EPAM Cloud Infrastructure

Using Ansible with EPAM Orchestrator

User Guide

March 2018

Version 1.4

CONTENT

1 ENVIRONMENT CONFIGURATION

To start working with Ansible via EPAM Cloud you should download and configure Maestro CLI.

Due to Ansible specifics, it can be used only on Linux-based workstaions.

To use Ansible more effectively, we strongly recommend to set up several workspaces, so that you could use different settings for different purposes.

To enable this, provide the following strings for the **~/.bashrc** file:

```
export MAESTRO_CLI_HOME={path_to_cli_dir}

export PATH=$PATH:$MAESTRO_CLI_HOME/bin
```

Please note that if necessary, you can use different credentials in different workspaces. By default, the credentials are taken from the default.cr file stored in **%MAESTRO_CLI_HOME%** folder. To set up specific credentials for a certain workspace, run **or2access** command from this workspace.

After the preparation is completed, invoke **or2-ansible-init (or2ai)** command. An example output is given in the screenshot bellow:

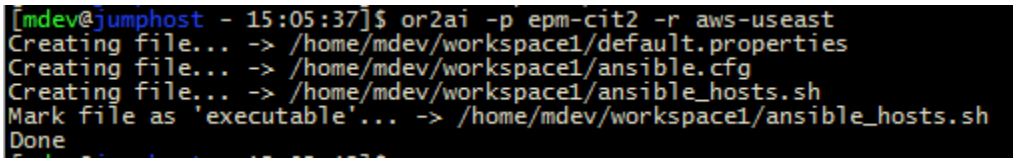

Figure 1 - or2-ansible-init response

The command sets up all the required configuration files in the current user directory:

- ansible.cfg – Ansible configuration file
- default.properties – contains default values for the required CLI parameters
- ansible_hosts.sh – executable script to get Ansible dynamic inventory

Below is an example screenshot of a current directory content:

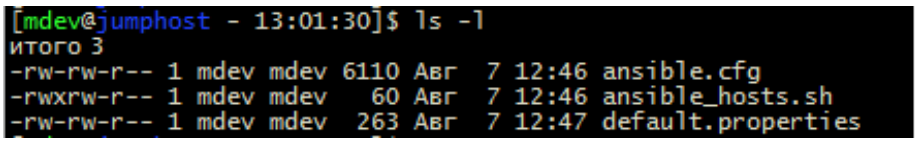

Figure 2 - Ansible configuration files

2 ANSIBLE HOSTS

One can describe, include/exclude hosts to/from groups using **or2-ansible-hosts.** All hosts are included into ‘default’ group by default.

To include a host into group use **or2-ansible-hosts (or2ah)** with **include** option. See the example in the screenshot below:

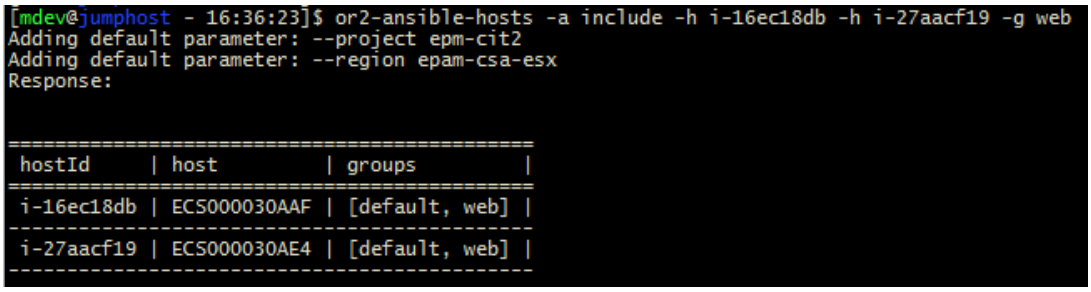

Figure 3 - Include host into group. Example

To exclude a host from a group use **or2-ansible-hosts (or2ah)** with **exclude** option. See the example in the screenshot below:

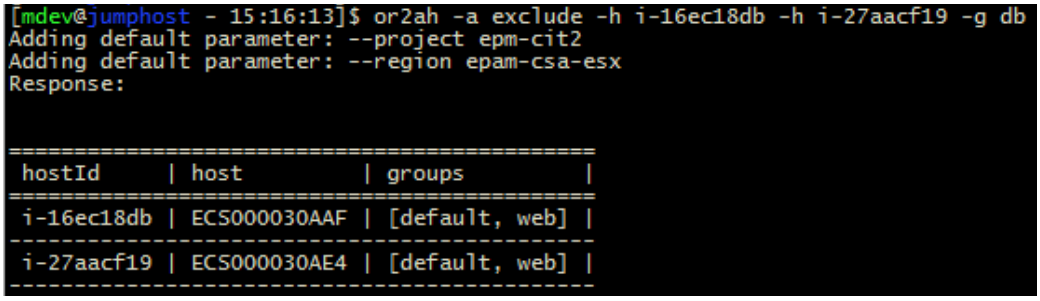

Figure 4 - Exclude host from a group. Example

To describe all hosts from a group use **or2-ansible-hosts (or2ah)** with **describe** option. See the example in the screenshot below:

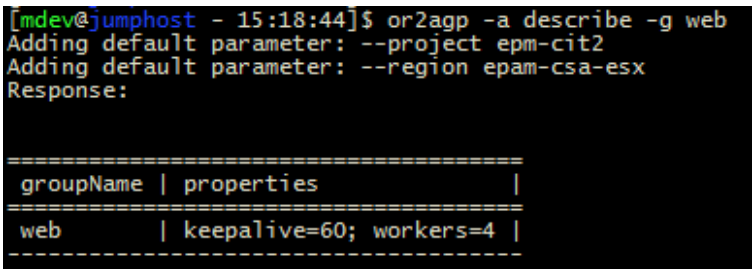

Figure 5 - Describe group hosts. Example

You cannot exclude host(s) from the ‘default’ group. A warning message will appear if you try to exclude host(s) from the ‘default’ group.

3 ANSIBLE GROUPS

Groups are used to organize hosts. All hosts are included into the 'default' group by default. You can create, delete and describe groups using **or2-ansible-group (or2ag)** command**.**

To create a group use **or2-ansible-group (or2ag)** with **create** option. See the example in the screenshot below:

```
[mdev@jumphost - 16:33:00]$ or2ag -a create -g loadbalancers
Adding default parameter: --project epm-cit2
Adding default parameter: --region epam-csa-esx
Response: Group with name 'loadbalancers' was successfully created
```

Figure 6 - Create new group. Example

To delete an existing group use **or2-ansible-groups (or2ag)** with **delete** option. See the example in the screenshot below:

```
[mdev@jumphost - 16:34:19]$ or2ag -a DELETE -g loadbalancers
Adding default parameter: --project epm-cit2
Adding default parameter: --region epam-csa-esx
Response: Group with name 'loadbalancers' was successfully deleted
```

Figure 7 - Delete a group. Example

To describe all groups use **or2-ansible-groups (or2ag)** with **describe** option. See the example in the screenshot below:

```
[mdev@jumphost - 16:34:49]$ or2-ansible-groups
Adding default parameter: --project epm-cit2
Adding default parameter: --region epam-csa-esx
Response:


==========
 name    |
==========
 web     |
----------
 db      |
----------
 default |
----------
```

Figure 8 - Describe groups. Example

You cannot delete the 'default' group from the list. A warning message will appear in case you try to delete the 'default' group.

4 DYNAMIC INVENTORY

To retrieve Ansible dynamic inventory, use the **or2-ansible-dynamic-inventory (or2adi)** command specifying the project and region for which data should be returned. Optionally, you can add the **-t (--pretty)** parameter to obtain the JSON data in the easily readable format. The command returns dynamic inventory data consisting of two major parts: *Meta* & *Groups*.

A fragment of the meta part of dynamic inventory is shown in the screenshot below:

```
{
  "_meta": {
    "hostvars": {
      "ECS00001****.epam.com": {
        "ansible_ssh_user": "Auto_Automation@epam.com",
        "ep_chefattributes": "graylog_server_name=ecs00001****.epam.com,log_files=/var/log/chef/client.log",
        "ep_chefrole": "graylog-client",
        "description": "log service client",
        "ansible_ssh_private_key_file": "~/.***********.pem",
        "ansible_user": "Auto_Automation@epam.com"
      },

      "ECS00001****.epam.com": {
        "stack-id": "GraylogServer-6dfeb80b",
        "ansible_ssh_user": "Auto_Automation@epam.com",
        "ep_chefrole": "graylog2_epc,graylog-client-remove",
        "description": "Project Graylog Server",
        "eo_service": "graylog-server",
        "ansible_ssh_private_key_file": "~/.***********.pem",
        "ansible_user": "Auto_Automation@epam.com"
      },
```


Figure 9 - Ansible dynamic inventory. Meta

Meta contains host variables that can be used in playbooks. To add new host variable use **or2-set-instance-properties (or2setp)** command.

Inventory groups are used to organize several hosts into a group with group attributes that can be used in playbooks. All hosts are automatically included in the default group. Additionally, groups can be created manually (see Ansible Groups above). Also, each **or2adi** command run returns dynamic groups – groups of hosts organized by a common attribute and existing at time of the command run. The following attributes are used to form dynamic groups:

- Instance shape
- Instance image
- Instance SSH key
- User for connection
- Instance property (only non-secure properties are used in forming dynamic groups)
- Instance tag
- Instance ID

The screenshot below shows the default group, and two dynamic groups formed by a property and a user for connection:

Figure 10 - Ansible dynamic inventory. Groups

Note: Ansible dynamic inventory can also be retrieved by running the **ansible_hosts.sh** script. This script invokes the **or2adi** command removing the strings with the “Adding default parameter” information from the output.

Ansible connects to VMs via **SSH**. Make sure the private key file is located in **~/.ssh/** directory.

5 ANSIBLE GROUP PROPERTIES

To set group properties use **or2-ansible-group-properties (or2agp)** with **set** option. See the example in the screenshot below:

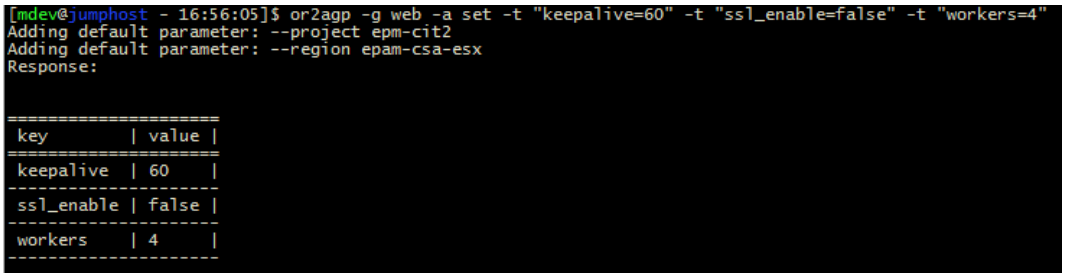

Figure 11 - Set group properties. Example

To delete the properties of an existing group use **or2-ansible-group-properties (or2agp)** with **delete** option. See the example in the screenshot below:

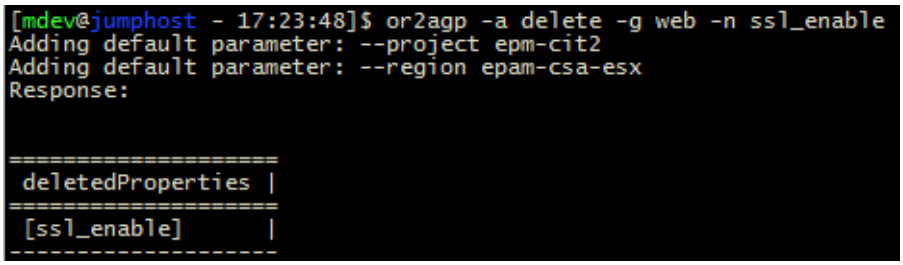

Figure 12 - Delete group properties. Example

To describe all group properties use **or2-ansible-group-properties (or2agp)** with **describe** option. See the example in the screenshot below:

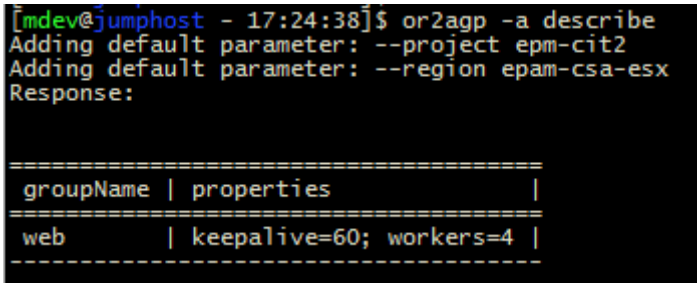

Figure 13 - Describe group properties. Example

6 ANSIBLE AUDIT

To see all the detailed audit use **or2audit** as described below:

```
or2audit –g acs
```

The example output is described in the screenshot below:

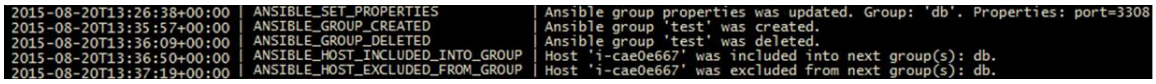

Figure 14 - Ansible audit events. Response example

7 TESTING ANSIBLE SERVICE

To verify that Ansible service correctly configured your VMs execute **ansible –m ping** command, specifying the name of the group (or host name) as shown below:

```
ansible web –m ping
```

Example output is shown in the screenshot below:

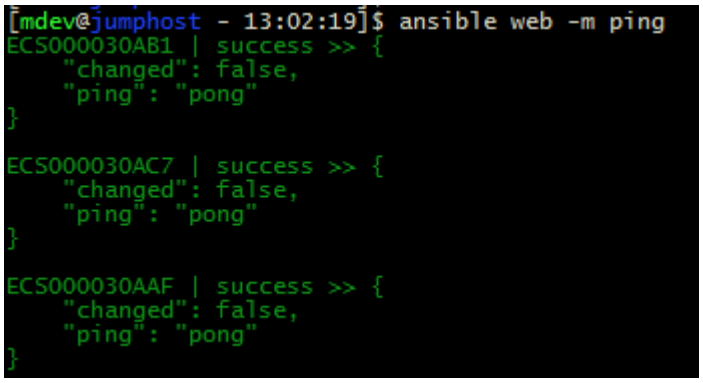

Figure 15 - Ansible ping. Example

To execute commands on remote hosts, run **ansible -a** command, specifying the name of the group (or host name) as shown below:

```
ansible web –a “cat /etc/issue”
```

Example output is shown in the screenshot below:

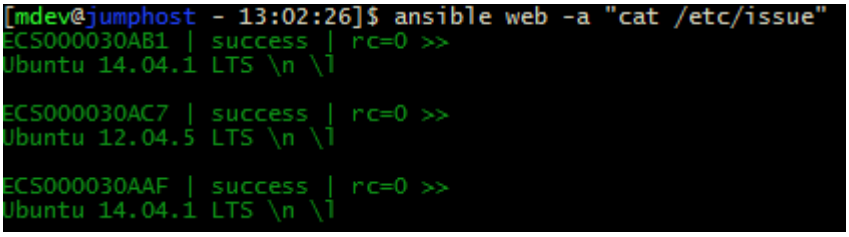

Figure 16 - Ansible remote command execution. Example

TABLE OF FIGURES

VERSION HISTORY

Version	Date	Summary
1.4	March, 2018	The usage limitation (for Linux only) is described.
1.3	June, 2017	Ansible dynamic inventory info updated
1.2	December 16, 2016	Classification changed from Confidential to Public, approved by Dzmitry Pliushch
1.1	June, 2016	Configuration info updated
1.0	September 5, 2015	Initially published