitted to Moab.

7. Back up your Moab Workload Manager home directory (`/opt/moab/` by default) before continuing.

8. If you are using green computing, or if you are using a resource manager other than Torque, run the `make perldeps` command to install the necessary perl modules using CPAN.

i CPAN is installed on SUSE-based systems by default. If upgrading on a Red Hat-based system, you will need to install CPAN `[root]# yum install perl-CPAN` if you have not already done so. When first running CPAN, you will be asked for configuration information. It is recommended that you choose an automatic configuration.

```
[root]# make perldeps
```

9. Install Moab.

```
[root]# make install
```

i Default configuration files are installed during `make install`. Existing configuration files are not overwritten and the new files are given a `.dist` extension.

10. If you use ODBC, you must confirm the database schema compatibility. For example, if you are upgrading Moab 8.1 to 9.0 no schema changes were made; however if you upgrade from Moab 8.0 and prior, you will need to upgrade your database. See [Migrating Your Database to Newer Versions of Moab](#) in the *Moab Workload Manager Administrator Guide* for more information.
11. Verify the version number is correct before starting the new server version.

```
[root]# moab --about

Defaults:  server=:42559  cfgdir=/opt/moab (env)  vardir=/opt/moab
Build dir:  /tmp/jenkins/workspace/MWM-9.0.0/label/build-sles12
Build host: us-devops-build10
Build date: Fri Oct 09 13:00:00 MST 2015
Build args: NA
Compiler Flags:  -D__M64 -D_BUILDDATETIME="2015100913" -DMUSEZEROMQ -
DMUSEWEBSERVICES -DMUSEMONGODB -DMMAX_GRES=512 -DMMAX_RANGE=2048 -DMMAX_TASK=32768
-fPIC -gdwarf-3 -Wall -Wextra -DVALGRIND -Og -x c++ -std=c++11 -DDMAX_PJOB=512 -D_
GNU_SOURCE
Compiled as little endian.
Version: moab server 9.0.0 (revision 2015100913, changeset
14dee972ebcee919207e48054e9f285db9f6a555)
```

12. If you are using Moab Accounting Manager with the native interface (**TYPE=native**), remove all entries in `moab.cfg` with the form `(AMCFG[*]*URL=exec://*)`, except for those that you have customized. See [AMCFG Parameters and Flags](#) in the *Moab Workload Manager Administrator Guide* for more information.

i In Moab Workload Manager 8.1 and after, Moab defaults to using a set of stock scripts that no longer need to be explicitly configured in the server configuration file.

13. Start Moab.

- Red Hat 6-based or SUSE 11-based systems

```
[root]# service moab start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start moab.service
```

Upgrading Moab Accounting Manager

This topic provides instructions to upgrade MAM to the latest release version. It includes instructions for migrating your database schema to a new version if necessary.

Moab Accounting Manager uses the standard `configure`, `make`, and `make install` steps for upgrades. This document provides a number of sample steps referenced to a particular installation on a Linux platform using the bash shell. These steps indicate the user ID in brackets performing the step. The exact commands to be performed and the user that issues them will vary based on the platform, shell, installation preferences, and other factors.

Upgrade Moab Accounting Manager

On the MAM Server Host, do the following:

1. Determine the MAM Accounting admin user and change to that user.
 - If you are upgrading MAM from a version *prior* to 9.0, use `glsuser`.

```
[root]# glsuser | grep 'Accounting Admin'
mam      True
Accounting Admin
[root]# su - mam
```

- If you are upgrading MAM at or after 9.0, use `mam-list-users`.

```
[root]# mam-list-users | grep 'Accounting Admin'
mam      True
Accounting Admin
[root]# su - mam
```

2. Determine whether you need to migrate your database.

a. Determine your database version.

- If you are upgrading MAM from a version *prior* to 9.0, run `goldsh System Query`.

```
[mam]$ goldsh System Query
```

- If you are upgrading MAM at or after 9.0, run `mam-shell System Query`.

```
[mam]$ mam-shell System Query
```

- ### b. If the current version is lower than 9.0, you must migrate your database. The steps required to do so are incorporated in the remaining steps for this topic.

3. Stop the server daemon.

- If you are upgrading MAM from a version *prior* to 9.0, run `goldd -k`.

```
[mam]$ goldd -k
```

- If you are upgrading MAM at or after 9.0, run `mam-server -k`.

```
[mam]$ mam-server -k
```

4. If you determined that you must migrate your database, create a database backup.

- PostgreSQL database.

```
[mam]$ pg_dump -U <mam_database_user> -W <old_database_name> > /tmp/<old_database_name>.sql
```

- MySQL database.

```
[mam]$ mysqldump -u <mam_database_user> -p <old_database_name> > /tmp/<old_database_name>.sql
```

5. Verify that each of the prerequisites listed in [Installing Moab Accounting Manager on page 39](#) have been satisfied.

6. Download the latest MAM build (`mam-<version>.tar.gz`) from the [Adaptive Computing](#) website.

 The variable marked `<version>` indicates the build's version.

7. Unpack the tar archive and change directory into the top directory of the distribution.

```
[mam]$ tar -zxvf mam-9.0.1.tar.gz
[mam]$ cd mam-9.0.1
```

8. Configure Moab Accounting Manager by running the `configure` script with the desired options.

It is recommended that you use the same configure options that were used in the previous installation. You can examine the `config.log` file where you unpacked your previous distribution to help determine the configuration options that were used to install the prior version of MAM.

 Client and server command names changed beginning with 9.0. If you want to create symbolic links to enable you to continue to use the old client and server command names, use the `--with-legacy-links` option with `configure`. When running a command under its old name, the command will issue a deprecation warning. This warning can be disabled by setting `client.deprecationwarning = false` in the `mam-client.conf` file.

```
[mam]$ ./configure
```

9. To compile the program, type `make`.

```
[mam]$ make
```

 If you only need to install the clients on a particular system, replace `make` with `make clients-only`. If you only need to install the web GUI on a particular system, replace `make` with `make gui-only`.

10. Run `make install` as root to install Moab Accounting Manager.

```
[mam]$ su -c "make install"
```

 If you only need to install the clients on a particular system, replace `"make install"` with `"make install-clients-only"`. If you only need to install the web GUI on a particular system, replace `"make install"` with `"make install-gui-only"`.

11. Edit the configuration files as necessary. You may want to compare your existing configuration files with those distributed with the new release to determine if you want to merge and change any of the new options within your configuration files.
- If you are upgrading MAM from a version *prior* to 9.0, the install process will have saved your prior configuration files to `{goldd,gold,goldg}.conf.pre-9.0` and written new default server configuration file as `mam-{server,client,gui}.conf`. You will need to merge any non-default parameters from your prior config files to the new default config files.

```
[mam]$ diff /opt/mam/etc/goldd.conf.pre-9.0 /opt/mam/etc/mam-server.conf
[mam]$ vi /opt/mam/etc/mam-server.conf
[mam]$ diff /opt/mam/etc/gold.conf.pre-9.0 /opt/mam/etc/mam-client.conf
[mam]$ vi /opt/mam/etc/mam-client.conf
[mam]$ diff /opt/mam/etc/goldg.conf.pre-9.0 /opt/mam/etc/mam-gui.conf
[mam]$ vi /opt/mam/etc/mam-gui.conf
```

- If you are upgrading MAM at or after 9.0, merge and change any of the new options supplied in the new default configuration files (saved in mam-`{server,client,gui}.conf.dist`) into your existing configuration files (mam-`{server,client,gui}.conf`).

```
[mam]$ diff /opt/mam/etc/mam-server.conf /opt/mam/etc/mam-server.conf.dist
[mam]$ vi /opt/mam/etc/mam-server.conf
[mam]$ diff /opt/mam/etc/mam-client.conf /opt/mam/etc/mam-client.conf.dist
[mam]$ vi /opt/mam/etc/mam-client.conf
[mam]$ diff /opt/mam/etc/mam-gui.conf /opt/mam/etc/mam-gui.conf.dist
[mam]$ vi /opt/mam/etc/mam-gui.conf
```

Verify that your current path points to your newly installed clients and server.

```
[mam]$ which mam-server
/opt/mam/sbin/mam-server
```

12. Start the server daemon back up.

```
[mam]$ mam-server
```

13. If you are migrating your database to 9.0, you will do so by running one or more migration scripts. You must run every incremental migration script between the version you are currently using and the new version (9.0). These scripts are designed to be rerunnable, so if you encounter a failure, resolve the failure and rerun the migration script. If you are unable to resolve the failure and complete the migration, contact Support.

For example, if you are migrating from Moab Accounting Manager version 7.2, you must run five migration scripts: the first to migrate the database schema from 7.2 to 7.3, the second to migrate from 7.3 to 7.5, the third to migrate the database schema from 7.5 to 8.0, the fourth to migrate the database schema from 8.0 to 8.1, and the fifth to migrate the database schema from 8.1 to 9.0.

```
[mam]$ sbin/migrate_7.2-7.3.pl
[mam]$ sbin/migrate_7.3-7.5.pl
[mam]$ sbin/migrate_7.5-8.0.pl
[mam]$ sbin/migrate_8.0-8.1.pl
[mam]$ sbin/migrate_8.1-9.0.pl
```

14. Verify that the resulting database schema version is 9.0.

```
[mam]$ mam-shell System Query
```

| Name | Version | Description |
|-------------------------|---------|--------------------|
| Moab Accounting Manager | 9.0 | Commercial Release |

15. Verify that the executables have been upgraded to 9.0.1.

```
[mam]$ mam-server -v
Moab Accounting Manager version 9.0.1
```

Upgrading Moab Web Services

This topic provides instructions to upgrade Moab Web Services to the latest release version. Depending on which version of MWS you are presently running, upgrade instructions may vary.

i You must deploy Moab Web Services on the *same* host as Moab Server (Moab Server Host). For documentation clarity, these instructions refer to the host for Moab Server and MWS Server as the MWS Server Host.

Before You Upgrade

MWS requires Tomcat 7. It is also recommended that you upgrade to Java 8 and MongoDB 2.4.x.

Upgrade to Tomcat 7

Tomcat 7 is required to run MWS 9.0 and after. On the MWS Server Host, do the following:

1. Check your Tomcat version.

```
[root]# rpm -qa tomcat
tomcat-7.0.33-4.el6.noarch
```

2. If your Tomcat version is prior to 7, upgrade Tomcat.

- Red Hat 6-based systems

```
[root]# service tomcat6 stop
[root]# chkconfig tomcat6 off
[root]# yum install tomcat
```

- Red Hat 7-based or SUSE 12-based systems

Support for Red Hat 7-based or SUSE 12-based systems began with 8.1.1.2

If upgrading from this version or after, Tomcat 7 is installed by default.

Upgrade to Java 8

i Oracle Java 8 Runtime Environment is the recommended Java environment, but Oracle Java 7 is also supported. All other versions of Java, including OpenJDK/IcedTea, GNU Compiler for Java, and so on cannot run MWS.

If you wish to upgrade to Java 8, refer to the [Install Java on page 57](#) instructions.

Upgrade the MongoDB

⚠ It is highly recommended that you perform a full database backup before updating your database. This can be done using the `mongodump` utility documented in the [MongoDB documentation](#) (<http://www.mongodb.org/display/DOCS/Backups>).

On the host where the MWS MongoDB database resides, do the following:

1. Check your MongoDB version.

```
[root]# mongo --version
```

2. If your MongoDB version is prior to 2.4, upgrade the database. When upgrading, you must add 'exclude=mongodb-org mongodb-org-server' to the `/etc/yum.repos.d/mongodb.repo` file to maintain 2.4.x. Depending on your MongoDB version, the file path may be `/etc/yum.repos.d/10gen.repo`.

```
[mongodb]
name=MongoDB Repository
baseurl=http://downloads-distro.mongodb.org/repo/redhat/os/x86_64/
gpgcheck=0
enabled=1
exclude=mongodb-org mongodb-org-server
```

Upgrade Moab Web Services

i You *must* complete the tasks in [Before You Upgrade on page 112](#) before upgrading MWS.

On the MWS Server Host, do the following:

1. Create a directory for which you will extract the contents of the MWS download tarball. For example:

```
[root]# mkdir /tmp/mws-install
[root]# cd /tmp/mws-install
```

2. Download the latest MWS build (`mws-<version>.tar.gz`) from the [Adaptive Computing](#) website.

i The variable marked `<version>` is the desired version of the suite; for example, 9.0.1.

3. In the directory you created earlier, extract the contents of the MWS download tarball and then change directory into the extracted directory. For example:

```
[root]# tar xvzf mws-9.0.1.tar.gz
[root]# cd mws-9.0.1
```

4. Deploy the updated `mws.war` to Tomcat.

- Red Hat 6-based systems

i If your prior MWS version had tomcat 6, you should have stopped the tomcat6 service when you upgraded to Tomcat 7 (required). See [Upgrade to Tomcat 7 on page 112](#) for more information.

```
[root]# service tomcat stop
[root]# rm -rf /usr/share/tomcat/webapps/mws /usr/share/tomcat/webapps/mws.war
[root]# cp mws.war /usr/share/tomcat/webapps/
[root]# chown -R tomcat:tomcat /usr/share/tomcat/webapps/mws.war
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl stop tomcat.service
[root]# rm -rf /usr/share/tomcat/webapps/mws /usr/share/tomcat/webapps/mws.war
[root]# cp mws.war /usr/share/tomcat/webapps/
[root]# chown -R tomcat:tomcat /usr/share/tomcat/webapps/mws.war
```

5. Back up the MWS home directory and create the required destination directory structure.

```
[root]# cp -r /opt/mws /opt/mws-9.0-backup
[root]# mkdir -p \
/opt/mws/etc/mws.d \
/opt/mws/hooks \
/opt/mws/log \
/opt/mws/plugins \
/opt/mws/spool/hooks \
/opt/mws/utils
[root]# chown -R tomcat:tomcat /opt/mws
[root]# chmod -R 555 /opt/mws
[root]# chmod u+w \
/opt/mws/log \
/opt/mws/plugins \
/opt/mws/spool \
/opt/mws/spool/hooks \
/opt/mws/utils
```

6. Copy the extracted utility files to the utility directory created above and give the tomcat user ownership of the directory.

```
[root]# cd utils
[root]# cp * /opt/mws/utils
[root]# chown tomcat:tomcat /opt/mws/utils/*
```

7. Merge the changes in the `/tmp/mws-install/mws-9.0.1/mws-config.groovy` file into your existing `/opt/mws/etc/mws-config.groovy`. Depending on your current MWS version, do the following as needed:
 - If Insight is part of your configuration, add the Insight configuration information (`insight.username`, `insight.password`, `insight.url`); prior version to 9.0
 - If Viewpoint is part of your configuration, register Viewpoint as client; prior to version 9.0
 - Change the `moab.messageQueue.port` to 5570; prior to version 8.0
 - Configure and appender for the audit log; prior to version 8.0
 - Change the layout to `"new com.ace.mws.logging.ACPatternLayout()"` for the output format of each log entry; prior to version 8.0
 - Remove the `mws.suite` parameter and the `mam.*` parameters (they have been moved to `/opt/mws/etc/mws.d/`); prior to version 8.0
 - Confirm the value for `moab.messageQueue.secretKey` matches the value located in `/opt/moab/etc/moab-private.cfg`; if you have not yet configured a secret key, see [Secure communication using secret keys](#).

The following is an example of the merged `mws-config.groovy` file for MWS 9.0.1:

```

// Any settings in this file may be overridden by any
// file in the mws.d directory.

// Change these to be whatever you like.
auth.defaultUser.username = "moab-admin"
auth.defaultUser.password = "changeme!"

// Moab Workload Manager configuration.
moab.secretKey = "<ENTER-KEY-HERE>"
moab.server = "localhost"
moab.port = 42559
moab.messageDigestAlgorithm = "SHA-1"

// MongoDB configuration.
// grails.mongo.username = "mws_user"
// grails.mongo.password = "<ENTER-KEY-HERE>"

// Insight configuration.
// dataSource_insight.username = "mws"
// dataSource_insight.password = "changeme!"
// dataSource_insight.url = "jdbc:postgresql://127.0.0.1:5432/moab_insight"

// Message bus configuration.
moab.messageQueue.port = 5570
// moab.messageQueue.secretKey = "<ENTER-KEY-HERE>"
mws.messageQueue.address = "*"
mws.messageQueue.port = 5564

// Sample OAuth Configuration
grails.plugin.springsecurity.oauthProvider.clients = [
    [
        clientId          : "viewpoint",
        clientSecret      : "<ENTER-CLIENTSECRET-HERE>",
        authorizedGrantTypes: ["password"]
    ]
]

// Sample LDAP Configurations

// Sample OpenLDAP Configuration
//ldap.server = "192.168.0.5"
//ldap.port = 389
//ldap.baseDNs = ["dc=acme,dc=com"]
//ldap.bindUser = "cn=Manager,dc=acme,dc=com"
//ldap.password = "*****"
//ldap.directory.type = "OpenLDAP Using InetOrgPerson Schema"

// Sample Active Directory Configuration
//ldap.server = "192.168.0.5"
//ldap.port = 389
//ldap.baseDNs = ["CN=Users,DC=acme,DC=com", "OU=Europe,DC=acme,DC=com"]
//ldap.bindUser = "cn=Administrator,cn=Users,DC=acme,DC=com"
//ldap.password = "*****"
//ldap.directory.type = "Microsoft Active Directory"

```



```

// Individual logger configurations
debug 'com.ace.mws',
    'grails.app.conf.BootStrap',
    'grails.app.controllers.com.ace.mws',
    'grails.app.domain.com.ace.mws',
    'grails.app.filters.com.ace.mws',
    'grails.app.services.com.ace.mws',
    'grails.app.tagLib.com.ace.mws',
    'grails.app.jobs.com.ace.mws',
    'grails.app.gapiParser',
    'grails.app.gapiRequest',
    'grails.app.gapiSerializer',
    'grails.app.translator',
    'plugins' // MWS plugins

info 'com.ace.mws.gapi.Connection',
    'com.ace.mws.gapi.parsers',
    'grails.app.service.grails.plugins.reloadconfig',
    'com.ace.mws.gapi.serializers'

off 'org.codehaus.groovy.grails.web.errors'

warn additivity: false, jdbc: 'org.apache.tomcat.jdbc'

// Logs event information to the events log, not the rootLog
trace additivity: false, events: 'com.ace.mws.events.EventFlatFileWriter'

// Logs audit information to the audit log, not the rootLog
trace additivity: false, audit: 'mws.audit'
}

```

8. Merge any changes supplied in the new `mws-config-hpc.groovy` file in to your installed `/opt/mws/etc/mws.d/mws-config-hpc.groovy`.
9. Remove unused MWS plugins. Unused plugins must be removed as their presence will prevent MWS from starting up.
 - Remove all plugins from `/opt/mws/plugins` except for the diagnostics, native, and power-management plugins.

```

[root]# cd /opt/mws/plugins
[root]# rm plugins-reports.jar plugins-storage.jar plugins-vcenter.jar

```

10. Verify the Tomcat user has read access to the `/opt/mws/etc/mws-config.groovy` and `/opt/mws/etc/mws.d/mws-config-hpc.groovy` file.
11. Verify the following lines are added to the end of `/etc/tomcat/tomcat.conf`.

```

CATALINA_OPTS="-DMWS_HOME=/opt/mws -Xms256m -Xmx3g -XX:MaxPermSize=384m -
Dfile.encoding=UTF8"
JAVA_HOME="/usr/java/latest"

```

12. Upgrade the schema of the `mws` database in MongoDB.

 You *must* perform this step, regardless of whether you upgraded MongoDB to version 2.4.x.

Run the database migration script provided with MWS. (It is safe to run this script more than once. If for any reason, errors occur during the execution of the script, run it again.)

```
[root]# mongo -u mws_user mws /opt/mws/utils/db-migrate.js -p
```

i The script might take several minutes to execute.

13. Start Tomcat.

⚠ You will need to start the "tomcat" service. Starting the "tomcat6" service will install the wrong version of Tomcat.

- Red Hat 6-based systems

```
[root]# service tomcat start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start tomcat.service
```

14. Visit <http://localhost:8080/mws/> in a web browser to verify that MWS is running again.

You will see some sample queries and a few other actions.

15. Log into MWS to verify configuration. (The credentials are the values of `auth.defaultUser.username` and `auth.defaultUser.password` set in `/opt/mws/etc/mws-config.groovy`.)

i If you encounter problems, or if MWS does not seem to be running, see the steps in [Moab Web Services Issues on page 269](#).

Migrating the MAM Database from MySQL to PostgreSQL

PostgreSQL is the preferred DBMS for MAM. Customers who have already installed MySQL as the DBMS for MAM are not required to migrate their database to use PostgreSQL at this time. However, MySQL is considered deprecated and new installations will only use PostgreSQL.

i PostgreSQL does not provide a standard procedure for migrating an existing database from MySQL to PostgreSQL. Adaptive Computing has had success using the `py-mysql2pgsql` tools for migrating/converting/exporting data from MySQL to PostgreSQL. See <https://github.com/philipsoutham/py-mysql2pgsql> for additional details.

To Migrate the MAM Database

This procedure was successfully tested on an actual customer MySQL database with millions of transactions on CentOS 6.4. It completed in less than an hour.

1. Make a backup copy of your MySQL mam database.

```
[root]# mysqldump mam > /archive/mam.mysql
```

2. Follow the instructions to Install PostgreSQL.

- **Manual Install** - [Install and Initialize the MAM PostgreSQL Server on page 43](#)
- **RPM Install** - [Install and Initialize PostgreSQL Server on page 146](#)

3. Install the prerequisite packages.

```
[root]# yum install git postgresql-devel gcc MySQL-python python-psycopg2 PyYAML
termcolor python-devel
```

4. Install pg-mysql2pgsql (from source).

```
[root]# cd /software
[root]# git clone git://github.com/philipsoutham/py-mysql2pgsql.git
[root]# cd py-mysql2pgsql
[root]# python setup.py install
```

5. Run pg-mysql2pgsql once to create a template yaml config file.

```
[root]# py-mysql2pgsql -v
```

6. Edit the config file to specify the MySQL database connection information and a file to output the result.

```
[root]# vi mysql2pgsql.yml
```

```
mysql:
hostname: localhost
port: 3306
socket:
username: mam
password: changeme
database: mam
compress: false
destination:
# if file is given, output goes to file, else postgres
file: /archive/mam.pgsql
postgres:
hostname: localhost
port: 5432
username:
password:
database:
```

7. Run the pg-mysql2pgsql program again to convert the database.

```
[root]# py-mysql2pgsql -v
```

8. Create the mam database in PostgreSQL.

```
[root]# su - postgres
[postgres]$ psql
postgres=# create database "mam";
postgres=# create user mam with password 'changeme!';
postgres=# \q
[postgres]$ exit
```

9. Import the converted data into the PostgreSQL database.

```
[root]# su - mam
[mam]$ psql mam < /archive/mam.pgsql
```

10. Point MAM to use the new postgresql database.

```
[mam]$ cd /software/mam-latest
[mam]$ ./configure # This will generate an etc/mam-
server.conf.dist file
[mam]$ vi /opt/mam/etc/mam-server.conf # Merge in the database.datasources from
etc/mam-server.conf.dist
```

11. Restart Moab Accounting Manager.

```
[mam]$ mam-server -r
```

Chapter 3 RPM installation Method

This chapter contains an introduction to the RPM Installation method and explains how to prepare your component hosts (physical machines in your cluster) for the RPM installations and upgrades. Information and configuration information for each Moab HPC Suite product or module using the RPM Installation method, is also provided.

In this chapter:

- [About RPM Installations and Upgrades on page 123](#)
- [Preparing the Host – Typical Method on page 125](#)
- [Creating the moab-offline Tarball on page 127](#)
- [Preparing the Host – Offline Method on page 130](#)
- [RPM Installations on page 132](#)
- [RPM Upgrades on page 240](#)

About RPM Installations and Upgrades

This topic contains information useful to know and understand when using RPMs for installation and upgrading.

Adaptive Computing provides RPMs to install or upgrade the various component servers (such as Moab Server, MWS Server, Torque Server). The Moab HPC Suite RPM bundle contains all the RPMs for the Moab HPC Suite components and modules. However, not every component may be installed or upgraded on the same host (for example, it is recommended that you install the Torque Server on a different host from the Moab Server).

In this topic:

- [RPM Installation and Upgrade Methods on page 123](#)
- [Special Considerations on page 124](#)
- [Installation and Upgrade Process on page 124](#)

RPM Installation and Upgrade Methods

Depending on your configuration, you may install many servers on a single host, or a single server on its own host. In addition, you can install various clients and GUIs on the same host you installed the server or on another host. For example, you have the Moab Server and the MWS Server on the same host (required) and you install the Torque Server on a different host (recommended).

i Be aware that the same host may be called by different names. For example, even though the Moab Server and the MWS Server are installed on the same host, the MWS instructions will call it the MWS Server Host, not the Moab Server Host.

Adaptive Computing provides two different types of RPM installation or upgrade methods.

- The typical method is the original RPM method in which you download the Moab HPC Suite RPM bundle to each host in your Moab HPC Suite environment.
- The offline method is available for configurations where the hosts in your Moab HPC Suite environment do *not* have internet access in order to download the Moab HPC Suite RPM dependencies. This method requires an authorized user to download the Moab HPC Suite RPM bundle and other related dependencies and create a moab-offline tarball. That tarball is then copied (using scp, DVD, USB drive, or similar) to each host in your

Moab HPC Suite environment. See [Creating the moab-offline Tarball on page 127](#) for instructions on how to create the tarball.

Special Considerations

Be aware of the following:

- On RHEL systems, you must be registered for a Red Hat subscription in order to have access to required rpm package dependencies.
- On SLES systems, you must be registered for a SUSE Linux Enterprise subscription in order to have access to required rpm package dependencies.
- Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges. You will see that the instructions execute commands as the root user. Also be aware that the same commands will work for a non-root user with the `sudo` command.
- If using the offline method, the internet-enabled host *must* have the *exact* same OS as the hosts within your Moab HPC Suite environment. As the Moab HPC Suite can have several hosts, and each host may not use the same OS, you may need to repeat this procedure for each OS used.

Installation and Upgrade Process

Each host (physical machine) will need to have the Moab HPC Suite RPM bundle and the Adaptive Computing repository enabled. This is referred to as preparing the host. Again this can be done using the typical or the offline method.

Once each host has been prepared, you can install or upgrade the individual components on the designated hosts. It is recommended that you or upgrade the software components in the following order.

1. Torque Resource Manager. See [Installing Torque Resource Manager on page 132](#) or [Upgrading Torque Resource Manager \(RPM\) on page 240](#).
2. Moab Workload Manager. See [Installing Moab Workload Manager on page 137](#) or [Upgrading Moab Workload Manager \(RPM\) on page 243](#).
3. Moab Accounting Manager. See [Installing Moab Accounting Manager on page 142](#) or [Upgrading Moab Accounting Manager \(RPM\) on page 247](#).
4. Moab Web Services. See [Installing Moab Web Services on page 154](#) or [Upgrading Moab Web Services \(RPM\) on page 250](#).
5. Moab Insight. See [Installing Moab Insight on page 165](#) or [Upgrading Moab Insight \(RPM\) on page 253](#).

6. Moab Viewpoint. See [Installing Moab Viewpoint on page 178](#) or [Upgrading Moab Viewpoint \(RPM\) on page 256](#).
7. Install the RLM Server (if using Viewpoint's Remote Visualization or Nitro and you do not already have an RLM Server). See [Installing RLM Server on page 195](#)
8. Remote Visualization. See [Installing Remote Visualization on page 198](#).
9. Integrate Nitro with your Moab HPC Suite. See [Nitro Integration on page 217](#).

i RLM Server, Remote Visualization, and Nitro integration are new in the 9.0.1 release. Upgrade instructions are not applicable at this time.

Related Topics

- [Chapter 3 RPM installation Method on page 122](#)

Preparing the Host – Typical Method

This topic contains instructions on how to download the Moab HPC Suite RPM bundle and enable the Adaptive Computing repository for all the hosts in your configuration.

The Moab HPC Suite RPM bundle contains all the RPMs for the Moab HPC Suite components and modules. However, not every component may be installed on the same host (for example, it is recommended that you install the Torque Server on a different host from the Moab Server).

i Whether you are installing RPMs on one host or on several hosts, each host (physical machine) on which a server is installed (Torque Server Host, Moab Server Host, etc) *must* have the Adaptive Computing Package Repository enabled. If Remote Visualization is part of your configuration, the Adaptive Computing Package Repository must also be enabled on the Torque MOM Hosts (compute nodes); otherwise is not necessary to enable the Adaptive Computing repository on the Torque MOM Hosts or client hosts.

On each host (physical machine), do the following:

1. If your site uses a proxy to connect to the Internet, do the following:

```
export http_proxy=http://<proxy_server_id>:<port>
export https_proxy=http://<proxy_server_id>:<port>
```

2. Download the 9.0.1 Moab HPC Suite RPM bundle from the [Adaptive Computing](#) website.
3. Untar the RPM bundle.

```
[root]# tar zxf <RPM bundle>
```

4. Change directories into the untarred directory.

i Consider reviewing the README file for additional details on using the RPM distribution tarball.

5. Install the suite repositories. The `-y` option installs with the default settings for the RPM suite.

i For a description of the options of the repository installer script, run:

```
[root]# ./install-rpm-repos.sh -h
```

```
[root]# ./install-rpm-repos.sh [<repository-directory>] [-y]
```

i If the installation returns the following warning line:

```
Warning: RPMDB altered outside of yum.
```

This is normal and can safely be ignored.

The [`<repository-directory>`] option is the directory where you want to copy the RPMs. If no argument is given, run "`install-rpm-repos.sh -h`" to view usage information and identify the default directory location. If the [`<repository-directory>`] already exists, RPMs will be added to the existing directory. No files are overwritten in [`<repository-directory>`].

A repository file is also created and points to the [`<repository-directory>`] location. For Red Hat 6-based or Red Hat 7-based systems, the repository file is created in `/etc/yum.repos.d/`. For SUSE 12-based systems, the repository files is created in `/etc/zypp/repos.d/`.

For ease in repository maintenance, the install script fails if Adaptive Computing RPMs are copied to different directories. If a non-default [`<repository-directory>`] is specified, please use the same directory for future updates.

The script installs the `createrepo` package and its dependencies. You must answer "y" to all the questions in order for the RPM install of the suite to work.

Additionally, the script installs:

- The EPEL and 10gen repositories on Red Hat 6-based and Red Hat 7-based systems.
- The openSUSE Apache:Modules, devel:languages:python, devel:languages:perl, and server:database repositories on SUSE 12-based systems.

6. Test the repository.

- Red-Hat 6-based or Red Hat 7-based systems

```
[root]# yum search moab
```

- SUSE 12-based systems

```
[root]# zypper search moab
```

If no error is given, the repository is correctly installed. The following is an example of the output after verifying the repository:

```
...
moab-accounting-manager.x86_64 : Moab Accounting Manager for Moab HPC Suite
moab-hpc-enterprise-suite.noarch : Moab HPC Suite virtual package
moab-insight.x86_64 : Moab Insight
moab-perl-RRDs.noarch : Moab RRDs
moab-tomcat-config.x86_64 : Tomcat Configuration for Web Services
moab-web-services.x86_64 : Moab Web Services
moab-workload-manager.x86_64 : Moab Workload Manager
moab-workload-manager-client.x86_64 : Moab Workload Manager Client
moab-workload-manager-common.x86_64 : Moab Workload Manager Common Files
moab-perl-data.noarch : Perl Configuration for perl packages by Adaptive Computing
moab-torque-client.x86_64 : Torque Client
moab-torque-common.x86_64 : Torque Common Files
moab-torque-devel.x86_64 : Torque Development Files
moab-torque-mom.x86_64 : Torque MOM agent
moab-torque-server.x86_64 : Torque Server
...
```

7. Continue with instructions to install or upgrade the Moab HPC Suite components. See [Installation and Upgrade Process on page 124](#) for more information.

Creating the moab-offline Tarball



The Moab Offline Tarball is *only* created if you are using the RPM Installation – Offline Method. See [RPM Installation and Upgrade Methods on page 123](#) for more information.

This topic contains instructions on how to create a moab-offline tarball on a web-enabled host outside of your Moab HPC Suite environment. This is the tarball that is then copied (using either by scp, DVD, USB or similar) to each host within your Moab HPC Suite environment.



The internet-enabled host *must* have the *exact* same OS as the hosts within your Moab HPC Suite environment. As the Moab HPC Suite can have several hosts, and each host may not use the same OS, you may need to repeat this procedure for each OS used.

These instructions assume the user is non-root, but has sudo rights.

On a web-enabled host, do the following:

1. If the host uses a proxy to connect to the Internet, do the following:

```
export http_proxy=http://<proxy_server_id>:<port>
export https_proxy=http://<proxy_server_id>:<port>
```

2. Download the 9.0.1 Moab HPC Suite RPM bundle from the [Adaptive Computing](#) website.
3. Untar the RPM bundle.

```
tar xzf <RPM bundle>
```

4. Change directories into the untarred directory.

i Consider reviewing the README file for additional details on using the RPM distribution tarball.

5. Install the suite repositories.

```
sudo ./install-rpm-repos.sh -y
```

i If the installation returns the following warning line:

```
Warning: RPMDB altered outside of yum.
```

This is normal and can safely be ignored.

The script installs the `createrepo` package and its dependencies. You must answer "y" to all the questions in order for the RPM install of the suite to work.

Additionally, the script installs:

- The EPEL and 10gen repositories on Red Hat 6-based and Red Hat 7-based systems.
- The openSUSE Apache:Modules, devel:languages:python, devel:languages:perl, and server:database repositories on SUSE 12-based systems.

6. Confirm you own /opt.

```
sudo chown <user>:<user> /opt
```

7. Create the moab-offline directory in which to store the RPMs.

```
mkdir /opt/moab-offline
```

8. Download the Moab HPC Suite RPMs into the moab-offline directory.

- Red Hat 6-based or Red Hat 7-based systems, do the following:
 - i. Symlink all the Moab HPC Suite RPMs to your moab-offline directory. This enables the repotrack utility to copy them.

```
ln -s /opt/adaptive-rpm-repository/rpm/*.rpm /opt/moab-offline/
```

- ii. Use repotrack to download all dependency RPMs.

```
repotrack -a x86_64 -p /opt/moab-offline moab-hpc-suite
```

- SUSE 12-based systems, do the following:

- i. Use zypper to download all packagers and dependencies with the `--download-only` option. The files will be downloaded into `/var/cache/zypp/packages`.

```
sudo zypper install --download-only moab-hpc-suite
```

- ii. Copy the downloaded RPMs into the moab-offline directory.

```
find /var/cache/zypp/packages/ -type f|xargs -I '{}' cp '{}' /opt/moab-offline/
```

9. Download the Java RPM into the moab-offline directory.

i The Java version may vary depending on the Moab HPC Suite components in your configuration. See [Component Requirements on page 9](#) for more information.

```
cd /opt/moab-offline
wget <java_url>
```

10. Create a repository file for the moab-offline directory.

The `createrepo` package and its dependencies should have been installed when you ran `./install-rpm-repos.sh -y`.

```
echo "[moab-offline]
name=moab-offline
baseurl=file:///opt/moab-offline
failovermethod=priority
enabled=1
gpgcheck=0" > moab-offline.repo
```

11. Create the moab-offline tarball. The "h" option ensures the symlinked targets will be copied, instead of just the links.

```
tar hczvf moab-offline.tgz moab-offline
```

This tarball can now be copied (using scp, DVD, USB drive, or similar) to *each* host within your Moab HPC Suite environment.

Preparing the Host – Offline Method

The offline method is available for configurations where the hosts in your environment do not have internet access in order to download the Moab HPC Suite RPM dependencies.

This topic describes how to deploy the moab-offline tarball so that you can install various Moab HPC Suite components and their dependencies on all the hosts in your environment.

On *each* host (physical machine), do the following:

1. If you have not already done so, copy the moab-offline tarball to the host. For example, copy it from a CD, USB drive, or Shared network drive. See [Creating the moab-offline Tarball on page 127](#) for instructions on how to create the tarball.
2. Place the moab-offline tarball in the /opt directory and enter that directory.

```
mv moab-offline.tgz /opt
cd /opt
```

3. Untar the moab-offline directory.

```
tar xvzf moab-offline.tgz
```

4. Copy the moab-offline.repo into place.

- Red Hat 6-based or Red Hat 7-based systems
 - Copy to yum.repos.d.

```
cp moab-offline/moab-offline.repo /etc/yum.repos.d/
```

- Update the cache.

```
yum clean all
```

- SUSE 12-based systems
 - Copy to zypp.repos.d.

```
cp moab-offline/moab-offline.repo /etc/zypp.repos.d/
```

- Update the cache.

```
zypper clean
```

5. Continue with instructions to install or upgrade the Moab HPC Suite components. See [Installation and Upgrade Process on page 124](#) for more information.

RPM Installations

This section provides instructions and other information for installing your Moab HPC Suite component using the RPM installation method.

In this section:

- [Preparing the Host – Typical Method on page 125](#)
- [Creating the moab-offline Tarball on page 127](#)
- [Preparing the Host – Offline Method on page 130](#)
- [Installing Torque Resource Manager on page 132](#)
- [Installing Moab Workload Manager on page 137](#)
- [Installing Moab Accounting Manager on page 142](#)
- [Installing Moab Web Services on page 154](#)
- [Installing Moab Insight on page 165](#)
- [Installing Moab Viewpoint on page 178](#)
- [Installing RLM Server on page 195](#)
- [Installing Remote Visualization on page 198](#)
- [Installing Nitro on page 218](#)
- [Installing Nitro Web Services on page 222](#)

Related Topics

- [About RPM Installations and Upgrades on page 123](#)

Installing Torque Resource Manager



If you intend to use Torque Resource Manager 6.0.1 with Moab Workload Manager, you must run Moab version 8.0 or later. However, some Torque 6.0 functionality requires Moab 9.0 or later.

This topic contains instructions on how to install, configure, and start Torque Resource Manager (Torque).

i For Cray systems, Adaptive Computing recommends that you install Moab and Torque Servers (head nodes) on commodity hardware (*not* on Cray compute/service/login nodes).

However, you must install the Torque pbs_mom daemon and Torque client commands on Cray login and "mom" service nodes since the pbs_mom must run on a Cray service node within the Cray system so it has access to the Cray ALPS subsystem.

See [Installation Notes for Moab and Torque for Cray](#) in the *Moab Workload Manager Administrator Guide* for instructions on installing Moab and Torque on a non-Cray server.

In this topic:

- [Prerequisites on page 133](#)
- [Install Torque Server on page 135](#)
- [Install Torque MOMs on page 136](#)
- [Configure Data Management on page 137](#)

Prerequisites

In this section:

- [Open Necessary Ports on page 133](#)
- [Verify the hostname on page 135](#)

Open Necessary Ports

Torque requires certain ports to be open for essential communication.

- For client and pbs_mom communication to pbs_server, the default port is 15001.
- For pbs_server communication to pbs_mom, the default port is 15002.
- For pbs_mom communication to pbs_mom, the default port is 15003.

For more information on how to configure the ports that Torque uses for communication, see [Configuring Ports](#) in the *Torque Resource Manager Administrator Guide* for more information.

If you have a firewall enabled, do the following:

1. On the Torque Server Host:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following line immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 15001 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=15001/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following port to the FW_SERVICES_EXT_TCP parameter
FW_SERVICES_EXT_TCP="15001"

[root]# service SuSEfirewall2 restart
```

2. On each Torque MOM Host (Compute Hosts):

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 15002 -j ACCEPT
-A INPUT -p tcp --dport 15003 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=15002/tcp --permanent
[root]# firewall-cmd --add-port=15003/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following ports to the FW_SERVICES_EXT_TCP parameter
FW_SERVICES_EXT_TCP="15002 15003"

[root]# service SuSEfirewall2 restart
```

Verify the hostname

On the Torque Server Host, confirm your host (with the correct IP address) is in your `/etc/hosts` file. To verify that the hostname resolves correctly, make sure that `hostname` and `hostname -f` report the correct name for the host.

Install Torque Server

i You *must* complete the prerequisite tasks earlier in this topic before installing the Torque Server. See [Prerequisites on page 133](#).

On the Torque Server Host, do the following:

1. If you are installing the Torque Server on its own host (recommend) and *not* on the same host where you installed another server (such as Moab Server), verify you completed the steps to prepare the host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).

2. Install the Torque Server RPM.

- Red Hat 6-based or Red Hat 7-based systems

```
yum install moab-torque-server
```

- SUSE 12-based systems

```
zypper install moab-torque-server
```

3. Verify `/usr/local/bin` and `/usr/local/sbin` are included in the `PATH`; if not present, add them.

```
[root]# cat > /etc/profile.d/torque.sh << EOL
export PATH=/usr/local/bin:/usr/local/sbin:\$PATH
EOL
[root]# . /etc/profile.d/torque.sh
```

4. Add the hostnames of your Torque MOMs (which is commonly all of your compute nodes) to the `/var/spool/torque/server_priv/nodes` file. You can remove the hostname entry for the Torque server node *unless* you will be running a Torque MOM daemon on this host. See [Managing Nodes](#) in the *Torque Resource Manager Administrator Guide* for information on syntax and options for specifying compute nodes.

Example:

```
vi /var/spool/torque/server_priv/nodes

node01 np=16
node02 np=16
...
```

5. Start the Torque server.

- Red Hat 6-based systems

```
[root]# service pbs_server start
[root]# service trqauthd start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start pbs_server.service
[root]# systemctl start trqauthd.service
```

Install Torque MOMs

In most installations, you will install a Torque MOM on each of your compute nodes.

Do the following:

1. From the Torque Server Host, copy the `moab-torque-common` and `moab-torque-mom` RPM files to each MOM node. It is also recommended that you install the `moab-torque-common` RPM so you can use client commands and submit jobs from compute nodes.

```
[root]# scp RPMs/moab-torque-common-*.rpm <torque-mom-host>:
[root]# scp RPMs/moab-torque-mom-*.rpm <torque-mom-host>:
[root]# scp RPMs/moab-torque-client-*.rpm <torque-mom-host>:
```

2. On each Torque MOM Host, install the RPMs; `moab-torque-common` is installed *first*.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# ssh root@<torque-mom-host>
[root]# yum install moab-torque-common-*.rpm moab-torque-mom-*.rpm moab-torque-client-*.rpm
```

- SUSE 12-based systems

```
[root]# ssh root@<torque-mom-host>
[root]# zypper install moab-torque-common-*.rpm moab-torque-mom-*.rpm moab-torque-client-*.rpm
```

3. On each Torque MOM Host, create or edit the `/var/spool/torque/server_name` file to contain the hostname of the Torque server.

```
[root]# echo <torque_server_hostname> > /var/spool/torque/server_name
```

4. On each Torque MOM Host, edit the `/var/spool/torque/mom_priv/config` file. This file is identical for all compute nodes and can be created on the Torque Server and distributed in parallel to all systems.

```
[root]# vi /var/spool/torque/mom_priv/config
$pbsserver      <torque_server_hostname> # hostname running pbs server
$logevent       225                     # bitmap of which events to log
```

5. On each Torque MOM Host, start the pbs_mom daemon.

- Red Hat 6-based systems

```
[root]# service pbs_mom start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start pbs_mom.service
```

6. If you installed the Torque Client RPM on the MOMs, then on each Torque MOM Host, start the trqauthd daemon.

- Red Hat 6-based systems

```
[root]# service trqauthd start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start trqauthd.service
```

Configure Data Management

When a batch job completes, stdout and stderr files are generated and placed in the spool directory on the master Torque MOM Host for the job instead of the submit host. You can configure the Torque batch environment to copy the stdout and stderr files back to the submit host. See [Configuring Data Management](#) in the *Torque Resource Manager Administrator Guide* for more information.

Related Topics

[Chapter 3 RPM installation Method on page 122](#)

Installing Moab Workload Manager

This topic contains instructions on how to install, configure, and start Moab Workload Manager (Moab).

i For Cray systems, Adaptive Computing recommends that you install Moab and Torque Servers (head nodes) on commodity hardware (*not* on Cray compute/service/login nodes).

However, you must install the Torque pbs_mom daemon and Torque client commands on Cray login and "mom" service nodes since the pbs_mom must run on a Cray service node within the Cray system so it has access to the Cray ALPS subsystem.

See [Installation Notes for Moab and Torque for Cray](#) in the *Moab Workload Manager Administrator Guide* for instructions on installing Moab and Torque on a non-Cray server.

In this topic:

- [Open Necessary Ports on page 138](#)
- [Install Moab Server on page 139](#)
- [Configure Torque to Trust Moab on page 141](#)
- [Verify the Installation on page 141](#)

Open Necessary Ports

Moab uses a configurable server port (default 42559) for client-server communication. If you intend to run client commands on a host different from the Moab Server Host, or if you will be using Moab in a grid, and if you have a firewall enabled, then you will need to configure the firewall to allow the server port.

On the Moab Server Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

# Needed on the Moab server for off-host client communication
-A INPUT -p tcp --dport 42559 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=42559/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following ports to the FW_SERVICES_EXT_TCP parameter as required

# Needed on the Moab server for off-host client communication
FW_SERVICES_EXT_TCP="42559"

[root]# service SuSEfirewall2 restart
```

Install Moab Server

i If your configuration uses firewalls, you must open the necessary ports before installing the Moab Server. See [Open Necessary Ports on page 138](#).

On the Moab Server Host do the following:

1. If your configuration uses firewalls, confirm you have opened the necessary ports. See [Open Necessary Ports on page 138](#).
2. If you have not already done so, complete the steps to prepare the Moab Server Host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).
3. Install RPM packages.

- a. Install the Moab Server RPMs.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install moab-workload-manager moab-workload-manager-hpc-configuration
```

- SUSE 12-based systems

```
[root]# zypper install moab-workload-manager moab-workload-manager-hpc-configuration
```

- b. If you are using Torque as a resource manager and installed the Torque Server on a different host (Torque Server Host; recommended) from the Moab Server (Moab Server Host), you will need to install the Torque client RPM on the Moab Server Host in order for Moab to interact with Torque.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install moab-torque-client
```

- SUSE 12-based systems

```
[root]# zypper install moab-torque-client
```

- c. If you are using Moab Accounting Manager and will be using the Native (custom script) accounting manager interface, and are installing the Moab Accounting Manager Server on a different host from the Moab Server (Moab Server Host) you will need to the install Moab Accounting

Manager client on the Moab Server Host in order for the custom scripts to use the MAM API.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install moab-accounting-manager
```

- SUSE 12-based systems

```
[root]# zypper install moab-accounting-manager
```

4. Source the following file to add the Moab executable directories to your current shell `$PATH` environment.

```
[root]# . /etc/profile.d/moab.sh
```

5. Copy your license file into the same directory as `moab.cfg` (`/opt/moab/etc/` by default). For example:

```
[root]# cp moab.lic $MOABHOMEDIR/etc/moab.lic
```

To verify the current status of your license, use `moab --about`.

Moab checks the status of the license every day just after midnight. At 60 and 45 days before, and daily from 30 days before license expiration to and including the license expiration date, Moab sends an e-mail to all level 1 administrators informing them of the pending Moab license expiration. A log record is also made of the upcoming expiration event. For the notifications to occur correctly, you must enable administrator email notification (see [Notifying Administrators of Failures](#) in the *Moab Workload Manager Administrator Guide*) and `moab.cfg` must contain email addresses for level 1 administrators. For example:

```
ADMINCFG[1] USERS=u1,u2,u3[,...]

USERCFG[u1] EMAILADDRESS=u1@company.com
USERCFG[u2] EMAILADDRESS=u2@company.com
USERCFG[u3] EMAILADDRESS=u3@company.com

MAILPROGRAM DEFAULT
```

i Moab will not run without a license. For information about obtaining a trial license, please contact [Adaptive Computing](#).

6. If you are using Torque as your resource manager and you installed the Torque Server on a different host (Torque Server Host) from the Moab Server (Moab Server Host), do the following:
 - a. Create or edit the `/var/spool/torque/server_name` file to contain the hostname of the Torque Server.

```
[root]# echo <Torque_server_hostname> > /var/spool/torque/server_name
```

- b. Verify that the Torque Server hostname used is *exactly* the name returned by a reverse hostname lookup.

```
[root]# cat /var/spool/torque/server_name | perl -lpe '$_=(gethostbyname($_))
[0]'
```

If different, take the necessary steps to make them match. For example, it may be necessary to add the Torque Server hostname to the `/etc/hosts` file on the Moab Server Host.

```
[root]# vi /etc/hosts
<Torque_server_ip_address> <Torque_server_hostname> <Torque_server_FQDN>
```

- c. Start the `trqauthd` daemon.

- Red Hat 6-based systems

```
[root]# service trqauthd start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start trqauthd.service
```

7. Start Moab (assumes Moab configured with the `--with-init` option).

- Red Hat 6-based systems

```
[root]# service moab start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start moab.service
```

Configure Torque to Trust Moab

If you are using Torque as a resource manager and you installed the Torque Server on a different host (Torque Host); recommended, do the following:

- On the *Torque* Host, add the name of the Moab Server Host (where Moab Server is installed) as a manager, and submit the host.

```
[root]# qmgr
Qmgr: set server managers += root@<moab_server_hostname>
Qmgr: set server submit_hosts += <moab_server_hostname>
Qmgr: exit
```

Verify the Installation

If you have a resource manager configured, verify that the scheduler is able to schedule a job.

- Submit a sleep job as a non-root user (adaptive is used in this example) and verify the job is running.

```
[root]# su - adaptive
[adaptive]$ echo sleep 150 | msub
[adaptive]$ showq
[adaptive]$ exit
```

Related Topics

[Chapter 3 RPM installation Method on page 122](#)

Installing Moab Accounting Manager

This topic contains instructions on how to install, configure, and start Moab Accounting Manager (MAM).

Perform the following:

1. [Plan Your Installation](#)
2. [Confirm Requirements](#)
3. [Open Necessary Ports](#)
4. [Install Dependencies, Packages, or Clients](#)
5. [Install MAM Server](#)
6. [Configure the MAM GUI](#)
7. [Access the MAM GUI](#)
8. [Configure Moab Workload Manager to use Moab Accounting Manager](#)
9. [Initialize Moab Accounting Manager](#)

Plan Your Installation

The first step is determining the number of different hosts (physical machines) required for your MAM installation.

Your MAM installation includes:

- MAM Server
- MAM Database
- MAM GUI (optional)
- MAM Clients (possibly several hosts)

Each of these components can be installed on their own hosts (meaning the actual physical machine) or can be combined on same hosts. For example, the MAM Database can be installed on the same *host* as the MAM Server. Or the MAM Server may be installed on the same host you installed the Moab Server.



If Moab Insight is part of your configuration, Insight requires a PostgreSQL 9.3. database. If your configuration will have the MAM PostgreSQL database on the *same* host as the Insight PostgreSQL database, you must install version 9.3 for the MAM PostgreSQL database.

Once you have determined which components are installed on which hosts, complete the rest of the instructions for the MAM installation.

i The instructions that follow in this topic will use the term Host after each component to reflect installing on a host (again, meaning the physical machine). For example, MAM Server Host and MAM Database Host. Depending on your configuration, Host may refer to as installed on its own machine or installed on the same machine as another component.

Confirm Requirements

In this section:

- [Hardware Requirements on page 143](#)
- [Supported Operating Systems on page 143](#)
- [Supported Databases on page 143](#)

Hardware Requirements

- Dual or Quad core Intel/AMD x86-64 processor
- At least 8 GB of RAM
- 1-2 TB disk space

i MAM is commonly installed on the same host as Moab; however, in some cases you might obtain better performance by installing them on different hosts.

Supported Operating Systems

MAM has been tested on the following variants of Linux:

- CentOS (6.x, 7.x)
- RHEL (6.x, 7.x)
- Scientific Linux (6.x, 7.x)
- SLES (12)

Supported Databases

MAM uses an RDBMS as a back end. If this is a new installation, use the following database:

- PostgreSQL 7.2 or higher

Adaptive Computing recommends that the database used by MAM does *not* reside on the same host as the database used by Insight. However, if you

choose to install the MAM PostgreSQL database on the *same* host where the Insight PostgreSQL database, then the MAM PostgreSQL database *must* be version 9.3.

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

Do the following as needed.

1. If you will be installing the MAM Server on a different host from where you installed the Moab Server *or* you will be installing the MAM Clients on other hosts, then on the MAM Server Host, open the MAM Server port (7112) in the firewall.

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod
# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 7112 -j ACCEPT
[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=7112/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="7112"
[root]# service SuSEfirewall2 restart
```

2. If using the MAM GUI, then on the MAM GUI Host, open the https port in the firewall for secure browser communication.

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod
# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 443 -j ACCEPT
[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=https/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="443"
[root]# service SuSEfirewall2 restart
```

3. If you will be installing the MAM Database on a different host from the MAM Server, then on the MAM Database Host, open the postgres port (5432) in the firewall.

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod
# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 5432 -j ACCEPT
[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=postgres/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5432"
[root]# service SuSEfirewall2 restart
```

Install Dependencies, Packages, or Clients

In this section:

- [Install and Initialize PostgreSQL Server on page 146](#)
- [Install Perl ReadLine \(Optional\) on page 147](#)

Install and Initialize PostgreSQL Server

Moab Accounting Manager uses a database for transactions and data persistence.

The MAM PostgreSQL database may be installed on:

- the same host as the MAM Server.
- a separate PostgreSQL database host.
- a separate *shared* PostgreSQL database host. If this shared database host *will* include the Insight PostgreSQL database, then the MAM PostgreSQL database *must* be version 9.3.

On the host where the MAM PostgreSQL database will reside, do the following:



These instructions assume you will be installing the MAM PostgreSQL database on a *different host* from where the Insight PostgreSQL database will reside.

If you wish to install *both* the MAM and the Insight PostgreSQL databases on the same host, different instructions are required. For example, you will need to enable the postgresql-9.3 RPM repo by following the RPM instructions to prepare the host (see [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#)) and you will need to modify the MAM PostgreSQL install instructions to reflect the 9.3 naming (see [Install PostgreSQL](#) for an example of how to install PostgreSQL 9.3 for Insight).

1. Install and initialize PostgreSQL.

- Red Hat 6-based systems

```
[root]# yum install postgresql-server
[root]# service postgresql initdb
```

- Red Hat 7-based systems

```
[root]# yum install postgresql-server
[root]# postgresql-setup initdb
```

- SUSE 12-based systems

```
[root]# zypper install postgresql-server
[root]# systemctl start postgresql.service
```

2. Configure trusted connections.

Edit or add a "host" line in the `pg_hba.conf` file for the interface from which the MAM Server will be connecting to the database and ensure that it specifies a secure password-based authentication method (for example, `md5`).

```
[root]# vi /var/lib/pgsql/data/pg_hba.conf

# Replace 127.0.0.1 with the IP address of the MAM Server Host if the
# MAM PostgreSQL server is on a separate host from the MAM server.
host    all             all             127.0.0.1/32         md5
host    all             all             ::1/128              md5
---
```

3. If the MAM Database Host is installed on a *different* host from where you will install the MAM Server, configure PostgreSQL to accept connections from the MAM Server Host.

```
[root]# vi /var/lib/pgsql/data/postgresql.conf

# Replace <mam-server-host> with the interface name from which the MAM server
# will be connecting to the database.
listen_addresses = '<mam-server-host>'
---
```

4. Start or restart the database.

- Red Hat 6-based systems

```
[root]# chkconfig postgresql on
[root]# service postgresql restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable postgresql.service
[root]# systemctl restart postgresql.service
```

Install Perl ReadLine (Optional)

Moab Accounting Manager can be optionally configured to provide command history editing functionality in the `mam-shell` command.

The `perl-Term-ReadLine-Gnu` package is recommended; however, depending on your OS version, it may not be already installed as a prerequisite package. You can install the RPM from a reputable third-party rpm site, or install from CPAN.

- Red Hat 6-based systems

No action needed. perl-Term-ReadLine-Gnu is installed as a prerequisite package.

- Red Hat 7.0-based systems

```
[root]# cpan Term::ReadLine::Gnu
```

- Red Hat 7.1-based systems

```
[root]# yum install perl-Term-ReadLine-Gnu
```

- SUSE 12-based systems

```
[root]# zypper install perl-Term-ReadLine-Gnu
```

Install MAM Server

i You *must* complete all the previous sections in this topic before installing MAM server. See the list of steps at the beginning of this topic.

On the MAM Server Host do the following:

1. If you are installing the MAM Server on its own host and *not* on the same host where you installed another server (such as Moab Server), verify you completed the steps to prepare the host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).
2. Install the MAM Server RPM.
 - Red Hat 6-based and Red Hat 7-based systems

```
yum install moab-accounting-manager
```

- i** If installing on RHEL, some packages may not be found in the standard RHEL distribution repositories.
- One way to overcome this problem is to install the missing dependencies from a CentOS repository using a reputable rpm download site (such as rpmfind.net or pbone.net). For example (for RHEL 7.1):

```
[root]# cat > /etc/yum.repos.d/centos.repo <<'EOF'
[centos]
name=CentOS $releasever - $basearch
baseurl=http://rpmfind.net/linux/centos/7.1.1503/os/$basearch/
enabled=0
gpgcheck=0
EOF

[root]# yum install --enablerepo=centos moab-accounting-manager
```

- Alternatively*, you can install the missing prerequisites from CPAN and then rerun the install.

```
[root]# yum install perl-CPAN
[root]# cpan Crypt::CBC Crypt::DES
[root]# yum install moab-accounting-manager
```

- SUSE 12-based systems

```
zypper install moab-accounting-manager
```

- As the database user, create a database called `mam` and grant database privileges to the `mam` user.

- i** PostgreSQL was installed and initialized earlier in this topic. See [Install and Initialize PostgreSQL Server on page 146](#).

```
[root]# su - postgres
[postgres]$ psql

create database mam;
create user mam with password 'changeme!';
\q

[postgres]$ exit
```

The *password* you define must be synchronized with the `database.password` value in `/opt/mam/etc/mam-server.conf`.

```
[root]# vi /opt/mam/etc/mam-server.conf

database.password = changeme!
```

4. Run the `hpc.sql` script to populate the Moab Accounting Manager database with objects, actions, and attributes necessary to function as an Accounting Manager.

```
[root]# su - mam
[mam]$ psql mam < /usr/share/moab-accounting-manager/hpc.sql
[mam]$ exit
```

5. Start the `mam` service.

- Red Hat 6-based systems

```
[root]# chkconfig --add mam
[root]# service mam start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable mam.service
[root]# systemctl start mam.service
```

Configure the MAM GUI

If you plan to use the web GUI, then on the MAM GUI Host, do the following:

1. As `root`, add or edit the SSL virtual host definition as appropriate for your environment. To do so, configure the `cgi-bin` directory in `ssl.conf`. Below the `cgi-bin` directory element, create an alias for `/cgi-bin` pointing to your `cgi-bin` directory. If you chose to install to a `cgi-bin` sub-directory, you might want to create an alias for that as well. Also, add `index.cgi` to the `DirectoryIndex` so you can use the shorter sub-directory name.
- Red Hat 6-based or Red Hat 7-based systems

```
[root]# vi /etc/httpd/conf.d/ssl.conf

<Directory "/var/www/cgi-bin">
## Add these lines
  Options ExecCGI
  AddHandler cgi-script .cgi
  AllowOverride All
  Order allow,deny
  Allow from all
</Directory>

# Aliases for /cgi-bin
Alias /cgi-bin/ /var/www/cgi-bin/
Alias /mam /var/www/cgi-bin/mam/

# Make shorter sub-dir name available
DirectoryIndex index.cgi
```

- SUSE 12-based systems

```
[root]# a2enflag SSL
[root]# cp /etc/apache2/vhosts.d/vhost-ssl.template /etc/apache2/vhosts.d/mam-ssl.conf
[root]# vi /etc/apache2/vhosts.d/mam-ssl.conf

<Directory "/srv/www/cgi-bin">
  Options ExecCGI
  AddHandler cgi-script .cgi
  AllowOverride All
  Require all granted
</Directory>

# Aliases for /cgi-bin
Alias /cgi-bin/ /srv/www/cgi-bin/
Alias /mam /srv/www/cgi-bin/mam/

# Make shorter sub-dir name available
DirectoryIndex index.cgi
```

2. For Red Hat-based systems where Security Enhanced Linux (SELinux) is enforced, you may need to customize SELinux to allow the web server to make network connections, use `setuid` for authentication, and write to the log file.

- a. Determine the current mode of SELinux.

```
[root]# getenforce
Enforcing
```

- If the command returns a mode of **Disabled** or **Permissive**, or if the `getenforce` command is not found, you can skip the rest of this step.
- If the command returns a mode of **Enforcing**, you can choose between options of customizing SELinux to allow the web GUI to perform its required functions or disabling SELinux on your system.

- b. If you choose to customize SELinux, do the following:

i SELinux can vary by version and architecture and that these instructions may not work in all possible environments.

i If you used the `--prefix=<prefix>` configuration option when you configured Moab Accounting Manager, you must replace references to `/opt/mam` in the example below with the `<prefix>` you specified. See [Moab Accounting Manager Configuration Options on page 94](#).

- Red Hat 6-based systems

```
[root]# cat > mamgui.te <<EOF
module mamgui 1.0;
require {
    type httpd_sys_script_t;
    type port_t;
    class capability setuid;
    class tcp_socket name_connect;
}
allow httpd_sys_script_t port_t:tcp_socket name_connect;
allow httpd_sys_script_t self:capability setuid;
EOF
[root]# checkmodule -M -m -o mamgui.mod mamgui.te
[root]# semodule_package -m mamgui.mod -o mamgui.pp
[root]# semodule -i mamgui.pp
[root]# setenforce 0
[root]# chcon -v -t httpd_sys_content_t /opt/mam/log
[root]# setenforce 1
```

- Red Hat 7-based systems

```
[root]# yum install checkpolicy policycoreutils-python
[root]# cat > mamgui.te <<EOF
module mamgui 1.0;
require {
    type httpd_sys_script_t;
    type unreserved_port_t;
    class tcp_socket name_connect;
}
allow httpd_sys_script_t unreserved_port_t:tcp_socket name_connect;
EOF
[root]# checkmodule -M -m -o mamgui.mod mamgui.te
[root]# semodule_package -m mamgui.mod -o mamgui.pp
[root]# semodule -i mamgui.pp
[root]# setenforce 0
[root]# chcon -v -t httpd_sys_rw_content_t /opt/mam/log
[root]# setenforce 1
```

3. For the highest security, it is recommended that you install a public key certificate that has been signed by a certificate authority. The exact steps to do this are specific to your distribution and the chosen certificate authority. An overview of this process for CentOS 7 is documented at https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/ch-Web_Servers.html#s2-apache-mod_ssl.

Alternatively, if your network domain can be secured from man-in-the-middle attacks, you could use a self-signed certificate. Often this does not require any additional steps since in many distributions, such as Red Hat, the Apache SSL configuration provides self-signed certificates by default.

If your configuration uses self-signed certificates, do the following:

- Red Hat 6-based and Red Hat 7-based systems

No action required. RedHat ships with ready-made certificates.

- SUSE 12-based systems

```
[root]# cd /etc/apache2
[root]# openssl genrsa -out ssl.key/server.key 1024
[root]# openssl req -new -key ssl.key/server.key -x509 -out ssl.crt/server.crt
```

4. Start or restart the HTTP server daemon.

- Red Hat 6-based systems

```
[root]# chkconfig httpd on
[root]# service httpd restart
```

- Red Hat 7-based systems

```
[root]# systemctl enable httpd.service
[root]# systemctl restart httpd.service
```

- SUSE 12-based systems

```
[root]# systemctl enable apache2.service
[root]# systemctl restart apache2.service
```

Access the MAM GUI

If you plan to use the web GUI, then on the MAM Server Host, do the following:

1. Create a password for the `mam` user to be used with the MAM Web GUI.

```
[root]# su - mam
[mam]$ mam-set-password
```

2. Verify the connection.

- a. Open a web browser and navigate to `https://<mam-server-host>/mam`.
- b. Log in as the `mam` user with the password you set in step 1.

Configure Moab Workload Manager to use Moab Accounting Manager

Do the following, where applicable:

1. On the *Moab* Server Host, edit the Moab configuration file.

```
[root]# vi /opt/moab/etc/moab.cfg
AMCFG[mam] TYPE=MAM HOST=<mam_server_host>
```

- a. Uncomment the AMCFG lines and customize as needed. See [Accounting, Charging, and Allocation Management](#) in the *Moab Workload Manager Administrator Guide*.

- b. If the Moab Server and the MAM Server are on the *same* host, set HOST to 'localhost'; otherwise, set HOST to the host name for the MAM Server (MAM Server Host).
2. Configure Moab to authenticate with MAM using the MAM secret key.
 - a. On the *MAM Server Host*, copy the auto-generated secret key from the `token.value` value in the `/opt/mam/etc/mam-site.conf` file.
 - b. On the *Moab Server Host*, add the secret key to the `moab-private.cfg` file as the value of the CLIENTCFG KEY attribute.

```
[root]# vi /opt/moab/etc/moab-private.cfg
CLIENTCFG[AM:mam] KEY=<MAMSecretKey>
```

3. Restart Moab

- Red Hat 6-based systems

```
[root]# service moab restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl restart moab.service
```

Initialize Moab Accounting Manager

You will need to initialize Moab Accounting Manager to function in the way that is most applicable to the needs of your site. See [Initial Setup](#) in the *Moab Accounting Manager Administrator Guide* to set up Moab Accounting Manager for your desired accounting mode.

Related Topics

[Chapter 3 RPM installation Method on page 122](#)

Installing Moab Web Services



You must deploy Moab Web Services on the *same* host as Moab Server (Moab Server Host). For documentation clarity, these instructions refer to the host for Moab Server and MWS Server as the MWS Server Host.

This topic contains instructions on how to install, configure, and start Moab Web Services (MWS).

In this topic:

- [Open Necessary Ports on page 155](#)
- [Install Dependencies, Packages, or Clients on page 156](#)

- [Install MWS Server on page 159](#)
- [Verify the Installation on page 164](#)

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

In this section:

- [Open the Tomcat Port \(8080\) on page 155](#)
- [Open the MWS MongoDB Database Port \(27017\) on page 155](#)

Open the Tomcat Port (8080)

On the MWS Server Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 8080 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=8080/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="8080"

[root]# service SuSEfirewall2 restart
```

Open the MWS MongoDB Database Port (27017)

i Depending on your system configuration, your MongoDB databases may not be installed on the same host as their corresponding component servers. For example, you may choose to install the MWS MongoDB database on the same host where you have installed other MongoDB databases instead of on the MWS Server Host.

Do the following, as needed:

- If you have chosen to install the MWS MongoDB database on the *same* host you installed other MongoDB databases (for example, the same host you installed the Moab MongoDB database), confirm the firewall port (27017) is already opened on that host.
- If you have chosen to install the MWS MongoDB database on a *different* host from other MongoDB databases, you will need to open the MWS MongoDB database port in firewall for that host. To open the port in the firewall, do the following:
 - Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

Add the following line immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 27017 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=27017/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 11-based or SUSE 12-based systems using SuSEfirewall2 (SUSE 11 is not supported on the MWS Server Host)

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="27017"

[root]# service SuSEfirewall2 restart
```

Install Dependencies, Packages, or Clients

In this section:

- [Install Java on page 156](#)
- [Install MongoDB on page 157](#)

Install Java

Install the Linux x64 RPM version of Oracle® Java® 8 Runtime Environment.

i Oracle Java 8 Runtime Environment is the recommended Java environment, but Oracle Java 7 is also supported. All other versions of Java, including OpenJDK/IcedTea, GNU Compiler for Java, and so on cannot run MWS.

On the MWS Server Host, do the following:

1. Install the Linux x64 RPM version of Oracle Java SE 8 JRE.
 - a. Go to the to the [Oracle Java download page](http://java.com/en/download/linux_manual.jsp) (http://java.com/en/download/linux_manual.jsp).
 - b. Copy the URL for the Linux x64 RPM version, and run the following command:
 - Red Hat 6-based or Red Hat 7-based systems

```
[root]# rpm -Uh <URL>
```

- SUSE 12-based systems

```
[root]# ln -s /usr/sbin/update-alternatives /usr/sbin/alternatives
[root]# rpm -Uh <URL>
```

Install MongoDB

To install and enable MongoDB, on the MWS Server Host, do the following:

1. Install MongoDB
 - Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install mongo-10gen-server
```

- SUSE 12-based systems

```
[root]# zypper install mongod
```

2. Start MongoDB.

i There may be a short delay (approximately 3 minutes) for Mongo to start the first time.

- Red Hat 6-based systems

```
[root]# chkconfig mongod on
[root]# service mongod start
```

- Red Hat 7-based systems

```
[root]# cat > /usr/lib/systemd/system/mongodb.service <<End-of-file
[Unit]
Description=High-performance, schema-free document-oriented database
After=syslog.target network.target

[Service]
Type=forking
User=mongod
Group=mongod
Environment=CONFIG=/etc/mongod.conf
Environment=OPTIONS=
EnvironmentFile=-/etc/sysconfig/mongod
ExecStart=/usr/bin/mongod -f \${CONFIG} \${OPTIONS}
PrivateTmp=true
LimitNOFILE=65536
TimeoutStartSec=180
StandardOutput=syslog
StandardError=syslog

[Install]
WantedBy=multi-user.target
End-of-file
[root]# rm -f /etc/init.d/mongod
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
```

- SUSE 12-based systems

```
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
```

3. Prepare the MongoDB database by doing the following:

a. Add the required MongoDB users.

i The passwords used below (`secret1`, `secret2`, and `secret3`) are examples. Choose your own passwords for these users.

```
[root]# mongo
> use admin;
> db.addUser("admin_user", "secret1");
> db.auth ("admin_user", "secret1");

> use moab;
> db.addUser("moab_user", "secret2");
> db.addUser("mws_user", "secret3", true);

> use mws;
> db.addUser("mws_user", "secret3");
> exit
```

i Because the `admin_user` has read and write rights to the `admin` database, it also has read and write rights to all other databases. See [Control Access to MongoDB Instances with Authentication](http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication/) (<http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication/>) for more information.

b. Enable authentication in MongoDB.

- Red Hat 6-based systems

```
[root]# vi /etc/mongod.conf
auth = true
[root]# service mongod restart
```

- Red Hat 7-based systems

```
[root]# vi /etc/mongod.conf
auth = true
[root]# systemctl restart mongod.service
```

- SUSE 12-based systems

MongoDB authentication is already enabled. No further action is needed.

Install MWS Server

i You *must* complete the tasks to install the dependencies, packages, or clients before installing MWS Server. See [Install Dependencies, Packages, or Clients on page 156](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing MWS Server. See [Open Necessary Ports on page 155](#).

On the MWS Host, do the following:

1. Install the MWS RPMs.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install moab-web-services moab-web-services-hpc-configuration
```

- SUSE 12-based systems

```
[root]# zypper install moab-web-services moab-web-services-hpc-configuration
```

2. Connect Moab to MongoDB

i The USEDATABASE parameter is unrelated to the MongoDB configuration.

- a. Set the **MONGOSERVER** parameter in `/opt/moab/etc/moab.cfg` to the MongoDB server hostname. Use localhost as the hostname if Moab and MongoDB are on the same host.

```
MONGOSERVER <host>[:<port>]
```

If your **MONGOSERVER** host is set to anything other than localhost, edit the `/etc/mongod.conf` file on the MongoDB Server host and either comment out any `bind_ip` parameter or set it to the correct IP address.

```
# Listen to local interface only. Comment out to listen on all interfaces.
#bind_ip=127.0.0.1
```

- b. In the `/opt/moab/etc/moab-private.cfg` file, set the **MONGOUSER** and **MONGOPASSWORD** parameters to the MongoDB `moab_user` credentials you set. See [Install MongoDB on page 157](#) earlier in this topic.

```
MONGOUSER    moab_user
MONGOPASSWORD secret2
```

- c. Verify that Moab is able to connect to MongoDB.

- Red Hat 6-based systems

```
[root]# service moab restart
[root]# mdiag -S | grep Mongo

Mongo connection (localhost) is up (credentials are set)
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl restart moab.service
[root]# mdiag -S | grep Mongo

Mongo connection (localhost) is up (credentials are set)
```

3. Secure communication using secret keys

- a. (Required) Moab and MWS use Message Authentication Codes (MAC) to ensure messages have not been altered or corrupted in transit. Generate a key and store the result in `/opt/moab/etc/.moab.key`.

- Red Hat 6-based systems

```
[root]# service moab stop
[root]# dd if=/dev/urandom count=24 bs=1 2>/dev/null | base64 >
/opt/moab/etc/.moab.key
[root]# chown root:root /opt/moab/etc/.moab.key
[root]# chmod 400 /opt/moab/etc/.moab.key
[root]# service moab start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl stop moab.service
[root]# dd if=/dev/urandom count=24 bs=1 2>/dev/null | base64 >
/opt/moab/etc/.moab.key
[root]# chown root:root /opt/moab/etc/.moab.key
[root]# chmod 400 /opt/moab/etc/.moab.key
[root]# systemctl start moab.service
```

- b. (Optional) Moab supports message queue security using AES. This feature requires a Base64-encoded 16-byte (128-bit) shared secret.

- a. Generate a key and append the result to `/opt/moab/etc/moab-private.cfg`.

- Red Hat 6-based systems

```
[root]# service moab stop
[root]# echo "MESSAGEQUEUESECRETKEY $(dd if=/dev/urandom count=16 bs=1
2>/dev/null | base64)" >> /opt/moab/etc/moab-private.cfg
[root]# service moab start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl stop moab.service
[root]# echo "MESSAGEQUEUESECRETKEY $(dd if=/dev/urandom count=16 bs=1
2>/dev/null | base64)" >> /opt/moab/etc/moab-private.cfg
[root]# systemctl start moab.service
```

i If MWS is configured to encrypt the message queue and Moab is not (or vice versa), then MWS will ignore the messages from Moab. Furthermore, all attempts to access the MWS service resource will fail.

- b. Verify that encryption is on for the ZeroMQ connection.

```
[root]# mdiag -S|grep 'ZeroMQ MWS'

ZeroMQ MWS connection is bound on port 5570 (encryption is on)
```

4. Set up the MWS configuration file.

- a. In the `/opt/mws/etc/mws-config.groovy` file, change these settings:

- **moab.secretKey**: Must match the Moab secret key you generated earlier (contained in `/opt/moab/etc/.moab.key`).
- **auth.defaultUser.username**: Any value you like, or leave as is.
- **auth.defaultUser.password**: Any value you like, but choose a strong password.
- **moab.messageQueue.secretKey**: If you opted to configure a message queue security key in MWS, this parameter value should match exactly that key specified in `/opt/moab/etc/moab-private.cfg` for the `MESSAGEQUEUESECRETKEY` Moab configuration parameter you generated earlier.

⚠ If MWS is configured to encrypt the message queue and Moab is not (or vice versa), then the messages from Moab will be ignored. Furthermore, all attempts to access the MWS service resource will fail.

```
[root]# vi /opt/mws/etc/mws-config.groovy

// Replace <ENTER-KEY-HERE> with the contents of /opt/moab/etc/.moab.key.

moab.secretKey = "<ENTER-KEY-HERE>"
moab.server = "localhost"
moab.port = 42559

// Replace <ENTER-KEY-HERE> with the value of MESSAGEQUEUESECRETKEY in
/opt/moab/etc/moab-private.cfg.

moab.messageQueue.secretKey = "<ENTER-KEY-HERE>"

// Change these to be whatever you like.

auth.defaultUser.username = "moab-admin"
auth.defaultUser.password = "changeme!"
```

 If you do not change **auth.defaultUser.password**, your MWS will not be secure (because anyone reading these instructions would be able to log into your MWS). Here are some [tips](#) for choosing a good password.

b. If you are using Moab Accounting Manager, change these settings in `/opt/mws/etc/mws.d/mws-config-hpc.groovy`:

- **mam.secretKey**: needs to match the MAM secret key in `/opt/mam/etc/mam-site.conf` on the MAM Server (as **token.value**)
- **mam.server**: set to the hostname of the MAM Server
- **mam.port**: set to the port of the MAM Server

```
[root]# vi /opt/mws/etc/mws.d/mws-config-hpc.groovy

mam.secretKey = "<ENTER-KEY-HERE>"
mam.server = "localhost"
mam.port = 7112
```

c. Do *one* of the following:

 You can configure only one authentication method in `mws-config.groovy`—LDAP or PAM, but not both. If you have configured both LDAP and PAM, MWS defaults to using LDAP.

If you need multiple authentication methods, you must add them to your local PAM configuration. See your distribution documentation for details.

- If you are configuring an MWS connection to your LDAP server, add the following parameters to the `mws-config.groovy` file:

```
ldap.server = "192.168.0.5"
ldap.port = 389
ldap.baseDNs = ["dc=acme,dc=com"]
ldap.bindUser = "cn=Manager,dc=acme,dc=com"
ldap.password = "*****"
ldap.directory.type = "OpenLDAP Using InetOrgPerson Schema"
```

This is just an example LDAP connection. Be sure to use the appropriate domain controllers (dc) and common names (cn) for your environment.

i If you followed the Adaptive Computing tutorial, [Setting Up OpenLDAP on CentOS 6 on page 231](#), your `ldap.directory.type` should be set to "OpenLDAP Using InetOrgPerson Schema." However, the use of other schemas is supported. For more information see [LDAP Configuration Using mws-config.groovy](#).

i To see how to configure a secure connection to the LDAP server, see [Securing the LDAP Connection](#).

- If you are configuring MWS to use PAM, add the the `pam.configuration.service` parameter to the `mws-config.groovy` file. For example:

```
pam.configuration.service = "login"
```

This is just an example PAM configuration file name. Make sure you specify the name of the configuration file you want MWS to use.

! Configuring MWS to authenticate via PAM using local `passwd` and `shadow` files presents a significant security risk. To make local authentication work, you would need to run Tomcat as root or give Tomcat read access to `/etc/shadow`. This configuration is highly discouraged and is not supported by Adaptive Computing.

The recommended approach is to configure PAM and NSS to authenticate against NIS or LDAP. For example, to make sure users with both local and NIS accounts are authenticating against NIS, configure the `nsswitch.conf` file as shown below.

```
passwd: nis files
shadow: nis files
group: nis files
```

i For more information about PAM configuration with MWS, see [PAM \(Pluggable Authentication Module\) Configuration Using mws-config.groovy](#).

- d. Add the `grails.mongo.username` and `grails.mongo.password` parameters to the `mws-config.groovy` file. Use the MWS credentials you added to MongoDB.

```
...
grails.mongo.username = "mws_user"
grails.mongo.password = "secret3"
```

5. Start or restart Tomcat.

- Red Hat 6-based systems

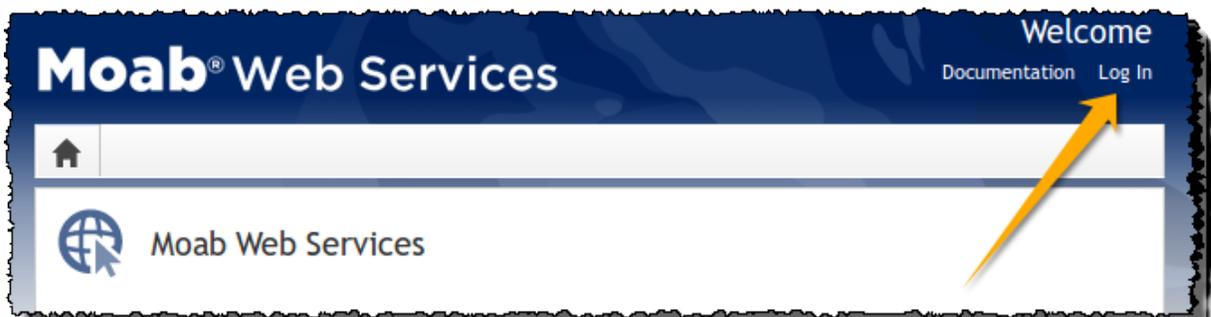
```
[root]# chkconfig tomcat on
[root]# service tomcat restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable tomcat.service
[root]# systemctl restart tomcat.service
```

Verify the Installation

1. Open a web browser.
2. Navigate to `http://<server>:8080/mws/`. You will see some sample queries and a few other actions.
3. Log in to MWS to verify that your credentials are working. (Your login credentials are the `auth.defaultUser.username` and `auth.defaultUser.password` values you set in the `/opt/mws/etc/mws-config.groovy` file.)



i If you encounter problems, or if the application does not seem to be running, see the steps in [Moab Web Services Issues on page 269](#).

Related Topics

- [Chapter 3 RPM installation Method on page 122](#)
- [Installing Moab Workload Manager on page 137](#)

Installing Moab Insight

This topic contains instructions on how to install Moab Insight (Insight).

Because Insight accumulates data for one cluster at a time, one Insight Server (daemon) should service one Moab instance. However, you can configure PostgreSQL to aggregate data using database replication mechanisms if you desire cross-cluster data.

i Moab Workload Manager and Insight both tend to heavily consume system resources. Therefore, Adaptive Computing *requires* that the Insight Server and the Moab Workload Manager Server run on different hosts. For these installation instructions, the "Moab Server Host" refers to one host and the "Insight Server Host" refers to another host.

In this topic:

- [Prerequisites on page 165](#)
- [Dependencies, Packages, or Client Installations on page 168](#)
- [Install Insight on page 174](#)

Prerequisites

In this section:

- [Open Necessary Ports on page 165](#)
- [Verify the hostname on page 168](#)

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

In this section:

- [Open the Insight Server Port \(5568\)](#)
- [Open the Insight PostgreSQL Database Port \(5432\)](#)
- [Open the MongoDB Database Port \(27017\)](#)
- [Open the Moab Server Ports \(5574 and 5575\)](#)

Open the Insight Server Port (5568)

On the Insight Server Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following line immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 5568 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=5568/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="5568"

[root]# service SuSEfirewall2 restart
```

Open the Insight PostgreSQL Database Port (5432)

Insight requires access to the Insight PostgreSQL database. Depending on your system configuration, your PostgreSQL databases may not be installed on the same host as their corresponding component servers. For example, you may choose to install the Insight PostgreSQL database on the same host where you have installed the Moab PostgreSQL database instead of on the Insight Server Host.

On the Insight Database Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 5432 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=5432/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5432"
[root]# service SuSEfirewall2 restart
```

Open the MongoDB Database Port (27017)

i Insight also requires access to the Moab MongoDB database. Depending on your system configuration, your MongoDB databases may not be installed on the same host as their corresponding component servers. For example, you may choose to install the Insight MongoDB on the same host where you have installed other MongoDB databases instead of on the Insight Server Host.

Do the following, as needed:

- If you have chosen to install the Insight MongoDB database on the *same* host you installed the Moab MongoDB database, confirm the firewall port (27017) is already opened on that host.
- If you have chosen to install the Insight MongoDB database on a *different* host from the Moab MongoDB database (for example if you installed the Insight MongoDB database on the Insight Server Host or on still a different host), you will need to make sure that *both* the Moab MongoDB database host and the Insight MongoDB database host have the firewall port (27017) open. To open the port in the firewall, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 27017 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=27017/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="27017"
[root]# service SuSEfirewall2 restart
```

Open the Moab Server Ports (5574 and 5575)

On the Moab Server Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 5574 -j ACCEPT
-A INPUT -p tcp --dport 5575 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=5574/tcp --permanent
[root]# firewall-cmd --add-port=5575/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="5574 5575"

[root]# service SuSEfirewall2 restart
```

Verify the hostname

On the Insight Server Host, confirm your host (with the correct IP address) is in your `/etc/hosts` file. To verify that the hostname resolves correctly, make sure that `hostname` and `hostname -f` report the correct name for the host.

Dependencies, Packages, or Client Installations

In this section:

- [Install Java on page 168](#)
- [Install MongoDB on page 169](#)
- [Install PostgreSQL on page 172](#)

Install Java

Install the Linux x64 RPM version of Oracle® Java® 8 Runtime Environment.

i Oracle Java 8 Runtime Environment is the recommended Java environment, but Oracle Java 7 is also supported.. All other versions of Java, including OpenJDK/IcedTea, GNU Compiler for Java, and so on cannot run Insight.

On the Insight Server Host, do the following:

1. Install the Linux x64 RPM version of Oracle Java SE 8 JRE.
 - a. Go to the to the [Oracle Java download page](http://java.com/en/download/linux_manual.jsp) (http://java.com/en/download/linux_manual.jsp).
 - b. Copy the URL for the Linux x64 RPM version, and run the following command:
 - Red Hat 6-based or Red Hat 7-based systems

```
[root]# rpm -Uh <URL>
```

- SUSE 12-based systems

```
[root]# ln -s /usr/sbin/update-alternatives /usr/sbin/alternatives
[root]# rpm -Uh <URL>
```

Install MongoDB

i The Insight MongoDB may be installed on the Insight Server Host or on different host. If you will install on a different host, and your configuration uses firewalls, open the necessary port. See [Open Necessary Ports on page 165](#)

To install and enable MongoDB, do the following:

1. On the host you have chosen to install the Insight MongoDB database, do the following:
 - a. Install `mongo-10gen-server`.
 - Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install mongo-10gen-server --exclude mongodb-org,mongodb-org-server
```

i Running yum upgrade will replace MongoDB 2.4.x with a more recent, and incompatible version. Consider using yum version lock to maintain MongoDB 2.4.x.

- SUSE 12-based systems

```
[root]# zypper install mongodb
```

- b. Start MongoDB.

i There may be a short delay (approximately 3 minutes) for Mongo to start the first time.

- Red Hat 6-based systems

```
[root]# chkconfig mongod on
[root]# service mongod start
```

- Red Hat 7-based systems

```
[root]# cat > /usr/lib/systemd/system/mongodb.service <<End-of-file
[Unit]
Description=High-performance, schema-free document-oriented database
After=syslog.target network.target

[Service]
Type=forking
User=mongod
Group=mongod
Environment=CONFIG=/etc/mongod.conf
Environment=OPTIONS=
EnvironmentFile=-/etc/sysconfig/mongod
ExecStart=/usr/bin/mongod -f \${CONFIG} \${OPTIONS}
PrivateTmp=true
LimitNOFILE=65536
TimeoutStartSec=180
StandardOutput=syslog
StandardError=syslog

[Install]
WantedBy=multi-user.target
End-of-file
[root]# rm -f /etc/init.d/mongod
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
```

- SUSE 12-based systems

```
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
```

2. Add the required MongoDB users to Insight MongoDB and Moab MongoDB; regardless of whether they share a host.

i These instructions show password examples (`secret1`, `secret2`, and `secret3`). Choose your own passwords for these users.

- Insight MongoDB

```
[root]# mongo
> use admin;
> db.addUser("admin_user", "secret1");
> db.auth("admin_user", "secret1");
> use insight;
> db.addUser("insight user", "secret4");
> db.addUser("mws_user", "secret3", true);

> exit
```

- Moab MongoDB

```
[root]# mongo
> use admin;
> db.auth("admin_user", "secret1");
> use moab;
> db.addUser ("insight_user", "secret4", true);
> exit
```

i Because the `admin_user` has read and write rights to the `admin` database, it also has read and write rights to all other databases. See [Control Access to MongoDB Instances with Authentication](http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication/) (<http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication/>) for more information.

3. Edit the MongoDB configuration.

Verify user authentication is enabled in MongoDB and that the mongo server is listening for external connections. By default some versions of MongoDB listen only for connections from localhost. Commenting out `bind_ip` causes the mongo server to listen on all interfaces.

i If Insight MongoDB resides on a different host from Moab MongoDB, you will need to perform this procedure twice (once for each host).

- Red Hat 6-based systems

```
[root]# vi /etc/mongod.conf
#bind_ip = <local_host>
...
auth = true
[root]# service mongod restart
```

- Red Hat 7-based systems

```
[root]# vi /etc/mongod.conf
#bind_ip = <local_host>
...
auth = true
[root]# systemctl restart mongod.service
```

- SUSE 12-based systems

```
[root]# vi /etc/mongod.conf
#bind_ip = <local_host>
...
auth = true
[root]# systemctl restart mongod.service
```

Install PostgreSQL

i The Insight PostgreSQL database may be installed on the Insight Server Host or on different host. If you will install on a different host, and your configuration uses firewalls, open the necessary port. See [Open Necessary Ports on page 165](#).

On the host you have chosen to install the Insight PostgreSQL database, do the following:

i These instructions are for installing PostgreSQL 9.3; however Insight may support other versions. See [Component Requirements on page 9](#) for requirements information.

1. Install PostgreSQL.

- Red Hat 6-based systems

```
[root]# yum install postgresql93-server
[root]# service postgresql-9.3 initdb
```

- Red Hat 7-based systems

```
[root]# yum install postgresql93-server
[root]# /usr/pgsql-9.3/bin/postgresql93-setup initdb
```

- SUSE 12-based systems

```
[root]# zypper install postgresql93-server
[root]# systemctl start postgresql
```

2. Edit the PostgreSQL configuration file to listen for connections on all interfaces. See the documentation provided in the postgresql.conf file if you want to be more restrictive.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# vi /var/lib/pgsql/9.3/data/postgresql.conf
# Uncomment the listen addresses line in the configuration:
listen_addresses = '*'
```

- SUSE 12-based systems

```
[root]# vi /var/lib/pgsql/data/postgresql.conf
# Uncomment the listen addresses line in the configuration:
listen_addresses = '*'
```

3. Edit the PostgreSQL host-based authentication (HBA) configuration file to enable TCP connections using encrypted passwords (change ident to md5) and add a line for each host that needs to connect to the database. See the "ADDRESS" documentation provided in pg_hba.conf for details.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# vi /var/lib/pgsql/9.3/data/pg_hba.conf
```

- SUSE 12-based systems

```
[root]# vi /var/lib/pgsql/data/pg_hba.conf
```

If you are using MWS, add the IP address of the host on which MWS Server is installed. This is shown in the following example as `<MWS_host_address>`.

i Using "0.0.0.0/0" in place of "`<MWS_host_address>`" will allow connections from all hosts.

i If the "host" lines are not present, add them as they appear in the example.

```
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local all all peer
# IPv4 local connections:
host all all <MWS_host_address>/32 md5 # If
using MWS
host all all <Insight_host_address>/32 md5
host all all 127.0.0.1/32 md5
# IPv6 local connections:
host all all ::1/128 md5
```

4. Start or restart the PostgreSQL database.

- Red Hat 6-based systems

```
[root]# chkconfig postgresql-9.3 on
[root]# service postgresql-9.3 start
```

- Red Hat 7-based systems

```
[root]# systemctl enable postgresql-9.3.service
[root]# systemctl restart postgresql-9.3.service
```

- SUSE 12-based systems

```
[root]# systemctl enable postgresql.service
[root]# systemctl restart postgresql.service
```

Install Insight

i You *must* complete the tasks to install the dependencies, packages, or clients before installing Insight Server. See [Dependencies, Packages, or Client Installations on page 168](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing Insight Server. See [Open Necessary Ports on page 165](#).

! These instructions contain steps to edit the `/opt/insight/etc/config.groovy` file.

Commented out values in the `config.groovy` file are not necessarily the default values.

It is recommended that anytime you edit the `config.groovy` file that you first stop Insight, edit the file and then restart Insight.

1. If you have not already done so, complete the steps to prepare the Insight Server Host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).

2. Install the Insight RPM.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install moab-insight
```

- SUSE 12-based systems

```
[root]# zypper install moab-insight
```

i If the installation returns the following warning line:

```
warning: rpmts_HdrFromFdno: Header V4 RSA/SHA1
Signature, key ID 952741e1: NOKEY
```

```
Retrieving key from file:///opt/adaptive-rpm-
repository/key/GPG_ADAPTIVE_COMPUTING_INC_EL_6_KEY
```

```
Importing GPG key 0x952741E1:
```

```
Userid: "Adaptive Computing Enterprises, Inc. (EL 6 key)
<info@adaptivecomputing.com>"
```

```
From : /opt/adaptive-rpm-repository/key/GPG_ADAPTIVE_
COMPUTING_INC_EL_6_KEY
```

This is normal. You can safely input `y` and continue.

3. Create the Insight user and PostgreSQL database.

 PostgreSQL was installed earlier in this topic. See [Install PostgreSQL on page 172](#).

 This instructions show the default changeme! password. Change this password according to your password security process.

a. Change to the postgres user.

```
su - postgres
```

b. Do the following:

```
[postgres]$ psql
CREATE USER moab_insight WITH PASSWORD 'changeme!';
CREATE DATABASE moab_insight WITH OWNER=moab_insight;
CREATE DATABASE moab_insight_reference WITH OWNER=moab_insight;
\q
```

c. Initialize the `moab_insight` database. This sets up event triggers so that database schema validation works correctly in Insight.

```
[postgres]$ psql moab_insight -f /opt/insight/db/initialize.sql
```

d. If you are also using MWS, create and grant permissions for the MWS user to query the database.

i. Log in to the `moab_insight` database.

```
[postgres]$ psql moab_insight
```

ii. Create the MWS user and grant the MWS user select permissions on all tables in the public schema.

```
CREATE USER mws WITH PASSWORD 'changeme!';
GRANT SELECT ON ALL TABLES IN SCHEMA public TO mws;
```

iii. Connect to the `moab_insight` database as the `moab_insight` user.

```
\connect moab_insight moab_insight localhost
```

iv. Grant the MWS user select permissions on all tables in the public schema.

```
ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT SELECT ON TABLES TO mws;
\q
[postgres]$ exit
```

4. If you are using MWS, on the MWS Server Host:

- a. Add or edit the `dataSource.insight.url` parameter in the `mws-config.groovy` file to specify the JDBC database connection information for the Insight Server. See [Configuration](#) in the *Moab Web Services Reference Guide* for more information on the MWS configuration properties.

i In the following example, `<insight_postgresql_server_ip_address>` represents the DNS name for the host on which the Insight PostgreSQL database resides. This example also uses the default PostgreSQL port number (5432).

```
[root]# vi /opt/mws/etc/mws-config.groovy

dataSource.insight.url = "jdbc:postgresql://<insight_postgresql_server_ip_
address>:5432/moab_insight"
dataSource.insight.password = "<postgresql_mws_user_password>"
```

b. Restart Tomcat.

- Red Hat 6-based systems

```
[root]# service tomcat restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl restart tomcat.service
```

5. Configure Insight's connection to the Insight PostgreSQL database. On the Insight Server Host, edit `/opt/insight/etc/config.groovy` as follows:

```
jdbc.referenceUrl = "jdbc:postgresql://<insight_postgresql_server_ip_address>/moab_
insight_reference"
jdbc.url = "jdbc:postgresql://<insight_postgresql_server_ip_address>/moab_insight"
jdbc.username = "moab_insight"
jdbc.password = "changeme!"
```

6. Configure Insight's connection to the Insight MongoDB database *and* the Moab MongoDB database. On the Insight Server Host, edit `/opt/insight/etc/config.groovy` as follows:

```
mongo.host="<insight mongo host>"
mongo.port=<insight mongo port>
mongo.username="insight user"
mongo.password="secret4"

moab.mongo.host="<moab mongo host>"
moab.mongo.port=<moab mongo port>
moab.mongo.username="insight user"
moab.mongo.password="secret4"
```

i "secret4" is the password you specified when installing the mongoDB. See [Install MongoDB on page 169](#).

7. On the Insight Server Host, configure the `moab.host` and `messageQueue.secretKey` parameters in the Insight configuration file `/opt/insight/etc/config.groovy`.

```
moab.host = "<moab server>"
messageQueue.secretKey = "<secret key>"
```

The `<secret key>` must match the secret key configured in `moab-private.cfg` on the Moab server for the **MESSAGEQUEUESECRETKEY** configuration parameter.

8. On the Insight Server Host, verify that Insight runs on startup.

- Red Hat 6-based systems.

```
[root]# chkconfig insight on
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable insight.service
```

9. On the Moab Server Host, configure Moab's connection to Insight.

- a. In `/opt/moab/etc/moab.cfg`, configure the **INSIGHTENDPOINT** parameter so that Moab can connect to Insight. See [Moab Parameters](#) in the *Moab Workload Manager Administrator Guide* for parameter information.

```
INSIGHTENDPOINT <hostname>[:<port>]
```

`<hostname>` is the server where Insight is located. `<hostname>` is required, `<port>` is optional.

- b. In `/opt/moab/etc/moab-private.cfg`, configure the **MESSAGEQUEUESECRETKEY** parameter so that Moab can connect to Insight. See [Secure communication using secret keys on page 160](#).

```
MESSAGEQUEUESECRETKEY <secret key>
```

The `<secret key>` is required when updating the Insight configuration file later in this procedure.

- c. Restart Moab in order for the new configuration parameters to take effect.

- Red Hat 6-based systems

```
service moab restart
```

- Red Hat 7-based or SUSE 12-based systems

```
systemctl restart moab.service
```

d. Verify that Moab is properly configured to connect to Insight.

```
mdiag -S | grep Insight
```

You should see something similar to the following:

```
[root]# mdiag -S | grep Insight
ZeroMQ Insight connection is bound on port 5574 (reliability port 5575) on host
* using Insight endpoint <the insight hostname displays here>:5568
encryption is on)
ZeroMQ Insight reliable message delivery is using store file(s) up to 1024 MB in
/opt/moab/spool/insight_store/
```

10. On the Insight Server Host, start Insight.

- Red Hat 6-based systems

```
[root]# service insight start
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl start insight.service
```



The first time you start Insight it will take a minute or two to create the database schema. Although 'service insight start' will quickly return OK, it is not safe to terminate Insight while this initialization is taking place. Rebooting or terminating Insight during this initialization may cause the database to not be initialized correctly.

You will know it is safe to reboot or terminate Insight if you see the following line in `/opt/insight/log/insight.log`.

```
2014-12-11T18:36:08.059-0700    main    INFO
com.ace.insight.app.Application 0          Started Application in 89.502
seconds (JVM running for 89.882)
```

Related Topics

[Chapter 3 RPM installation Method on page 122](#)

Installing Moab Viewpoint

This topic contains instructions on how to install Moab Viewpoint (Viewpoint).

In this topic:

- [Prerequisites on page 179](#)
- [Install Viewpoint Server on page 183](#)

- [Enable Access to the Viewpoint File Manager on page 188](#)
- [License and Configure Viewpoint on page 189](#)
- [Grant Users Access to Viewpoint on page 195](#)

i Viewpoint requires a connection to Moab Server and MWS installed on the shared host. Viewpoint may also be installed on that shared host or on a different host. For documentation clarity, the instructions refer to the shared Moab Server and MWS Server host as the Moab Server Host and the host on which you install Viewpoint Server as the Viewpoint Server Host.

Prerequisites

In this section:

- [Security Enhanced Linux on page 179](#)
- [Open Necessary Ports on page 180](#)
- [Configure the ViewpointQueryHelper Plugin on page 182](#)

Security Enhanced Linux

For Red Hat-based systems where Security Enhanced Linux (SELinux) is enforced, you need to adjust SELinux to allow the web server to make network connections and create and write to the log file.

For Red Hat 6-based or Red Hat 7-based systems, do the following on the Viewpoint Server Host:

1. To determine the current mode of SELinux, run `getenforce`.

```
[root]# getenforce
```

2. If the command returns a mode of Disabled or Permissive, or if the `getenforce` command is not found, you can skip the rest of this procedure.
3. If the command returns a mode of Enforcing, you can choose between options of customizing SELinux to allow the web GUI to perform its required functions or disabling SELinux on your system.
 - If you choose to customize SELinux:

i SELinux can vary by version and architecture and that these instructions may not work in all possible environments.

```
[root]# yum install policycoreutils-python
[root]# semanage permissive -a httpd_t
```

- If you choose to disable SELinux:

```
[root]# vi /etc/sysconfig/selinux
SELINUX=disabled
[root]# setenforce 0
```

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

In this section:

- [Open the Viewpoint PostgreSQL Database Port \(5432\)](#)
- [Open the Apache Web Server Port \(8081\)](#)
- [Open the Viewpoint File Manager Port \(8443\)](#)

Open the Viewpoint PostgreSQL Database Port (5432)

Viewpoint requires access to the Viewpoint PostgreSQL database. Depending on your system configuration, your PostgreSQL databases may not be installed on the same host as their corresponding component servers. For example, you may choose to install the Viewpoint PostgreSQL database on the same host where you have installed the Insight PostgreSQL database instead of on the Viewpoint Server Host.

If you choose to install the Viewpoint PostgreSQL database on a *different* host from where you will install Viewpoint Server, do the following on the Viewpoint Database Host:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 5432 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=5432/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5432"
[root]# service SuSEfirewall2 restart
```

Open the Apache Web Server Port (8081)

On the Viewpoint Server Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 8081 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=8081/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="8081"

[root]# service SuSEfirewall2 restart
```

Open the Viewpoint File Manager Port (8443)

On the Moab Server Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 8443 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=8443/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="8443"

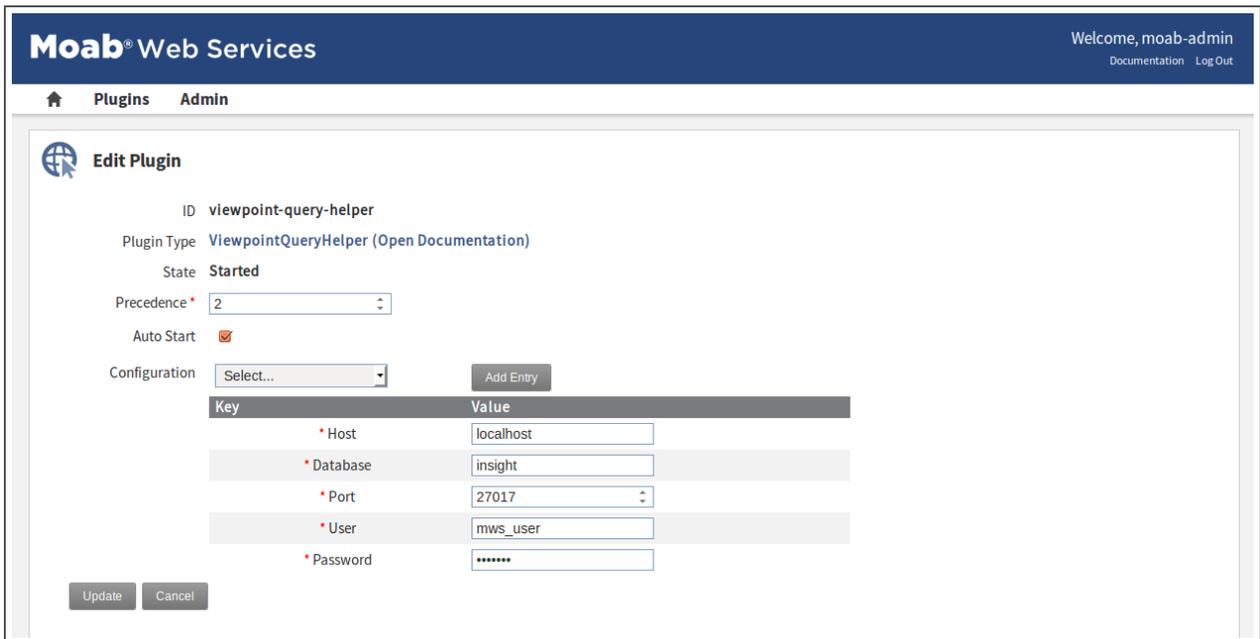
[root]# service SuSEfirewall2 restart
```

Configure the ViewpointQueryHelper Plugin

You will need to configure the MWS ViewpointQueryHelper plugin to allow Viewpoint to query the Insight MongoDB (MongoDB host, database, port, and user information).

Do the following:

1. Using a web browser, navigate to your MWS instance (`http://<server>:8080/mws/`) and then log in as the MWS administrative user (`moab-admin`, by default).
2. Select **Plugins** and then from the drop-down select **Plugins** to display the list of MWS plugins (displays Plugin List page).
3. Click the `viewpoint-query-helper` plugin to view this plugin's information (displays Show Plugin page).
4. Click **Edit** to modify the Configuration table fields (displays Edit Plugin page). The following is an example of the Edit Plugin page.



5. Modify the values as needed. The following table describes the required information.

| Key | Value Description |
|----------|--|
| host | Name or IP address of the host on which Insight MongoDB resides. |
| database | Name of the MongoDB database to which Insight writes. |

| Key | Value Description |
|----------|---|
| port | Port number for Insight MongoDB (typically 27017). |
| user | User name with which MWS connects to Insight MongoDB. |
| password | Password used by the user listed in the value for the "user" key. |

i This is the user name and password you specified when installing Insight. See the step ["Add the required MongoDB users to Insight MongoDB and Moab MongoDB; regardless of whether they share a host."](#) for the user and password information.

- When finished, click **Update** to save your changes and close this page (return to the Show Plugin page); otherwise click **Cancel** to reset all the changes.
- When satisfied with the values, on the Show Plugin page, confirm that the State is "Started". If it is not, go to Plugins, select Plugin Monitoring, and start the plugin using the green start button.
- Log out of your MWS instance and close the web browser.

See also [About Plugins](#) in the *Moab Web Services Reference Guide* for more information.

Install Viewpoint Server

i You *must* complete the prerequisite tasks earlier in this topic before installing the Viewpoint Server. See [Prerequisites on page 179](#).

Do the following:

- If you are installing Viewpoint on its own host *or* on a host that does not have another RPM installation, complete the steps to prepare the host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).

2. Set up PostgreSQL for Viewpoint.

i These instructions assume you will install the Viewpoint PostgreSQL database on the same host as the Insight PostgreSQL database (strongly recommended). Depending on your system confirmation, this may be the on the Insight Server Host or on the PostgreSQL Database Host.

If you choose to install the Viewpoint PostgreSQL database on a host that does not already have a PostgreSQL database, you will need to install the Viewpoint PostgreSQL database. See [Install PostgreSQL on page 172](#) for more information.

On the host containing the Insight PostgreSQL, do the following:

```
[root]# su - postgres
[postgres]$ psql
CREATE USER moab_viewpoint WITH PASSWORD 'changeme!';
CREATE DATABASE moab_viewpoint WITH OWNER=moab_viewpoint;
\q
[postgres]$ exit
```

3. On the Moab Server Host, install the moab-viewpoint-filemanager package.

a. If installing on a SUSE 12-based system, update zypper.

```
[root]# zypper update zypper
```

b. Install the package.

- Red Hat 6-based and Red Hat 7-based systems

```
[root]# yum install moab-viewpoint-filemanager
[root]# yum install python-setuptools
```

- SUSE 12-based systems

```
[root]# zypper install moab-viewpoint-filemanager
```

c. Using the instructions in `/opt/acfileman/utils/certs-handling/Readme.txt`, follow these steps:

Step 1. Create CA (Certificate Authority).

Step 2. Create server (WebDav server) certificate and key.

Step 3. Create client certificate and key.

Step 4. Configure WebDav server.

Step 5. Configure WebDav client (Viewpoint); skip for now.

For example:

```
[root]# cd /opt/acfileman/utils/certs-handling
[root]# ./ac-cert-tool.sh create-ca
[root]# ./ac-cert-tool.sh create-server-cert --altnames 127.0.0.1,localhost
<moab_host>
[root]# ./ac-cert-tool.sh create-client-cert
[root]# bash certs/servers/<moab_host>/install-server-certs.sh -u root:root -p
600 /opt/acfileman/etc/
[root]# vi /opt/acfileman/etc/uwsgi.ini
      [uwsgi]
      https = 0.0.0.0:8443,/opt/acfileman/etc/server-
cert.pem,/opt/acfileman/etc/server-
key.pem,HIGH:!DSS:!3DES:!SRP:!PSK:!DHE:!EDH:!ECDH:!aNULL,!/opt/acfileman/etc/ca-
cert.pem
```

d. Configure the moab-viewpoint-filemanager package to start up at system boot and start the moab-viewpoint-filemanager.

- Red Hat 6-based systems

```
[root]# chkconfig acfileman on
[root]# service acfileman restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl enable acfileman.service
[root]# systemctl restart acfileman.service
```

4. On the Moab Server Host, enable negative job priority and remote visualization features.

a. Set the ENABLENEGJOBPRIORITY parameter in /opt/moab/etc/moab.cfg.

```
[root]# vi /opt/moab/etc/moab.cfg
ENABLENEGJOBPRIORITY TRUE
```

i You must set this Moab parameter to support Viewpoint features that enable users to specify user priorities for their jobs. See [Advanced Settings](#) in the *Viewpoint Reference Guide* for more information on enabling user priorities for jobs.

b. If using the Remote Visualization features, set the USEMOABJOBID parameter in /opt/moab/etc/moab.cfg.

```
[root]# vi /opt/moab/etc/moab.cfg
USEMOABJOBID TRUE
```

c. Restart Moab.

- Red Hat 6-based systems

```
[root]# service moab restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl restart moab.service
```

5. On the Moab Server Host, register Viewpoint as a client in MWS.

- a. Edit the `grails.plugin.springsecurity.oauthProvider.clients` array in `/opt/mws/etc/mws-config.groovy` and specify a client id and a client secret. Leave the `authorizedGrantTypes` field unchanged.

i The following is a suggested script for generating the client secret:

```
dd if=/dev/urandom count=24 bs=1 2>/dev/null | base64
```

```
[root]# vi /opt/mws/etc/mws-config.groovy
grails.plugin.springsecurity.oauthProvider.clients = [
    [
        clientId: "viewpoint",
        clientSecret: "<ENTER-CLIENTSECRET-HERE>",
        authorizedGrantTypes: ["password"]
    ]
]
```

b. Restart Tomcat.

- Red Hat 6-based systems

```
[root]# service tomcat restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl restart tomcat.service
```

6. On the Viewpoint Server Host, do the following:

a. Install the `moab-viewpoint` package.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install moab-viewpoint
```

- SUSE 12-based systems

```
[root]# zypper install moab-viewpoint
```

b. (Optional) Configure virtual hosts. The `moab-viewpoint` package installs a file for Apache.

- Red Hat 6-based and Red Hat 7-based systems

```
/etc/httpd/conf.d/viewpoint.conf
```

- SUSE 12-based systems

```
/etc/apache2/conf.d/viewpoint.conf
```

Virtual host configurations should be made within this file. See <http://httpd.apache.org/docs/2.2/vhosts/> for more information.

- c. Edit the `/opt/viewpoint/etc/viewpoint.cfg` values as needed. The following is an example of the `viewpoint.cfg` file with the default values.

```
[admin]
username = viewpoint-admin
password = pbkdf2_
sha256$20000$ZHeToCJgrSUH$+xmzYdhpqZCJokxO9eGzyr2B6jrfCgLlBT+pBgMis4w=

[environment]
IRIS_DATABASE_NAME = moab_viewpoint
IRIS_DATABASE_PASSWORD = changeme!
IRIS_DATABASE_USER = moab_viewpoint
IRIS_DATABASE_HOST = localhost
IRIS_DATABASE_PORT = 5432

[settings]
past_hours = 24
future_hours = 4
```

Be aware of the following:

- **[admin]:** For security purposes, the admin password is encrypted. In the example, the default is the encrypted equivalent to "changeme!", which is the default for the Viewpoint instance. Change this default password to a different encrypted password.

To encrypt the password, do the following (substituting "changeme!" with your password):

```
[root]# echo -n 'changeme!' | /opt/viewpoint/bin/viewpoint makehash
Using default hasher
pbkdf2_sha256$20000$ZHeToCJgrSUH$+xmzYdhpqZCJokxO9eGzyr2B6jrfCgLlBT+pBgMis4w=
```

i The default hashing algorithm is `pbkdf2_sha256`. To show the other available algorithms, run `/opt/viewpoint/bin/viewpoint makehash --help`. `bcrypt_sha256` and `bcrypt` are *not* supported on Red Hat 7-based systems.

- **[environment]:** "changeme!", although unencrypted, is the default for the Viewpoint database password. If you do not change this password, your Viewpoint database will not be secure. For tips on choosing a good password, see <https://www.us-cert.gov/ncas/tips/ST04-002>.
- **[settings]:** These values are used to limit the threshold for the Resource Job Timeline. See [Resource Job Timeline Page](#) in the *Moab Viewpoint Reference Guide*.

d. Initialize Viewpoint's PostgreSQL database.

```
[root]# /opt/viewpoint/bin/viewpoint migrate
```

e. Start (or restart) the Apache service.

• Red Hat 6-based systems

```
[root]# chkconfig httpd on
[root]# service httpd restart
```

• Red Hat 7-based systems

```
[root]# systemctl enable httpd.service
[root]# systemctl restart httpd.service
```

• SUSE 12-based systems

```
[root]# systemctl enable apache2.service
[root]# systemctl restart apache2.service
```

Enable Access to the Viewpoint File Manager

This section finishes the SSL authentication steps you began when you installed `moab-viewpoint-filemanager` -- that is, Step 5 of `/opt/acfileman/utils/certs-handling/Readme.txt` that you skipped earlier.

Do the following:

1. On the Moab Server Host, do the following:

```
[root]# cd /opt/acfileman/utils/certs-handling/certs
[root]# scp ca/ca-cert.pem client/client-cert.pem client/client-key.pem
root@<viewpoint_host>:/opt/viewpoint/lib/viewpoint/webdav_client
```

2. On the Viewpoint Server Host, set the mode, owner, and group of the files you copied over.

• Red Hat 6-based systems

```
[root]# cd /opt/viewpoint/lib/viewpoint/webdav_client
[root]# chmod 600 ca-cert.pem client-key.pem client-cert.pem
[root]# chown apache:apache ca-cert.pem client-key.pem client-cert.pem
[root]# service httpd restart
```

• Red Hat 7-based systems

```
[root]# cd /opt/viewpoint/lib/viewpoint/webdav_client
[root]# chmod 600 ca-cert.pem client-key.pem client-cert.pem
[root]# chown apache:apache ca-cert.pem client-key.pem client-cert.pem
[root]# systemctl restart httpd.service
```

- SUSE 12-based systems

```
[root]# cd /opt/viewpoint/lib/viewpoint/webdav_client
[root]# chmod 600 ca-cert.pem client-key.pem client-cert.pem
[root]# chown wwwrun:www ca-cert.pem client-key.pem client-cert.pem
[root]# systemctl restart apache2.service
```

License and Configure Viewpoint

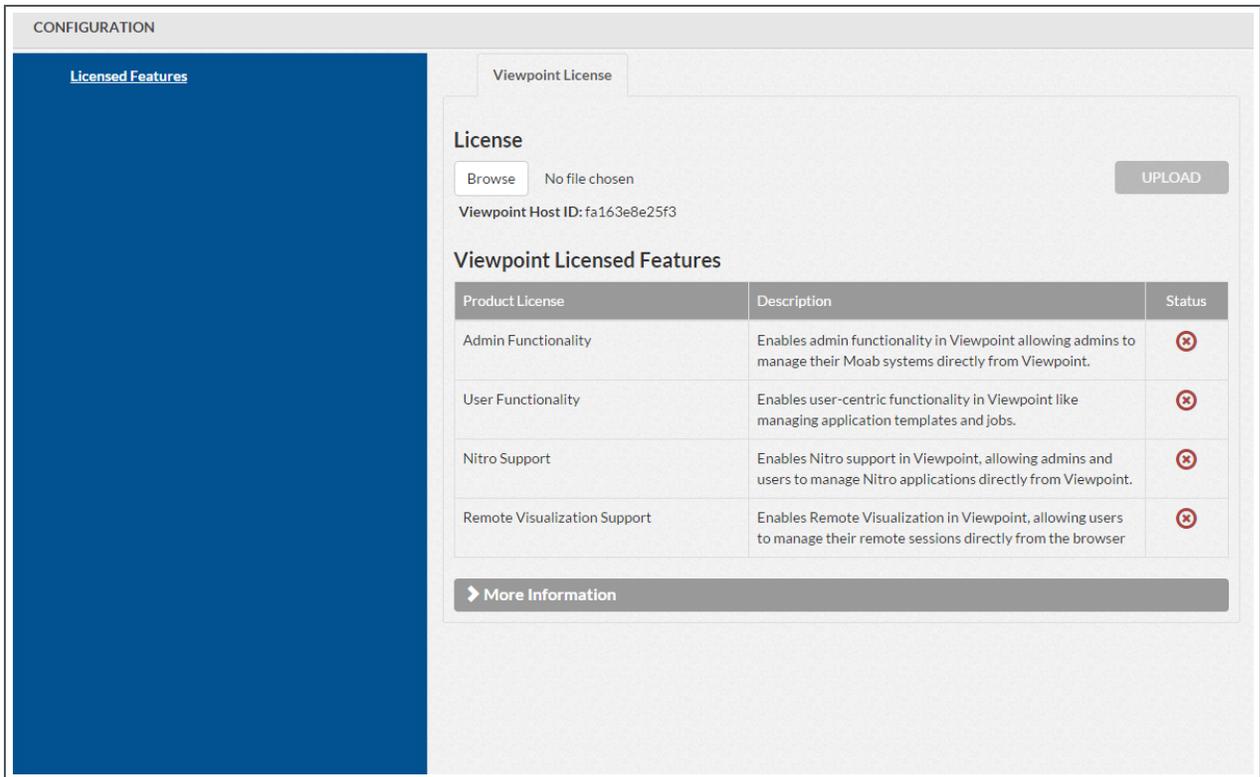
The next step in the Viewpoint installation is to obtain the Viewpoint license and provide that license and other configuration information using the Configuration page from the Viewpoint portal.

Do the following:

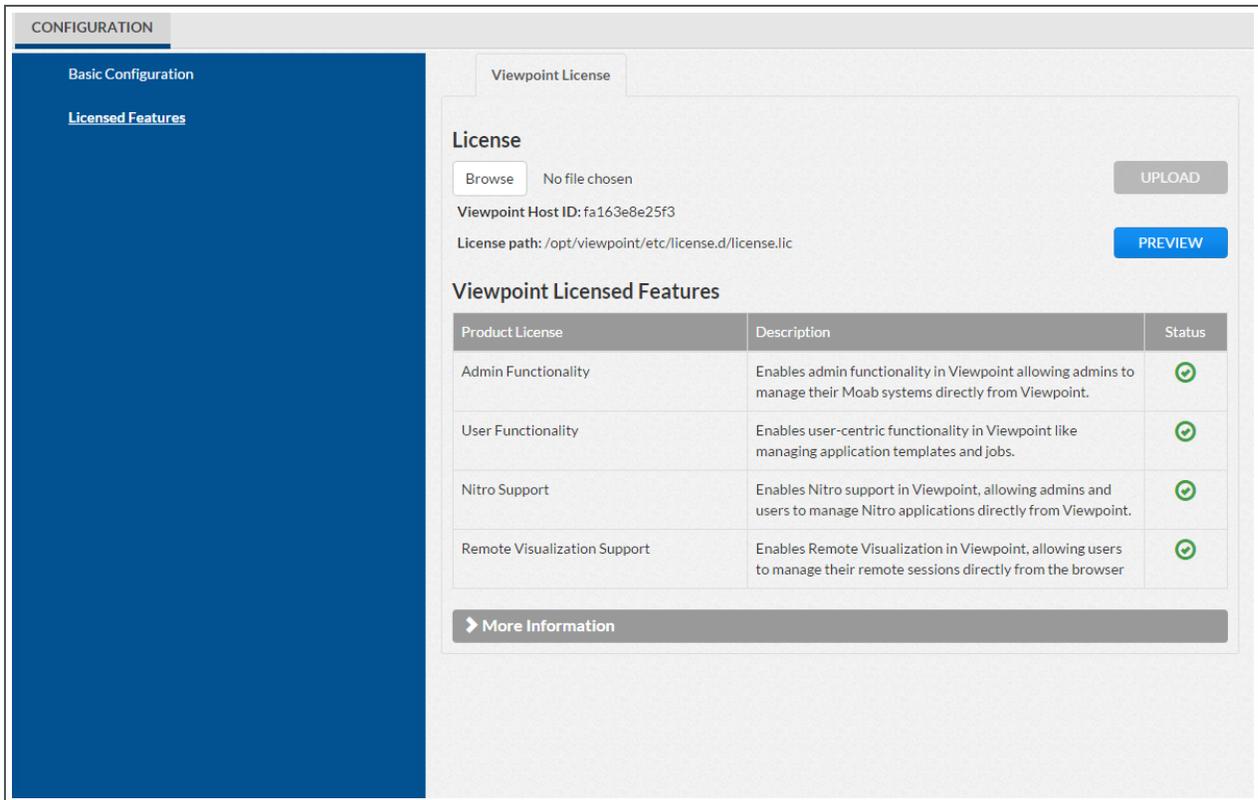
1. Using a web browser, navigate to your Viewpoint instance. (http://<viewpoint_host>:8081; where <viewpoint_host> is the IP address or name of the Viewpoint Server Host).
2. Log in as the Viewpoint administrative user (viewpoint-admin, by default) using the password you set in the Viewpoint installation instructions.
- 3.

i The Viewpoint administrative user has very limited rights: it has permission upload the license file and to set up the connection with MWS.

The Configuration page with *only* the Licensed Features/Viewpoint License information displayed. The following is an example of what you will see once you first log in.



4. In the License section, locate the Viewpoint Host ID.
5. Email licenses@adaptivecomputing.com with that hostid.
6. Adaptive Computing will generate the license and send you the Viewpoint license (.lic) file in a return email.
7. Save the Viewpoint license in a safe location.
8. Go back to your Viewpoint instance and log in again as the Viewpoint administrative user.
9. When the Configuration page with *only* the Licensed Features/Viewpoint License information displays, do the following:
 - a. Click **Browse**, navigate to where you saved the Viewpoint License file, and then click **Open**.
 - b. Click **Upload**.
10. Once the license file has uploaded, the Viewpoint License information shows green check boxes for your licensed features and displays the path to your uploaded license file under the Viewpoint Host ID information. The following is an example of what you will see once the license file is uploaded.



- Click **Preview** to view the contents of the license file you uploaded
- You can also expand the More Information section to see expiration information.
- In addition, the Basic Configuration page is now available for you to provide MWS, File Manager and other configuration information. If you have not already done so, select the Basic Configuration page.

The following image is an example of the Basic Configuration page.

The screenshot displays the 'CONFIGURATION' tab in the Viewpoint interface. On the left is a blue sidebar with navigation links: 'Basic Configuration' (selected), 'Roles', 'Principals', 'Remote Visualization Services', 'Nitro Services', and 'Licensed Features'. The main content area is divided into three sections:

- MWS Configuration:** Includes input fields for 'Server' (http://10.2.184.128:8080), 'Username' (moab-admin), 'Password' (masked with dots), 'Path' (/mws/), 'Client Id' (irisClientId), and 'Client Secret' (masked with dots). There is a 'Reset Permissions' checkbox and a 'TEST' button.
- File Manager Configuration:** Includes input fields for 'Accessible Roots' (/home:/tmp), 'Maximum Upload Size (bytes)' (-1), 'Server Protocol' (http), 'Server Verify SSL' (checkbox), 'SSL Certificate File', 'SSL Certificate Key', 'CA Bundle File', 'Server Host' (127.0.0.1), 'Server Port' (8001), and 'Server Root Path' (/).
- Misc Options:** Includes a 'Node Names to Ignore' field (DEFAULT,GLOBAL) and a 'Use Google Analytics to help improve this product' checkbox.

At the bottom right is a 'SAVE' button. At the bottom left is the 'Viewpoint Build Information' section:

```

Viewpoint Build Information
Version: 9.0
Revision: 755e52cd88ba79d5cc784be1316e8a2fffd8734b
Branch: remotes/origin/9.0.x
Build Date: 2016-02-18 01:16:17
    
```

11. In the MWS Configuration area, do the following:
 - a. In the Server field, enter the URL for MWS on the Moab Server Host. For example: http://server:8080

i If your configuration uses a secure connection between Viewpoint and MWS, the URL must contain "https" and the secure port.

- b. In the Username and Password fields, enter the MWS administrator credentials. You can find these credentials in `/opt/mws/etc/mws-config.groovy` on the Moab Server Host. Look for `auth.defaultUser.username` and `auth.defaultUser.password`.
- c. In the Path field, the default value (`/mws/`) is already filled in. Leave it as is unless you have installed MWS with a non-default path.
- d. In the Client Id and Client Secret fields, enter the values that you set during the Viewpoint installation. Refer back to the step ([On the Moab Server Host, register Viewpoint as a client in MWS.](#)) earlier in this topic.
- e. Click **TEST** to confirm the settings are correct. If successful,  appears outside the Server field.

12. In the File Manager Configuration area, modify the values as needed. The following table describes the required information.

| Field | Description |
|-----------------------------|---|
| Accessible Roots | <p>The root folders that users can access from the File Manager page. This can be used to limit users' access to certain directories, without giving them access to the "/" folder on the remote file system (RFS). Separate root folders with a colon (:).</p> <p>For example, if you define <code>/home</code> and <code>/user/share/groups</code> as accessible roots, although users will be able to see a tree similar to the following, the users will not be able to see (access) anything inside <code>/usr</code> other than "share" and anything inside "share" other than "groups".</p> <pre> - /home/ user1/ user2/ youruser/ - /usr/ - share/ groups/ </pre> |
| Maximum Upload Size (bytes) | Total amount of data that can be uploaded in a single file. A value of '-1' means unlimited. |
| Server Protocol | The connection protocol (either http or https). |

| Field | Description |
|----------------------|---|
| Server Verify SSL | When enabled: <ul style="list-style-type: none"> The client SSL certificate will be verified. Viewpoint will use the given certificate when connecting to File Manager Service. |
| SSL Certificate File | The location of the SSL certificate file on the Viewpoint Server. Usually, <code>/opt/viewpoint/lib/viewpoint/webdav_client/client-cert.pem</code> . |
| SSL Certificate Key | The location of the SSL certificate key on the Viewpoint Server. Usually, <code>/opt/viewpoint/lib/viewpoint/webdav_client/client-key.pem</code> . |
| CA Bundle File | The location of the CA bundle file on the Viewpoint Server. Usually, <code>/opt/viewpoint/lib/viewpoint/webdav_client/ca-cert.pem</code> . |
| Server Host | The name of the Moab Server host on which you installed the File Manager Service. |
| Server Port | The port number for the File Manager Service. |
| Server Root Path | The root URL path where File Manager Service publishes its API (usually it is simply <code>"/</code>). |

13. In the Misc Options area, do the following:
 - a. In the Node Names to Ignore field, enter the nodes that you want Viewpoint to ignore. Separate node names with a comma (,).
 - b. Choose whether you wish to use Google Analytics to help improve this product.
14. Click **SAVE** to submit the information. You will be logged out of Viewpoint. This ends the configuration as the Viewpoint administrator user. The MWS administrative user (moab-admin, by default) is now active and be used to log into Viewpoint.

i If you are unable to log in as the MWS administrative user, confirm the MWS connection information is correct and test the connection as described earlier in this procedure.

Grant Users Access to Viewpoint

For a user to be able to access Viewpoint, he or she must be a member of a principal.

Do the following:

1. Using a web browser, navigate to your Viewpoint instance. (http://<viewpoint_host>:8081; where <viewpoint_host> is the IP address or name of the Viewpoint Server Host)
2. Log in as the MWS administrative user (moab-admin, by default).
3. Click **Configuration** from the menu. The Basic Configuration page displays.
4. Click **Principals** from the left pane.
5. Create one or more principals. See [Creating or Editing Principals](#) in the *Moab Viewpoint Reference Guide* for instructions on setting up principals.

 Viewpoint comes configured with an admin and a user role that you can assign to the principals. You can also modify the default roles and create new roles as needed. See [About Roles](#) in the *Moab Viewpoint Reference Guide* for more information.

Related Topics

[Chapter 3 RPM installation Method on page 122](#)

Installing RLM Server

Access to a Reprise License Manager (RLM) server is required when using Remote Visualization and/or Nitro.

 The RLM Server can run multiple licenses. If your company already uses an RLM Server, you do not need to install a new one for Remote Visualization or Nitro. However, Remote Visualization and Nitro will use a different port than the default RLM Server port (5053). Skip this topic and follow the instructions in [Installing Remote Visualization on page 198](#) or [Installing Nitro on page 218](#) as applicable.

This topic contains instructions on how to install an RLM Server.

In this topic:

- [Open Necessary Ports on page 196](#)
- [Install the RLM Server on page 197](#)
- [Change the Default Passwords on page 197](#)

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

i These instructions assume you are using the default ports. If your configuration will use other ports, then substitute your port numbers when opening the ports.

On the RLM Server, do the following:

1. Open the RLM Server port (5053) and the RLM Web Interface port (5054).

- Red Hat 6-based systems using iptables

```
[root]# iptables -A INPUT -p tcp --dport 5053 -j ACCEPT
[root]# iptables -A INPUT -p tcp --dport 5054 -j ACCEPT
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=5053-5054/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5053 5054"
[root]# service SuSEfirewall2 restart
```

2. If Remote Visualization is part of your configuration, open the Remote Visualization port (57889).

- Red Hat 6-based systems using iptables

```
[root]# iptables -A INPUT -p tcp --dport 57889 -j ACCEPT
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=57889/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="57889"
[root]# service SuSEfirewall2 restart
```

3. If Nitro is part of your configuration, open the ISV adaptiveco port for the Adaptive license-enabled products (for example: 5135).

- Red Hat 6-based systems using iptables

```
[root]# iptables -A INPUT -p tcp --dport 5135 -j ACCEPT
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=5135/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="5135"
[root]# service SuSEfirewall2 restart
```

Install the RLM Server

i If your configuration uses firewalls, you *must also* open the necessary ports before installing Nitro. See [Open Necessary Ports on page 196](#).

On the host on where the RLM Server will reside, do the following:

1. If you are installing RLM Server on its own host *or* on a host that does not have another RPM installation, complete the steps to prepare the host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).
2. Install the RPM.
 - Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum install ac-rlm
```

- SUSE 12-based systems

```
[root]# zypper install ac-rlm
```

Change the Default Passwords

The RLM Web interface includes two usernames (admin and user) by default. These usernames have the default password "changeme!".

! If you do not change this password, RLM, and Remote Visualization, will not be secure. For tips on choosing a good password, see <https://www.us-cert.gov/ncas/tips/ST04-002>.

Do the following for *both* the user and the admin usernames:

1. Using a web browser, navigate to your RLM instance (http://<RLM_host>:5054; where <RLM_host> is the IP address or name of the RLM Server Host).

 If you have problems connecting using the web browser, on the RLM server check /opt/rlm/rlm.dll for error information.

2. Log in.
3. Select **Change Password** and change the password according to your password security process.

 The password for "user" will be needed as part of the Remote Visualization installation.

Installing Remote Visualization

This topic contains instructions on how to install Remote Visualization, including licensing and configuration information.

 Remote Visualization uses the FastX product. The Remote Visualization installation includes installing the Remote Visualization Server (gateway server) and Remote Visualization on the Torque MOM Hosts (session servers).

 Remote Visualization Server (gateway server) and the Remote Visualization Session Servers, must be configured in order for Remote Visualization to work.

In this topic:

- [Open Necessary Ports on page 199](#)
- [Obtain and Install the Remote Visualization License on page 200](#)
- [Configure the RLM Plugin on page 202](#)
- [Configure Moab to use Moab Web Services as a Resource Manager on page 204](#)
- [Install Remote Visualization on page 204](#)
- [Configure Remote Visualization - Gateway Server on page 206](#)
- [Configure Remote Visualization - Session Server on page 210](#)
- [Copy the Session Server Configuration to the Remaining Torque MOM Hosts on page 214](#)

- [\(Optional\) Install Graphical Packages on Each Torque MOM Host on page 215](#)
- [Configure Moab for Remote Visualization on page 215](#)
- [Configure Viewpoint for Remote Visualization on page 216](#)

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to Remote Visualization.

Do the following:

1. On the Remote Visualization Server (also known as the gateway server), open port 3443.

- Red Hat 6-based systems using iptables

```
[root]# iptables -A INPUT -p tcp --dport 3443 -j ACCEPT
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
firewall-cmd --add-port=3443/tcp --permanent
firewall-cmd --reload
```

- SUSE 12-based systems

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following ports to the FW_SERVICES_EXT_TCP parameter as required

# Needed on the Moab server for off-host client communication
FW_SERVICES_EXT_TCP="3443"

[root]# service SuSEfirewall2 restart
```

On each Remote Visualization Session Server (Torque MOM Host), do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables -A INPUT -p tcp --dport 3443 -j ACCEPT
[root]# iptables -A INPUT -p tcp --dport 6000:6005 -j ACCEPT
[root]# iptables -A INPUT -p tcp --dport 16001 -j ACCEPT #if using gnome
[root]# iptables -A INPUT -p tcp --dport 35091 -j ACCEPT #if using gnome
[root]# iptables -A INPUT -p udp -m udp --dport 117 -j ACCEPT

[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
firewall-cmd --add-port=3443/tcp --permanent
firewall-cmd --add-port=6000:6005/tcp --permanent
firewall-cmd --add-port=16001/tcp --permanent #if using gnome
firewall-cmd --add-port=35091/tcp --permanent #if using gnome
firewall-cmd --add-port=117/udp --permanent
firewall-cmd --reload
```

- SUSE 12-based systems

```
[root]# vi /etc/sysconfig/SuSEfirewall2

# Add the following ports to the FW_SERVICES_EXT_TCP parameter as required

# Needed on the Moab server for off-host client communication
FW_SERVICES_EXT_TCP="3443 6000:6005 16001 35091" #ports 16001 and 35091 are only
needed if using gnome

FW_SERVICES_EXT_UDP="117"

[root]# service SuSEfirewall2 restart
```

Obtain and Install the Remote Visualization License

Remote Visualization uses the RLM to validate the amount of open and available sessions.



These instructions assume you already have access to an RLM Server. See [Installing RLM Server on page 195](#) for instructions on how to set up a new RLM Server.

Do the following:

1. Email licenses@adaptivecomputing.com and request an activation key. Adaptive Computing will send you the activation key in a return email.
2. Once you have your activation key, do the following on the RLM Server:
 - a. Install the license activation script.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum -y install StarNetFastX2
```

- SUSE 12-based systems

```
[root]# zypper install StarNetFastX2
```

- b. Run the license activation script.

```
/usr/lib/fastx2/install/activate
```

c. When prompted:

- Enter the activation key.
- Enter how many seats (sessions) you want for this license.

When the license has generated you will see something similar to the following:

```
License activated and saved in /usr/lib/fastx2/rlm/FastX2-<date>.lic
```

d. Move the license file to the /opt/rlm directory.

```
mv /usr/lib/fastx2/rlm/FastX2-<date>.lic /opt/rlm
```

i This license file references the default RLM Server port (5053). If the RLM Server in your configuration uses a different port, you will need to modify the license file to reflect the actual port.

e. If you did *not* install an RLM Server using the file available from Adaptive Computing (for example, because your system configuration already uses one), do the following:

- If you have not already done so, download and unpack the Nitro - Tarball Bundle (`nitro-tarball-bundle-<version>-<OS>.tar.gz`).

i The `starnet.set` file is provided in the RLM tarball (`ac-rlm-<version>.tar.gz`). This tarball is packaged in the Nitro - Manual Tarball. You do not need to install Nitro to use Remote Visualization.

- Identify and unpack the RLM tarball (`ac-rlm-<version>.tar.gz`).
- Locate the 'adaptiveco.set' file and copy it into the same directory where the Nitro license resides (`/opt/rlm`).

f. Remove the license activation script.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum -y remove StarNetFastX2
```

- SUSE 12-based systems

```
[root]# zypper remove StarNetFastX2
```

g. Restart RLM.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# service rlm restart
```

- SUSE 12-based systems

```
[root]# systemctl restart rlm.service
```

Configure the RLM Plugin

Moab can schedule available remote visualization sessions by querying the RLM server for the number of active and total available sessions.

i In order for Moab to schedule remote visualization sessions, Moab also needs to be configured to use Moab Web Services as a resource manager. See [Configuring Moab Workload Manager](#) in the *Moab Web Services Reference Guide* for more information.

Do the following:

1. Using a web browser, navigate to your MWS instance (`http://<server>:8080/mws/`) and then log in as the MWS administrative user (moab-admin, by default).
2. Select **Plugins** and then from the drop-down select **Plugins** to display the list of MWS plugins (displays Plugin List page).
3. Click **Add Plugin** (displays Create Plugin page).
4. Select **RLM** from the Plugin Type drop-down.
5. Click **Continue** (displays the already built information for this plugin on the Create Plugin page).
6. In the Configuration field, select **Resource** from the drop-down and then click **Add Entry** (adds the Resource key to the table). The following is an example of what your Create Plugin page should look like.

Create Plugin

Plugin Type **RLM (Open Documentation)**

ID *

Precedence

Poll Interval *

Auto Start

Configuration

| Key | Value |
|------------|---|
| * URL | <input type="text" value="http://server:5054"/> |
| * Username | <input type="text" value="user"/> |
| * Password | <input type="text" value="....."/> |
| * ISV | <input type="text" value="starnet"/> |
| * Product | <input type="text" value="fastx2"/> |
| Resource | <input type="text" value="remote_visualization"/> <input type="button" value="Remove"/> |

7. Enter the key values. The following table describes the required information.

| Key | Value Description |
|----------|---|
| URL | Hostname (or IP address) <i>and</i> port number for the RLM Server. |
| Username | The username in the RLM Web interface; typically user. |
| Password | Password used by the user listed in the Username key. This is the password you set when you install the RLM. See Change the Default Passwords . |
| ISV | Independent software vender for Remote Visualization. This value must be starnet. |
| Product | Name of the licensed product for Remote Visualization. This value must be fastx2. |
| Resource | Name of the resource to report to Moab Workload Manager. This value must be remote_visualization. |

- When finished, click **Save** to save your changes and close this page; otherwise click **Cancel** to reset all the changes.

i The state should be "Started". If the state says "Errored", click Edit, modify the values as needed, click Update. Then from the Plugin Monitoring page, locate the RLM plugin and click the play icon.

- Log out of your MWS instance and close the web browser.

Configure Moab to use Moab Web Services as a Resource Manager

In order for Moab to schedule remote visualization sessions, Moab also needs to be configured to use Moab Web Services as a resource manager.

On the Moab Server Host, do the following:

- Add the following lines to `/opt/moab/etc/moab.cfg`:

```
RMCFG[mws]                TYPE=MWS
RMCFG[mws]                BASEURL=http://localhost:8080/mws
```

The BASEURL must match the configured URL of MWS.

- Add the following line to `/opt/moab/etc/moab-private.cfg`:

```
CLIENTCFG[RM:mws] USERNAME=moab-admin PASSWORD=changeme!
```

i **USERNAME** and **PASSWORD** must match the values of `auth.defaultUser.username` and `auth.defaultUser.password`, respectively, found in the MWS configuration file. The MWS RM contacts MWS directly using the base URL, username, and password configured.

- Restart Moab.

- Red Hat 6-based systems

```
[root]# chkconfig moab on
[root]# service moab restart
```

- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl restart moab.service
```

Install Remote Visualization

Remote Visualization needs to be installed on the gateway server and on *all* the session servers (Torque MOM Hosts).

i You *must* complete all the tasks earlier in this topic before installing Remote Visualization.

Do the following:

1. Prepare the hosts for RPM installation. If you will be installing Remote Visualization on a host that does not have another RPM installation, complete the steps to prepare the host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).
2. On the Remote Visualization Gateway Server Host and each Session Server Host, do the following:

- a. Install FastX and all its dependencies.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# yum -y install ImageMagick-perl perl-Crypt-SSLeay perl-X11-Protocol
StarNetFastX2
```

- SUSE 12-based systems

```
[root]# zypper --non-interactive install perl-PerlMagick perl-Crypt-SSLeay
perl-X11-Protocol StarNetFastX2
```

- b. Create or use an unprivileged account to login into fastx with admin privileges.

i The following example uses the ace user and password. You can use an existing user, as long as that user can ssh into this host with a username/password pair.

```
[root]# useradd ace
...
[root]# passwd ace
...
```

- c. If installing on a SUSE 12-based system, add the node.js repository and refresh zypper.

```
[root]# zypper addrepo
http://download.opensuse.org/repositories/devel:languages:nodejs/SLE_
12/devel:languages:nodejs.repo
[root]# zypper refresh
```

- d. Run the install.sh script on the Remote Visualization Gateway Server and on all of the Session Servers (Torque MOMs).

```
[root]# export LICENSE_SERVER_HOST="<RLM Server IP address>"
[root]# export ADMIN_USER="ace" # the Remote Visualization administrator user
[root]# printf "y\n${LICENSE_SERVER_HOST}\ny\ny\n${ADMIN_USER}\ny\n" |
/usr/lib/fastx2/install.sh
```

3. On each Session Server, do the following:
 - a. Set the following parameters in `/etc/ssh/sshd_config`:

```
PasswordAuthentication yes
ChallengeResponseAuthentication no
```

- b. Restart the sshd service.
 - Red Hat 6-based systems
 - Red Hat 7-based or SUSE 12-based systems

```
[root]# service sshd restart
```

```
[root]# systemctl restart sshd.service
```

Configure Remote Visualization - Gateway Server

Do the following:

1. Using a web browser, navigate to your *secure* Remote Visualization instance. (**https://<gateway_host>:3443**; where *<gateway_host>* is the IP

- address or name of the Remote Visualization Server Host).
- 2. Log in as the fastX admin user you created earlier in this topic.

Adaptive
COMPUTING

User Name
ace

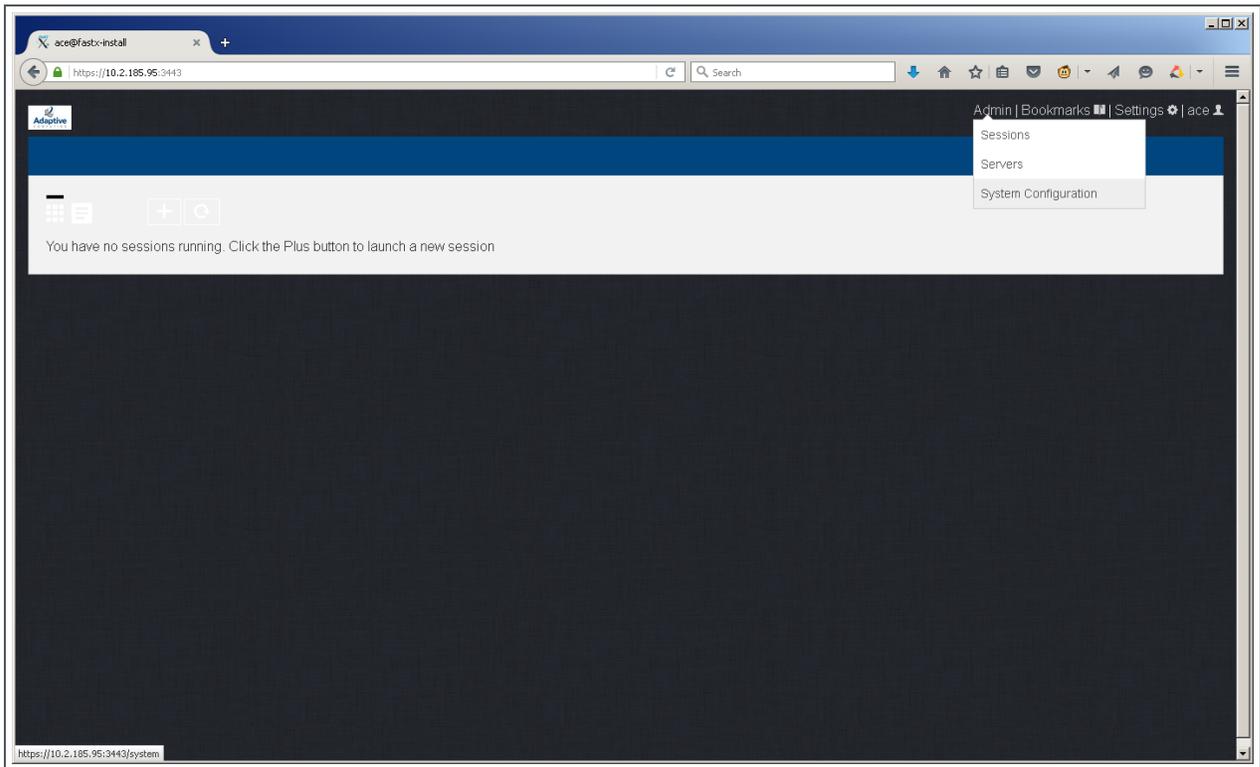
Password
••••

Use Public Key Authentication

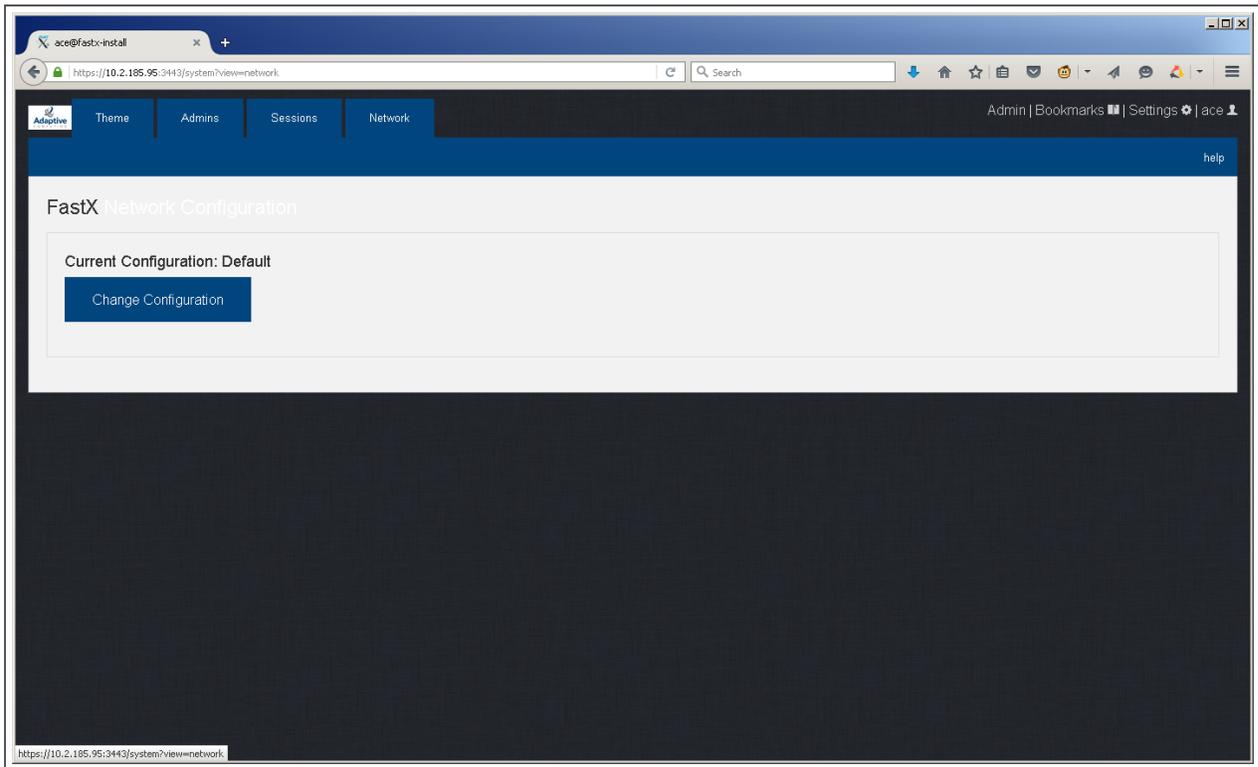
Log In

Version: 2.1.55

3. Select **Admin\System Configuration**.



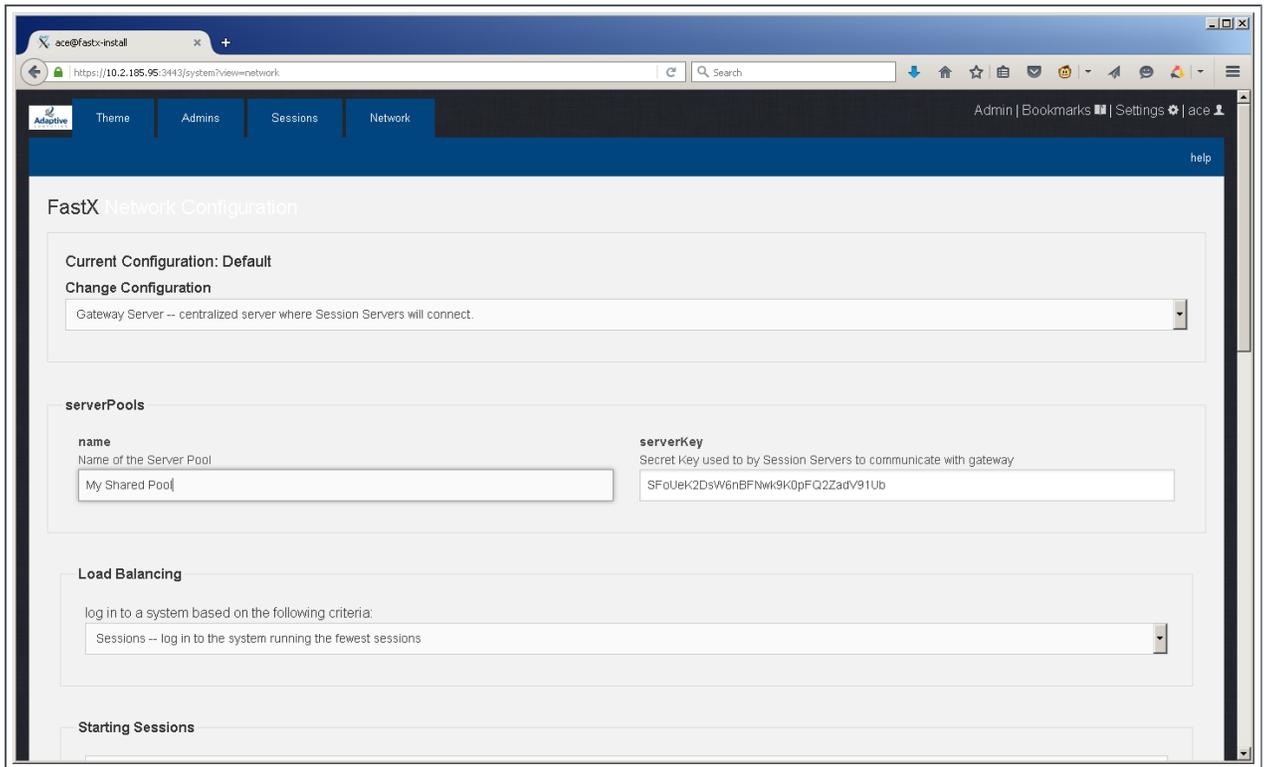
4. Select the Network tab and click **Change Configuration**.



5. Do the following:

- a. From the Change Configuration drop-down, pick the option for "Gateway Server -- centralized server where Session Servers will connect".
- b. In the Server Pools section, enter a server pool name (for example, My Shared Pool).
- c. Copy the generated server key to your clipboard or other location where you can access it when you are configuring the Remote Visualization

sessions on the Torque MOM Hosts.



d. Click **Save**.

6. Start the FastX service.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# service fastx restart
Starting fastx: stx: [ OK ]
```

- SUSE 12-based systems

```
[root]# systemctl restart fastx.service
```

Configure Remote Visualization - Session Server

Do the following:

1. Using a web browser, navigate to your *secure* Remote Visualization instance. (**https://<session-host>:3443**; where <session_host> is the IP

- address or name of the Remote Visualization Host).
- 2. Log in as the fastX admin user you created earlier in this topic.

Adaptive
COMPUTING

User Name
ace

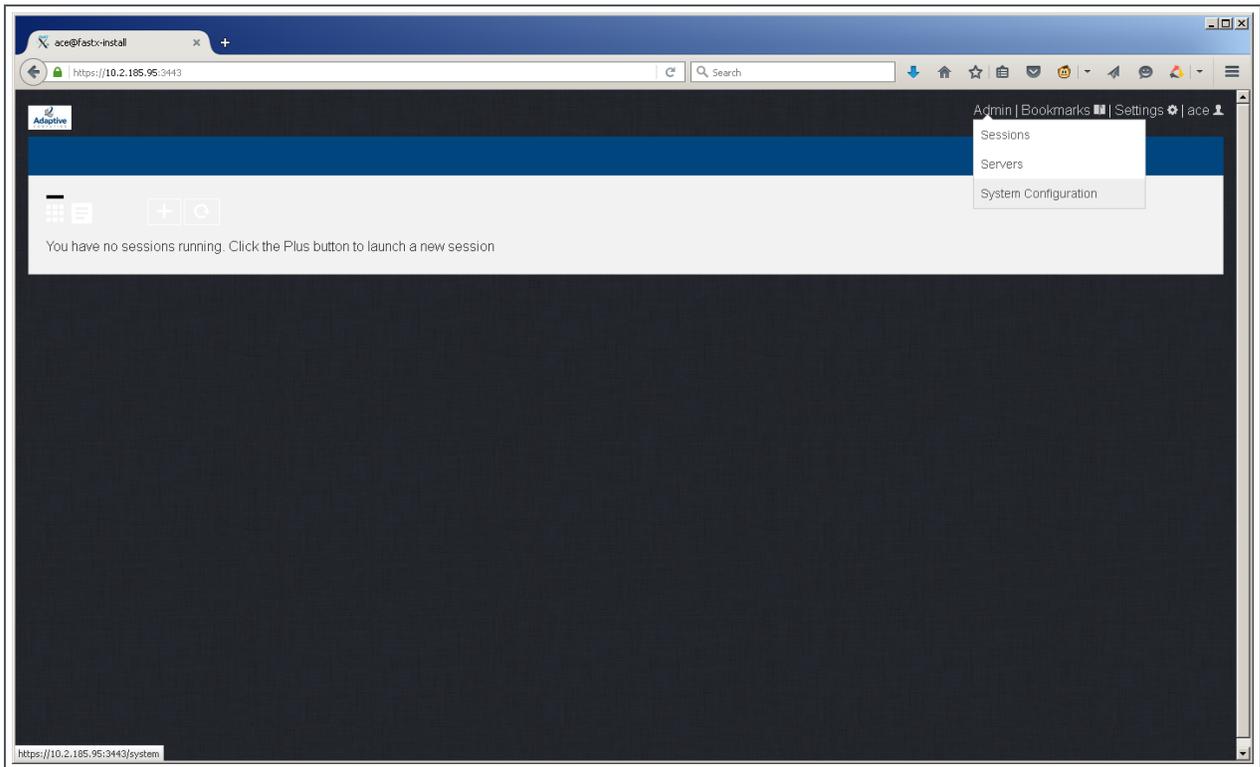
Password
•••

Use Public Key Authentication

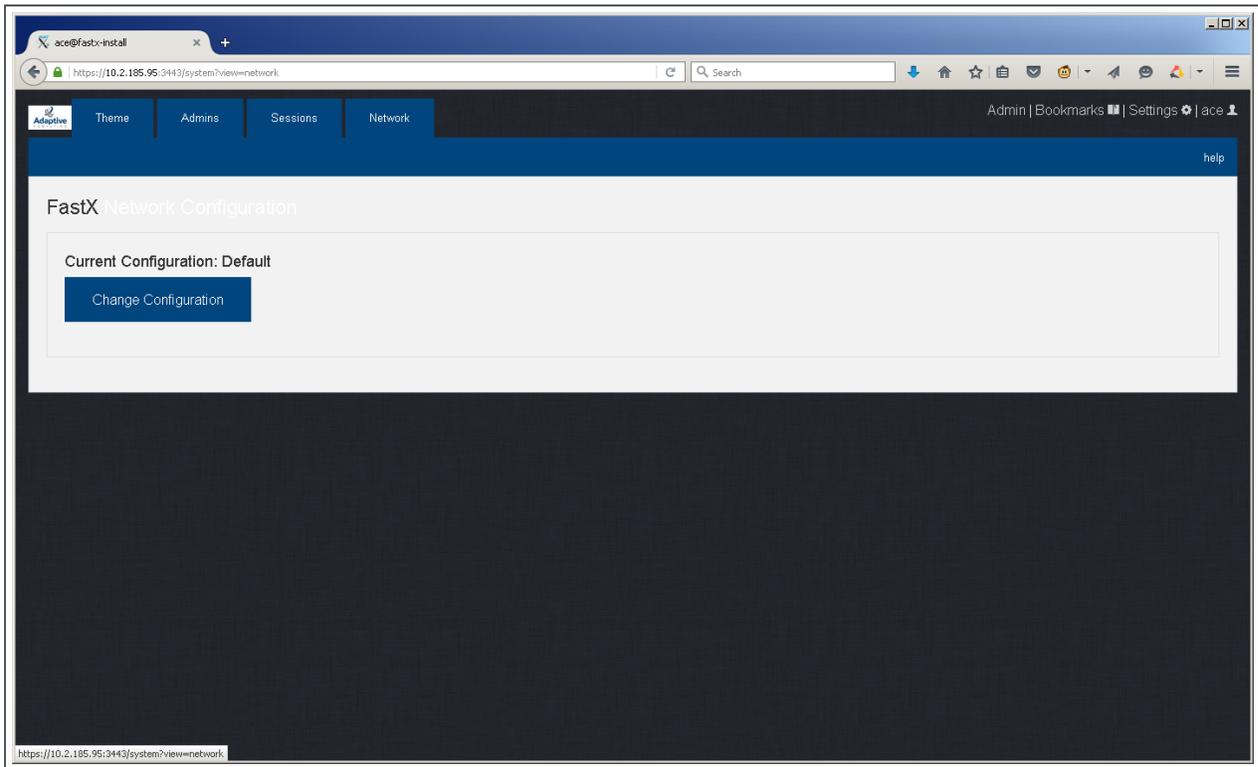
Log In

Version: 2.1.55

3. Select **Admin\System Configuration**.



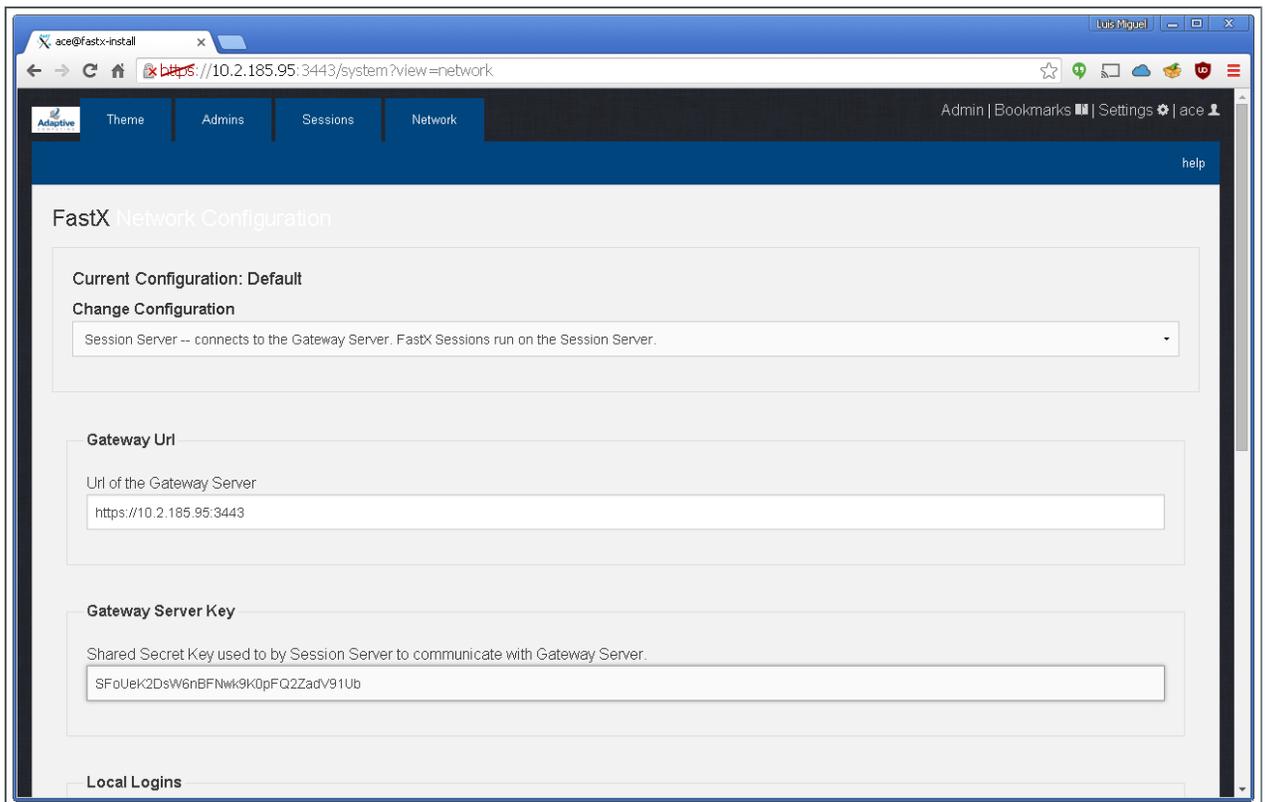
4. Select the Network tab and click **Change Configuration**.



5. Do the following:

- a. From the Change Configuration drop-down, pick the option for "Session Server -- connects to the Gateway Server. FastX Sessions run on the Session Server will connect".
- b. In the Url of the Gateway Server field, enter the URL for the gateway server you previously created (for example `https://<gateway_host>:3443`).
- c. Type or enter the server key that was generated for your gateway server

earlier in this topic.



d. Click **Save**.

6. Start the FastX service.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# service fastx restart
Starting fastx: stx: [ OK ]
```

- SUSE 12-based systems

```
[root]# systemctl restart fastx.service
```

Copy the Session Server Configuration to the Remaining Torque MOM Hosts

After you configured the initial Session Server, the settings are saved in a json file. You can use this json file to easily configure the additional Session Servers in your configuration.

On *each additional* Session Server (Torque MOM Host), do the following:

1. Copy the json file from the first Session Server (using the previous procedure) onto the additional Session Server.

```
scp root@<initial session server ip_address>:/usr/lib/fastx2/config/session-
server.json /usr/lib/fastx2/config
chown fastx. /usr/lib/fastx2/config/session-server.json
```

2. Restart the FastX service.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# service fastx restart
Starting fastx: stx: [ OK ]
```

- SUSE 12-based systems

```
[root]# systemctl restart fastx.service
Starting fastx: stx: [ OK ]
```

(Optional) Install Graphical Packages on Each Torque MOM Host

A few graphical packages are available to let you easily submit remote visualization jobs from Viewpoint (install a desktop environment).

One each Torque MOM Host, run the following command:

- Red Hat 6-based systems

```
[root]# yum -y groupinstall "Desktop" "Desktop Platform" "X Window System"
"Fonts"
[root]# yum -y install xterm
```

- Red Hat 7-based systems

```
[root]# yum -y groupinstall "GNOME Desktop" "X Window System" "Fonts"
[root]# yum -y install xterm
```

- SUSE 12-based systems

```
[root]# zypper install gnome-desktop
[root]# zypper install -t pattern gnome-basic x11
```

Configure Moab for Remote Visualization

On the Moab Server Host, verify the `/opt/moab/etc/moab.cfg` file contains the following uncommented parameter:

```
JOBCFG[remote_visualization] FLAGS=usemoabjobid SELECT=TRUE
```

i This parameter configuration specifies that Moab will reference remote visualization jobs by their internal Moab job id. However, the job's output and error files will still be generated by your resource manager (for example, Torque). This means that, even though your job will get assigned a Moab job id, your job's output and error file names will reference the resource manager's job id (for example, job.oX).

If you need the job's output files to match the same job id as your Moab job, append the following parameters to your moab.cfg:

```
RMCFG[torque] SYNCJOBID=TRUE  FLAGS=ProxyJobSubmission
RMCFG[internal] JOBIDFORMAT=integer
```

Be advised that these appended parameters are *not* recommended for all systems; especially if your configuration includes customizations. If your system is not working as expected, contact Adaptive Computing support for assistance.

If you have made changes to the moab.cfg file, make sure you restart Moab.

- Red Hat 6-based systems

```
[root]# service moab restart
```

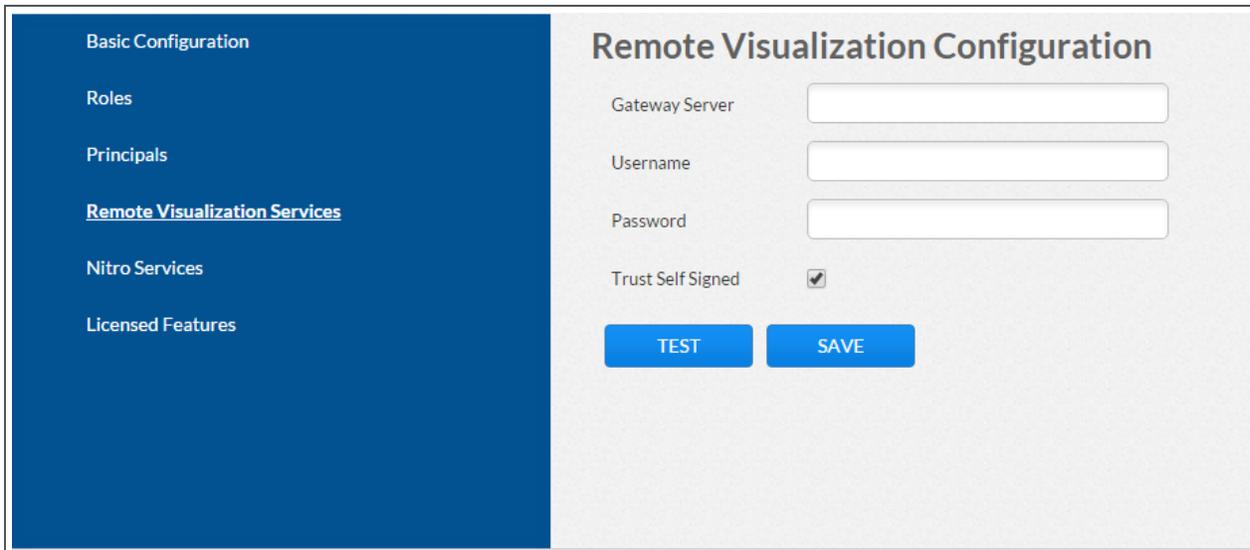
- Red Hat 7-based or SUSE 12-based systems

```
[root]# systemctl restart moab.service
```

Configure Viewpoint for Remote Visualization

Do the following:

1. Using a web browser, navigate to your Viewpoint instance (`http://<server>:8081`) and then log in as the MWS administrative user (moab-admin, by default).
2. Click **Configuration** from the menu and then click **Remote Visualization Services** from the left pane. The following is an example of the Remote Visualization Configuration page.



3. Enter the configuration information. The following table describes the required information.

| Field | Description |
|-------------------|--|
| Gateway Server | Hostname (or IP address) and port number for the FastX gateway server. For example, <code>https://<server>:3443</code> . |
| Username | Name of the FastX admin user you specified when you installed the Remote Visualization Server. For example, <code>ace</code> . |
| Password | The FastX admin user's password. |
| Trust Self Signed | Indicates whether the Remote Visualization was set up using self-signed certificates. |

4. Click **TEST** to confirm the settings are correct.
5. Click **SAVE**.

Nitro Integration

This section provides instructions on integrating Nitro as part of your Moab HPC Suite configuration.

In this section:

- [Installing Nitro on page 218](#)
- [Installing Nitro Web Services on page 222](#)

Installing Nitro

This topic contains instructions on how to install Nitro, including licensing and configuration information.

Nitro

- needs to be available to all of the nodes that will be used as part of the Nitro job.
- can be installed either to each node individually *or* to a shared file system that each node can access.
- can be installed to integrate with a scheduler, such as Moab, or without (Nitro standalone). The instructions are the same.

In this topic:

- [Obtain a Nitro License on page 218](#)
- [Open Necessary Ports on page 220](#)
- [Install Nitro on page 220](#)
- [Verify Network Communication on page 222](#)

Obtain a Nitro License

The Nitro license file is installed on an RLM Server.



These instructions assume you already have access to an RLM Server. See [Installing RLM Server on page 195](#) for instructions on how to set up a new RLM Server.

Do the following:

1. On the RLM server, obtain the hostid and hostname.

- hostid

```
[root]# /opt/rlm/rlmhostid
```

You should see output similar to the following.

```
rlmhostid v12.0
Copyright (C) 2006-2015, Reprise Software, Inc. All rights reserved.

Hostid of this machine: 00259096f004
```

- hostname

```
[root]# /opt/rlm/rlmhostid host
```

You should see output similar to the following.

```

rlmhostid v12.0
Copyright (C) 2006-2015, Reprise Software, Inc. All rights reserved.

Hostid of this machine: host=<your-host-name>

```

2. Email licenses@adaptivecomputing.com for a license and include the hostid and hostname you just obtained.
3. Adaptive Computing will generate the license and send you the Nitro license file (.lic) file in a return email.
4. On the RLM server, do the following:
 - a. Download and install the license file.

```

[root]# cd /opt/rlm
[root]# chown rlm:rlm <licenseFileName>.lic

```

- b. If the RLM Server in your configuration uses a firewall, edit the license file to reference the ISV adaptiveco port for the Adaptive license-enabled products. This is the same port number you opened during the RLM Server installation. See the instructions to open necessary ports in the [Installing RLM Server on page 68](#) (manual installation method) or [Installing RLM Server on page 195](#) (RPM installation method) for more information.

```

[root]# vi /opt/rlm/nitro.lic

ISV adaptiveco port=5135

```

The license file already references the RLM Server port (5053 by default).

i If the RLM Server in your configuration uses different ports, you will need to modify the license file to reflect the actual ports. See the instructions to open necessary ports in the [Installing RLM Server on page 68](#) (manual installation method) or [Installing RLM Server on page 195](#) (RPM installation method) for more information.

- c. If you did *not* install an RLM Server using the file available from Adaptive Computing (for example, because your system configuration already uses one), do the following:
 - i. If you have not already done so, download and unpack the Nitro - Tarball Bundle (nitro-tarball-bundle-<version>-<OS>.tar.gz).
 - ii. Identify and unpack the RLM tarball (ac-rlm-<version>.tar.gz).
 - iii. Locate the 'adaptiveco.set' file and copy it into the same directory where the Nitro license resides (/opt/rlm).

- d. Perform a reread to update the RLM Server with you license.

```
[root]# /opt/rlm/rlmreread
```

Open Necessary Ports

Nitro uses several ports for communication between the workers and the coordinator.

The default port is 47000, and up to four ports are used in running Nitro (ports 47000-47003).

On each compute node (coordinator), open the necessary ports.

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"
-A INPUT -p tcp --dport 47000 -j ACCEPT
-A INPUT -p tcp --dport 47001 -j ACCEPT
-A INPUT -p tcp --dport 47002 -j ACCEPT
-A INPUT -p tcp --dport 47003 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=47000-47003/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="47000 47001 47002 47003"

[root]# service SuSEfirewall2 restart
```

Install Nitro

i You *must* complete the tasks to obtain a Nitro license before installing Nitro. See [Obtain a Nitro License on page 218](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing Nitro. See [Open Necessary Ports on page 220](#).

On the host on where Nitro will reside, do the following:

1. If you are installing Nitro on its own host *or* on a host that does not have another RPM installation, complete the steps to prepare the host. See _

[Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).

2. Install the RPM.

- Red Hat 6-based or Red Hat 7-based systems

```
[root] # yum install nitro
```

- SUSE 12-based systems

```
[root] # zypper install nitro
```

3. Copy the license file you generated earlier in this topic to each compute node (coordinator). On each compute node, *or* on the shared file system, do the following:

```
[root]# cp <licenseFileName>.lic /opt/nitro/bin/
```

4. Identify the `launch.nitro.sh` script version for your resource manager. This script will be copied to the bin directory from where user job scripts will execute Nitro. See the *Nitro Administrator Guide* for more information.

Reference scripts are provided in `/opt/nitro/scripts`.

```
[root]# find /opt/nitro -name launch_nitro.sh
./scripts/lsf/launch_nitro.sh
./scripts/torque/launch_nitro.sh
./scripts/slurm/launch_nitro.sh
./scripts/alps/torque/launch_nitro.sh
./scripts/alps/slurm/launch_nitro.sh
```

5. Copy the launch script to the bin directory. (This example uses the Torque-based launch script.)

```
[root]# cp /opt/nitro/scripts/torque/launch_nitro.sh /opt/nitro/bin/
```

i This is a "copy" file operation and not a "move" operation. This allows you to customize your version of the script and always have the factory version available for consultation and/or comparison.

6. Customize the `bin/launch.nitro.sh` script as needed for your site's administrative policies. For example, to enable the Nitro coordinator's host to always execute a local Nitro worker, modify the `bin/launch.nitro.sh` script version to always pass the `--run-local-worker` command line option to the coordinator. See the *Nitro Administrator Guide* for more information on editing the launch script.

7. If you are *not* using a shared file system, copy the Nitro installation directory to *all* hosts.

```
[root]# scp -r /opt/nitro root@host002:/opt
```

Verify Network Communication

Verify that the nodes that will be running Nitro are able to communicate with the Nitro ports *and* that the nodes are able to communicate with one another.

Related Topics

- [Nitro Integration on page 217](#)

Installing Nitro Web Services

This topic contains instructions on how to install Nitro Web Services.

Do the following in the order presented:

1. [Open Necessary Ports](#)
2. [Install MongoDB](#)
3. [Install and Configure Nitro Web Services](#)
4. [Configure Viewpoint for Nitro Web Services](#)
5. [Configure Nitro Compute Nodes](#)

Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

In this section:

- [Open the Tornado Web Port \(9443\) and the ZMQ Port \(47100\) on page 222](#)
- [Open the MongoDB Database Port \(27017\) on page 223](#)

Open the Tornado Web Port (9443) and the ZMQ Port (47100)

On the Nitro Web Services Host, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 9443 -j ACCEPT
-A INPUT -p tcp --dport 47100 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=9443/tcp --permanent
[root]# firewall-cmd --add-port=47100/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="9443 47100"

[root]# service SuSEfirewall2 restart
```

Open the MongoDB Database Port (27017)

i Nitro Web Services requires access to a MongoDB database. Depending on your system configuration, your MongoDB databases may not be installed on the same host as their corresponding component servers. For example, you may choose to install the Nitro Web Services MongoDB on the same host where you have installed other MongoDB databases.

Do the following, as needed:

- If you have chosen to install the Nitro Web Services MongoDB database on the *same* host you installed other MongoDB databases, confirm the firewall port (27017) is already opened on that host.
- If you have chosen to install the Nitro Web Services MongoDB database on a *different* host from other MongoDB databases, you will need to open the Nitro Web Services MongoDB database port in the firewall for that host. To open the port in the firewall, do the following:

- Red Hat 6-based systems using iptables

```
[root]# iptables-save > /tmp/iptables.mod
[root]# vi /tmp/iptables.mod

# Add the following lines immediately *before* the line matching
# "-A INPUT -j REJECT --reject-with icmp-host-prohibited"

-A INPUT -p tcp --dport 27017 -j ACCEPT

[root]# iptables-restore < /tmp/iptables.mod
[root]# service iptables save
```

- Red Hat 7-based systems using firewalld

```
[root]# firewall-cmd --add-port=27017/tcp --permanent
[root]# firewall-cmd --reload
```

- SUSE 12-based systems using SuSEfirewall2

```
[root]# vi /etc/sysconfig/SuSEfirewall2

FW_SERVICES_EXT_TCP="27017"

[root]# service SuSEfirewall2 restart
```

Install MongoDB

To install and enable MongoDB on the Nitro Web Services Host, do the following:

1. Install MongoDB.

- Red Hat 6-based or Red Hat 7-based systems

```
[root]# cat > /etc/yum.repos.d/mongodb.repo <<End-of-file
[mongodb]
name=MongoDB Repository
baseurl=http://downloads-distro.mongodb.org/repo/redhat/os/x86_64
gpgcheck=0
enabled=1
exclude=mongodb-org mongodb-org-server
End-of-file
[root]# yum install mongo-10gen-server
```

- SUSE 12-based systems

```
[root]# zypper ar --refresh -r
http://download.opensuse.org/repositories/server:/database/SLE_
12/server:database.repo
[root]# zypper install mongodb
```

2. Start MongoDB

i There may be a short delay (approximately 3 minutes) for Mongo to start the first time.

- Red Hat 6-based systems

```
[root]# chkconfig mongod on
[root]# service mongod start
```

- Red Hat 7-based systems

```
[root]# cat > /usr/lib/systemd/system/mongodb.service <<End-of-file
[Unit]
Description=High-performance, schema-free document-oriented database
After=syslog.target network.target

[Service]
Type=forking
User=mongod
Group=mongod
Environment=CONFIG=/etc/mongod.conf
Environment=OPTIONS=
EnvironmentFile=-/etc/sysconfig/mongod
ExecStart=/usr/bin/mongod -f %CONFIG %OPTIONS
PrivateTmp=true
LimitNOFILE=65536
TimeoutStartSec=180
StandardOutput=syslog
StandardError=syslog

[Install]
WantedBy=multi-user.target
End-of-file
[root]# rm -f /etc/init.d/mongod
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
[root]# systemctl daemon-reload
```

- SUSE 12-based systems

```
[root]# systemctl enable mongodb.service
[root]# systemctl start mongodb.service
[root]# systemctl daemon-reload
```

3. Prepare the MongoDB database by doing the following:

a. Add the required MongoDB users.

i The password used below (secret1) is an example. Choose your own password for this user.

```
[root]# mongo
> use admin;
> db.addUser("admin_user", "secret1");
> db.auth("admin_user", "secret1");
> exit
```

i Because the `admin_user` has read and write rights to the `admin` database, it also has read and write rights to all other databases. See [Control Access to MongoDB Instances with Authentication](http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication/) (<http://docs.mongodb.org/manual/tutorial/control-access-to-mongodb-with-authentication/>) for more information.

b. Enable authentication in MongoDB.

- Red Hat 6-based systems

```
[root]# vi /etc/mongod.conf
auth = true
[root]# service mongod restart
```

- Red Hat 7-based systems

```
[root]# vi /etc/mongod.conf
auth = true
[root]# systemctl restart mongod.service
```

- SUSE 12-based systems

MongoDB authentication is already enabled. No further action is needed.

Install and Configure Nitro Web Services

i You *must* complete the tasks earlier in this topic before installing Nitro Web Services.

On the host where Nitro Web Services will reside, do the following:

1. If you are installing Nitro Web Services on its own host *or* on a host that does not have another RPM installation, complete the steps to prepare the host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).

2. Install the Nitro Web Services RPM.

- Red Hat 6-based or Red Hat 7-based systems

```
[root] yum install -y nitro-web-services
```

- SUSE 12-based systems

```
[root] zypper --non-interactive install nitro-web-services
```

3. Understand and edit the configuration files, including clarify what each configuration file is for and what to expect the first time the NWS service is started vs. each subsequent start.

MongoDB user, table, and index creation is performed at initial startup. Many of the options defined in the Nitro Web Service configuration files influence Mongo user/password and index creation.



Username and passwords are created *only* if they do not yet exist. Changing a password in the configuration file after initial startup will not update the password in Mongo.

The installation provides two configuration files.

- /opt/nitro-web-services/etc/nitro.cfg

This is the Nitro Web Services web application configuration file.

Before initial startup, set "admin_username" and "admin_password" to the MongoDB admin username and password you used when setting up MongoDB. It is also recommended that you change all other default passwords before starting Nitro Web Services. If you do not change the passwords at this point, it will be more difficult to change them later.

By default, NWS uses an auto-generated self-signed SSL certificate. However, you can use your own certfile, keyfile, and ca_certs files if you wish.

- /opt/nitro-web-services/etc/zmq_job_status_adapter.cfg

This is the Nitro ZMQ Job Status Adapter configuration file.

The Nitro ZMQ Job Status Adapter listens to job status updates on the ZQM bus and publishes them to MongoDB using the Nitro Web Services REST API. The username and password must be set to a MongoDB user with write permissions. At minimum, set the password for nitro-writeonly-user to the password defined in /opt/nitro-web-services/etc/nitro.cfg and make sure the SSL options are set correctly based on SSL settings in /opt/nitro-web-services/etc/nitro.cfg.

4. Start the services and configure Nitro Web Services to start automatically at system boot.

- Red Hat 6-based systems

```
[root] chkconfig --add nitro-web-services
[root] chkconfig --add nitro-zmq-job-status-adapter
[root] service nitro-web-services start
[root] service nitro-zmq-job-status-adapter start
```

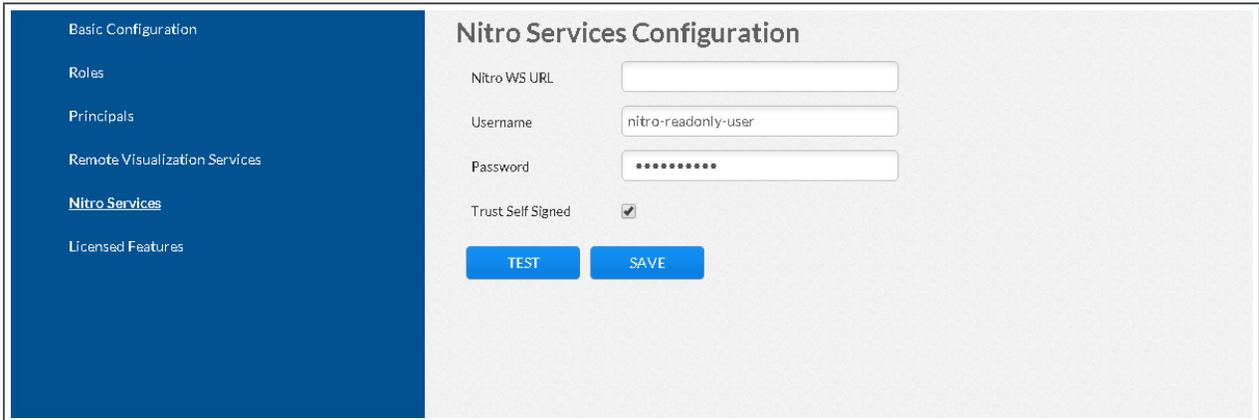
- Red Hat 7-based or SUSE 12-based systems

```
[root] systemctl enable nitro-web-services.service
[root] systemctl enable nitro-zmq-job-status-adapter.service
[root] systemctl daemon-reload
[root] systemctl start nitro-web-services.service
[root] systemctl start nitro-zmq-job-status-adapter.service
```

Configure Viewpoint for Nitro Web Services

Do the following:

1. Using a web browser, navigate to your Viewpoint instance (`http://<server>:8081`) and then log in as the MWS administrative user (`moab-admin`, by default).
2. Click **Configuration** from the menu and then click **Nitro Services** from the left pane. The following is an example of the Nitro Services Configuration page.



3. Enter the configuration information. The following table describes the required information.

| Field | Description |
|-------------------|--|
| Nitro WS URL | Hostname (or IP address) and port number for the host on which you installed Nitro Web Services. For example, <code>https://<hostname>:9443</code> |
| Username | Name of the user. This typically <code>nitro-readonly-user</code> . |
| Password | The user's password. |
| Trust Self Signed | Indicates whether Nitro Web Services was set up using self-signed certificates. |

4. Click **TEST** to confirm the settings are correct. This confirms whether Nitro Web Services is up and receiving connections.
5. Click **SAVE**.
6. (Recommended) Use curl to test Nitro Web Services connectivity.

```
curl --insecure --data '{"username": "nitro-admin", "password": "ChangeMe2!"}' \
https://<hostname>:9443/auth
```

You should see output similar to the following.

```

{
  "status": 200,
  "data": {
    "nitro-key": "3e0fb95e9a0e44ae91daef4deb500dcc67a3714880e851d781512a49",
    "user": {
      "username": "nitro-admin",
      "last_updated": "2016-02-26 23:34:55.604000",
      "name": "Nitro Admin",
      "created": "2016-02-26 23:34:55.604000",
      "auth": {
        "job": [
          "read",
          "write",
          "delete"
        ],
        "user": [
          "read",
          "write",
          "delete"
        ]
      }
    }
  }
}

```

Configure Nitro Compute Nodes

You need to configure the Nitro coordinators to send job status updates to the Nitro Web Services's ZMQ Job Status Adapter. The ZMQ Job Status Adapter is responsible for reading job status updates off of the ZMQ bus and persisting them to Mongo. Nitro Web Services can then be used to access Nitro job status.

Each Nitro job has a Nitro Coordinator. Nitro Coordinators can be configured to publish job status updates to ZMQ by setting the "nws-connector-address" configuration option in Nitro's `nitro.cfg` file. Each compute node allocated/scheduled to a Nitro Job can play the role of a Nitro coordinator. Therefore, you must update the "nws-connector-address" in each compute node's `nitro.cfg` file.

i Configuring `nws-connector-address` is simplified if each node is sharing Nitro's configuration over a shared filesystem. If you are *not* using a shared filesystem, update the Nitro configuration on *each* compute node.

Do the following:

1. If you have not already done so, on the Nitro Web Services Host, locate the `msg_port` number in the `/opt/nitro-web-services/etc/zmq_job_status_adapter.cfg` file. This is the port number you need to specify for the `nws-connector-address`.
2. On *each* Nitro compute note (Torque MOM Host), specify the `nws-connector-address` in the `/opt/nitro/etc/nitro.cfg` file.

```
...  
# Viewpoint connection allows Nitro to communicate job status information  
# to viewpoint. This option indicates name and port of the remote server  
# in the form: <host>:<port>  
nws-connector-address <nitro-web-services-hostname>:47100  
...
```

Related Topics

- [Nitro Integration on page 217](#)

Additional Configuration

In this section:

- [Configuring SSL in Tomcat on page 231](#)
- [Setting Up OpenLDAP on CentOS 6 on page 231](#)
- [Trusting Servers in Java on page 238](#)

Configuring SSL in Tomcat

To configure SSL in Tomcat, please refer to the Apache Tomcat [documentation](http://tomcat.apache.org/tomcat-7.0-doc/ssl-howto.html) (<http://tomcat.apache.org/tomcat-7.0-doc/ssl-howto.html>).

Setting Up OpenLDAP on CentOS 6

These instructions are intended to help first-time LDAP administrators get up and running. The following procedures contain instructions for getting started using OpenLDAP on a CentOS 6 system. For more complete information on how to set up OpenLDAP see the [OpenLDAP documentation](http://www.openldap.org/doc/admin24/) (<http://www.openldap.org/doc/admin24/>).

In this topic:

- [Installing and Configuring OpenLDAP on Centos 6 on page 231](#)
- [Adding an Organizational Unit \(OU\) on page 235](#)
- [Adding a User on page 236](#)
- [Adding a Group on page 237](#)
- [Adding a User to a Group on page 237](#)

i Adaptive Computing is not responsible for creating, maintaining, or supporting customer LDAP or Active Directory configurations.

Installing and Configuring OpenLDAP on Centos 6

First, you will need to install OpenLDAP. These instructions explain how you can do this on a CentOS 6 system.

To install and configure OpenLDAP on Centos 6

1. Run the following command:

```
[root]# yum -y install openldap openldap-clients openldap-servers
```

2. Generate a password hash to be used as the admin password. This password hash will be used when you create the root user for your LDAP installation. For example:

```
[root]# slappasswd
New password : p@ssw0rd
Re-enter new password : p@ssw0rd
{SSHA}51PFVw19zeh7LT53hQH69znzj8TuBrLv
```

3. Add the root user and the root user's password hash to the OpenLDAP configuration in the `olcDatabase={2}bdb.ldif` file. The root user will have permissions to add other users, groups, organizational units, etc. Do the following:

- a. Run this command:

```
[root]# cd /etc/openldap/slapd.d/cn\=config
[root]# vi olcDatabase=\{2\}bdb.ldif
```

- b. If the **olcRootPW** attribute does not already exist, create it. Then set the value to be the hash you created from `slappasswd`. For example:

```
olcRootPW: {SSHA}51PFVw19zeh7LT53hQH69znzj8TuBrLv
...
```

4. While editing this file, change the distinguished name (DN) of the **olcSuffix** to something appropriate. The suffix typically corresponds to your DNS domain name, and it will be appended to the DN of every other LDAP entry in your LDAP tree.

For example, let's say your company is called Acme Corporation, and that your domain name is "acme.com". You might make the following changes to the `olcDatabase={2}bdb.ldif` file:

```
olcSuffix: dc=acme,dc=com
...
olcRootDN: cn=Manager,dc=acme,dc=com
...
olcRootPW: {SSHA}51PFVw19zeh7LT53hQH69znzj8TuBrLv
...
```

 Do not set the cn of your root user to "root" (`cn=root,dc=acme,dc=com`), or OpenLDAP will have problems.

 Throughout the following examples in this topic, you will see `dc=acme,dc=com`. "acme" is only used as an example to illustrate what you would use as your own domain controller if your domain name was "acme.com". You should replace any references to "acme" with your own organization's domain name.

5. Modify the DN of the root user in the `olcDatabase={1}monitor.ldif` file to match the **olcRootDN** line in the `olcDatabase={2}bdb.ldif` file. Do the following:

- a. Run this command to edit the `olcDatabase={2}bdb.ldif` file:

```
[root]# vi olcDatabase=\{1\}monitor.ldif
```

- b. Modify the **olcAccess** line so that the **dn.base** matches the **olcRootDN** from the `olcDatabase={2}bdb.ldif` file. (In this example, **dn.base** should be `"cn=Manager,dc=acme,dc=com"`.)

```
olcAccess: {0}to * by
dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth" read by
dn.base="cn=Manager,dc=acme,dc=com" read by * none
```

- c. Now the root user for your LDAP is `cn=Manager,dc=acme,dc=com`. The root user's password is the password that you entered using `slappasswd` earlier in this procedure, which, in this example, is **p@ssw0rd**
6. Hide the password hashes from users who should not have permission to view them.

i A full discussion on configuring access control in OpenLDAP is beyond the scope of this tutorial. For help, see [the OpenLDAP Access Control documentation](http://www.openldap.org/doc/admin24/access-control.html) (<http://www.openldap.org/doc/admin24/access-control.html>).

- a. Run this command to edit the `olcDatabase=\{2\}bdb.ldif` file:

```
[root]# vi olcDatabase=\{2\}bdb.ldif
```

- b. Add the following two lines to the end of the file to restrict users from viewing other users' password hashes.

```
olcAccess: {0}to attrs=userPassword by self write by
dn.base="cn=Manager,dc=acme,dc=com" write by anonymous auth by * none
olcAccess: {1}to * by dn.base="cn=Manager,dc=acme,dc=com" write by self write by
* read
```

These lines allow a user to read and write his or her own password. It also allows a manager to read and write anyone's password. Anyone, including anonymous users, is allowed to view non-password attributes of other users.

7. Make sure that OpenLDAP is configured to start when the machine starts up, and start the OpenLDAP service.

```
[root]# chkconfig slapd on
[root]# service slapd start
```

8. Now, you must manually create the `"dc=acme,dc=com"` LDAP entry in your LDAP tree.

An LDAP directory is analogous to a tree. Nodes in this tree are called LDAP "entries" and may represent users, groups, organizational units, domain controllers, or other objects. The attributes in each entry are determined by the LDAP schema. In this tutorial we will build entries based on the InetOrgPerson schema (which ships with OpenLDAP by default).

In order to build our LDAP tree we must first create the root entry. Root entries are usually a special type of entry called a domain controller (DC). Because we are assuming that the organization is called Acme Corporation, and that the domain is "acme.com," we will create a domain controller LDAP entry called `dc=acme,dc=com`. Again, you will need to replace "acme" with your organization's domain name. Also note that `dc=acme,dc=com` is what is called an LDAP distinguished name (DN). An LDAP distinguished name uniquely identifies an LDAP entry.

Do the following:

- a. Create a file called `acme.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi acme.ldif
```

- b. Add the following lines in `acme.ldif`:

```
dn: dc=acme,dc=com
objectClass: dcObject
objectClass: organization
dc: acme
o : acme
```

- c. Now add the contents of this file to LDAP. Run this command:

```
[root]# ldapadd -f acme.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

- d. Verify that your entry was added correctly.

```
[root]# ldapsearch -x -LLL -b dc=acme,dc=com
dn: dc=acme,dc=com
objectClass: dcObject
objectClass: organization
dc: acme
o: acme
```

9. Run the following:

```
[root]# sudo iptables -L
[root]# sudo service iptables save
```

10. By default, the CentOS 6 firewall will block external requests to OpenLDAP. In order to allow MWS to access LDAP, you will have to configure your firewall to allow connections on port 389. (Port 389 is the default LDAP port.)

Configuring your firewall is beyond the scope of this tutorial; however, it may be helpful to know that the default firewall on CentOS is a service called `iptables`. For more information, see the documentation on [iptables](http://wiki.centos.org/HowTos/Network/IPTables) (<http://wiki.centos.org/HowTos/Network/IPTables>). In the most basic case, you may be able to add a rule to your firewall that accepts connections to port 389 by doing the following:

- a. Edit your `iptables` file:

```
[root]# vi /etc/sysconfig/iptables
```

- b. Add the following line *after* all the **ACCEPT** lines but *before* any of the **REJECT** lines in your `iptables` file:

```
# ... lines with ACCEPT should be above
-A INPUT -p tcp --dport 389 -j ACCEPT
# .. lines with REJECT should be below
```

For example, here is a sample `iptables` file with this line added:

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -p tcp --dport 389 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited

COMMIT
```

- c. Now reload `iptables`.

```
[root]# service iptables reload
```

i Although providing instructions is beyond the scope of this tutorial, it is also highly recommended that you set up OpenLDAP to use SSL or TLS security to prevent passwords and other sensitive data from being sent in plain text. For information on how to do this, see the [OpenLDAP TLS documentation](http://www.openldap.org/doc/admin24/tls.html) (<http://www.openldap.org/doc/admin24/tls.html>).

Now that you have installed and set up Open LDAP, you are ready to add organizational units. See [Adding an Organizational Unit \(OU\) on page 235](#).

Adding an Organizational Unit (OU)

These instructions will describe how to populate the LDAP tree with organizational units (OUs), groups, and users, all of which are different types of LDAP entries. The examples that follow also presume an `InetOrgPerson`

schema, because the InetOrgPerson schema is delivered with OpenLDAP by default.

To add an organizational unit (OU) entry to the LDAP tree

In this example, we are going to add an OU called "Users".

1. Create a temporary file called `users.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi users.ldif
```

2. Add these lines to `users.ldif`:

```
dn: ou=Users,dc=acme,dc=com
objectClass: organizationalUnit
ou: Users
```

3. Add the contents of `users.ldif` file to LDAP.

```
[root]# ldapadd -f users.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

Adding a User

To add a user to LDAP

In this example, we will add a user named "Bob Jones" to LDAP inside the "Users" OU.

1. Create a temporary file called `bob.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi bob.ldif
```

2. Add these lines to `bob.ldif`:

```
dn: cn=Bob Jones,ou=Users,dc=acme,dc=com
cn: Bob Jones
sn: Jones
objectClass: inetOrgPerson
userPassword: p@ssw0rd
uid: bjones
```

3. Add the contents of `bob.ldif` file to LDAP.

```
[root]# ldapadd -f bob.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

Adding a Group

To add a group to LDAP

In this example, we will add a group called "Engineering" to LDAP inside the "Users" OU.

1. Create a temporary file called `engineering.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi engineering.ldif
```

2. Add these lines to `engineering.ldif`:

```
dn: cn=Engineering,ou=Users,dc=acme,dc=com
cn: Engineering
objectClass: groupOfNames
member: cn=Bob Jones,ou=Users,dc=acme,dc=com
```

3. Add the contents of `engineering.ldif` file to LDAP.

```
[root]# ldapadd -f engineering.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

Adding a User to a Group

To add a user to an LDAP group

In this example, we will add an LDAP member named "Al Smith" to the "Engineering" LDAP group. This example assumes that user, Al Smith, has already been added to LDAP.

i Before you add a user to an LDAP group, the user must first be added to LDAP. For more information, see [Adding a User on page 236](#).

1. Create a temporary file called `addUserToGroup.ldif`. (You can delete this file once its content has been added to LDAP, so in this example, we will create it in the `/tmp` folder.)

```
[root]# cd /tmp
[root]# vi addUserToGroup.ldif
```

2. Add these lines to `addUserToGroup.ldif`:

```
dn: cn=Engineering,ou=Users,dc=acme,dc=com
changetype: modify
add: member
member: cn=Al Smith,ou=Users,dc=acme,dc=com
```

3. Now add the contents of `addUserToGroup.ldif` file to LDAP.

```
[root]# ldapadd -f addUserToGroup.ldif -D cn=Manager,dc=acme,dc=com -w p@ssw0rd
```

Trusting Servers in Java

In this topic:

[Prerequisites on page 238](#)

[Retrieve the Server's X.509 Public Certificate on page 238](#)

[Add the Server's Certificate to Java's Keystore on page 238](#)

Prerequisites

Some of these instructions refer to `JAVA_HOME`, which must point to the same directory that Tomcat uses. To set `JAVA_HOME`, do this:

```
[root]# source /etc/tomcat/tomcat.conf
```

Your system administrator might have defined Tomcat's `JAVA_HOME` in a different file.

Retrieve the Server's X.509 Public Certificate

To retrieve the server's certificate, use the following command:

```
[root]# $JAVA_HOME/bin/keytool -printcert -rfc -sslserver <servername>:<port> > /tmp/public.cert.pem
```

Replace `<servername>` with the server's host name and `<port>` with the secure port number. The default port for https is 443. The default port for ldaps is 636. If successful, `/tmp/public.cert.pem` contains the server's public certificate. Otherwise, `/tmp/public.cert.pem` contains an error message. This message is typical: `keytool error: java.lang.Exception: No certificate from the SSL server.` This message suggests that the server name or port is incorrect. Consult your IT department to determine the correct server name and port.

Add the Server's Certificate to Java's Keystore

Java stores trusted certificates in a database known as the keystore. Because each new version of Java has its own keystore, you need to add the server certificate to the Java keystore (using the steps below) every time you install a new version of Java.

Java's keystore is located at `$JAVA_HOME/lib/security/cacerts`. If Tomcat's `JAVA_HOME` points to a JDK, then the keystore is located at `$JAVA_`

HOME/jre/lib/security/cacerts. To add the server certificate to the keystore, run the following command:

```
[root]# $JAVA_HOME/bin/keytool -import -trustcacerts -file /tmp/public.cert.pem -alias <servername> -keystore $JAVA_HOME/lib/security/cacerts
```

You will be prompted for the keystore password, which is "changeit" by default.

i Your system administrator might have changed this password.

After you've entered the keystore password, you'll see the description of the server's certificate. At the end of the description it prompts you to trust the certificate.

```
Trust this certificate? [no]:
```

Type *yes* and press **Enter** to add the certificate to the keystore.

RPM Upgrades

This section provides instructions and other information when upgrading your Moab HPC Suite components using the RPM upgrade method.

In this section:

- [Preparing the Host – Typical Method on page 125](#)
- [Creating the moab-offline Tarball on page 127](#)
- [Preparing the Host – Offline Method on page 130](#)
- [Upgrading Torque Resource Manager \(RPM\) on page 240](#)
- [Upgrading Moab Workload Manager \(RPM\) on page 243](#)
- [Upgrading Moab Accounting Manager \(RPM\) on page 247](#)
- [Upgrading Moab Web Services \(RPM\) on page 250](#)
- [Upgrading Moab Insight \(RPM\) on page 253](#)
- [Upgrading Moab Viewpoint \(RPM\) on page 256](#)
- [Migrating the MAM Database from MySQL to PostgreSQL on page 263](#)

Upgrading Torque Resource Manager (RPM)

This topic provides instructions to upgrade Torque Resource Manager to the latest release version using the RPM upgrade method. It includes instructions for migrating your database schema to a new version if necessary.

i Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

Upgrade Steps

Do the following:

1. If you installed Torque Server on its own host *or* if Torque Server is the first component being upgraded on a host with other RPM installations, complete the steps to prepare the host.

Do the same as needed for each Torque MOM Host (compute node).

See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).

2. Stop all Torque Server, Torque MOM, and Torque Client Services. See [Stop Torque Services on page 241](#).
3. Upgrade Torque Server, Torque MOMs, and Torque Clients. See [Upgrade Torque Server, MOMs, and Clients on page 242](#).
4. Start all Torque Server, Torque MOM, and Torque Client Services. See [Start Torque Services on page 242](#).

Stop Torque Services

Do the following:

1. On the Torque Server Host, shut down the Torque server.

- Red Hat 6-based systems

```
[root]# service pbs_server stop
```

- Red Hat 7-based

```
[root]# systemctl stop pbs_server.service
```

2. On each Torque MOM Host, shut down the Torque MOM service.

 Confirm all jobs have completed before stopping pbs_mom. You can do this by typing "momctl -d3". If there are no jobs running, you will see the message "NOTE: no local jobs detected" towards the bottom of the output. If jobs are still running and the MOM is shutdown, you will only be able to track when the job completes and you will not be able to get completion codes or statistics.

- Red Hat 6-based systems

```
[root]# service pbs_mom stop
```

- Red Hat 7-based

```
[root]# systemctl stop pbs_mom.service
```

3. On each Torque Client Host (including the Moab Server Host, the Torque Server Host, and the Torque MOM Hosts, if applicable), shut down the trqauthd service.

- Red Hat 6-based systems

```
[root]# service trqauthd stop
```

- Red Hat 7-based

```
[root]# systemctl stop trqauthd.service
```

Upgrade Torque Server, MOMs, and Clients

i You *must* complete all the previous upgrade steps in this topic before upgrading Torque Server, MOMs, and Clients. See the list of steps at the beginning of this topic.

Do the following:

1. Upgrade Torque Server.

On the Torque Server Host, install the upgrade.

```
[root]# yum update moab-torque*
```

2. Upgrade Torque MOMs.

Do the following:

- a. On the Torque Server Host, locate the directory where the rpm distro tarball was unpacked and copy the moab-torque-common and moab-torque-mom RPM files to each Torque MOM Host (excluding a Torque MOM Host that also resides on the Torque Server Host). It is also recommended that you install the moab-torque-common RPM so you can use client commands and submit jobs from the Torque MOM Hosts (compute nodes).

```
[root]# scp RPMs/moab-torque-common-*.rpm <torque-mom-host>:
[root]# scp RPMs/moab-torque-mom-*.rpm <torque-mom-host>:
[root]# scp RPMs/moab-torque-client-*.rpm <torque-mom-host>:
```

- b. On each Torque MOM Host (excluding a Torque MOM Host that also resides on the Torque Server Host), use the uploaded RPMs to update the host.

```
[root]# ssh root@<torque-mom-host>
[root]# yum update moab-torque-*
```

3. Upgrade Torque Clients.

If you have any Torque Client Hosts that are different from the Torque Server Host or the Torque MOM Hosts, update those Torque Client Hosts.

```
[root]# yum update moab-torque*
```

Start Torque Services

Do the following:

1. On the Torque Server Host, start up the Torque server.

- Red Hat 6-based systems

```
[root]# service pbs_server start
```

- Red Hat 7-based

```
[root]# systemctl daemon-reload
[root]# systemctl start pbs_server.service
```

2. On each Torque MOM Host, start up the Torque MOM service.

- Red Hat 6-based systems

```
[root]# service pbs_mom start
```

- Red Hat 7-based

```
[root]# systemctl daemon-reload
[root]# systemctl start pbs_mom.service
```

3. On each Torque Client Host (including the Moab Server Host, Torque Server Host and Torque MOM Hosts, if applicable), start up the trqauthd service.

- Red Hat 6-based systems

```
[root]# service trqauthd start
```

- Red Hat 7-based

```
[root]# systemctl daemon-reload
[root]# systemctl start trqauthd.service
```

Upgrading Moab Workload Manager (RPM)

This topic provides instructions to upgrade Moab Workload Manager to the latest release version using the RPM upgrade method. It includes instructions for migrating your database schema to a new version if necessary.

i Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

Upgrade Steps

Do the following:

1. If you installed Moab Server on its own host *or* if Moab Server is the first component being upgraded on a host with other RPM installations, complete the steps to prepare the host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).
2. If you use ODBC, confirm the database schema compatibility. For example, if you are upgrading Moab 8.1 to 9.0 no schema changes were made; however if you upgrade from Moab 8.0 and prior, you will need to upgrade your database. See [Migrating Your Database to Newer Versions of Moab](#) in the *Moab Workload Manager Administrator Guide* for more information.
3. Upgrade Moab Server. See [Upgrade Moab Server on page 244](#).

Upgrade Moab Server

i You *must* complete all the previous upgrade steps in this topic before upgrading Moab Server. See the list of steps at the beginning of this topic.

i The Moab RPM automatically creates a backup of all relevant files. These backups are stored in `/var/tmp/backup-<rpmName>-<timestamp>.tar.gz`.

If changes are detected between any existing configuration files and new configuration files, a version of the new configuration file will be saved under `<configurationFileLocation>/<fileName>.rpmnew`.

On the Moab Server Host, do the following:

1. Stop Moab.

- Red Hat 6-based systems

```
[root]# service moab stop
```

- Red Hat 7-based

```
[root]# systemctl stop moab.service
```

2. Install the upgrade.

```
[root]# yum update moab-workload-manager*
```

3. Merge the configuration files.

i You will need to decide whether to start with the old configuration file and add newer configuration options (or vice versa). Typically it depends on the amount of customization you previously made in earlier versions. In instances where you have modified very little, you should consider using the newer configuration and merging site-specific settings from the old file into the new one. The following steps highlight important changes between the 7.2.x default configuration and the 9.0.1 default configuration. Also note that new configuration files may have auto-generated content for secret keys and default passwords—be careful to ensure that secret keys shared between components are configured correctly.

i The recommended layout for the `/opt/moab/etc/` directory appears as follows:

```
[root]# ls -l /opt/moab/etc
total 29
-rw-r--r--. 1 root moab 2323 Nov 13 13:41 config.moab.pl
-rw-r--r--. 1 root moab 989 Nov 13 13:41 config.sql.pl
lrwxrwxrwx. 1 root root 14 Nov 13 15:46 moab.cfg -> moab.hpc.cfg
-rw-r--r--. 1 root moab 23500 Nov 13 15:43 moab.hpc.cfg
drwxr-xr-x. 2 root moab 4096 Nov 13 15:41 moab.d
-rw-r--r--. 1 root moab 391 Nov 13 13:41 moab.dat
-r--r--r--. 1 root root 493 Nov 6 16:14 moab.lic
-rw-----. 1 root moab 288 Nov 13 15:39 moab-private.cfg
lrwxrwxrwx. 1 root root 14 Nov 13 15:46 nami.cfg -> nami.hpc.cfg
-rw-r--r--. 1 root moab 563 Nov 13 15:43 nami.hpc.cfg
```

- a. Merge the `/opt/moab/etc/moab-private.cfg` file. Make sure that unique items in `/opt/moab/etc/moab-private.cfg.rpmnew` are added to the existing `/opt/moab/etc/moab-private.cfg` file. Include the new MWS RM credentials if you configure MWS as a resource manager:

```
CLIENTCFG[RM:mws] USERNAME=moab-admin PASSWORD=changeme!
```

i The default MWS credentials in 7.2.x were `admin:adminpw`. For releases after 7.2.x, the default credentials were changed to `moab-admin:changeme!`. Use whatever credentials you have configured in `/opt/mws/etc/mws-config.groovy`.

- b. Merge customizations from `/opt/moab/etc/moab.cfg` and `/opt/moab/etc/moab.d/*` into `/opt/moab/etc/moab.hpc.cfg`.

i If you are upgrading from a version prior to 9.0, the Torque RPMs will have moved the Torque binaries from `/usr` to `/usr/local`. Make sure that your `RMCFG[] SUBMITCMD` parameter is set to the correct path for `qsub`.

```
[root]# vi /opt/moab/etc/moab.cfg
RMCFG[pbs]      TYPE=PBS  SUBMITCMD=/usr/local/bin/qsub
```

- Although there are several ways to configure and merge changes into the `/opt/moab/etc/moab.cfg` file, the following instructions outline the recommended best practices. *Deviations from these best practices may result in unexpected behavior or added difficulty in future upgrades.*
- It is best to use the new default configuration file (`/opt/moab/etc/moab.hpc.cfg`) and merge changes from previous files into that one. You will notice that content from the `/opt/moab/etc/moab.d/` directory has been merged into `/opt/moab/etc/moab.hpc.cfg`. Ensure that custom configuration options in all files located in `/opt/moab/etc/moab.d/` directory get merged in to `/opt/moab/etc/moab.hpc.cfg`.
- You should avoid `#include` configurations.
- Although the upgrade should have created a backup of the `moab.cfg` file (in `/var/tmp/backup-<rpmName>-<timestamp>.tar.gz`), it is best to create your own backup until you can confirm the updated configuration behaves as expected.

```
[root]# cp /opt/moab/etc/moab.cfg /opt/moab/etc/moab.cfg.bak
```

- c. If you are upgrading from a version prior to 8.0, once the changes have been merged to `/opt/moab/etc/moab.hpc.cfg`, configure Moab to use the new file. The recommended configuration is to use a symlink called `/opt/moab/etc/moab.cfg` that points to `/opt/moab/etc/moab.hpc.cfg`.

```
[root]# ln -s /opt/moab/etc/moab.hpc.cfg /opt/moab/etc/moab.cfg
```

4. Start Moab.

- Red Hat 6-based systems

```
[root]# service moab start
```

- Red Hat 7-based

```
[root]# systemctl start moab.service
```

Upgrading Moab Accounting Manager (RPM)

This topic provides instructions to upgrade Moab Accounting Manager to the latest release version using the RPM upgrade method. It includes instructions for migrating your database schema to a new version if necessary.

i Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

Upgrade Steps

Do the following:

1. If you installed MAM Server on its own host *or* if MAM Server is the first component being upgraded on a host with other RPM installations, complete the steps to prepare the host.

Do the same as needed for the MAM GUI Host and each MAM Client Host.

See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).

2. Upgrade MAM Server. See [Upgrade MAM Server on page 247](#).
3. Upgrade MAM GUI. See [Upgrade MAM GUI on page 249](#).
4. Upgrade MAM Clients. See [Upgrade MAM Clients on page 250](#).

Upgrade MAM Server

i You *must* complete all the previous upgrade steps in this topic before upgrading MAM Server. See the list of steps at the beginning of this topic.

On the MAM Server Host, do the following:

1. Stop MAM.
 - Red Hat 6-based systems

```
[root]# service mam stop
```

- Red Hat 7-based systems

```
[root]# systemctl stop mam.service
```

2. Install the upgrade.

i The MAM RPM name has changed between version 8.1 and 9.0. The RPM obsoleted process removes the old RPM and installs the new RPM separately; this results in removing the mam user and not preserving the customized configuration files. A special process must be followed when upgrading from an RPM version prior to 9.0.

- If you are upgrading MAM from an RPM version prior to 9.0, do the following:

```
for i in /opt/mam/etc/{gold,goldd,goldg,site}.conf
do
cp -p ${i} ${i}.rpmsave
done
rpm -e --nopostun moab-hpc-enterprise-suite moab-hpc-accounting-manager
yum install moab-accounting-manager
for i in /opt/mam/etc/mam-*.conf
do
cp -p ${i} ${i}.rpmnew
done
\cp -f /opt/mam/etc/gold.conf.rpmsave /opt/mam/etc/mam-client.conf
\cp -f /opt/mam/etc/goldd.conf.rpmsave /opt/mam/etc/mam-server.conf
\cp -f /opt/mam/etc/goldg.conf.rpmsave /opt/mam/etc/mam-gui.conf
\cp -f /opt/mam/etc/site.conf.rpmsave /opt/mam/etc/mam-site.conf
```

- If you are upgrading MAM from an RPM version at or after 9.0, do the following:

```
[root]# yum update moab-accounting-manager*
```

3. Compare your existing configuration files (/opt/mam/etc/mam-*.conf) with those distributed with the new release (/opt/mam/etc/mam-*.conf.rpmnew) and merge the differing lines into your configuration files.

4. Start the mam service.

- Red Hat 6-based system

```
[root]# service mam start
```

- Red Hat 7-based system

```
[root]# systemctl start mam.service
```

5. If you are upgrading MAM from a version prior to 8.0, add the new mam user as a MAM Accounting Admin.

```
[root]# su -c "mam-create-user -u mam -d \"Accounting Admin\"" moab
[root]# su -c "mam-modify-role -r SystemAdmin --add-user mam" moab
[root]# perl -p -i -e 's/moab/mam/ if /^super.user/' /opt/mam/etc/mam-server.conf
```

6. Migrate the Moab Accounting Manager database from your current version to 9.0, running the migration script.

- a. Run one or more migration scripts. You must run every incremental migration script between the version you are currently using and the new version (9.0). The migration scripts are located in the `/usr/share/moab-accounting-manager/` directory. These scripts are designed to be rerunnable, so if you encounter a failure, resolve the failure and rerun the migration script. If you are unable to resolve the failure and complete the migration, contact Support.

i The migration scripts *must* be run as the mam user.

For example, if you are migrating from Moab Accounting Manager version 7.2, you must run five migration scripts: the first to migrate the database schema from 7.2 to 7.3, the second to migrate from 7.3 to 7.5, the third to migrate the database schema from 7.5 to 8.0, the fourth to migrate the database schema from 8.0 to 8.1, and the fifth to migrate the database schema from 8.1 to 9.0.

```
[root]# su - mam
[mam]$ /usr/share/moab-accounting-manager/migrate_7.2-7.3.pl
[mam]$ /usr/share/moab-accounting-manager/migrate_7.3-7.5.pl
[mam]$ /usr/share/moab-accounting-manager/migrate_7.5-8.0.pl
[mam]$ /usr/share/moab-accounting-manager/migrate_8.0-8.1.pl
[mam]$ /usr/share/moab-accounting-manager/migrate_8.1-9.0.pl
```

- b. Verify that the resulting database schema version is 9.0.

```
[mam]$ mam-shell System Query
```

| Name | Version | Description |
|-------------------------|---------|--------------------|
| Moab Accounting Manager | 9.0 | Commercial Release |

- c. Verify that the executables have been upgraded to 9.0.1.

```
[mam]$ mam-server -v
Moab Accounting Manager version 9.0.1
```

Upgrade MAM GUI

If you are using the MAM GUI and the MAM GUI Host is different from the MAM Server Host, then do the following on the MAM GUI Host:

1. Install the upgrade.
 - If you are upgrading the MAM RPM from a version prior to 9.0, do the following:

```
cp -p /opt/mam/etc/goldg.conf /opt/mam/etc/goldg.conf.rpmsave
rpm -e --nopostun moab-hpc-accounting-manager
yum install moab-accounting-manager
cp -p /opt/mam/etc/mam-gui.conf /opt/mam/etc/mam-gui.conf.rpmnew
\cp -f /opt/mam/etc/goldg.conf.rpmsave /opt/mam/etc/mam-gui.conf
```

- If you are upgrading the MAM RPM from a version at or after 9.0, do the following:

```
[root]# yum update moab-accounting-manager*
```

2. Compare your current gui configuration file (`/opt/mam/etc/mam-gui.conf`) with the one distributed with the new release (`/opt/mam/etc/mam-gui.conf.rpmnew`) and merge the differing lines into your current configuration file.

Upgrade MAM Clients

If you are have any MAM Client Hosts that are different from the MAM Server Host or MAM GUI Hosts, then do the following on each MAM Client Host:

1. Install the upgrade.

- If you are upgrading the MAM RPM from a version prior to 9.0, do the following:

```
cp -p /opt/mam/etc/gold.conf /opt/mam/etc/gold.conf.rpmsave
rpm -e --nopostun moab-hpc-accounting-manager
yum install moab-accounting-manager
cp -p /opt/mam/etc/mam-client.conf /opt/mam/etc/mam-client.conf.rpmnew
\cp -f /opt/mam/etc/gold.conf.rpmsave /opt/mam/etc/mam-client.conf
```

- If you are upgrading the MAM RPM from a version at or after 9.0, do the following:

```
[root]# yum update moab-accounting-manager*
```

2. Compare your current client configuration file (`/opt/mam/etc/mam-client.conf`) with the one distributed with the new release (`/opt/mam/etc/mam-client.conf.rpmnew`) and merge the differing lines into your current configuration file.

Upgrading Moab Web Services (RPM)

This topic provides instructions to upgrade Moab Web Services to the latest release version using the RPM upgrade method. It includes instructions for migrating your database schema to a new version if necessary.



If upgrading Moab Web Services from a version prior to 8.0, this upgrade removes all MWS roles and permissions and recreates the default roles. If you have modified any MWS permissions or roles, you will need to recreate them after the upgrade is complete.

i Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

Upgrade Steps

Do the following:

1. Confirm the Moab Server RPM upgrade has completed on the host on which MWS Server is also installed. See [Upgrading Moab Workload Manager \(RPM\) on page 243](#).
2. Upgrade your MongoDB database version to 2.4.x (recommended). See [1.1 Upgrading the MongoDB Database](#).
3. Upgrade to Java 8 (recommended). See [Upgrade to Java 8 on page 251](#)
4. Upgrade MWS Server. See [Upgrade MWS Server on page 251](#).

Upgrade to Java 8

i Oracle Java 8 Runtime Environment is the recommended Java environment, but Oracle Java 7 is also supported. All other versions of Java, including OpenJDK/IcedTea, GNU Compiler for Java, and so on cannot run MWS.

If you wish to upgrade to Java 8, refer to the [Install Java on page 57](#) instructions.

Upgrade MWS Server

i You *must* complete all the previous upgrade steps in this topic before upgrading MWS server. See the list of steps at the beginning of this topic.

i The MWS RPM automatically creates a backup of all relevant files. These backups are stored in `/var/tmp/backup-<rpmName>-<timestamp>.tar.gz`.

If changes are detected between any existing configuration files and new configuration files, a version of the new configuration file will be saved under `<configurationFileLocation>/<fileName>.rpmnew`.

On the MWS Server Host, do the following:

1. Stop Tomcat.

- Red Hat 6-based systems.

- If your prior MWS Server version used Tomcat 6, disable the tomcat 6 service.

```
[root]# service tomcat6 stop
[root]# chkconfig tomcat6 off
```

i Tomcat 6 is not supported for MWS 9.0 and after. The MWS RPM will automatically install Tomcat 7.

- If your prior MWS Server version used Tomcat 7, stop the tomcat 7 service.

```
[root]# service tomcat stop
```

- Red Hat 7-based systems, stop the tomcat 7 service.

```
[root]# systemctl stop tomcat.service
```

2. Back up the MWS home directory.

```
[root]# cp -r /opt/mws /opt/mws-8.1-backup
```

3. Install the upgrade.

```
[root]# yum update moab-web-services*
```

4. Upgrade the schema of the `mws` database in MongoDB.

⚠ You *must* perform this step, regardless of whether you upgraded MongoDB to version 2.4.x.

Run the database migration script provided with MWS. (It is safe to run this script more than once. If for any reason, errors occur during the execution of the script, run it again.)

```
[root]# mongo -u mws_user mws /opt/mws/utils/db-migrate.js -p
```

i You may be prompted for the mongo password. The password can be found in the `/opt/mws/etc/mws-config.groovy` file under the "grails.mongo.password" key.

i The script might take several minutes to execute.

5. Merge the configuration files.

- Merge new configuration options from `/opt/mws/etc/mws-config.groovy.rpmnew` into `/opt/mws/etc/mws-config.groovy`.
- Merge new configuration options from `/opt/mws/etc/mws.d/mws-config-hpc.groovy.rpmnew` into `/opt/mws/etc/mws.d/mws-config-hpc.groovy`.

See the step to merge the changes in [Upgrading Moab Web Services on page 112](#) for more details.

6. Remove unused MWS plugins. Unused plugins must be removed as their presence will prevent MWS from starting up.

- Remove all plugins from `/opt/mws/plugins` except for the diagnostics, native, and power-management plugins.

```
[root]# cd /opt/mws/plugins
[root]# rm plugins-reports.jar plugins-storage.jar plugins-vcenter.jar
```

7. Verify the Tomcat user has read access to the `/opt/mws/etc/mws-config.groovy` and `/opt/mws/etc/mws.d/mws-config-hpc.groovy` file.

8. Verify the following lines are added to the end of `/etc/tomcat/tomcat.conf`.

```
CATALINA_OPTS="-DMWS_HOME=/opt/mws -Xms256m -Xmx3g -Dfile.encoding=UTF8"
JAVA_HOME="/usr/java/latest"
```

9. Start Tomcat

- Red Hat 6-based systems

```
[root]# service tomcat start
```

- Red Hat 7-based systems

```
[root]# systemctl start tomcat.service
```

Upgrading Moab Insight (RPM)

This topic provides instructions to upgrade Moab Viewpoint to the latest release version using the RPM upgrade method. It includes instructions for migrating your database schema to a new version if necessary.

i Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

Upgrade the Insight Server

Do the following:

1. On the Moab Server Host, stop Moab from sending messages to Insight.

```
[root]# mschedctl -s
```

2. On the Insight Server Host, do the following:

- a. If you have not already done so, complete the steps to prepare the Insight Server Host for the upgrade. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).

- b. Stop Insight

- Red Hat 6-based systems

```
[root]# service insight stop
```

- Red Hat 7-based

```
[root]# systemctl stop insight.service
```

- c. Back up the Insight home directory.

```
[root]# cp -r /opt/insight /opt/insight-9.0.0-backup
```

- d. Back up your Insight databases

```
[root]# su - postgres
[postgres]$ pg_dump moab_insight > /tmp/moab_insight_9.0.0.dump
[postgres]$ pg_dump moab_insight_reference > /tmp/moab_insight_reference_9.0.0.dump
[postgres]$ exit
[root]# mv /tmp/moab_insight_9.0.0.dump /opt
[root]# mv /tmp/moab_insight_reference_9.0.0.dump /opt
[root]# mongodump --username insight_user --password secret4 -d insight --out /opt/insight_9.0.0.mongo.dump
```

- e. Install the upgrade

```
[root]# yum update moab-insight*
```

- f. Merge the new configuration from `/opt/insight/etc/config.groovy.rpmnew` into `/opt/insight/etc/config.groovy`.
- g. Verify the insight user has read access to the `/opt/insight/etc/config.groovy` file.

```
[root]# ls -l /opt/insight/etc/config.groovy
-rw-r--r--. 1 insight insight 3787 Jan 24 17:51 /opt/insight/etc/config.groovy
```

- h. Verify the following line is added to the end of `/opt/insight/etc/insight.conf`:

```
JAVA_HOME="/usr/java/latest"
```

- i. Start Insight.

i When upgrading, Insight will take some time before it will start processing messages from Moab.

- Red Hat 6-based systems

```
[root]# service insight start
```

- Red Hat 7-based

```
[root]# systemctl daemon-reload
[root]# systemctl start insight.service
```

- j. Wait for and confirm the database upgrade completed. *All jobs must be transferred before the upgrade is complete; this could take a long time if you have a lot of jobs.*

When the upgrade is completed, you will see the following lines in your `/opt/insight/log/insight.log` file:

```
19:03:43-0700 com.ace.insight.data.service.dbinit.DbUpgradeService
Moving 1231907 archived jobs to MongoDB
19:03:48-0700 com.ace.insight.data.service.dbinit.DbUpgradeService
Successfully transferred 100 of 1231907 jobs to MongoDB
19:03:48-0700 com.ace.insight.data.service.dbinit.DbUpgradeService
Successfully transferred 200 of 1231907 jobs to MongoDB
...
19:03:49-0700 com.ace.insight.data.service.dbinit.DbUpgradeService
Successfully transferred 1231907 of 1231907 jobs to MongoDB
```

3. On the Moab Server Host, have Moab resume sending messages to Insight.

```
mschedctl -r
```

Upgrading Moab Viewpoint (RPM)

This topic provides instructions to upgrade Moab Viewpoint to the latest release version using the RPM upgrade method. It includes instructions for migrating your database schema to a new version if necessary.

i Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

In this topic:

- [Upgrade the Viewpoint Server on page 256](#)
- [Upgrade the Viewpoint File Manager Service on page 259](#)
- [License and Configure Viewpoint on page 260](#)
- [Reset Viewpoint Roles on page 262](#)

Upgrade the Viewpoint Server

On the Viewpoint Server Host, do the following:

1. If you installed Viewpoint Server on its own host *or* if Viewpoint Server is the first component being upgraded on a host with other RPM installations, complete the steps to prepare the host. See [Preparing the Host – Typical Method on page 125](#) or [Preparing the Host – Offline Method on page 130](#).
2. Stop the Apache service.
 - Red Hat 6-based systems

```
[root]# service httpd stop
```

- Red Hat 7-based systems

```
[root]# systemctl stop httpd.service
```

3. Back up configuration files.

i Beginning with the 9.0.1 release, these variables are obsolete:

- IRIS_LOGS_FILENAME
- IRIS_LOGS_PATH
- IRIS_SESSION_FILE_PATH
- IRIS_TEMPLATE_DEBUG

Also be aware that the IRIS_DEBUG variable must *not* be used in production.

Therefore, remove these variables from /etc/httpd/conf.d/viewpoint.conf before backing it up.

```
[root]# cp -p /opt/viewpoint/config/config.json /etc/httpd/conf.d/viewpoint.conf /tmp
```

4. Back up certificates to connect to the file manager (if Viewpoint connects to file manager over SSL).

```
[root]# cp -p /opt/viewpoint/webdav_client/client-cert.pem /opt/viewpoint/webdav_client/client-key.pem /opt/viewpoint/webdav_client/ca-cert.pem /tmp
```

5. Remove your existing Viewpoint installation and some packages that are no longer needed.

```
[root]# rpm -e --nodeps moab-viewpoint
[root]# rpm -q --quiet python-importlib && rpm -e python-importlib
[root]# rpm -q --quiet mod_wsgi && rpm -e mod_wsgi
```

6. Remove some leftover files.

```
[root]# rm -rf /var/log/viewpoint /opt/viewpoint /etc/httpd/conf.d/viewpoint.conf /etc/cron.daily/viewpoint.sh
```

7. Install the new Viewpoint RPM.

```
[root]# yum install moab-viewpoint
```

8. Restore the config.json file to its new location

```
[root]# cp -p /tmp/config.json /opt/viewpoint/lib/viewpoint/config/config.json
```

9. Restore certificates to their new location

```
[root]# cp -p /tmp/client-cert.pem /tmp/client-key.pem /tmp/ca-cert.pem /opt/viewpoint/lib/viewpoint/webdav_client/
```

10. If you have any customizations to /etc/httpd/conf.d/viewpoint.conf, merge them into the new /etc/httpd/conf.d/viewpoint.conf.

i All IRIS_DATABASE* SetEnv entries in /etc/httpd/conf.d/viewpoint.conf are now obsolete. Database environment variables are now stored in /opt/viewpoint/etc/viewpoint.cfg. Therefore, move all your uncommented database SetEnv entries into the environment section of /opt/viewpoint/etc/viewpoint.cfg. For example:

```
/etc/httpd/conf.d/viewpoint.conf (old)
    SetEnv IRIS_DATABASE_NAME moab_viewpoint
/opt/viewpoint/etc/viewpoint.cfg (new)
    IRIS_DATABASE_NAME = moab_viewpoint
```

Verify or add values in the /opt/viewpoint/etc/viewpoint.cfg file for these parameters:

- IRIS_DATABASE_HOST
- IRIS_DATABASE_HOST
- IRIS_DATABASE_NAME
- IRIS_DATABASE_USER
- IRIS_DATABASE_PASSWORD

After you are finished, your /opt/viewpoint/etc/viewpoint.cfg will look something like this:

```
[admin]
username = viewpoint-admin
password = pbkdf2_
sha256$20000$ZHeToCJgrSUH$+xmzYdhpqZCJokx09eGzyr2B6jrfCgLlBT+pBgMis4w=

[environment]
IRIS_DATABASE_HOST = localhost
IRIS_DATABASE_PORT = 5432
IRIS_DATABASE_NAME = moab_viewpoint
IRIS_DATABASE_USER = moab_viewpoint
IRIS_DATABASE_PASSWORD = changeme!

[settings]
past_hours = 24
future_hours = 4
```

11. Change the admin password in `/opt/viewpoint/etc/viewpoint.cfg`.



For security purposes, the admin password is encrypted. In the example above, the default is the encrypted equivalent to "changeme!", which is the default for the Viewpoint instance. Change this default password to a different encrypted password. To encrypt the password, do the following (substituting "changeme!" with your password):

```
[root]# echo -n 'changeme!' | /opt/viewpoint/bin/viewpoint makehash
Using default hasher
pbkdf2_sha256$20000$ZHeToCJgrSUH$+xmzYdhpqZCJokxO9eGzyr2B6jrfCgLLBT+pBgMis4w=
```



The default hashing algorithm is `pbkdf2_sha256`. To show the other available algorithms, run `/opt/viewpoint/bin/viewpoint makehash --help`. `bcrypt_sha256` and `bcrypt` are *not* supported on Red Hat 7-based systems.

12. Initialize Viewpoint's PostgreSQL database.

```
[root]# /opt/viewpoint/bin/viewpoint migrate --fake-initial
```

13. Update Viewpoint's configuration file.

```
[root]# /opt/viewpoint/bin/update-viewpoint-config
```

14. Start the Apache service.

- Red Hat 6-based systems

```
[root]# service httpd start
```

- Red Hat 7-based systems

```
[root]# systemctl start httpd.service
```

Upgrade the Viewpoint File Manager Service

On the Moab Server Host where the Viewpoint File Manager Service resides, do the following:

1. Install the `moab-viewpoint-filemanager` package.

```
[root]# yum install moab-viewpoint-filemanager
[root]# yum install python-setuptools
```

2. Restart the File Manager Service.

- Red Hat 6-based systems

```
[root]# service acfileman restart
```

- Red Hat 7-based

```
[root]# systemctl restart acfileman.service
```

License and Configure Viewpoint

The next step in the Viewpoint installation is to obtain the Viewpoint license and provide that license and other configuration information using the Configuration page from the Viewpoint portal.

Do the following:

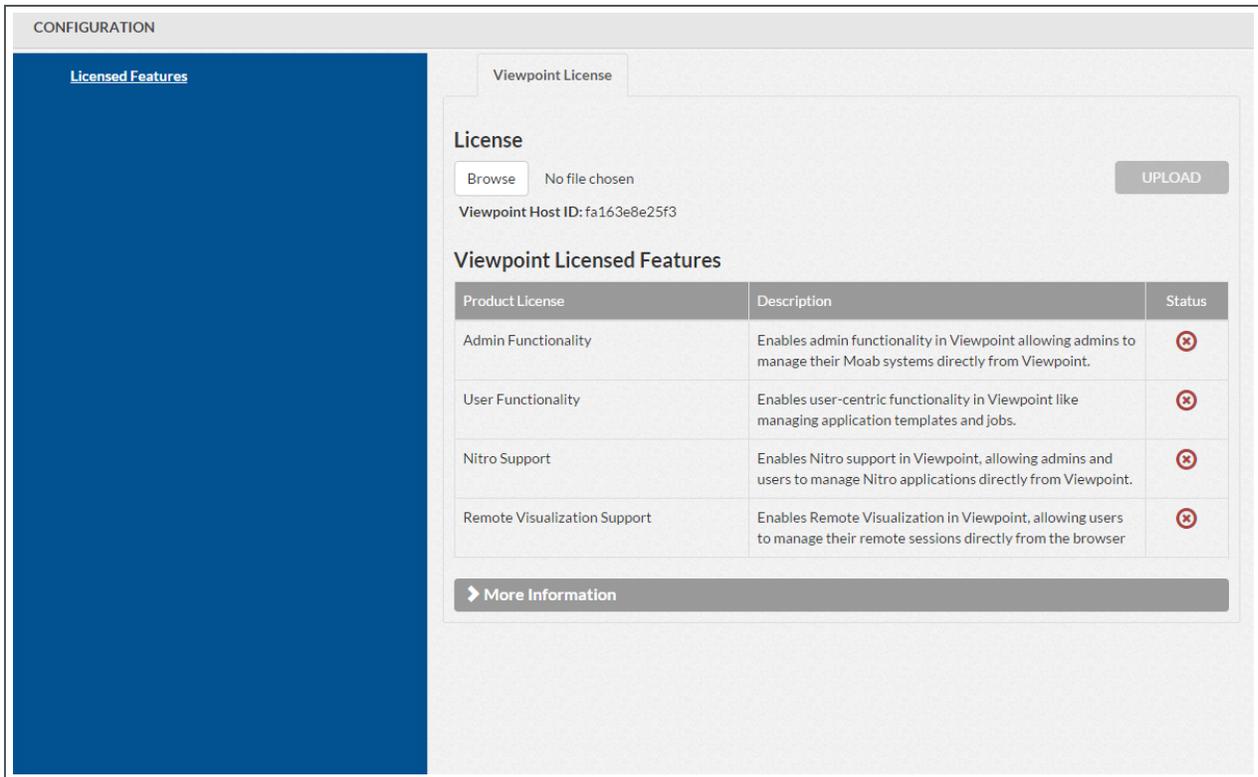
1. Using a web browser, navigate to your Viewpoint instance. (http://<viewpoint_host>:8081; where *<viewpoint_host>* is the IP address or name of the Viewpoint Server Host).

i If you see a "404 ... Page Not Found" error rather than a login page, this is expected since a license is now required to access viewpoint. Use the Sign Out link to return to access the login page.

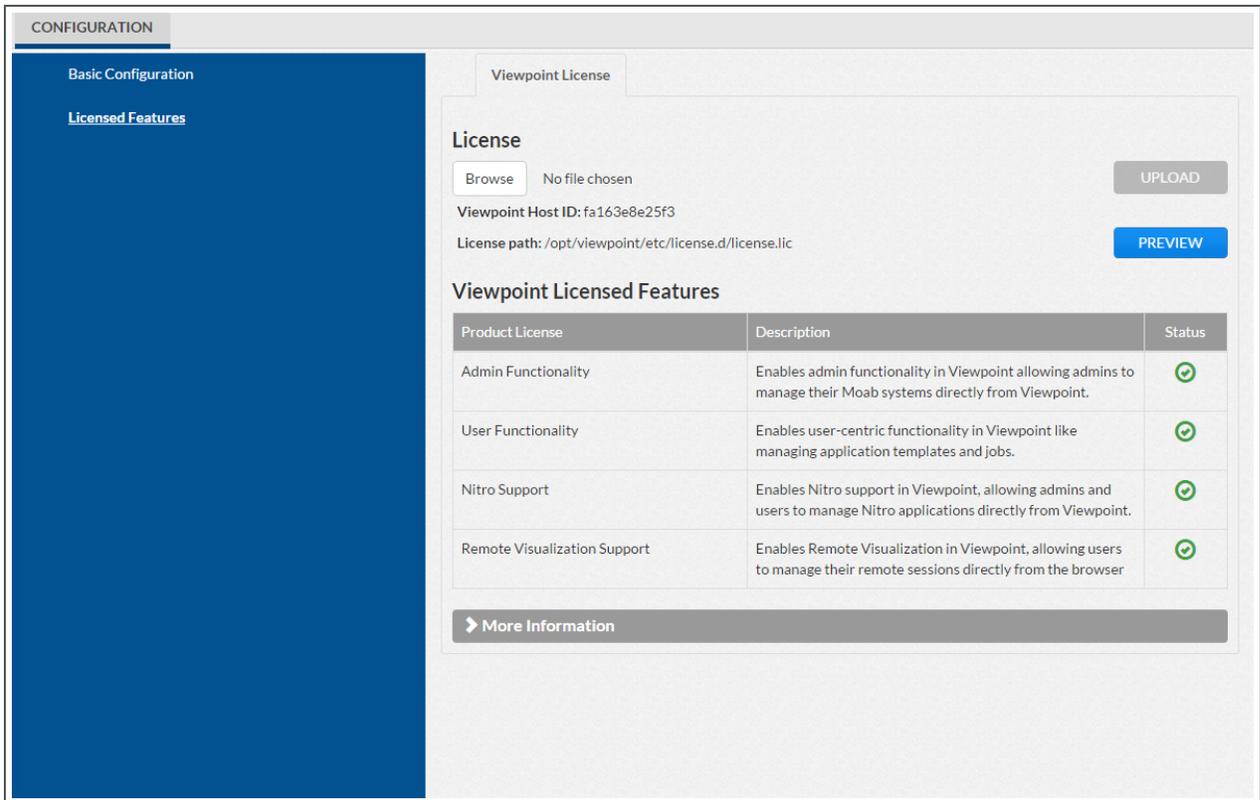
2. Log in as the Viewpoint administrative user (viewpoint-admin, by default) using the password you set in the Viewpoint installation instructions.

i The Viewpoint administrative user has very limited rights: it has permission upload the license file and to set up the connection with MWS.

The Configuration page with *only* the Licensed Features/Viewpoint License information displayed. The following is an example of what you will see once you first log in.



3. In the License section, locate the Viewpoint Host ID.
4. Email licenses@adaptivecomputing.com with that hostid.
5. Adaptive Computing will generate the license and send you the Viewpoint license (.lic) file in a return email.
6. Save the Viewpoint license in a safe location.
7. Go back to your Viewpoint instance and log in again as the Viewpoint administrative user.
8. When the Configuration page with *only* the Licensed Features/Viewpoint License information displays, do the following:
 - a. Click **Browse**, navigate to where you saved the Viewpoint License file, and then click **Open**.
 - b. Click **Upload**.
9. Once the license file has uploaded, the Viewpoint License information shows green check boxes for your licensed features and displays the path to your uploaded license file under the Viewpoint Host ID information. The following is an example of what you will see once the license file is uploaded.



- Click **Preview** to view the contents of the license file you uploaded
- You can also expand the More Information section to see expiration information.
- The Basic Configuration page is now available for you to verify or change MWS, File Manager and other configuration information.

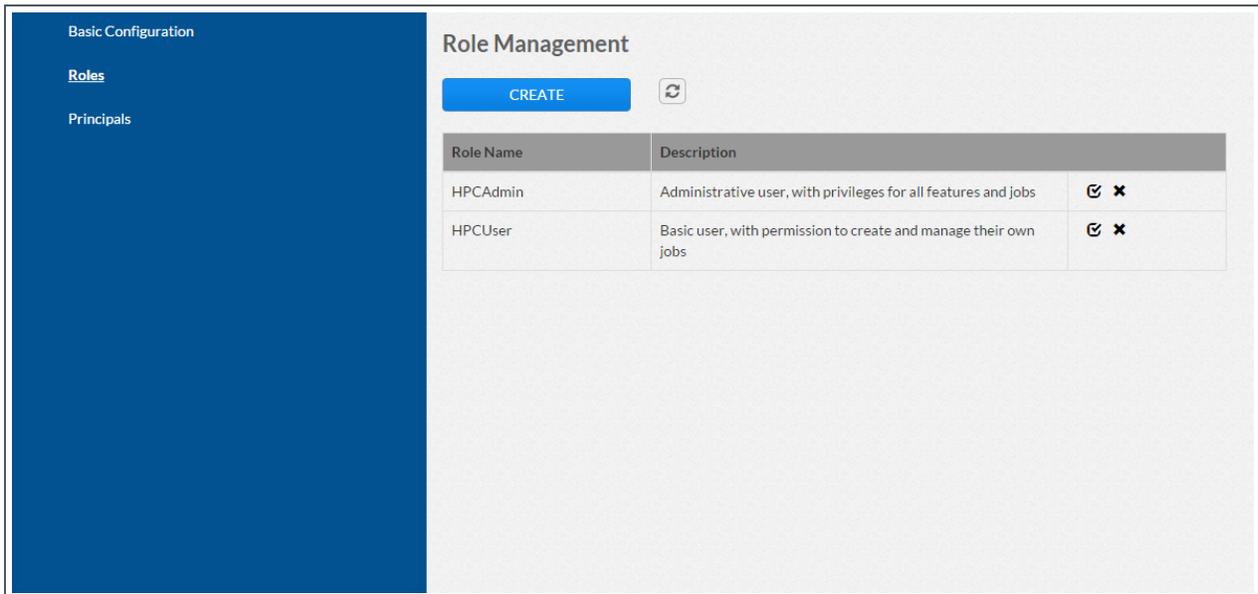
Reset Viewpoint Roles

Once you have successfully upgraded, you will need to reset the Viewpoint roles to reflect changes (for example, new permissions).

As the `MWS administrative` user (`moab-admin`, by default), do the following:

1. Using a web browser, navigate to your Viewpoint instance. (`http://<viewpoint_host>:8081`; where `<viewpoint_host>` is the IP address or name of the Viewpoint Server Host).
2. Log in as the MWS administrative user.
3. Click **Configuration** from the menu. The Basic Configuration page displays.

- Click **Roles** from the left pane. The Role Management page displays.



- Click next one of the default roles (HPCAdmin or HPCUser).
- When the Roles page displays, click **RESET**.
- When a message appears asking to your confirm the reset, click **OK**.

Migrating the MAM Database from MySQL to PostgreSQL

PostgreSQL is the preferred DBMS for MAM. Customers who have already installed MySQL as the DBMS for MAM are not required to migrate their database to use PostgreSQL at this time. However, MySQL is considered deprecated and new installations will only use PostgreSQL.

i PostgreSQL does not provide a standard procedure for migrating an existing database from MySQL to PostgreSQL. Adaptive Computing has had success using the `py-mysql2pgsql` tools for migrating/converting/exporting data from MySQL to PostgreSQL. See <https://github.com/philipsoutham/py-mysql2pgsql> for additional details.

To Migrate the MAM Database

This procedure was successfully tested on an actual customer MySQL database with millions of transactions on CentOS 6.4. It completed in less than an hour.

1. Make a backup copy of your MySQL mam database.

```
[root]# mysqldump mam > /archive/mam.mysql
```

2. Follow the instructions to Install PostgreSQL.

- **Manual Install** - [Install and Initialize the MAM PostgreSQL Server on page 43](#)
- **RPM Install** - [Install and Initialize PostgreSQL Server on page 146](#)

3. Install the prerequisite packages.

```
[root]# yum install git postgresql-devel gcc MySQL-python python-psycopg2 PyYAML
termcolor python-devel
```

4. Install pg-mysql2pgsql (from source).

```
[root]# cd /software
[root]# git clone git://github.com/philipsoutham/py-mysql2pgsql.git
[root]# cd py-mysql2pgsql
[root]# python setup.py install
```

5. Run pg-mysql2pgsql once to create a template yaml config file.

```
[root]# py-mysql2pgsql -v
```

6. Edit the config file to specify the MySQL database connection information and a file to output the result.

```
[root]# vi mysql2pgsql.yml
```

```
mysql:
hostname: localhost
port: 3306
socket:
username: mam
password: changeme
database: mam
compress: false
destination:
# if file is given, output goes to file, else postgres
file: /archive/mam.pgsql
postgres:
hostname: localhost
port: 5432
username:
password:
database:
```

7. Run the pg-mysql2pgsql program again to convert the database.

```
[root]# py-mysql2pgsql -v
```

8. Create the mam database in PostgreSQL.

```
[root]# su - postgres
[postgres]$ psql
postgres=# create database "mam";
postgres=# create user mam with password 'changeme!';
postgres=# \q
[postgres]$ exit
```

9. Import the converted data into the PostgreSQL database.

```
[root]# su - mam
[mam]$ psql mam < /archive/mam.pgsql
```

10. Point MAM to use the new postgresql database.

```
[mam]$ cd /software/mam-latest
[mam]$ ./configure # This will generate an etc/mam-
server.conf.dist file
[mam]$ vi /opt/mam/etc/mam-server.conf # Merge in the database.datasrc from
etc/mam-server.conf.dist
```

11. Restart Moab Accounting Manager.

```
[mam]$ mam-server -r
```

Chapter 4 Troubleshooting

This chapter details some common problems and general solutions. Additional troubleshooting may be found in the individual Moab HPC Suite component documentation.

In this chapter:

- [General Issues on page 266](#)
- [Moab Web Services Issues on page 269](#)
- [Moab Viewpoint Issues on page 273](#)

General Issues

This topic details some common problems and general solutions.

In this topic:

- [Moab error: "cannot determine local hostname" on page 266](#)
- [Moab error: "Moab will now exit due to license file not found" on page 267](#)
- [Other Moab issues on page 267](#)
- [Where do I change my passwords? on page 267](#)

Moab error: "cannot determine local hostname"

```
# service moab start
Starting moab: ERROR:      cannot determine local hostname - node is misconfigured
                        [FAILED]
```

If you encounter this error when starting Moab, check the `/opt/moab/etc/moab.cfg` file to make sure a valid host is configured. For example:

```
...
SCHEDCFG [Moab]                SERVER=<moab-hostname>:42559
...
```

Also check `/etc/hosts` to be sure the host name resolves, at least with localhost:

```
...
127.0.0.1  <moab-hostname> localhost localhost.localdomain localhost4
localhost4.localdomain4
...
```

Moab error: "Moab will now exit due to license file not found"

```
# service moab start
Starting moab: Moab will now exit due to license file not found
Please contact Adaptive Computing (sales@adaptivecomputing.com) to get a license for
your system
[FAILED]
```

If you encounter this error when starting Moab, make sure your Moab license file is named **moab.lic** and is located in the `/opt/moab/etc/` directory.

Also make sure the license is not expired. The expiration date is listed in the license file. For example:

```
# cat /opt/moab/etc/moab.lic
...
# Expires after Tue Dec 31 10:43:46 2013
...
```

Other Moab issues

See Troubleshooting and System Maintenance in the *Moab Workload Manager Administrator Guide*.

Where do I change my passwords?

In this section:

- [Moab Super User Username and Password on page 267](#)
- [MongoDB Passwords on page 268](#)

Moab Super User Username and Password

The default username and password for Moab are **moab-admin** and **changeme!** (respectively).

To change the username and/or the password for the Moab super user.

1. Stop the `tomcat` and `moab` services.

```
[root]# service moab stop
[root]# service tomcat stop
```

2. Change the respective values in the following files:

- `/opt/mws/etc/mws-config.groovy`:

```
auth.defaultUser.username = "moab-admin"
auth.defaultUser.password = "changeme!"
```

- `/opt/moab/etc/moab-private.cfg`:

```
CLIENTCFG[RM:mws] USERNAME=moab-admin PASSWORD=changeme!
```

- /opt/moab/etc/cloud.cfg:

```
CONFIG[default]      MWS_USERNAME=moab-admin
CONFIG[default]      MWS_PASSWORD=changeme!
```

3. Start the tomcat service.

```
[root]# service tomcat start
```

4. Start the moab service.

```
[root]# service moab start
```

MongoDB Passwords

To change the passwords for MongoDB:

1. Stop the tomcat and moab services.

```
[root]# service moab stop
[root]# service tomcat stop
```

2. Change the passwords for the MongoDB accounts (i.e., **moab_user** and/or **mws_user**). See the [MongoDB documentation](http://docs.mongodb.org/manual/tutorial/change-user-password/) (<http://docs.mongodb.org/manual/tutorial/change-user-password/>) for detailed instructions.

3. Edit the password values in the following files:

- /opt/moab/etc/moab-private.cfg:

```
MONGOUSER          moab_user
MONGOPASSWORD       secret2
```

- /opt/mws/etc/mws-config.groovy:

```
// MongoDB configuration.
grails.mongo.username = "mws_user"
grails.mongo.password = "secret3"
```

4. Start the tomcat service.

```
[root]# service tomcat start
```

5. Start the moab service.

```
[root]# service moab start
```

Moab Web Services Issues

This topic details some common problems and general solutions for Moab Web Services.

If something goes wrong with MWS, look in the following files:

- The MWS log file. By default this is `/opt/mws/log/mws.log`.
- The Tomcat `catalina.out` file, usually in `/var/log/tomcat` or `$CATALINA_HOME/logs`.

i If you remove the `log4j` configuration from `mws-config.groovy`, MWS writes its log files to `java.io.tmpdir`. For Tomcat, `java.io.tmpdir` is generally set to `$CATALINA_BASE/temp` or `CATALINA_TMPDIR`.

In this topic:

- [MongoDB: Errors during MWS startup on page 269](#)
- [MongoDB: Out of semaphores to get db connection on page 271](#)
- [MongoDB: Connection wait timeout after 120000 ms on page 271](#)
- [java.lang.OutOfMemoryError: Java heap space on page 271](#)
- [java.lang.OutOfMemoryError: PermGen space on page 272](#)
- [SEVERE: Context \[/mws\] startup failed due to previous errors on page 272](#)
- [MoabReached Maximum Number of Concurrent Client Connections on page 272](#)

MongoDB: Errors during MWS startup

If the application fails to start and gives error messages such as these:

```
Error creating bean with name 'mongoDatastore'
can't say something; nested exception is com.mongodb.MongoException
```

```
ERROR   rails.app.services.com.ace.mws.ErrorService    0
        Error encountered while attempting to authenticate account or query database; the
        MongoDB server is not available. Please verify connection to server '/127.0.0.1:27017'
        and that MongoDB is running.
```

MongoDB is most likely not running, or the MongoDB host and port are misconfigured.

In this case, there are a few things to verify:

- (Not relevant if MongoDB is installed on a different host) **Is MongoDB installed?**

Run the following commands to assess whether MongoDB is installed on the current host.

```
$ mongo
-bash: mongo: command not found
```

To remedy, install MongoDB, start the `mongod` service and then restart the `tomcat` service. See [Install MongoDB on page 57](#) (Manual Installation) or [Install MongoDB on page 157](#) (RPM Installation) for more information on how to install and configure MongoDB.

- (Only relevant if MongoDB is installed on a different host) **Is MWS configured to connect to the remote MongoDB host?**

Run the following commands to assess whether MongoDB is installed on the current host.

```
[root]# cat /opt/mws/etc/mws-config.groovy | grep 'grails.mongo'
// grails.mongo.username = "mws_user"
// grails.mongo.password = "<ENTER-KEY-HERE>"
// grails.mongo.host = "127.0.0.1"
// grails.mongo.port = 27017
```

Make sure that the `grails.mongo.*` options are configured in `/opt/mws/etc/mws-config.groovy` for the remote MongoDB server and then restart the `tomcat` service.

```
[root]# service tomcat restart
```

- **Is MWS configured to authenticate with MongoDB, and is MongoDB configured to enforce authentication?**

Run the following commands to assess the relevant MWS and MongoDB configurations.

```
[root]# cat /opt/mws/etc/mws-config.groovy | grep 'grails.mongo'
// grails.mongo.username = "mws_user"
// grails.mongo.password = "<ENTER-KEY-HERE>"

[root]# cat /etc/mongod.conf | grep 'auth'
#noauth = true
auth = true
```

The configuration above is problematic because the `grails.mongo` credentials are commented out in the `/opt/mws/etc/mws-config.groovy` file while MongoDB is configured to enforce authentication (`"auth = true"`). Similar connection issues will exist if the `grails.mongo` parameters do not match the credentials configured for the `"mws_user"` on both the `mws` and `moab` databases in MongoDB.

(For upgrade scenarios only) If the application fails to start and gives the following message in `/opt/mws/etc/log/mws.log`:

```
java.lang.Exception: The db-migrate.js script has not yet been run. Please see the
upgrade section of the installation guide for instructions.
```

Then the `db-migrate.js` script must be run to update the schema of the `mws` database in MongoDB.

MongoDB: Out of semaphores to get db connection

To resolve this error, adjust the values of `connectionsPerHost` or `threadsAllowedToBlockForConnectionMultiplier` by adding them to `mws-config.groovy`. For example:

```
grails.mongo.options.connectionsPerHost = 60
grails.mongo.options.threadsAllowedToBlockForConnectionMultiplier = 10
```

For more information on these options, refer to these documents:

- [Configuring Moab Web Services](#) in the *Moab Web Services Administrator Guide*, which briefly discusses a few MongoDB driver options.
- The [MongoOptions](#) documentation (<http://api.mongodb.org/java/current/com/mongodb/MongoOptions.html>), which contains full details on all MongoDB driver options.

i You must restart Tomcat after adding, removing, or changing **grails.mongo.options** parameters.

As shipped, `mws-config.groovy` does not contain any **grails.mongo.options** parameters. To adjust their values, you need to add them to `mws-config.groovy`.

The default value of **connectionsPerHost** is normally 10, but MWS sets it internally to 50.

The default value of **threadsAllowedToBlockForConnectionMultiplier** is 5.

Any of the options listed in `MongoOptions` can be specified in `mws-config.groovy`. Just use the prefix **grails.mongo.options** as shown above.

MongoDB: Connection wait timeout after 120000 ms

See [MongoDB: Out of semaphores to get db connection](#) above.

java.lang.OutOfMemoryError: Java heap space

Increase the size of the heap using JVM options **-Xms** and **-Xmx**. Here are the suggested values:

```
CATALINA_OPTS="-DMWS_HOME=/opt/mws -Xms256m -Xmx3g -XX:MaxPermSize=384m"
```

- **-Xms**: Set initial Java heap size.
- **-Xmx**: Set maximum Java heap size.

i Beginning with Java 8, the `MaxPermSize` option is ignored.

java.lang.OutOfMemoryError: PermGen space

(Recommended) Upgrade to Java. Java 8 has completely removed PermGen space and the `MaxPermSize` option is ignored.

For Java version prior to 8, you can increase the size of the permanent generation using JVM option **-XX:MaxPermSize**. Here are the suggested values:

```
CATALINA_OPTS="-DMWS_HOME=/opt/mws -Xms256m -Xmx3g -XX:MaxPermSize=384m"
```

SEVERE: Context [/mws] startup failed due to previous errors

If `catalina.out` contains this error, look in `/opt/mws/log/mws.log` and `/opt/mws/log/stacktrace.log` for more details on the error.

Also ensure that the `/opt/mws/etc/mws-config.groovy` file can be read by the Tomcat user. The permissions should appear as follows:

```
$ ls -al /opt/mws/etc/mws-config.groovy
-r----- 1 tomcat tomcat 4056 Dec  4 12:07 mws-config.groovy
```

MoabReached Maximum Number of Concurrent Client Connections

When this error message is encountered, simply add a new line to the `moab.cfg` file:

```
CLIENTMAXCONNECTIONS 256
```

This will change the Moab configuration when Moab is restarted. Run the following command to immediately use the new setting:

```
[root]# changeparam CLIENTMAXCONNECTIONS 256
```

i The number **256** above may be substituted for the desired maximum number of Moab client connections.

Moab Viewpoint Issues

This topic details some common problems and general solutions for Moab Viewpoint.

In this topic:

- [Viewpoint does not report any of my jobs or nodes on page 273](#)

Viewpoint does not report any of my jobs or nodes

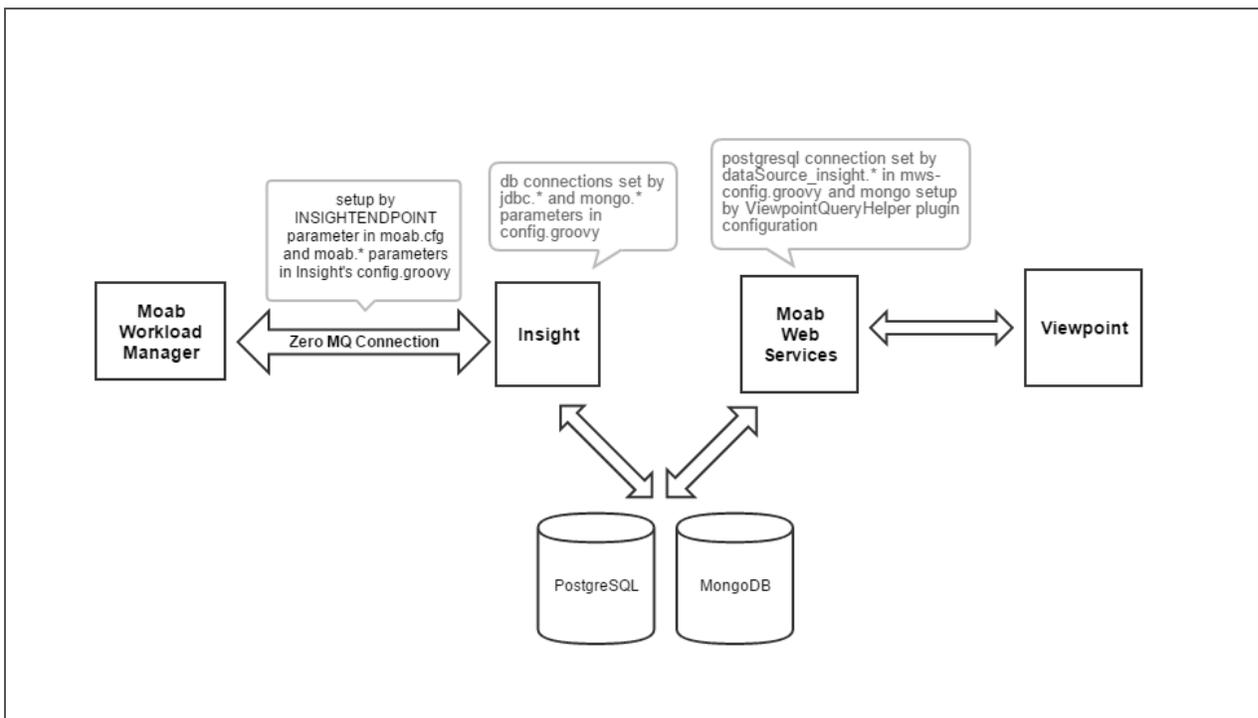
There are multiple reasons why jobs and nodes might not show up in Viewpoint.

Verify the following:

1. Moab HPC Suite Setup

Essentially, there are many communication points in our stack from the point that jobs get submitted to the point they get displayed in Viewpoint.

Please take a look at the following diagram describing our data flow architecture:



The Moab Workload Manager will push data into Insight using a ZeroMQ message queue pipe.

Then, Insight will parse that data and insert it into two distinct databases: a PostgreSQL relational database and a NoSQL MongoDB database.

When Viewpoint needs to query information on jobs and nodes, it will communicate with Moab Web Services, which in turn will consume the data directly from the databases (PostgreSQL and MongoDB) where Insight recorded Moab's events.

Failure to configure the communication channels between all these components will result in Viewpoint not being able to display job or node information.

2. **Hardware Specifications**

Another reason why Viewpoint might not be able to show job and node information is that you installed all Moab HPC components in a single machine that is too overloaded.

See [Server Hardware Requirements on page 4](#) for more information.

3. **RPM Versions**

One other common problem customers can experience is that they install incompatible versions of our software components.

Please make sure you are using the same major version across all components (e.g. Moab Workload Manager 9, Moab Web Services 9, Insight 9, etc.).