Automation Configuration

On this page

Overview

The Automation Agent uses an automation configuration to determine the desired state of a MongoDB deployment and to effect changes as needed. If you modify the deployment through the Ops Manager web interface, you never need manipulate this configuration.

If you are using the Automation Agent without Ops Manager, you can construct and distribute the configuration manually.

Optional fields are marked as such.

A field that takes a `<number>` as its value can take integers and floating point numbers.

Configuration Version

This lists the version of the automation configuration.

```
"version" : <integer>
```

Name	Type	Description
`version`	integer	The version of the configuration.

Download Base

The download base is the path to the directory where automatic version downloads will be targeted and scripts for starting processes will be created.

```
"options" : {
    "downloadBase" : <string>,
    "downloadBaseWindows" : <string>
}
```

Name	Type	Description
`options`	object	The `options` object is required and must contain both the `downloadBase` and `downloadBaseWindows` fields.
`options.downloadBase`	string	The directory on Linux and Unix (including Mac OS X) platforms for automatic version downloads and startup scripts.
`options.downloadBaseWindows`	string	The directory on Windows platforms for automatic version downloads and startup scripts.

MongoDB Versions Specifications

The `mongoDbVersions` array defines specification objects for the MongoDB instances found in the `processes` array. Each MongoDB instance in the `processes` array must have a specification object in this array.

```
"mongoDbVersions" : [
    {
        "name" : <string>,
        "builds" : [
            {
                "platform" : <string>,
                "url" : <string>,
                "gitVersion" : <string>,
                "modules" : [ <string>, ... ],
                "architecture" : <string>,
                "bits" : <integer>,
                "win2008plus" : <Boolean>,
                "winVCRedistUrl" : <string>,
                "winVCRedistOptions" : [ <string>, ... ],
                "winVCRedistDll" : <string>,
                "winVCRedistVersion" : <string>
            },
            ...
        ],
    },
    ...
]
```

Name	Type	Description

Name	Type	Description
`mongoDbVersions`	array of objects	The `mongoDbVersions` array is required and defines specification objects for the MongoDB instances found in the `processes` array. Each MongoDB instance in `processes` must have a specification object in `mongoDbVersions`.
`mongoDbVersions.name`	string	The name of the specification object. The specification object is attached to a MongoDB instance through the instance's `processes.version` field in this configuration.
`mongoDbVersions.builds`	array of objects	Objects that define the builds for this MongoDB instance.
`mongoDbVersions.builds.platform`	string	The platform for this MongoDB instance.
`mongoDbVersions.builds.url`	string	The URL from which to download MongoDB for this instance.
`mongoDbVersions.builds.gitVersion`	string	The commit identifier that identifies the state of the code used to build the MongoDB process. The MongoDB buildInfo command returns the gitVersion identifier.
`mongoDbVersions.builds.modules`	array	The list of modules for this version. Corresponds to the `modules` field returned by MongoDB 3.2+ `buildInfo` command.
`mongoDbVersions.builds.architecture`	string	The processor's architecture. Possible values are `amd64` or `ppc64le`.
`mongoDbVersions.builds.bits`	integer	*Deprecated*. The processor's bus width. Do not remove or make modifications to this field.
`mongoDbVersions.builds.win2008plus`	Boolean	*Optional*. Set to `true` if this is a Windows build that requires either Windows 7 later or Windows Server 2008 R2 or later.
`mongoDbVersions.builds.winVCRedistUrl`	string	*Optional*. The URL from which the required version of the Microsoft Visual C++ redistributable can be downloaded.
`mongoDbVersions.builds.winVCRedistOptions`	array	*Optional*. String values that list the command-line options to be specified when running the Microsoft Visual C++ redistributable installer. Each command-line option is a separate string in the array.

Name	Type	Description
`mongoDbVersions.builds.winVCRedistDll`	string	*Optional*. The name of the Microsoft Visual C++ runtime DLL file that the agent will check to determine if a new version of the Microsoft Visual C++ redistributable is needed.
`mongoDbVersions.builds.winVCRedistVersion`	string	*Optional*. The minimum version of the Microsoft Visual C++ runtime DLL that must be present to skip over the installation of the Microsoft Visual C++ redistributable.

Automation Agent

The `agentVersion` object is optional and specifies the version of Automation Agent.

```
"agentVersion" : {
    "name" : <string>,
    "directoryUrl" : <string>
}
```

Name	Type	Description
`agentVersion`	object	*Optional* The version of the Automation Agent to run. If the running version does not match this setting, the Automation Agent downloads the specified version, shuts itself down, and starts the new version.
`agentVersion.name`	string	The desired version of the Automation Agent (e.g. “1.8.1.1042-1”).
`agentVersion.directoryUrl`	string	The URL from which to download Automation Agent.

Monitoring Agent

The `monitoringVersions` array is optional and specifies the version of the Monitoring Agent.

```
"monitoringVersions" : [
    {
        "name" : <string>,
        "hostname" : <string>,
        "urls" : {
            <platform1> : {
                <build1> : <string>,
                ...,
                "default" : <string>
            },
            ...
        },
        "baseUrl" : <string>,
        "logPath" : <string>,
        "logRotate" : {
            "sizeThresholdMB" : <number>,
            "timeThresholdHrs" : <integer>,
            "numUncompressed": <integer>,
            "percentOfDiskspace" : <number>
        }
    },
    ...
]
```

Name	Type	Description
`monitoringVersions`	array of objects	*Optional*. Objects that define version information for each Monitoring Agent.
`monitoringVersions.name`	string	The desired version of the Monitoring Agent (e.g. “2.9.1.176-1”). For MongoDB compatibility with Automation, see MongoDB Compatibility.
`monitoringVersions.hostname`	string	The hostname of the machine that runs the Monitoring Agent. If the Monitoring Agent is not running on the machine, Ops Manager installs the agent from the location specified in `monitoringVersions.urls`.
`monitoringVersions.urls`	object	The platform- and build-specific URLs from which to download the Monitoring Agent.

Name	Type	Description
`monitoringVersions.urls.<platform>`	object	This field has a name that identifies an operating system and optionally a version. The field contains an object with key-value pairs, where each key is either the name of a build or `default` and each value is a URL for downloading the Monitoring Agent. The object must include the `default` key set to the default download URL for the platform.
`monitoringVersions.baseUrl`	string	The base URL used for the `mmsBaseUrl` setting in the Monitoring Agent Configuration.
`monitoringVersions.logPath`	string	*Optional*. The directory where the agent stores its logs. The default is to store logs in `/dev/null`. To update, see the monitoringAgentConfig endpoint.
`monitoringVersions.logRotate`	object	*Optional*. Enables log rotation for the MongoDB logs for a process. To update, see the monitoringAgentConfig endpoint.
`monitoringVersions.logRotate.sizeThresholdMB`	number	The maximum size in MB for an individual log file before rotation. To update, see the monitoringAgentConfig endpoint.
`monitoringVersions.logRotate.timeThresholdHrs`	integer	The maximum time in hours for an individual log file before rotation. To update, see the monitoringAgentConfig endpoint.
`monitoringVersions.logRotate.numUncompressed`	integer	*Optional*. The maximum number of total log files to leave uncompressed, including the current log file. The default is `5`. In earlier versions of Ops Manager, this field was named `maxUncompressed`. The earlier name is still recognized, though the new version is preferred. To update, see the monitoringAgentConfig endpoint.
`monitoringVersions.logRotate.percentOfDiskspace`	number	*Optional*. The maximum percentage of total disk space all log files should take up before deletion. The default is `.02`. To update, see the monitoringAgentConfig endpoint.

Backup Agent

The `backupVersions` array is optional and specifies the version of the Backup Agent.

```
"backupVersions" : [
    {
        "name" : <string>,
        "hostname" : <string>,
        "urls" : {
            <platform1> : {
                <build1> : <string>,
                ...,
                "default" : <string>
            },
            ...
        },
        "baseUrl" : <string>,
        "logPath" : <string>,
        "logRotate" : {
            "sizeThresholdMB" : <number>,
            "timeThresholdHrs" : <integer>,
            "numUncompressed": <integer>,
            "percentOfDiskspace" : <number>
        }
    },
    ...
]
```

Name	Type	Description
`backupVersions`	array of objects	*Optional*. Objects that define version information for each Backup Agent.
`backupVersions.name`	string	The desired version of the Backup Agent (e.g. “3.1.1.263-1”).
`backupVersions.hostname`	string	The hostname of the machine that runs the Backup Agent. If the Backup Agent is not running on the machine, Ops Manager installs the agent from the location specified in `backupVersions.urls`.
`backupVersions.urls`	object	The platform- and build-specific URLs from which to download the Backup Agent.

Name	Type	Description
`backupVersions.urls.<platform>`	object	This field has a name that identifies an operating system and optionally a version. The field contains an object with key-value pairs, where each key is either the name of a build or `default` and each value is a URL for downloading the Backup Agent. The object must include the `default` key set to the default download URL for the platform.
`backupVersions.baseUrl`	string	The base URL used for the `mothership` and `https` settings in the Backup Agent Configuration. For example, for `"baseUrl"=https://cloud.mongodb.com`, the backup configuration fields would have these values: `mothership=api-backup.mongodb.com` and `https"=true`.
`backupVersions.logPath`	string	*Optional*. The directory where the agent stores its logs. The default is to store logs in `/dev/null`. To update, see the backupAgentConfig endpoint.
`backupVersions.logRotate`	object	*Optional*. Enables log rotation for the MongoDB logs for a process. To update, see the backupAgentConfig endpoint.
`backupVersions.logRotate.sizeThresholdMB`	number	The maximum size in MB for an individual log file before rotation. To update, see the backupAgentConfig endpoint.
`backupVersions.logRotate.timeThresholdHrs`	integer	The maximum time in hours for an individual log file before rotation. To update, see the backupAgentConfig endpoint.
`backupVersions.logRotate.numUncompressed`	integer	*Optional*. The maximum number of total log files to leave uncompressed, including the current log file. The default is `5`. To update, see the backupAgentConfig endpoint.
`backupVersions.logRotate.percentOfDiskspace`	number	*Optional*. The maximum percentage of total disk space all log files should take up before deletion. The default is `.02`. To update, see the backupAgentConfig endpoint.

MongoDB Instances

The `processes` array determines the configuration of your MongoDB instances. You can also use the array to restore an instance.

```
"processes" : [
    {
        "name" : <string>,
        "processType" : <string>,
        "version" : <string>,
        "<args>" : <object>,
        "disabled" : <Boolean>,
        "manualMode" : <Boolean>,
        "hostname" : <string>,
        "cluster": <string>,
        "numCores": <integer>,
        "logRotate" : {
            "sizeThresholdMB" : <number>,
            "timeThresholdHrs" : <integer>,
            "numUncompressed": <integer>,
            "percentOfDiskspace" : <number>
        },
        "authSchemaVersion": <integer>,
        "alias": <string>,
        "backupRestoreUrl" : <string>
    },
    ...
]
```

Name	Type	Description
`processes`	array of objects	The `processes` array contains objects that define the `mongos` and `mongod` instances that Ops Manager monitors. Each object defines a different instance.
`processes.name`	string	A unique name to identify the instance.
`processes.processType`	string	Either `mongod` or `mongos`.
`processes.version`	string	The name of the `mongoDbVersions` specification used with this instance.
`processes.<args>`	object	This field is named either `args2_6`, for MongoDB versions 2.6 and higher (including 3.0 and higher), or `args2_4`, for versions 2.4 and earlier. The field contains a MongoDB configuration object in the format appropriate to the version. For information on format and supported MongoDB options, see supported configuration options.
`processes.disabled`	Boolean	*Optional*. Set to `true` to shut down the process.
`processes.manualMode`	Boolean	*Optional*. Set to `true` to operate this process in manual mode. The Automation Agent will take no actions on the process.

Name	Type	Description
`processes.hostname`	string	*Optional*. The name of the host this process should run on. This defaults to `localhost`.
`processes.cluster`	string	*Optional*. Required for a `mongos`. The name of the cluster. This must correspond to the `sharding.name` field in the `sharding` array for the `mongos`.
`processes.numCores`	integer	*Optional*. The number of cores the process should be bound to. The Automation Agent will spread processes out across the cores as evenly as possible.
`processes.logRotate`	object	*Optional*. Enables log rotation for the MongoDB logs for a process.
`processes.logRotate.sizeThresholdMB`	number	The maximum size in MB for an individual log file before rotation. The file rotates immediately if the file meets either this `sizeThresholdMB` or the `processes.logRotate.timeThresholdHrs` limit.
`processes.logRotate.timeThresholdHrs`	integer	The maximum time in hours for an individual log file before the next rotation. The time is since the last rotation. The log file rotates immediately if the file meets either this `timeThresholdHrs` or the `processes.logRotate.sizeThresholdMB` limit.
`processes.logRotate.numUncompressed`	integer	*Optional*. The maximum number of total log files to leave uncompressed, including the current log file. The default is `5`.
`processes.logRotate.percentOfDiskspace`	number	*Optional*. The maximum percentage of total disk space that can be used to store the log files. If this limit is exceeded, the compressed log files are deleted to meet this limit, starting with the oldest log files first. The default is `.02`.
`processes.authSchemaVersion`	integer	*Optional*. The schema version of the user credential objects. This should match all other elements of the `processes` array that belong to the same cluster. The possible values are `1`, `3`, and `5`. The default is `3` for `2.6` clusters and `1` for `2.4` clusters.

Name	Type	Description
`processes.alias`	string	*Optional*. A hostname alias (often a DNS CNAME) for the server on which the process runs. If an alias is specified, the Automation Agent prefers the alias over the host specified in `processes.hostname` when connecting to the server. You can also specify this alias in `replicaSets.host` and `sharding.configServer`.
`processes.backupRestoreUrl`	string	*Optional*. This is used only when creating a restore and specifies the delivery url for the restore. See Automate Backup Restoration through the API.

Replica Sets

The `replicaSets` array is optional and defines each replica set's configuration.

```
"replicaSets" : [
    {
        "_id" : <string>,
        "version" : <integer>
        "members" : [
            {
                "_id" : <integer>,
                "host" : <string>
            },
            ...
        ],
        "force" : {
            "currentVersion" : <integer>
        }
    },
    ...
]
```

Name	Type	Description
`replicaSets`	array of objects	*Optional*. Objects that define the configuration of each replica set. The Automation Agent uses the values in this array to create valid replica set configuration documents. The agent regularly checks that replica sets are configured correctly. If a problem occurs, the agent reconfigures the replica set according to its configuration document. The array can contain the following top-level fields from a replica set configuration document: `_id`; `version`; and `members`. For more information on the configuration documents, see replSetGetConfig in the MongoDB manual.
`replicaSets._id`	string	The name of the replica set.

Name	Type	Description
`replicaSets.version`	integer	The version of the replica set configuration.
`replicaSets.members`	array of objects	Objects that define each member of the replica set. The `members.host` field must specify the host's name as listed in `processes.name`. The Automation Agent expands the `host` field to create a valid replica set configuration. For more information on `members` objects, see replSetGetConfig in the MongoDB manual.
`replicaSets.force`	object	*Optional*. An object that contains the `currentVersion` field set to a version number. Automation will force a reconfiguration of the replica set if and only if the value of `currentVersion` equals the current version of the replica set. You can use `force` to reconfigure a replica set that has lost members and can't reach a majority of votes.

Sharded Clusters

The `sharding` array is optional and defines the configuration of each sharded cluster.

```
"sharding" : [
    {
        "name" : <string>,
        "configServer" : [ <string>, ... ],
        "collections" : [
            {
                "_id" : <string>,
                "key" : [
                    [ shard key ],
                    [ shard key ],
                    ...
                ]
            },
            ...
        ],
        "shards" : [
            {
                "_id" : <string>,
                "rs" : <string>
            },
            ...
        ]
    },
    ...
]
```

Name	Type	Description

Name	Type	Description
`sharding`	array of objects	*Optional*. Objects that define the configuration of each sharded cluster. Each object in the array contains the specifications for one cluster. The Automation Agent regularly checks each cluster's state against the specifications. If the specification and cluster don't match, the agent will change the configuration of the cluster, which might cause the balancer to migrate chunks.
`sharding.name`	string	The name of the cluster. This must correspond with the value in `processes.cluster` for a `mongos`.
`sharding.configServer`	array	String values that provide the names of each config server's hosts. The host names are the same names as are used in each host's `processes.name` field.
`sharding.collections`	array of objects	Objects that define the sharded collections and their shard keys.
`sharding.collections._id`	string	The namespace of the sharded collection. The namespace is the combination of the database name and the name of the collection. For example, `testdb.testcoll`.
`sharding.collections.key`	array of arrays	The collection's shard keys. This "array of arrays" contains a single array if there is a single shard key and contains multiple arrays if there is a compound shard key.
`sharding.shards`	array of objects	Objects that define the cluster's shards.
`sharding.shards._id`	string	The name of the shard.
`sharding.shards.rs`	string	The name of the shard's replica set, as specified in the `replicaSets._id` field.

Cluster Balancer

The `balancer` object is optional and defines balancer settings for each cluster.

```
"balancer": {
    "<clusterName1>": <object>,
    "<clusterName2>": <object>,
    ...
}
```

Name	Type	Description

Name	Type	Description
`balancer`	object	*Optional*. This object contains fields named according to clusters, each field containing an object with the desired balancer settings for the cluster. The object uses the `stopped` and `activeWindow` fields, as described in the procedure to schedule the balancing window in this tutorial in the MongoDB manual.

Authentication

The `auth` object is optional and defines authentication-related settings.

```
"auth" : {
    "autoUser": <string>,
    "autoPwd": <string>,
    "disabled": <Boolean>,
    "deploymentAuthMechanisms": [ <string>, <string>, ... ],
    "key" : <string>,
    "keyfile" : <string>,
    "usersDeleted" : [
        {
            "user" : <string>,
            "dbs" : [ <string>, ... ]
        }
    ],
    "usersWanted" : [
        {
            "db" : <string>,
            "user" : <string>,
            "roles" : [ <string>, ... ],
            "pwd" : <32-character hex string>,
            "initPwd" : <string>,
            "userSource" : <string>,
            "otherDBRoles" : {
                <string> : [ <string>, ... ]
            }
        }
    ]
}
```

Name	Type	Description
`auth`	object	*Optional*. Defines authentication-related settings.
`auth.autoUser`	string	The username that the Automation agent uses when connecting to an instance.
`auth.autoPwd`	string	The password that the Automation agent uses when connecting to an instance.

Name	Type	Description
`auth.disabled`	Boolean	Specifies whether authentication is enabled or disabled. Set to `true` to disable authentication, or `false` to enable authentication.
`auth.deploymentAuthMechanisms`	array	Lists the supported authentication mechanisms for the processes in the deployment. Specify `MONGODB-CR` for `MONGODB-CR` / `SCRAM-SHA-1` authentication, `MONGODB-X509` for x.509 Client Certificate authentication, `PLAIN` for LDAP authentication, and `GSSAPI` for authenticating with Kerberos.
`auth.disabled`	boolean	*Optional*. Indicates if auth is disabled. If not specified, `disabled` defaults to `false`.
`auth.key`	string	The contents of the key file that Ops Manager uses to authenticate to the MongoDB processes. The `key` is not required if `disabled` is `true`.
`auth.keyfile`	string	The path and name of the key file that Ops Manager uses to authenticate to the MongoDB processes. The `keyfile` is not required if `disabled` is `true`.
`auth.usersDeleted`	array of objects	*Optional*. Objects that define the authenticated users to be deleted from specified databases or from all databases. This array must contain two fields: the `auth.usersDeleted.user` field and the `auth.usersDeleted.dbs` field.
`auth.usersDeleted.user`	string	The user's name.
`auth.usersDeleted.dbs`	array	String values that list the names of the databases from which the authenticated user is to be deleted.
`auth.usersWanted`	array of objects	*Optional*. Contains objects that define authenticated users to add to specified databases. Each object must have the `auth.usersWanted.db`, `auth.usersWanted.user`, and `auth.usersWanted.roles` fields, and then have exactly one of the following fields: `auth.usersWanted.pwd`, `auth.usersWanted.initPwd`, or `auth.usersWanted.userSource`.
`auth.usersWanted.db`	string	The database to which to add the user.
`auth.usersWanted.user`	string	The name of the user.
`auth.usersWanted.roles`	array	String values that list the roles to be assigned the user from the user's database, which is specified in `auth.usersWanted.db`.
`auth.usersWanted.pwd`	32-character hex string	The MONGODB-CR hash of the password assigned to the user. If you set this field, **do not** set the `auth.usersWanted.initPwd` or `auth.usersWanted.userSource` fields.

Name	Type	Description
`auth.usersWanted.initPwd`	string	An initial cleartext password assigned to the user. If you set this field, **do not** set the `auth.usersWanted.pwd` or `auth.usersWanted.userSource` fields.
`auth.usersWanted.userSource`	string	If you use MongoDB version 2.4, you can use this field to specify the database that contains the user's credentials. See the Privilege Documents page in the MongoDB 2.4 manual. If you set this field, **do not** set the `auth.usersWanted.pwd` or `auth.usersWanted.initPwd` fields.
`auth.usersWanted.otherDBRoles`	object	*Optional*. If the `auth.usersWanted.db` field specifies `admin` as the user's database, then this object can assign to the user roles from other databases as well. The object contains key-value pairs where the key is the name of the database and the value is an array of string values that list the roles be assigned from that database.

SSL

The `ssl` object is optional and enables SSL for encrypting connections.

```
"ssl" : {
    "CAFilePath" : <string>
}
```

Name	Type	Description
`ssl`	object	*Optional*. Enables SSL for encrypting connections. To use SSL, be sure to choose a package that supports SSL. Starting in MongoDB 3.0, most MongoDB distributions now include support for SSL. All MongoDB Enterprise supported platforms include SSL support.
`ssl.CAFilePath`	string	The path to the certificate used to authenticate through SSL.

MongoDB Roles

The `roles` array is optional and describes user-defined roles.

```
"roles" : [
    {
        "role" : <string>,
        "db" : <string>,
        "privileges" : [
            {
                "resource" : { ... },
                "actions" : [ <string>, ... ]
            },
            ...
        ],
        "roles" : [
            {
                "role" : <string>,
                "db" : <string>
            }
        ]

    },
    ...
]
```

Name	Type	Description
`roles`	array of objects	*Optional*. The `roles` array contains objects that describe the cluster's user-defined roles. Each object describes a different user-defined role. Objects in this array contain the same fields as documents in the :manual:` system roles collection </reference/system-roles-collection>`, except for the `_id` field, which is not included here.

Kerberos

The `kerberos` object is optional and defines a kerberos service name used in authentication.

```
"kerberos": {
    "serviceName": <string>
}
```

Name	Type	Description
`kerberos`	object	*Optional*. A key-value pair that defines the kerberos service name agents use to authenticate via kerberos.
`kerberos.serviceName`	string	The service name agents use to authenticate to a `mongod` or `mongos` via kerberos. This name is also used to set the `saslServiceName` option in a MongoDB configuration, as described on the MongoDB Server Parameters page in the MongoDB manual.

Indexes

The `indexConfigs` array is optional and defines indexes to be built for specific replica sets.

```
"indexConfigs" : [
    {
        "key" : [
            [ <string> : <val> ],
            ...
        ],
        "rsName" : <string>,
        "dbName" : <string>,
        "collectionName" : <string>
    },
    ...
]
```

Name	Type	Description
`indexConfigs`	array of objects	*Optional*. Objects that define specific indexes to be built for specific replica sets.
`indexConfigs.key`	array of arrays	The index's keys. This "array of arrays" contains a single array if the index has just one key.
`indexConfigs.rsName`	string	The replica set that the index is build on.
`indexConfigs.dbName`	string	The database the index applies to.
`indexConfigs.collectionName`	string	The collection the index applies to.