

# VELOSTRATA RUNBOOK AUTOMATION TOOL

# Table of Contents

- Using Runbook Automation Tool..... 3
  - Overview ..... 4
  - Pre-requisites ..... 7
  - Creating the Cloud Details ..... 8
  - Creating a Runbook ..... 10
  - Modifying your Runbook..... 14
  - Rightsizing the Instances in your Runbook..... 17
  - Creating a Runbook Job..... 19
  - Aborting a Runbook Job ..... 21
  - Monitoring a Runbook Job ..... 22
  - Managing Cloud to Cloud Runbook via PowerShell/API ..... 24

# **Using Runbook Automation Tool**

# Overview

The Runbook Automation Tool simplifies the mass migration of VMs from on-premises VMware or AWS into GCP. It is available for use from the Velostrata Manager page ([https://IP\\_OF\\_VELOSTRATA\\_APPLIANCE](https://IP_OF_VELOSTRATA_APPLIANCE)). Typically, runbook automation is only used when migrating multiple VMs, sometimes tens or hundreds in one sprint. The purpose of runbook automation is to automate a migration of VMs in a pre-defined order. To do this, a runbook file (CSV) is used, where the RunGroup variable is defined as follows:

- -1 indicates this row (VM) will be skipped/ignored.
- All other values are migrated in ascending order. 5 VMs with RunGroup 100, for example, will move together in a group. Then another 3 VMs with RunGroup 200, for example, will move together next. And so on.

After defining the runbook inventory file, you'll also select a job, which is what actions are performed on these VMs. For some operations (run in cloud, full migration, offline migration) the first operation that takes place is a graceful shutdown of the VM at the source (on-prem or cloud). After that, when the runbook automation actions run, they do so in a sequence as defined, and are all re-entrant, that is, if there is any failure in the process, you can fix the problem and either click the re-run button from the runbook interface, or you can create a new job altogether with the same runbook file. The actions available via runbook are defined below:

## Test Clone

Creates clones of the selected VMs for the purpose of in-cloud testing. A test clone will look and feel like the live systems, and will leverage what appears to be the live data. However, no live data (nor the VM) are actually affected through usage of test clones. This is only available for on-prem to cloud migrations.

## Delete Test Clone

This will shut down and terminate the test clones that were previously created. This is only available for on-prem to cloud migrations, and where a test clone is already running.

## Run in Cloud

Moves the source VMs from on-prem or AWS to GCP.

1. Shuts down the source VMs in reverse order of the RunGroups.
2. Executes the Run in Cloud operation for each instance.
3. Waits for the management task to complete.
4. Waits for the successful completion of the TCP port probing (Optional).

*Note: For cloud-to-cloud migrations, the Writelsolation value will be ignored because TRUE is the only available option as write-back is not supported.*

## Move Back

Moves the instances in GCP back to their source, either on-prem or AWS.

1. Stops the VMs in reverse order of the RunGroups.
2. Moves back each instance.

## Full Migration

Runs a series of automation tool actions to migrate a group of VMs, as follows:

1. Runbook begins the full migration tasks
2. Waits for the VMs to be in the 'Cache on Demand' state.
3. Waits for the successful completion of the TCP port probing (optional)
4. Waits for the 'Ready to Detach' state.

*Note: For cloud-to-cloud migrations, the Writelsolation value will be ignored because TRUE is the only available option as write-back is not supported.*

## Offline Migration

Velostrata enables you to migrate workloads with operating systems that are not currently supported by Velostrata's streaming technology. During the offline migration process, the VM storage is migrated to the cloud first, and the VM is started in the cloud only after the migration is completed. Velostrata automatically performs the detach and cleanup of residue data associated with the VM.

## Prepare to Detach

This brings the VMs running in the cloud to a state where they are ready to be detached from their on-prem sources.

## Detach

Once the VMs are in a Ready to Detach migration state, and the preparing to detach tasks complete successfully, you can detach the VMs. This severs their ties to on-prem, and means you can no longer perform instant rollback via Velostrata.

**Note:** This job waits for the successful completion of the TCP port probing (Optional).

## Cleanup

Once the VMs are detached and you have completed any required validation, you can start the detach cleanup. For each VM, this marks the VM as unmanaged by Velostrata.

# Pre-requisites

You can use the Velostrata Runbook Automation Tool to plan, order, and execute your migration operations.

## On-prem to Cloud

- VMware:
  - Subnet with connectivity to GCP, AWS, or Azure.
  - Instances to migrate.
- Velostrata Cloud Extension properly deployed in GCP.

## AWS to GCP

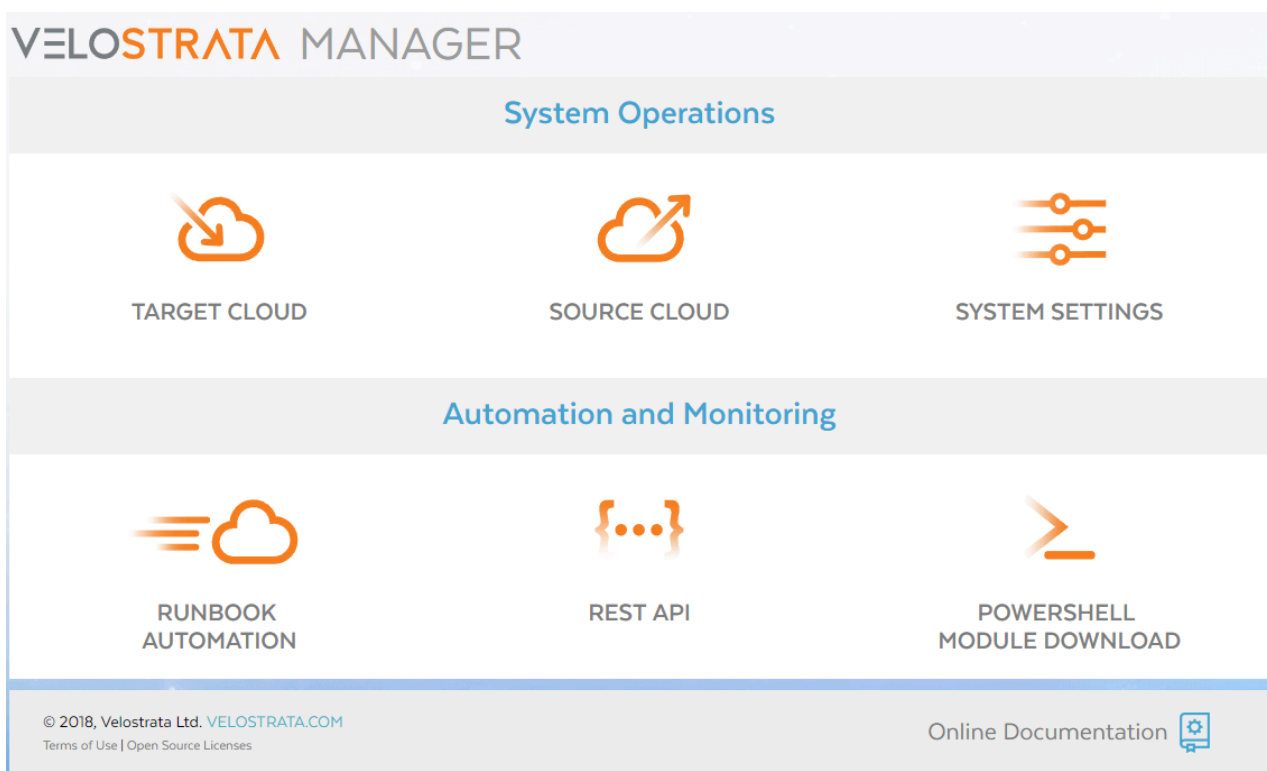
You must meet the requirements above, as well as the following:

- AWS:
  - Subnet with connectivity to GCP.
  - Instances to migrate.
- Velostrata Cloud Extension properly deployed in GCP.
- Cloud Details object that embeds the AWS credentials.
  - Covered on next page

# Creating the Cloud Details

*The following steps only need to be completed if you are performing migrations from AWS to GCP:*

1. Login to the Velostrata appliance via HTTPS://IP\_OF\_VELO\_APPLIANCE
  2. Click the Source Cloud icon
  3. Click on the Cloud Details tab
  4. Upload your AWS credentials file (it is a .JSON file which you generate via AWS)
  5. Define the remaining cloud details.
  6. Click finish.
1. Open a Web browser and navigate to the Velostrata Manager Virtual Appliance IP address, for example,
  2. Click the **System Settings** icon.
  3. Login when prompted with username 'localsupport'. Use your Velostrata Subscription ID or GCP Billing ID as your password.



4. Click the **Source Cloud** icon and then click the **Create** button.



### Create New Cloud Details

\*Name:

new-aws-source-cloud

\*Credentials:

awsCreds-ruC

\*Region:

US East (Ohio)

\*VPC:

VeloScale-US-Ohio | vpc-7029c119

\*Security Group:

default

Please select the subnets where Velostrata workers are created when migrating instances from the respective availability zones:

Worker subnet for availability zone: us-east-2a

12.0.3.0/24 | subnet-395cab50

Worker subnet for availability zone: us-east-2b

12.0.2.0/24 | subnet-fff9f087

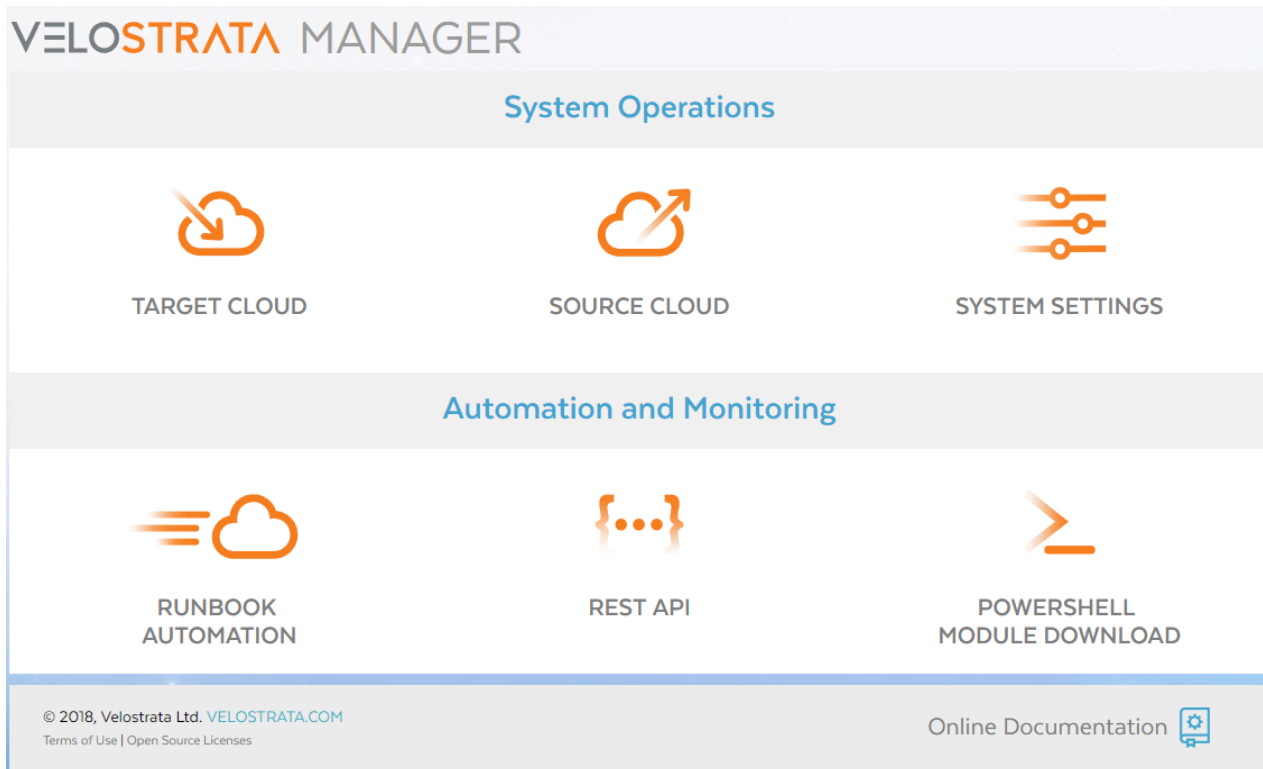
Ok

5. Give this new source cloud a name.
6. Pick the cloud credentials you wish to use for this source cloud.
  - A. The cloud credentials must have been already added, via the **Cloud Credentials** tab.
7. Define your **Region**, **VPC**, and **Security Group**.
8. Pick your **Velostrata Worker Subnets**, which is where these workers will be created, such that they reside in the relevant availability zone based on the location of the source machine.
9. Click OK.

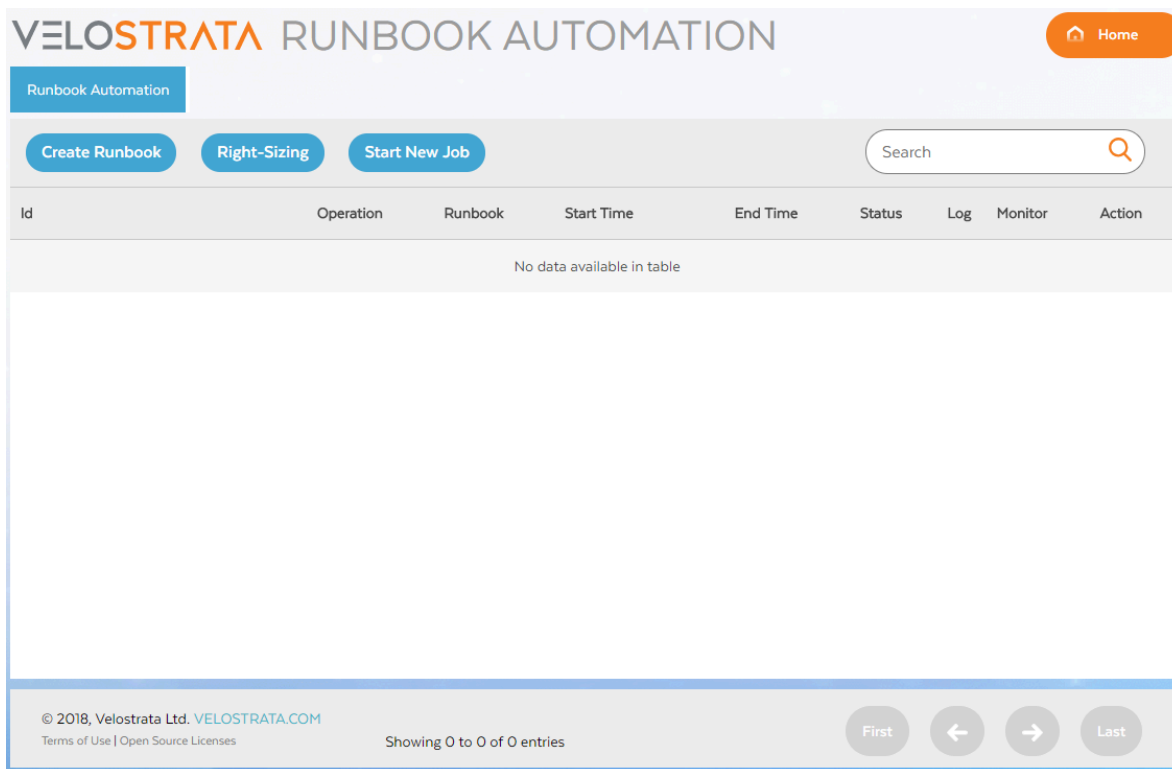
# Creating a Runbook

The runbook is the inventory file we'll use to catalog a group of VMs and define the parameters of our runbook automation task.

1. Navigate to the Velostrata Manager page at [https://VELO\\_APPLIANCE\\_IP](https://VELO_APPLIANCE_IP).



2. Click the **Runbook Automation** icon.
3. Log in as **apiuser**
  - A. Your password will be your Velostrata subscription ID or your GCP billing ID.
4. Once you're successfully logged in, click the **Create New Runbook** button.



5. When the menu appears, you'll select your **Source** which will either be **vSphere** or **AWS**. Continue along by expanding the appropriate section below:

## On-Prem as Source

1. [Mandatory] For **Source Datacenter**, select the on-premises VMware datacenter whose inventory you'd like to export. This list of options is automatically populated from the vCenter you registered this Velostrata appliance with.
2. [Optional] For **Target Cloud Extension**, select your cloud destination (where VMs will be migrated to). You can populate this later if you'd like.
3. [Optional] For **Target Network**, based on your target cloud extension selection, you can also choose the **Populate with Cloud Extension Defaults** option. You can populate these fields later, too, if desired.

Note: This pre-populates the target subnet and GCP network tags with CE defaults for workloads.

4. Click **Create**.

**Create Runbook**

Source: vSphere ▼

Source Datacenter: dc-test ▼

Target Cloud Extension: gcp\_datacenter-2\_1531153632\_bGW ▼

Target Network: ☒ Populate with Cloud Extension Defaults

Create

## AWS as Source

1. Select the required **Source Cloud Details**, which is where VMs will be migrated from.
2. [Mandatory] Enter one or more **Filter by Source Tags** by entering **Name** and **Value** pairs. This is how we'll pull the VM information out of AWS. Therefore, you must have this tag pair in AWS as well for any VM you wish to inventory. For example, you would need a tag named "WORKLOAD" with value "CRM" in AWS, and then you would define that here in this menu, too. You can also use operators, such that tag name equals "WORKLOAD" and the value you specify in Velostrata is the \* operator, which would inventory any VM with the WORKLOAD tag from this AWS environment.
3. [Optional] For **Target Network**, based on your target cloud extension selection, you can also choose the **Populate with Cloud Extension Defaults** option. You can populate these fields later, too, if desired.

Note: This pre-populates the target subnet and GCP network tags with CE defaults for workloads.

5. Click **Create**.

Create Runbook

Source:

AWS

Source Cloud Details:

awsCloudDetails-uEa

Filter by Source Tags:

Name

Value

workload

crm

Target Cloud Extension:

gcp\_datacenter-2\_1531153632\_bGW

Target Network:

☒ Populate with Cloud Extension Defaults

Create

After configuring your options and clicking complete, the runbook inventory file (.csv) is created and you will be prompted to save it locally.

It will look like this when you open it via your spreadsheet editor:

	A	B	C	D	E	F	G
1	RunGroup	BlockOnF	ProbeTCP	ProbeWai	VmID	SourceClo	OsType
2	-1	FALSE	0	0	i-0525f1b5	awsCreds	linux
3	-1	FALSE	0	0	i-0ed746a	awsCreds	linux
4	-1	FALSE	0	0	i-005fc91b	awsCreds	linux
5	-1	FALSE	0	0	i-0472daf8	awsCreds	linux
6	-1	FALSE	0	0	i-03c2179e	awsCreds	linux
7	-1	FALSE	0	0	i-09479270	awsCreds	linux
8	-1	FALSE	0	0	i-0a18d52	awsCreds	linux

# Modifying your Runbook

After clicking complete in the previous step, the runbook inventory file (.csv) is created. Navigate to this file and open with your preferred spreadsheet editor.

Identify and delete the rows for the VMs that you do not want to migrate -or- define their **RunGroup** value as **-1** (which is also the default in template) to instruct the automation tool to skip processing a VM.

For VMs you wish to include in your runbook tasks, you must assign a positive integer value. VMs with the same value will execute in the same group. Groups are executed in ascending order. For example, if you have RunGroup 200 with 3 VMs and RunGroup 400 with 2 VMs, the RunGroup 200 will execute first. When that RunGroup is completely finished, RunGroup 400 will begin.

There are many other fields you can edit and modify (or add, if desired) to the runbook.

**Important note:** when using runbooks to migrate to GCP, all values must be in **lowercase** (even boolean values). There are some cases where operations will fail solely because values are in all caps.

## Required Fields

**The following fields are always required to be populated:**

- **RunGroup:** Grouping of VMs that will be migrated. VMs in the same group will be migrated in parallel, VMs in different groups will be migrated sequentially in order (lowest group number to highest). A negative value means skips the row.
- **VmID:**
  - If migrating from on-prem VMware, this is the inventory path or VM ID.
  - If migrating from AWS, this is the AWS instance ID.
- **BlockOnFailure:** If set to **TRUE** and an action fails, the automation tool stops and the subsequent RunGroups are not executed. The default value is **FALSE**.

Required for RunInCloud, FullMigration, OfflineMigration:

- OSLicense (can be empty)
- SourceCloudDetails
- TargetCloudExtension
- TargetEdgeNode
- TargetInstanceType
  - You can use Velostrata's Rightsizing feature to assist with this. For more information [click here](#).
- TargetInstanceName
- TargetSubnet
- TargetStaticIP (can be empty)
- GcpNetworkTags (can be empty or removed from CSV)

- All the “tag:...” columns (can be empty or removed from CSV)

Required for FullMigration, OfflineMigration:

- GcpDiskType

Used by RunInCloud, Detach (a valid number must be always present):

- ProbeTCPPort
- ProbeWaitMinutes

Writesolation:

- When performing on-prem to cloud migrations, for Writesolation user can select either TRUE or FALSE (write-back mode).
- When performing cloud-to-cloud migrations, Writesolation field is ignored because TRUE is the only option. Write-back is not supported in cloud-to-cloud migrations currently.

## Additional Fields

Below is a list of all the additional fields that are officially supported by the runbook. You are free to add other custom fields for informational purposes only as well.

- **ProbeTCPPort:** TCP port to probe the VM on. This field is always zero and can be changed.
  - **NOTE:** The port should be accessible on the target GCP instance when running the automation tool.
- **ProbeWaitMinutes:** Number of minutes to wait for the port probe to respond, after which the probe check will be considered to have failed. The default is 0 minutes (no wait).
- **VmID:** ID of the source VM.
- **SourceCloudDetails:** The source cloud to migrate from.
- **OsType:** VM OS.
- **OsLicense:** VM license.
- **NumCPU:** Current number of CPUs of the source VM.
- **MemoryMB:** Current memory size of the source VM.
- **NumDisks:** Current number of disks of the source VM.
- **ProvisionedSpaceGB:** Total allocated storage space for the VM.
- **TargetCloudExtensionName:** Velosrata cloud extension name to be used for cloud migration.
- **TargetEdgeNode:** Edge node to run the VMs in the cloud, use **NodeA** or **NodeB**.
- **TargetInstanceType:** Instance type of the VM that will be created in cloud, for example, **n1-standard-1**.
- **TargetInstanceName:** If the AWS instance has a **Name** tag, its value is normalized according to the GCP naming limitations (lower case, and so on) and used. Otherwise the AWS instance ID is used.
- **TargetSubnet:** Subnet ID to be used by the instance in the cloud. An empty field means that the default subnet will be selected (as specified in the Cloud Extension creation). For example, **<https://www.googleapis.com/compute/v1/projects/<project ID>/regions/europe-west1/subnetworks/<subnet name>>**

- **TargetStaticIP:** A static IP to assign to the instance in the cloud.
- **TargetPublicIP:** A public IP to assign to the instance in the cloud.
- **Ephemeral:** Whether or not to assign an ephemeral IP address (FALSE/TRUE).
- **GcpNetworkTags:** The VM network tags. If this field is empty, then no tag will be added to the instance. If the default CE checkbox is marked on runbook creation, then the default tag will be populated in the csv and added to the instance.
- **GcpDiskType:** For the PrepareToDetach operation only. **Standard** or **SSD**.
- **tag[name]** and **tag[value]:** A column is added for every tag found on the AWS instances, except for Velostrata reserved tag names ("name", "managedbyvelostrata", "managedbyvelostratachangetime", "cloudextension", "vcenter", "originvmid", "velostrataapplianceaddresses", "velostratavmid"). The tag names and values are normalized to GCP naming limitations.

## Informational Fields

These fields are for informational purposes only and will be ignored by the runbook:

- OsType
- NumCPU
- MemoryGB
- NumDisks
- ProvisionedSpaceGB
- \*all columns added\* as a result of Right-sizing recommendations



# Rightsizing the Instances in your Runbook

For any Run in Cloud or Migration operations, you must define the target instance type in your runbook inventory file. You can do this manually, but you can also use Velostrata's Rightsizing capability to accomplish this. Using this capability only works for rows whose RunGroup value is set to a positive number.

Rightsizing can provide two types of recommendations for you, as follows:

1. **Performance-based:** this will create a suggestion based on the CPU and RAM options currently selected for the on-premises VM. You can get this recommendation without having previously toggled on rightsizing monitoring.
2. **Cost-based:** this will evaluate both the current CPU and RAM options of the on-premises VM alongside the average usage of this VM over a period of time. To use this option, you must have already toggled on 'rightsizing monitoring' for this group of VMs. It is recommended that monitoring run for at least 48 hours to provide accurate insight, though this could be shorter or longer depending on usage.

Your rightsizing options will automatically be added to your inventory file via a number of new columns that are added. There will be six recommendations made in total. 3 that reflect performance-based options and 3 that reflect cost-based options (assuming you've toggled on monitoring already). Each recommendation will include two columns: one for the recommended instance target and one that estimates the hourly cost to run that VM. There will also be another column that contains the information on when monitoring began. In total, there will be 13 new columns added to your runbook. The naming convention of these new columns will be as follows

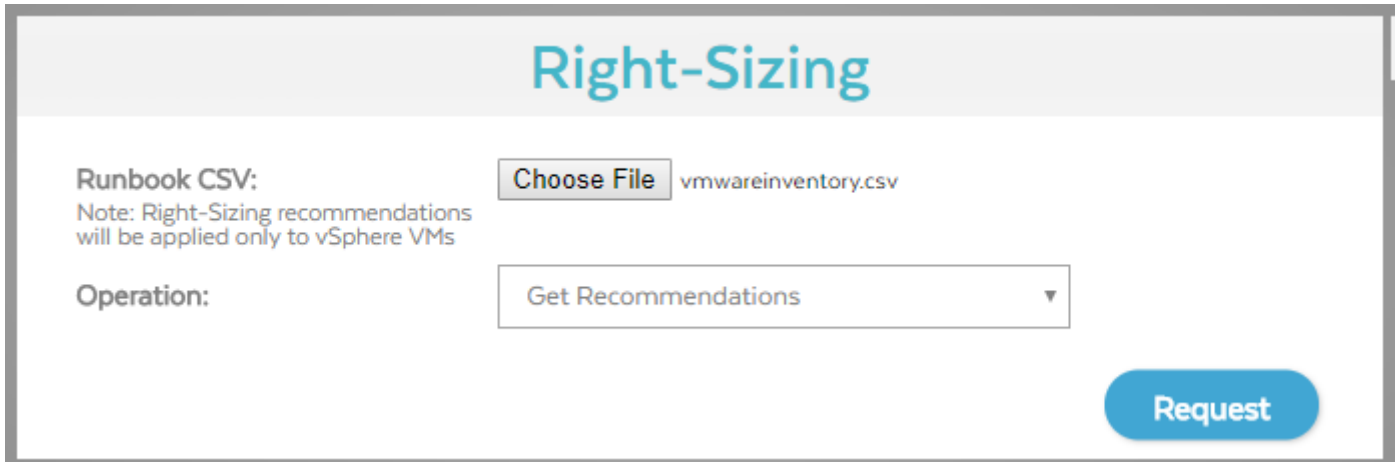
- **PerfOptRec\_type\_N** (N = 1, 2, 3) – the recommended performance-optimized' cloud instance type.
- **PerfOptRec\_cost\_N** (N = 1, 2, 3) – the list price for the recommended cloud instance type.
  - For GCP only, this price includes the [sustained use discount](#) given by Google.
- **VmUsageInfo** – the collected VM usage information – this is the base for cost-optimized recommendations.
- **CostOptRec\_type\_N** (N = 1, 2, 3) – the recommended cost-optimized cloud instance type.
- **CostOptRec\_cost\_N** (N = 1, 2, 3) – the list price for the recommended cloud instance type.
  - For GCP only, this price includes the [sustained use discount](#) given by Google.

Here is an example of what you'll see in your inventory file after running rightsizing:

PerfOptRec_type_1	PerfOptRec_cost_1	PerfOptRec_type_2	PerfOptRec_cost_2	PerfOptRec
basic_a2	0.075	standard_a2_v2	0.087	standard_d

Here is how to perform the appropriate rightsizing actions:

1. From within the Runbook Automation main menu, click the Rightsizing button.
2. Click the **Choose File** button, navigate to your inventory CSV file, and select it.



**Right-Sizing**

Runbook CSV: **Choose File** vmwareinventory.csv

Note: Right-Sizing recommendations will be applied only to vSphere VMs

Operation: **Get Recommendations** ▼

**Request**

3. Using the **Operation** drop-down menu, select the operation you'd like to perform:

- **Start Monitoring:** this will begin to monitor the on-premises VMs in this runbook inventory file so that you can receive cost-based recommendations at a later date.
- **Stop Monitoring:** this will turn monitoring off for the on-premises VMs in this runbook inventory file once you've successfully retrieved your recommendations. The maximum number of VMs that can be monitored at one time is 100, so it is important to turn monitoring off once recommendations are retrieved and runbook jobs are complete.
- **Get Recommendations:** this will take your runbook inventory file and populate the 12 recommendation columns (as detailed above).

4. Once you've successfully performed 'Start Monitoring' and 'Get Recommendations', you can proceed to making your selections by opening the runbook inventory file in your preferred spreadsheet editor.

5. Locate the recommendation columns (listed in the table in the middle of this article) and examine their suggestions. Pick the one that you would like to use for any given row and copy the instance type value.

6. Paste that instance type value (from step 5) into the column for VM that is titled 'TargetInstanceType'.

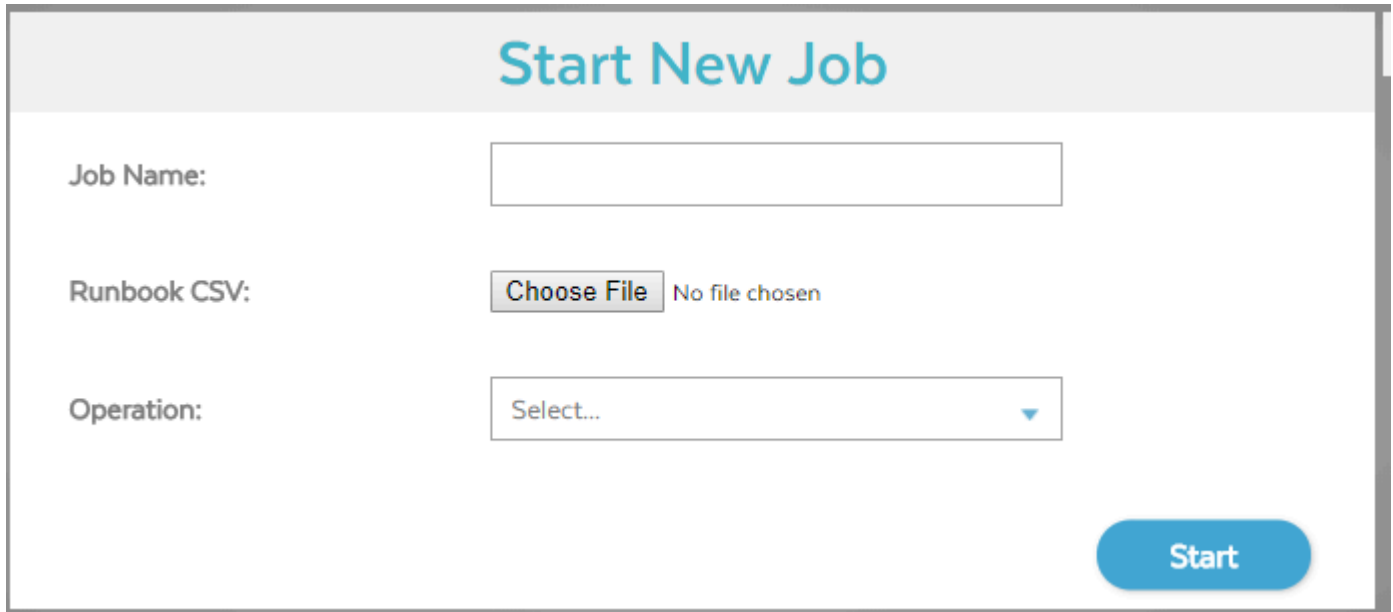
7. You do not need to delete the recommendation columns before executing your runbook job. The runbook will ignore them.

8. Proceed to use this modified runbook inventory file to run jobs (see next sections).

# Creating a Runbook Job

Once you've configured your Runbook Inventory File, it's time to execute that job using Velostrata. Follow these steps:

1. From the Velostrata Runbook Automation page, click **Start New Job**.



The screenshot shows a web form titled "Start New Job" in a light blue header. The form has three main input sections: "Job Name:" with a text box, "Runbook CSV:" with a "Choose File" button and "No file chosen" text, and "Operation:" with a dropdown menu showing "Select...". A blue "Start" button is located at the bottom right of the form.

2. Enter a **Job Name**.
3. Click **Choose File**, then select the inventory CSV file that you previously created and saved.
4. Select the required **Operation** (For more information on available operations, please revisit [this page](#)).
5. Click **Start**.




VELOSTRATA RUNBOOK AUTOMATION

Home

Runbook Automation

Create RunbookRight-SizingStart New Job

Search

Id	Operation	Runbook	Start Time	End Time	Status	Log	Monitor	Action
suse-run-in-cloud-001	RunInCloud		Jul 10, 2018 7:01 AM		Running			<div>Abort</div>

© 2018, Velostrata Ltd. VELOSTRATA.COM  
Terms of Use | Open Source Licenses

Showing 1-1 of 1

First

←

1

→

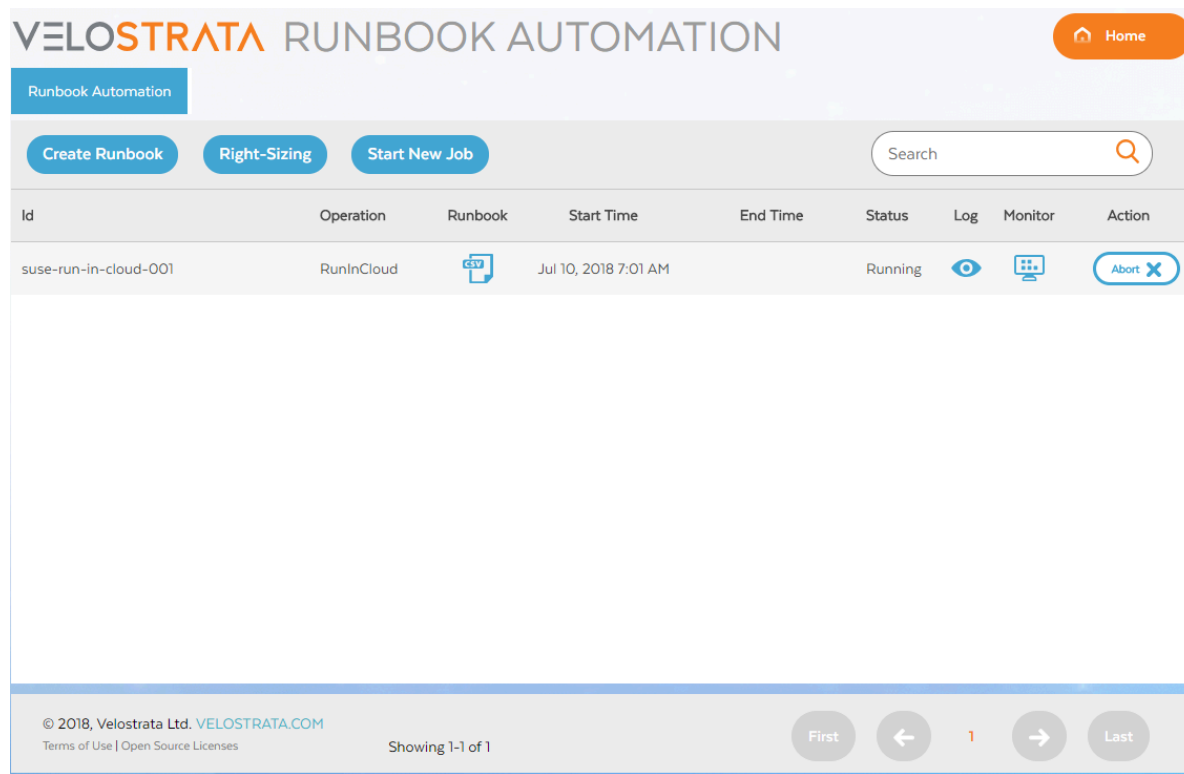
Last

Once you click start, your job will begin to execute. As mentioned previously, the runbook automation actions run in a sequence as defined, and are all re-entrant. That is, if there is any failure in the process, you can fix the problem and run a new job with the same runbook or click the **Re-Run** button that will appear from the screen above (far right).




To abort this runbook job, click the **Abort** button on the far right of the listed job.

# Aborting a Runbook Job

To stop a runbook job that is in progress simply select the required row and click the **Abort** button on the far right.



The screenshot shows the Velostrata Runbook Automation interface. At the top, there's a header with the logo and a 'Home' button. Below the header, there's a navigation bar with buttons for 'Create Runbook', 'Right-Sizing', and 'Start New Job', along with a search bar. The main area contains a table with columns: Id, Operation, Runbook, Start Time, End Time, Status, Log, Monitor, and Action. There is one row with the following data: Id: suse-run-in-cloud-001, Operation: RunInCloud, Runbook: (icon), Start Time: Jul 10, 2018 7:01 AM, End Time: (empty), Status: Running, Log: (eye icon), Monitor: (monitor icon), and Action: Abort (button with an 'X' icon). At the bottom, there's a footer with copyright information, a 'Showing 1-1 of 1' indicator, and navigation buttons for 'First', 'Previous', '1', 'Next', and 'Last'.

Id	Operation	Runbook	Start Time	End Time	Status	Log	Monitor	Action
suse-run-in-cloud-001	RunInCloud		Jul 10, 2018 7:01 AM		Running			<button>Abort X</button>

The reason to abort a runbook job might be because you have discovered an error taking place during this process. You need to resolve this issue and then re-attempt the rest of the actions defined in the runbook. This is easy to do because the runbooks are re-entrant, which means they can be 'aborted' and 're-run' such that the runbook picks up where it left off previously. Some additional details about what happens when you click abort:

- Aborting a Runbook stops the Runbook job execution but does **not** stop the Velostrata Manager tasks that are already running/completed.
- For example, if you have a runbook that is designed to start five test clones one after the other. If, at the time you click abort, two of those test clones are already running then those two test clones will continue to run. The final three test clones, however, will not begin after clicking abort.
- After clicking abort, you will see a new button appear in that row called "Re-run".
- If you click the re-run button (after aborting), the runbook will pick up where it left off. Meaning the two test clones already running will stay running, and two *\*new\** test clones will not begin in addition to those already running. The runbook will instead simply begin by starting the third test clone, which was next in line when the runbook was previously aborted. After that test clone is live, it will create the fourth and then the fifth.

# Monitoring a Runbook Job

To find out more about what is taking place during your runbook automation execution, Velostrata provides two ways to monitor your progress: viewing the log (static) and monitoring (interactive). More on each, below:

## Monitoring

To monitor progress and (potentially) take action during a runbook job, click the icon in the appropriate row under the 'monitor' column.

This brings up a new screen, below, which will auto-update every 5 seconds:

VELOSTRATA

JOB MONITORING

Home

Runbook Automation

Virtual Machines

Start

Stop

Reboot

Move Back

Search

VmID	Target Instance Name	Project	Zone	Private IP	Power State	Run Group	Migration Status
suse11sp3-09194025-ntb	suse11sp3-09194025-ntb	velos-auto-1	europa-west1-d	10.60.1.41	N/A	100	Moving To Target Cloud
suse11sp3-09194025-ntb-clone1531155059	suse11sp3-09194025-ntb-clone1531155059	velos-auto-1	europa-west1-d	10.60.1.46	N/A	100	Moving To Target Cloud

© 2018, Velostrata Ltd. VELOSTRATA.COM

Terms of Use | Open Source Licenses

Show 10 entries

Showing 1-2 of 2

First

←

1

→

Last

From here, you can see a list of all the VMs that were included in this runbook job. Each row contains specific information about what is taking place with that VM during the runbook job. You can select the row and then **perform operations** on that particular VM as well. These actions include:

- **Start:** this will boot up the VM in the cloud if it's currently off.
- **Stop:** this will turn the VM off in the cloud if it's currently on.
- **Restart:** this will restart the VM in the cloud if it's currently on.

## View Log

To view the log, click the 'eye' icon in the row whose runbook job you wish to see.

Viewing the log will give you detailed insight into exactly what is taking place with sections that you can expand in detail.

This is significantly more granular than the 'monitoring' option and should be used for expert troubleshooting, possibly while interacting directly with Velostrata support.

Time	Level	Logger	Message
2018-07-10 14:01:19	INFO	RunBase	Job starting...
2018-07-10 14:01:19	TRACE	VcenterAdapterImpl	translateToVmId("dc-test/vm/suse11sp3-09194025-nTB")
<input type="checkbox"/> details...			
2018-07-10 14:01:19	INFO	RunBase	dc-test/vm/suse11sp3-09194025-nTB - vm-19
2018-07-10 14:01:19	TRACE	VcenterAdapterImpl	translateToVmId("dc-test/vm/suse11sp3-09194025-nTB-clone1531155059")
<input type="checkbox"/> details...			
2018-07-10 14:01:19	INFO	RunBase	dc-test/vm/suse11sp3-09194025-nTB-clone1531155059 - vm-21
2018-07-10 14:01:20	INFO	RunBase	Initiating stop sequence on source VMs...
2018-07-10 14:01:20	INFO	RunBase	stop sequence on source VMs: starting group #100
2018-07-10 14:01:20	TRACE	VelosAdapterImpl	listManagedVms()
<input type="checkbox"/> details...			
2018-07-10 14:01:20	TRACE	VcenterAdapterImpl	getVmPowerStates(["vm-19", "vm-21"])
<input type="checkbox"/> details...			
2018-07-10 14:01:20	DEBUG	Wait	Waiting for stop VMs for PT20M, polling period: PT20S
<input type="checkbox"/> details...			
2018-07-10 14:01:20	INFO	RunBase	stop sequence on source VMs: finished group #100, can continue: true
2018-07-10 14:01:20	INFO	RunBase	Finished stop sequence on source VMs
2018-07-10 14:01:20	TRACE	VelosAdapterImpl	listAllCloudExtensions()
<input type="checkbox"/> details...			
2018-07-10 14:01:21	INFO	RunBase	Initiating run in cloud...
2018-07-10 14:01:21	INFO	RunBase	run in cloud: starting group #100
2018-07-10 14:01:21	TRACE	VelosAdapterImpl	listManagedVms()
<input type="checkbox"/> details...			
2018-07-10 14:01:21	DEBUG	Wait	Waiting for tasks to finish, polling period: PT30S
<input type="checkbox"/> details...			

# Managing Cloud to Cloud Runbook via PowerShell/API

You can manage the cloud-to-cloud runbook jobs using PowerShell or the REST API.

For more information on using these tools, please visit this [article](#).