Vincent Meoc

Constantly monitor your vRealize automation environment with the

“vRA deployment probe workflow”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Ver. | Author | Description | Reviewers |
| 8/1/2015 | 0.5 | Vincent Meoc | Draft done |  |
|  |  |  |  |  |

Contents

[What is it for? 3](#_Toc440037326)

[How to set this up 3](#_Toc440037327)

[1. Respect the pre requisites 4](#_Toc440037328)

[2. Import the vRO Workflow 4](#_Toc440037329)

[3. Create the REST Workflow 4](#_Toc440037330)

[4. Create the vR Ops custom group 4](#_Toc440037331)

[5. Set the input 4](#_Toc440037332)

[6. Create the vR Ops alert: 4](#_Toc440037333)

[7. Create the vR Ops dashboard & notification 6](#_Toc440037334)

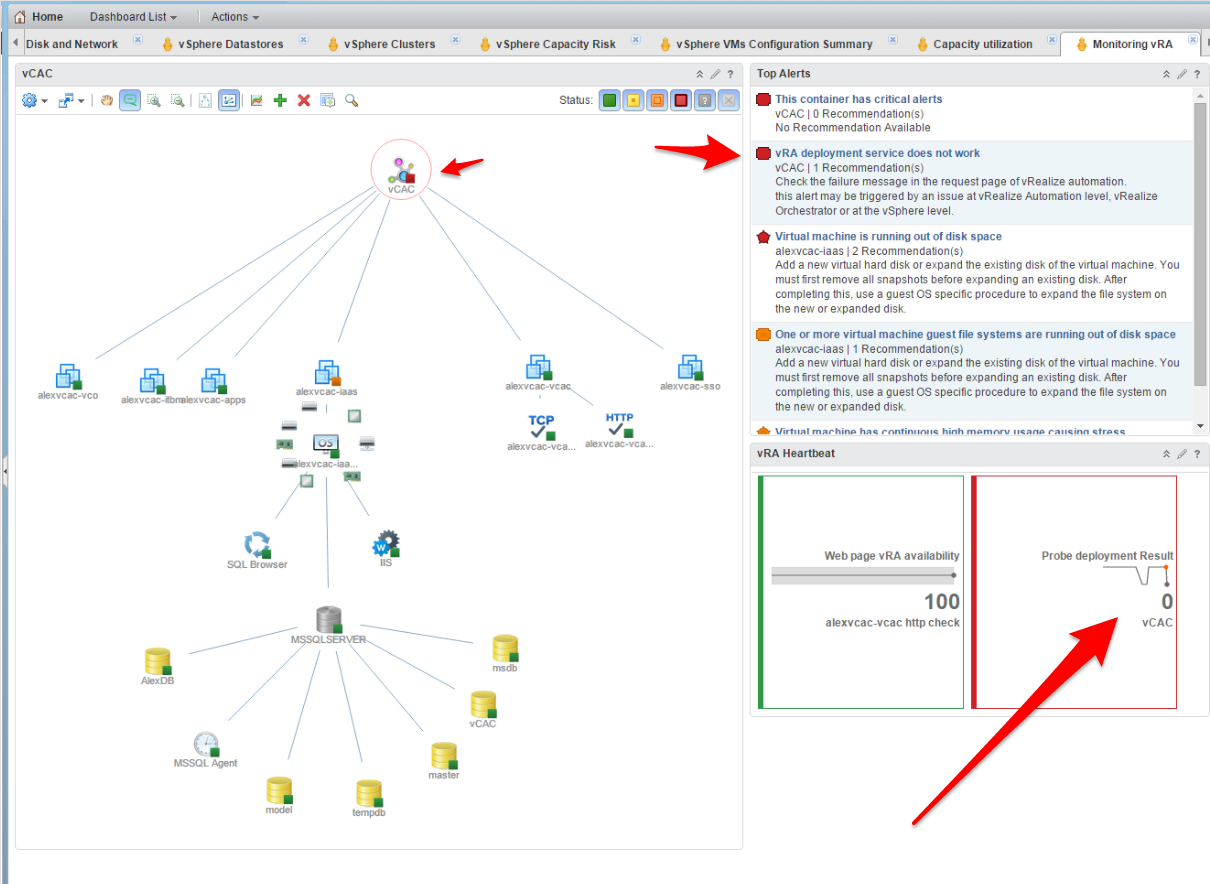
[8. 1 more thing! 7](#_Toc440037335)

[Conclusion 7](#_Toc440037336)

# What is it for?

In a nutshell: this workflow check for vRealize Automation deployment service availability. It launch a VM deployment in vRA, get the deployment result, and send it to vRealize Operations.

Then, from vRealize Operation, you can use this information to send a notification by email and display the impact on a dashboard such as this one:



# How to set this up

Here are the high level steps:

* Respect the pre requisites
* Import the vRO workflow
* Create the REST Workflow
* Create the vR Ops custom group
* Set the input
* Create the vR Ops alert
* Create the vR Ops dashboard & notification

## Respect the pre requisites

You will need vRealize Automation 6 (Not vRA 7), vRealize Operationss 6.x, vRealize Orchestrator 6.x

## Import the vRO Workflow

Nothing specific here. It’s all in the vRealize Orchestrator (vRO) documentation

## Create the REST Workflow

The REST workflow is necessary to add stats to the custom group in vR Ops. The method is named addstat and you will find more information in the vRealize Operations documentation (vR Ops) at the address https://<vROPSserver>/suite-api.

To create the REST workflow in vRO, the [documentation](http://pubs.vmware.com/orchestrator-70/index.jsp#com.vmware.vrealize.orchestrator-use-plugins.doc/GUID-D476E5F2-FEA5-4674-83EC-5110A7D620D6.html?resultof=%2522%2552%2545%2553%2554%2522%2520%2522%2572%2565%2573%2574%2522%2520) describe the process.

## Create the vR Ops custom group

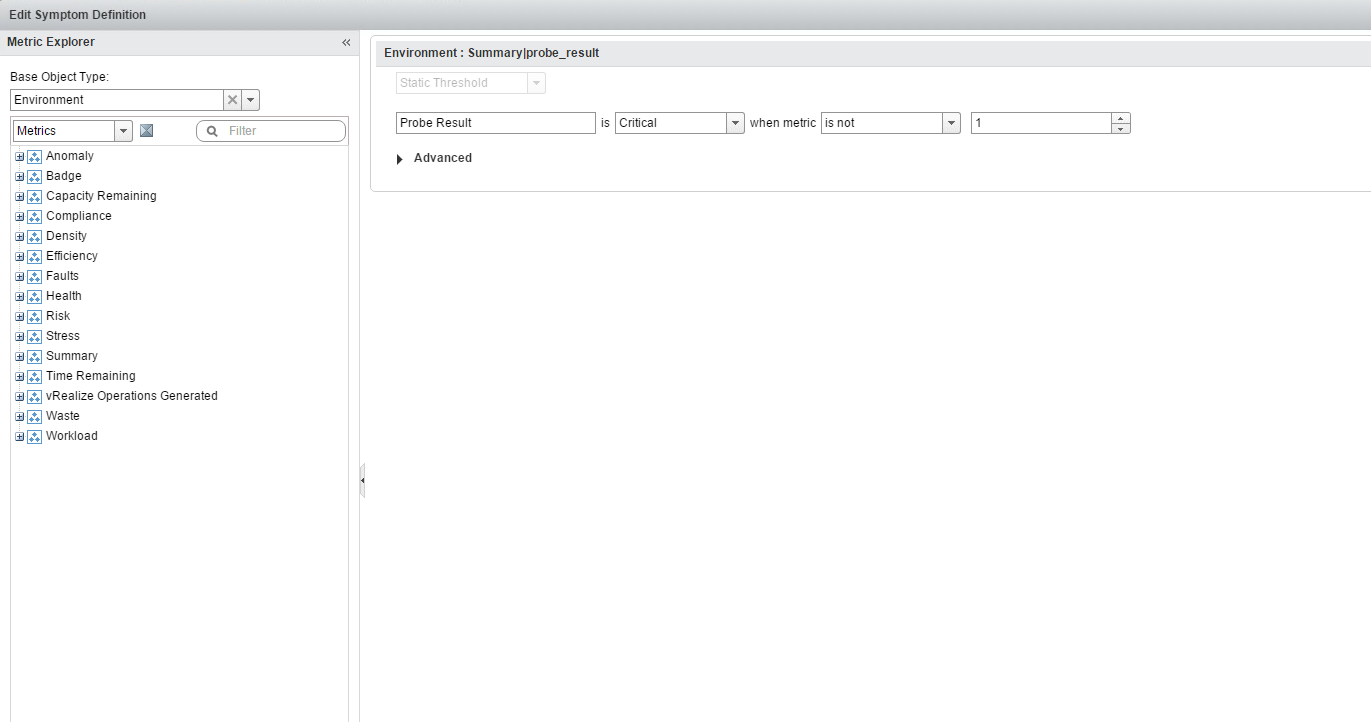
This is a standard action in vR Ops

## Set the input

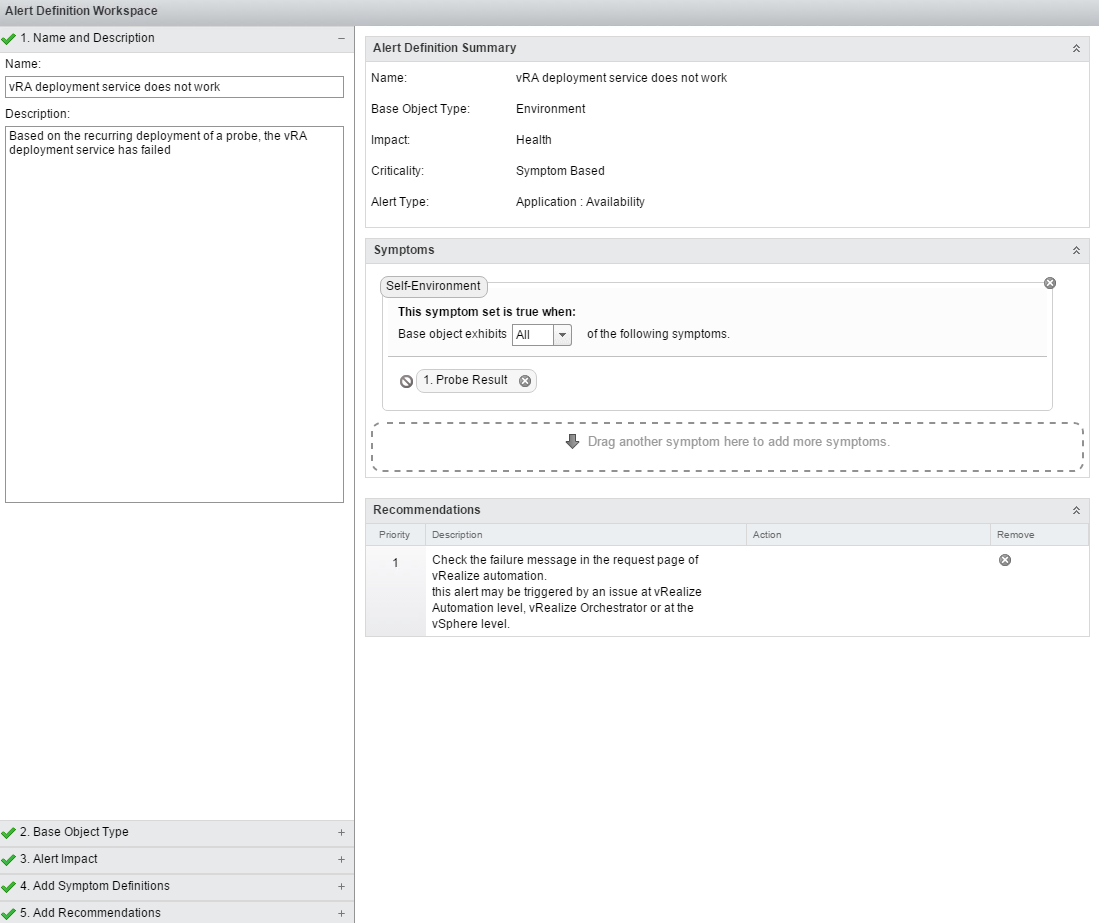
* **Item**: it’s the machine you will use as your probe. Make sure that this probe go through all the process your users’VM go through. By doing that, you will not only test the VM deployment but also the different workflow that this VM trigger during his provisioning.
* **vR Ops resource**: this is the ID of the custom group you will add the deployment results as a metric (summary/probe\_result)
* **restoperation**: you need to point to the REST operation you just created
* **VCACHOST**: the vRealize Automation host you want to monitor
* **DestroyrequestID**: This is the ID of the vRA destroy method to clean up what was provisioned. This is hard coded in the Data formatting task (I know it’s very bad!). I guess that in another vRA environment, the ID will not match and you’ll need to enter your own.

## Create the vR Ops alert:

First, create the symptom alert



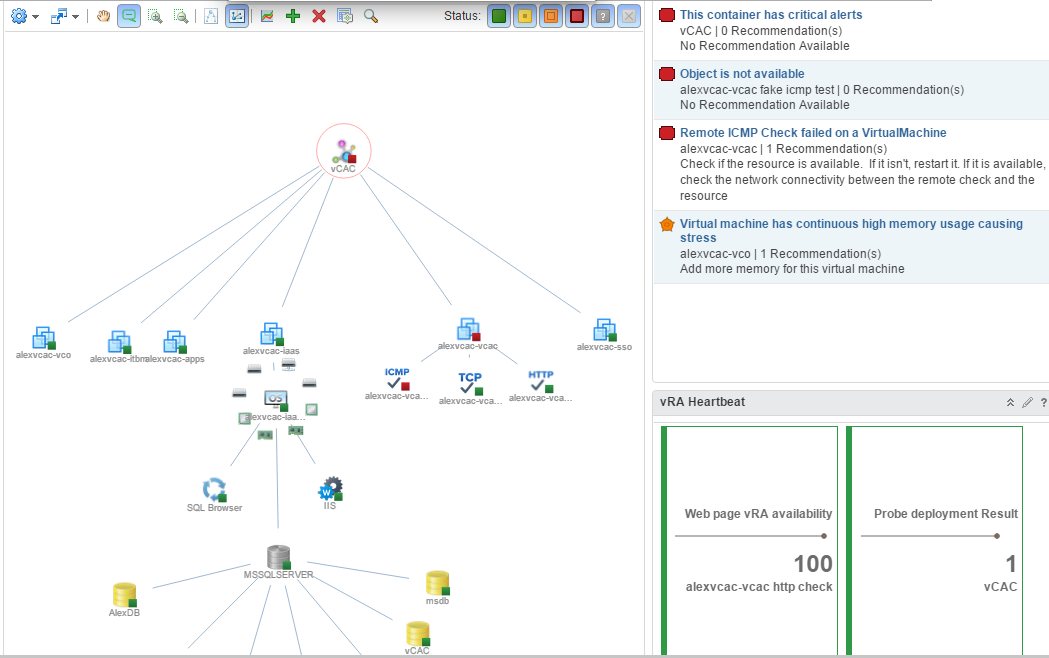
Then, create the alert definition. You’ll need to apply it to your objet type



## Create the vR Ops dashboard & notification

These are standard vRealize Opertions procedure so I won’t go in the details.

As an example, you can find below the dashoard I created.



## 1 more thing!

You will need to schedule this workflow to start every 1H. To do that, just right click on the vRO workflow and follow the Schedule Workflow wizard.

# Conclusion

That’s it. Enjoy!