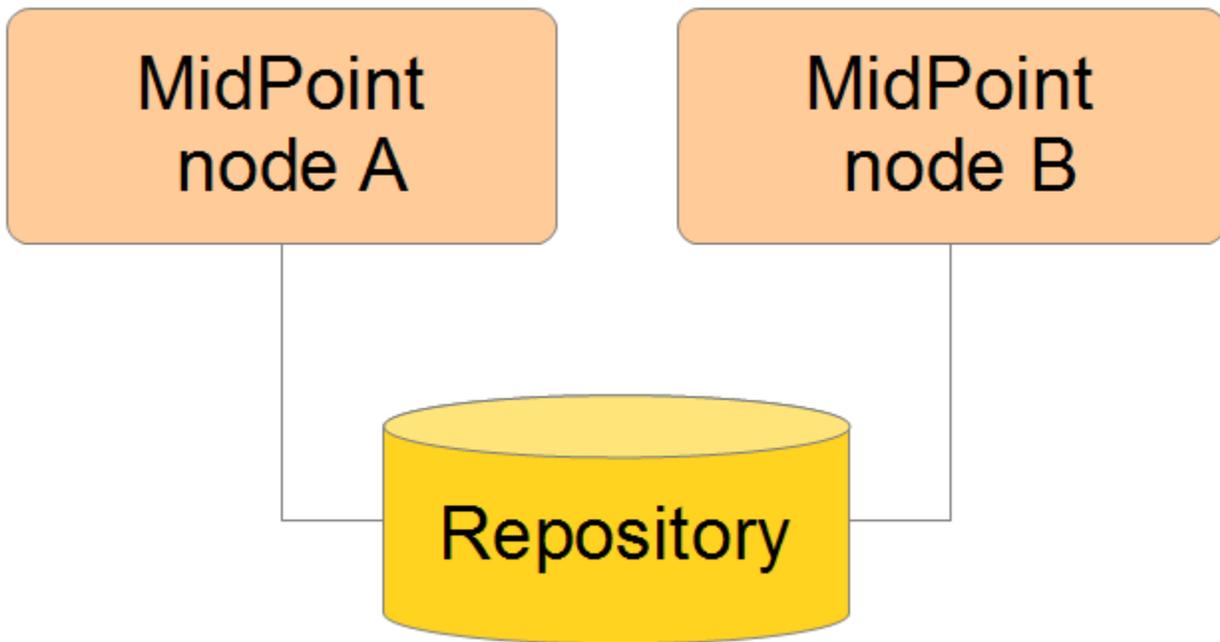


# Clustering / high availability setup

- [Introduction](#)
- [Other cluster configuration items \(midPoint 4.0 and above\)](#)
  - [How intra-cluster URL is determined](#)
- [Configuring the cluster before midPoint 4.0](#)
  - [Cluster infrastructure configuration](#)
  - [Troubleshooting JMX](#)
- [Using H2 when clustered](#)

## Introduction

[Clustering / high availability](#) can be achieved by setting up two midPoint nodes working against common midPoint repository.



In order to do this, it is necessary to set a couple of parameters in midPoint configuration.

An example, when using PostgreSQL database:

```
<repository>
  <repositoryServiceFactoryClass>com.evolveum.midpoint.repo.sql.SqlRepositoryFactory<
/repositoryServiceFactoryClass>
  <database>postgresql</database>
  <jdbcUsername>midpoint</jdbcUsername>
  <jdbcPassword>.....</jdbcPassword>
  <jdbcUrl>jdbc:postgresql://.../midpoint</jdbcUrl>
  <hibernateHbm2ddl>none</hibernateHbm2ddl>
  <missingSchemaAction>create</missingSchemaAction>
</repository>
<taskManager>
  <clustered>true</clustered>
</taskManager>
```

Typically you set the following configuration parameter only:

Parameter	Description
clustered	Determines if the installation is running in clustered (failover) mode. Default is false. If you need clustering/failover, set this to true.

In some circumstances the Quartz component in task manager needs to use separate database (usually only if you want to run it on H2). Then you need to configure that properly; see the section on H2 use below.

**Important:** if there are more nodes sharing a repository, all of them must have parameter `clustered` set to `true`. Otherwise, tasks will not be scheduled correctly. (`midPoint` will disable scheduling tasks on non-conformant nodes, i.e. on non-clustered nodes that are parts of such a system.) The best way how to ensure this is to have common config file. But if that's not possible or practical, make sure that all nodes have the same settings.

Also, ensure your **system time is synchronized** across all node members (using NTP or similar service), otherwise strange behaviour may occur such as tasks restart on different nodes.

## Other cluster configuration items (midPoint 4.0 and above)

The main difference between 4.0 and previous versions is that since `midPoint 4.0` the main mechanism for clusterwide task management is REST instead of JMX. (Although JMX is still supported, its use is now deprecated and users are strongly advised to switch to REST.) Configuration items are:

Command-line parameter	Configuration parameter	Meaning	Note
<code>-Dmidpoint.nodeld</code>	<code>&lt;nodeld&gt;</code>	The node identifier. The default is <code>DefaultNode</code> for non-clustered deployments. For clustered ones, either <code>nodeId</code> or <code>nodeIdSource</code> must be used.	
<code>-Dmidpoint.nodeldSource</code>	<code>&lt;nodeldSource&gt;</code>	Source of the node identifier. It is applied if explicit node ID is not defined. The source can be either <code>hostname</code> meaning that the host name is used as the node identifier or <code>random</code> meaning that the random value for node ID is generated when the node is started.	
<code>-Dmidpoint.hostname</code>	<code>&lt;hostname&gt;</code>	Overrides the local host name information. If not specified, the operation system is used to determine the host name.	Normally you do not need to specify this information.
<code>-Dmidpoint.httpPort</code>	<code>&lt;httpPort&gt;</code>	Overrides the local HTTP port information. If not specified, Tomcat/Catalina JMX objects are queried to determine the HTTP port information. This information is used only to construct URL address used for intra-cluster communication (see below).	Normally you do not need to specify this information.
<code>-Dmidpoint.url</code>	<code>&lt;url&gt;</code>	Overrides the intra-cluster URL information (see below).	Normally you do not need to specify this information.

Note that you can either use the command line options (`-Dxxx=yyy`) to set these parameters or you can use `config.xml` to do that. In the latter case the configuration parameters are to be specified directly under the `<midpoint>` element.

### How intra-cluster URL is determined

In order to minimize the configuration work needed while keeping the maximum level of flexibility, the node URL used for intra-cluster communication (e.g. <https://node1.acme.org:8080/midpoint>) is derived from the following items - in this order:

- `<urlOverride>` property in the Node object in the repository
- `-Dmidpoint.url / <url>` information in command line or `config.xml` file
- computed based on information in `infrastructure/intraClusterHttpUrlPattern` property, if defined; that property can use the following macros:
  - `$host` for host name (obtained dynamically from OS or overridden via `-Dmidpoint.hostname` or `<hostname>` config property)
  - `$port` for HTTP port (obtained dynamically from Tomcat JMX objects or overridden via `-Dmidpoint.httpPort` config property)
  - `$path` for `midPoint` URL path (obtained dynamically from the servlet container)
- computed based on protocol scheme (obtained dynamically from Tomcat JMX objects), host name, port, and servlet path, as `scheme://host:port/path`.

When troubleshooting these mechanisms you can set logging for `com.evolveum.midpoint.task.quartzimpl.cluster.NodeRegistrar` (or the whole task manager module) to `DEBUG`.

## Configuring the cluster before midPoint 4.0

Mainly because of JMX limitations, some parameters have to be set up via Java system properties. In the following we expect the Oracle JRE is used.

Parameter	Meaning
<code>midpoint.nodeld</code>	This is an identifier of the local node. It is not part of the <code>midPoint</code> configuration, because we assume that this configuration file will be shared among cluster members. The default value is: <code>DefaultNode</code> . However, when running in clustered mode, there is no default, and this property <b>must be</b> explicitly specified.
<code>midpoint.jmxHostName</code>	Host name on which this node wants to be contacted (via JMX) by other nodes in cluster. (It will be announced to other nodes via Node record in repository.) Usually not necessary to specify, as the default is the current host IP address.

com.sun.management.jmxremote.port	This is the port on which JMX agent will listen. It <b>must be specified</b> for clustered mode, because JMX is used to query status of individual nodes and to manage them (start/stop scheduler, stop tasks on that node). And, if you test a clustering/failover configuration (more midPoint nodes) on a single machine, be sure to set this parameter to <b>different</b> values for individual midPoint nodes. Otherwise, you will get " <a href="#">java.net.BindException: Address already in use: JVM_Bind</a> " exception on tomcat startup.
com.sun.management.jmxremote.ssl	Whether SSL will be used for JMX communication. For sample installations it can be set to <code>false</code> , however, <b>for production use we recommend setting it to <code>true</code></b> (alongside other SSL-related JMX properties, see <a href="http://docs.oracle.com/javase/1.5.0/docs/guide/management/agent.html#remote">http://docs.oracle.com/javase/1.5.0/docs/guide/management/agent.html#remote</a> ).
com.sun.management.jmxremote.password.file and com.sun.management.jmxremote.access.file	Names of the password and access files for JMX authentication and authorization. E.g. <code>d:\midpoint\config\jmxremote.password</code> , <code>d:\midpoint\config\jmxremote.access</code> . Examples of these files are in the <code>samples/jmx</code> directory in SVN.  <b>Beware, the <code>jmxremote.password</code> file must be readable only to its owner (i.e. user who starts the tomcat), otherwise the JVM refuses to start.</b> In Windows, you typically have to stop inheriting permissions to this file, and manually remove all entries that grant access to persons other than the owner.

Also, the following configuration items in `<taskManager>` section of `config.xml` have to be set:

Parameter	Meaning
jmxUsername, jmxPassword	Credentials used for JMX communication among cluster nodes. Default values are <code>midpoint</code> and <code>secret</code> respectively, but we strongly recommend changing at least the JMX password. Currently, all nodes should be accessible using the same credentials.

### An example

#### NodeA (in `catalogina.bat`)

```
SET CATALINA_OPTS=-Dmidpoint.nodeId=NodeA \
-Dmidpoint.home=d:\midpoint\home \
-Dcom.sun.management.jmxremote=true \
-Dcom.sun.management.jmxremote.port=20001 \
-Dcom.sun.management.jmxremote.ssl=false \
-Dcom.sun.management.jmxremote.password.file=d:\midpoint\home\jmxremote.password \
-Dcom.sun.management.jmxremote.access.file=d:\midpoint\home\jmxremote.access
```

#### NodeB (in `catalogina.bat`)

```
SET CATALINA_OPTS=-Dmidpoint.nodeId=NodeB \
-Dmidpoint.home=d:\midpoint\home \
-Dcom.sun.management.jmxremote=true \
-Dcom.sun.management.jmxremote.port=20002 \
-Dcom.sun.management.jmxremote.ssl=false \
-Dcom.sun.management.jmxremote.password.file=d:\midpoint\home\jmxremote.password \
-Dcom.sun.management.jmxremote.access.file=d:\midpoint\home\jmxremote.access
```

(Note: the jmx port is set to 20002 just to allow running both nodes on a single machine. If you are sure they will not be run on a single machine, we recommend setting the port to the same value, just for simplicity.)

(Note: when you have firewall, please also set `com.sun.management.jmxremote.urlport` to the same port as `com.sun.management.jmxremote.port`).

## Cluster infrastructure configuration

Even if 3.9 and below there are some types of information (e.g. reports) that are accessed using REST calls. So, midpoint needs to have an intra-cluster HTTP URL pattern specified. This should be the HTTP/HTTPS pattern which is used by midpoint nodes to communicate with each others. The pattern is in fact an URL prefix pointing to the root URL of the application. The pattern is specified in the system configuration object as present in the example below.

```
<systemConfiguration>
...
<infrastructure>
<intraClusterHttpUrlPattern>https://$host/midpoint</intraClusterHttpUrlPattern>
</infrastructure>
...
</systemConfiguration>
```

## Troubleshooting JMX

The following message(s) may appear in idm.log if there is a problem with JMX password:

```
2014-03-04 14:05:31,692 [TASKMANAGER] [http-bio-8080-exec-3] TRACE (com.evolveum.midpoint.task.quartzimpl.execution.ExecutionManager): Getting node and task info from the current node (Tomcat7_Node1)
2014-03-04 14:05:31,693 [TASKMANAGER] [http-bio-8080-exec-3] DEBUG (com.evolveum.midpoint.task.quartzimpl.execution.ExecutionManager): Getting running task info from remote node (Tomcat7_Node2, 127.0.1.1)
2014-03-04 14:05:31,700 [MODEL] [http-bio-8080-exec-3] ERROR (com.evolveum.midpoint.model.controller.ModelController): Couldn't search objects in task manager, reason: Authentication failed! Invalid username or password
2014-03-04 14:05:31,701 [] [http-bio-8080-exec-3] ERROR (com.evolveum.midpoint.web.page.admin.server.dto.NodeDtoProvider): Unhandled exception when listing nodes, reason: Subresult com.evolveum.midpoint.task.api.TaskManager..searchObjects of operation com.evolveum.midpoint.model.controller.ModelController.searchObjects is still UNKNOWN during cleanup; during handling of exception java.lang.SecurityException: Authentication failed! Invalid username or password
```

In that case, double-check your JMX passwords (in config.xml and in jmx.remote.password files) in all instances.

The following message(s) may appear in idm.log if there is problem with firewall between IDM nodes:

```
2014-05-26 09:07:38,438 [TASKMANAGER] [http-bio-8181-exec-1] ERROR (com.evolveum.midpoint.task.quartzimpl.execution.RemoteNodesManager): Cannot connect to the remote node node02 at 10.1.1.2:8123, reason: Failed to retrieve RMIServer stub: javax.naming.CommunicationException [Root exception is java.rmi.ConnectIOException: Exception creating connection to: 10.1.1.2; nested exception is: java.net.NoRouteToHostException: No route to host]
```

Please note that it seems that JMX communication needs more than the JMX port specified in Tomcat startup configuration (in this fragment, 8123)! I resolved the problem by simply allowing all TCP communication between the nodes. I will update this solution after I find a better one 🙄

The following message may appear if your clock is not synchronized between midPoint nodes:

```
2014-05-26 00:45:32,818 [TASKMANAGER] [QuartzScheduler_midPointScheduler-node02_ClusterManager] WARN (org.quartz.impl.jdbcjobstore.JobStoreTX): This scheduler instance (node02) is still active but was recovered by another instance in the cluster. This may cause inconsistent behavior.
```

## Using H2 when clustered

Using H2 in clustered mode is **not** recommended because of needless complexity. First, it needs to be specified to run in standalone process. And second, Quartz and midPoint need to use separate MVCC-related settings.

An example:

```
<repository>
  <repositoryServiceFactoryClass>com.evolveum.midpoint.repo.sql.SqlRepositoryFactory<
/ repositoryServiceFactoryClass>
  <baseDir>${midpoint.home}</baseDir>
  <embedded>>false</embedded>
  <asServer>>true</asServer>
  <driverClassName>org.h2.Driver</driverClassName>
  <jdbcUsername>sa</jdbcUsername>
  <jdbcPassword></jdbcPassword>
  <jdbcUrl>jdbc:h2:tcp://localhost:6000/~ /midpoint;LOCK_MODE=1;DB_CLOSE_ON_EXIT=FALSE;LOCK_TIMEOUT=10000<
/ jdbcUrl>
  <hibernateDialect>org.hibernate.dialect.H2Dialect</hibernateDialect>
  <hibernateHbm2ddl>update</hibernateHbm2ddl>
</repository>
<taskManager>
  <clustered>>true</clustered>
  <jdbcUrl>jdbc:h2:tcp://localhost:6000/~ /midpoint-quartz;MVCC=TRUE;DB_CLOSE_ON_EXIT=FALSE</jdbcUrl>
  <jmxUsername>midpoint</jmxUsername>
  <jmxPassword>secret</jmxPassword>
</taskManager>
```

The following task manager settings are relevant in this context:

Parameter	Meaning
jdbcUrl	If you are using H2, you have to set up use different database parameters from those used by midPoint repository. And, because <a href="#">MVCC mode</a> is to be enabled, task manager has to use a database instance different from the one used by the repository.  (If you are using database other than H2, you may skip setting special jdbcUrl in the <taskManager> configuration. The jdbcUrl from repository config will be used. As a result, Quartz tables will be stored in the same database instance as midPoint tables.)
dataSource	Uses specified data source to obtain DB connections. (See <a href="#">Repository Configuration</a> ).

Other task manager database settings (e.g. jdbc username and password, driver class name, hibernate dialect) are taken by default from <repository> configuration, but, of course, they may be overridden in task manager configuration.

H2 then has to be started independently of both nodes. In this case, it is expected to listen on port 6000. To do that, you can use e.g. this command line:

```
java -jar h2-1.3.171.jar -tcp -tcpPort 6000 -tcpAllowOthers
```