Java

OpCon CLI

Contents

[OpConCLI 3](#_Toc36616761)

[Supported Tasks 3](#_Toc36616762)

[Installation 3](#_Toc36616763)

[Windows 3](#_Toc36616764)

[Linux 4](#_Toc36616765)

[Configuration 4](#_Toc36616766)

[Task Definitions 5](#_Toc36616767)

[EncryptValue Utility 10](#_Toc36616768)

[APPENDIX A 11](#_Toc36616769)

[Table 1: Command Line Utilities Functions 8](#_Toc36616843)

[Figure 1: Connector.config example 4](#_Toc36616820)

# OpConCLI

Is a command utility that submits requests to OpCon using the SMA OpCon Rest-API.

The utility is available for both Windows and Linux.

## Supported Tasks

The utility provides the following tasks:

* AppToken Creates an application token
* Dependency Creates a dependency on a task in a remote OpCon System. It tracks the

execution of the remote job waiting for job completion. Retrieves the job log of the remote job and terminates with the same termination code as the remote job (requires OpCon 17.1 or greater).

* GetJobLog Retrieve the job log of a job in a schedule in the Daily tables (requires OpCon

17.1 or greater).

* JobAction Performs an action on a job in the Daily tables (available actions are hold,

cancel, skip, kill, start, restart, forceRestart, restartOnHold, release, markFinishedOk, markFailed, markUnderReview, markFixed).

* JobAdd Add a job to schedule in the Daily tables. Includes an option to wait for the

completion of the job that has been added.

* MachAction Performs an action on a machine or list of machines (available actions are up,

down, limited, wlimited). The option wlimited option, sets the machine into limited state and waits until all current executing jobs on the machine have completed before returning.

* MachAdd Adds a machine or list of machines to the OpCon System.
* MachGrpAdd Adds the machines to the machine group.
* MachGrpRemove Removes the machines from the machine group.
* MachUpdate Updates machine name, IP address or DNS address.
* PropExp Uses the expression evaluator to update properties. If the property does not

Exist it is created. Supports global and instance properties.

* PropUpdate Updates the value of a global property for OpCon 18.1 and beyond (supports

encryption).

* SchedAction Performs an action on a schedule (available actions are hold, release, start,

close).

* SchedBuild Build a schedule (requires OpCon 18.1 or greater).
* SchedRebuild Rebuilds schedules in the daily (requires OpCon 18.1 or greater).
* ThreshUpdate Updates the value of a threshold.
* Version Retrieves the version of a the SMA OpCon Rest-API

## Installation

The installer includes separate installers for Windows and UNIX systems.

### Windows

Download OpConCLI\_win.zip file from the desired release by selecting and saving the file. After download create an installation root folder and then extract the information from the downloaded file into the created installation directory.

### Linux

Download OpConCLI\_Linux.tar file from the desired release by selecting and saving the file. After download create an installation root folder and then extract the information from the downloaded file into the created installation directory.

### Configuration

The OpConCLI utility uses a configuration file Connector.config that contains the OpCon System connection information. The header name of the OpCon System connection information is used by the -o option to retrieve the connection information for the required system. This allows the utility to submit requests to multiple OpCon systems from a single installation.

The USER and PASSWORD information should be encrypted using the EncryptValue.exe program for Windows or EncryptValue.sh shell for Linux. The encryption tool provides basic encryption capabilities. If required this can be enhanced by downloading the source code and modifying the encryption routines.

[GENERAL]

DEBUG=ON

[OPCON001]

SERVER=OPCON001

PORT=9010

USING\_TLS=True

USER=62324e685a47303d

PASSWORD=6233426a6232353463484d3d

[OPCON002]

SERVER=OPCON002

PORT=9010

USING\_TLS=True

USER=62324e685a47303d

PASSWORD=6233426a6232353463484d3d

Figure : Connector.config example

The values in the configuration file are as follows:

SERVER is the address of the OpCon Server

PORT is the port being used by the OpCon RestAPI server (usually 9010 or 9000)

USING\_TLS is if the OpCon Rest-API server is using TLS connections (Either True or False)

USER is a valid OpCon user encrypted using the EncryptValue program.

PASSWORD is the password associated with the OpCon user encrypted using the EncryptValue

Program.

## Task Definitions

The OpConCLI program currently supports several tasks that require various arguments. The arguments associated with each task are defined in the table below.

|  |  |  |
| --- | --- | --- |
| *Task* | *Arguments* | *Description* |
| AppToken | -t AppToken | Can be used to create an application token. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -ap | Required field for AppToken and represents the name of the application to create the token for. If a token for this application name already exists, it returns the existing value. |
| Dependency | -t Dependency | Can be used create a remote dependency on a job in another OpCon system. Tracks the job and returns the termination code of the job being tracked. Also returns the job log of the remote job. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -d | Optional field that defines the Date of the request. If not present, the current date will be used. Date format is YYYY-MM-DD. |
|  | -sn | The name of the schedule in the Daily tables to add the job to. |
|  | -jn | The name of the job to add to the schedule in the Daily tables. |
| GetJobLog | -t GetJobLog | Retrieves the job log of an OpCon job. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -d | Optional field that defines the Date of the request. If not present, the current date will be used. Date format is YYYY-MM-DD. |
|  | -sn | The name of the schedule in the Daily tables containing the job to retrieve the job log for. |
|  | -jn | The name of the job in the schedule in the Daily tables to retrieve the job log for. |
|  | -jld | Optional field defining the full path name of the directory to write the job log into. The job log name created by the agent is used as the file name (i.e. JOB001\_0000004245.log).  If the field is not present the job log information will be added to the OpConCLI execution output. |
| JobAction | -t JobAction | Performs an action on a job on schedule in the daily tables. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -d | Optional field that defines the Date of the request. If not present current date will be used. Date format is YYYY-MM-DD. |
|  | -sn | The name of the schedule in the Daily tables that contains the job. |
|  | -jn | The name of the job. |
|  | -ja | The action to be performed on the job. Values are hold, cancel, skip, kill, start, restart, forceRestart, restartOnHold, release, markFinishedOk, markFailed, markUnderReview, markFixed. |
| JobAdd | -t JobAdd | Adds a job to an existing schedule in the daily tables. Includes the capability to wait for the job that was added to complete. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -d | Optional field that defines the Date of the request. If not present, the current date will be used. Date format is YYYY-MM-DD. |
|  | -sn | The name of the schedule in the Daily tables to add the job to. |
|  | -jn | The name of the job to add to the schedule in the Daily tables. |
|  | -jf | The name of the frequency to be used for the job add request. |
|  | -ip | Optional field defining instance properties that should be passed to the job during the job add process. The format is name=value,name=value |
|  | -joh | Indicates to add the job in a hold status. If not present, the job will be added in a release status. |
|  | -jw | Indicates to add the job and then track the execution of the job that was added. |
| MachAction | -t MachAction | Performs an action on a machine. Includes a special value wlimited which sets a machine to limited mode and then waits for all current tasks on the machine to complete before returning. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -mn | the machine(s) to perform the action on (value mach1 for single machine or mach1,mach2,mach3 for multiple machines). |
|  | -ma | the action to be performed on the machine(s). Values are up, down, limited. |
| MachAdd | -t MachAdd | Creates a new machine in the OpCon system. It is possible to use a template and use the -md, -mi & -mn arguments to change the values in the template.  Note : The JSON data presented consists of an array to allow the submission of more than one machine at a time. Therefore the JSON data must be wrapped in [ ] brackets. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -md | Fully Qualified DNS of the machine to be added. If present will override the value in the JSON file. (Note : only works if there is a single machine definition in the JSON file). |
|  | -mf | File name of JSON structure containing the definitions of the machine to add (see Appendix A for file template). |
|  | -mi | IP Address of the machine to be added. If present will override the value in the JSON file. (Note : only works if there is a single machine definition in the JSON file). |
|  | -mn | Name of the machine to be added. If present will override the value in the JSON file. (Note : only works if there is a single machine definition in the JSON file). |
| MachGrpAdd | -t MachGrpAdd | Add a machine or machines to a machine group. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -mg | The machine group name. |
|  | -mn | The machine(s) to add to the machine group (value mach1 for single machine or mach1,mach2,mach3 for multiple machines). |
| MachGrpRemove | -t MachGrpRemove | Remove a machine or machines from a machine group. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -mg | The machine group name. |
|  | -mn | the machine(s) to remove from the machine group (value mach1 for single machine or mach1,mach2,mach3 for multiple machines). |
| MachUpdate | -t MachUpdate | Update the manchine name, ip address or fully qualified DNS name of an existing machine. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -mn | Name of the machine to be updated. |
|  | -mnu | The new name of the machine if the name is to be changed. |
|  | -miu | The new IP Address of the machine if the IP Address is to be updated (in this case the fully qualified DNS name will be set to <Default>). |
|  | -mdu | The new DNS Address of the machine if the DNS Address is to be updated (in this case the IP Address will be set to <Default>). |
| PropExp | -t PropExp | Uses expression evaluation to create or update properties. Supports global and instance properties. When using instance properties use the fully qualified name. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -pn | The name of the property to update. |
|  | -pv | The value of the property. |
| PropUpdate | -t PropUpdate | Updates the contents of a global property. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -pn | The name of the property to update. |
|  | -pv | The value of the property. |
|  | -pe | If the property is encrypted (values true or false) |
|  |  |  |
| SchedAction | -t SchedAction | Performs an action on the schedule in the daily tables. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -d | Optional field that defines the Date of the request. If not present, the current date will be used. Date format is YYYY-MM-DD. |
|  | -sn | The name of the schedule to update. |
|  | -sa | The action to be performed on the schedule. Values are hold, release, start, close. |
| SchedBuild | -t SchedBuild | Builds a schedule. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value in the CommandLine.config file to identify the connection information for the task. |
|  | -d | Optional field that defines the Date of the request. If not present, the current date will be used. Date format is YYYY-MM-DD. |
|  | -sn | The name of the schedule to build. |
|  | -soh | Indicates to build schedule in a hold status. If not present, the schedule will be built in a release status. |
|  | -ip | Optional field defining instance properties that should be passed to the schedule during the build. The format of the instance properties is name=value,name=value |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value to identify the connection information for that system. |
| SchedRebuild | -t SchedRebuild | Can be used to rebuild schedules in the daily from a given date for a number of days. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value to identify the connection information for that system. |
|  | -d | Optional field that defines the Date of the request. If not present, the current date will be used. Date format is YYYY-MM-DD. |
|  | -sd | The number of days to rebuild the schedules for. |
|  | -sri | Is used to indicate which schedule should be rebuilt. Should be the first x number of characters of the schedule name or the entire name. If the schedule names contains spaces remember to enclose the argument in double quotes. |
| ThreshUpdate | -t ThreshUpdate | Updates the value of a threshold. |
|  | -o | The name of the OpCon system to submit the request to. Matches a header value to identify the connection information for that system. |
|  | -tn | The name of the threshold to update. |
|  | -tv | The value of the threshold. |

Table : Command Line Utilities Functions

Examples Windows:

Build the schedule TEST\_JOB\_ADD for today’s date on hold.

opconcli.exe -t SchedBuild -sn TEST\_JOB\_ADD -soh -o OPCONA

Build the schedule TEST\_JOB\_ADD for 12th May 2019 on hold setting schedule instance properties PROP1 & PROP2.

opconcli.exe -t SchedBuild -sn TEST\_JOB\_ADD -soh -d 2019-05-12 -ip PROP1=TEST,PROP2=ONE -o OPCONA

Set the machines MACH1,MACH2 to a down state.

opconcli.exe -t MachAction -mn BVHTEST02AMT\_AGENT,AMT\_AGENT -ma down -o OPCONA

Add job TEST\_ADD\_JOB to schedule TEST\_JOB\_ADD for today’s date using frequency AllDays and setting job instance properties TIME and ERROR.

opconcli.exe -t JobAdd -sn TEST\_JOB\_ADD -jn TEST\_ADD\_JOB -jf AllDays -ip TIME=15,ERROR=0 -o OPCONA

Add job TEST\_ADD\_JOB to schedule TEST\_JOB\_ADD for today’s date using frequency AllDays in a hold status and setting job instance properties TIME and ERROR.

opconcli.exe -t JobAdd -sn TEST\_JOB\_ADD -jn TEST\_ADD\_JOB -jf AllDays -ip TIME=15,ERROR=0 -o OPCONA -joh

Add job TEST\_ADD\_JOB to schedule TEST\_JOB\_ADD for today’s date using frequency AllDays, setting job instance properties TIME and ERROR and wait for job to complete before returning.

opconcli.exe -t JobAdd -sn TEST\_JOB\_ADD -jn TEST\_ADD\_JOB -jf AllDays -ip TIME=15,ERROR=0 -jw -o OPCONA

Retrieve job log for job TEST\_ADD\_JOB of schedule TEST\_JOB\_ADD for today’s date.

opconcli.exe -t GetJobLog -sn TEST\_JOB\_ADD -jn TEST\_ADD\_JOB -o OPCONA

Retrieve job log for job TEST\_ADD\_JOB tof schedule TEST\_JOB\_ADD for today’s date writing the log into file in directory c:\temp.

opconcli.exe -t GetJobLog -sn TEST\_JOB\_ADD -jn TEST\_ADD\_JOB -jld c:\temp -o OPCONA

Cancel job TEST\_ADD\_JOB of schedule TEST\_JOB\_ADD for today’s date.

opconcli.exe -t JobAction -sn TEST\_JOB\_ADD -jn TEST\_ADD\_JOB -ja cancel -o OPCONA

Update threshold THRESH001 value to 55.

opconcli.exe -t ThreshUpdate -tn THRESH001 -tv 55 -o OPCONA

Add machines TEST001 & TEST002 to machine group GRP001.

opconcli.exe -t MachGrpAdd -mg GRP001 -mn TEST001,TEST002 -o OPCONA

Create machine TEST001 in OpCon system OPCON from template win\_machadd.json overriding name and address.

opconcli.exe -t MachAdd -mn TEST001 -mf c:\templates\win\_machadd.json -mi 10.0.2.12 -o OPCON

Rebuild schedules in OpCon system OPCON from 1st March for 5 days.

opconcli.exe -t SchedRebuild -d 2020-03-01 -o OPCON -sd 5

Rebuild schedules starting with TEST in OpCon system OPCON from 1st March for 5 days.

opconcli.exe -t SchedRebuild -d 2020-03-01 -o OPCON -sd 5 -sri TEST

# EncryptValue Utility

The EncryptValue utility uses standard 64 bit encryption.

Supports a -v argument and displays the encrypted value.

Examples Windows:

Encrypt the value abcdefg.

EncryptValue.exe -v abcdefg

# APPENDIX A

Template JSON file for creating machines. (See the Online OpCon API Machines Model for latest information).

[

{

"requiresXMLEscape": false,

"agentSMACommunicationsProtocol": "NEW2",

"allowKillJob": false,

"tcpIpAddress": "<Default>",

"pollInterval": 1000,

"checkMachineStatusInterval": 120,

"connectionAttemptTimeout": 1000,

"noBufferRetryCount": 40,

"noBufferSleepTime": 250,

"maxConsecutiveSendMessages": 100,

"consecutiveSendSleepTime": 100,

"sendBufferCount": 25,

"receiveBufferCount": 25,

"maxBytesSentPerMessage": 1024,

"maxBytesReadPerMessage": 1024,

"responseTimeout": 30,

"agentCheckCRC": true,

"closeSocketDuringSynchronization": false,

"traceAllMessages": true,

"smaNetComIdentifier": "<Default>",

"gatewayName": "<None>",

"jorsPortNumber": 3220,

"fileTransferRole": "T",

"fileTransferIPAddress": "<Default>",

"availableProperties": [

[

{

"name": "PowerShellPath",

"value": "C:\\Windows\\system32\\WindowsPowerShell\\v1.0"

},

{

"name": "MyMachineProp",

"value": "MyMachineValue"

}

]

],

"fileTransferPortNumberForNonTLS": 3220,

"fileTransferPortNumberForTLS": 0,

"agentFileTransferPortNumberForTLS": "0",

"lsamTime": 0,

"lsamTimeDeltafromSAM": 0,

"fullyQualifiedDomainName": "qa2012r2sam",

"fullFileTransferSupport": false,

"timeOffsetfromUTC": -5,

"timeZoneName": "Central Standard Time",

"timeOffsetfromSAMInHours": 0,

"supportsHandshake": "False",

"tlsCertificateDistinguishedName": "<Default>",

"checkCertificateRevocationList": false,

"supportTLSForSMAFTAgent": false,

"agentSupportTLSForSMAFTAgent": "False",

"supportTLSForSMAFTServer": false,

"agentSupportTLSForSMAFTServer": "False",

"supportNonTLSForSMAFTAgent": true,

"agentSupportNonTLSForSMAFTAgent": "False",

"supportNonTLSForSMAFTServer": true,

"agentSupportNonTLSForSMAFTServer": "False",

"supportedAgentCapability": [],

"useTLSforSchedulingCommunications": false,

"name": "MyWinAgent",

"id": 88,

"type": {

"id": 3,

"description": "Windows"

},

"socket": 3240,

"gatewayId": 0,

"groups": [],

"lastUpdate": "2017-06-11T15:03:27.7480000-05:00",

"maximumJobs": 0,

"opconMaximumJobs": 0,

"currentJobs": 0,

"status": {

"state": "D",

"networkStatus": "D",

"operationStatus": "D"

}

}

]