Procedure to Modify Facter Parameter Values in CPS

Contents

Introduction

This document describes the procedure to modify facter parameter values in Cisco Policy Suite (CPS) Replica Set.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Linux
- CPS
- Puppet

Cisco recommends that you must have privilege Root access to CPS CLI.

Components used

The information in this document is based on these software and hardware versions:

- CPS 20.2
- MongoDB v3.6.17
- UCS-B

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

Facter is Puppet’s cross-platform system profile library. It discovers and reports per-node facts, which are available in your Puppet manifest as variables. Puppet supports the hold of multiple values as an environment variable. This feature is supported in Puppet by the use of faster. In Puppet, facter is a standalone tool that holds the environment level variable. It can be considered similar to the env variable of Bash or Linux. Sometimes there can be an overlap between the information stored in facts and the environment variable of the machine. In Puppet, the key-value pair is known as “fact”. Each resource has its own facts and in Puppet, the user has the leverage to build their own custom facts.

The command facter can be used to list all the different environment variables and their associated values.

The reason why facter is important for Puppet is that facter and facts are available throughout Puppet code as a **global variable**, which means they can be used in the code at any point in time without any other reference.

Problem

Suppose a CPS script uses facter parameter values for some purpose like to set a threshold value for alert generation, if you want to fine-tune the threshold values based on your network, then you must modify the respective facter parameter value accordingly.

Consider the sample CPS script **/var/qps/install/20.2.0/scripts/bin/support/snmp-traps/process-traps/gen-gx-drop-trap.sh** and the alert **Gx average Message CCR-X processing increased**.

As you can see, the script uses facter value at first if configured.

```
# If threshold configured in facter, get that value, else use default 20ms
TRAP_AVG_LEVEL=200
```

So, here the script doesn't consider the input provided as TRAP_AVG_LEVEL=200", whereas it considers facter value of 20ms.

Run this command from pcrfclient to display respective facter parameter values.

```
#facter | grep ccr
gx_alarm_ccr_i_avg_threshold => 20
gx_alarm_ccr_t_avg_threshold => 20
gx_alarm_ccr_u_avg_threshold => 20
```

Procedure to Modify Facter Parameter Values

Here are the steps to modify the facter parameter values to 200ms.

1. Approach for CPS hosted in OpenStack.

Step 1. Backup of current facter configuration.

Run this command from Cluster Manager (ensure that directory **/mnt/backup** exists).

```
# curl -X GET http://installer:8458/api/system/config/config/ -o /mnt/backup/ facter-
```

```
config_$(date +%Y-%m-%d).yaml
```

Step 2. Preparation of YAML file with proper threshold parameter values. Prepare a **.yaml** file in ClusterManager.

```
# vi gx_alarm_threshold.yaml
gxAlarmCcrIAvgThreshold: "200"
gxAlarmCcrUAvgThreshold: "200"
gxAlarmCcrTAvgThreshold: "200"
```

Step 3. Run this command from Cluster Manager to update the threshold parameters.

Note: This command must be executed from the same directory where gx_alarm_threshold.yaml file is placed. Here it's **/tmp**.

```
[root@installer tmp]# curl -i -X PATCH http://installer:8458/api/system/config/config -H
"Content-Type: application/yaml" --data-binary "@gx_alarm_threshold.yaml"
HTTP/1.1 200 OK
Date: Fri, 21 Jan 2022 06:45:20 GMT
Content-Length: 0

[root@installer tmp]
```

Step 4. Verify the change in threshold parameter values. Run this command from pcrfclient.

```
#facter | grep ccr

Expected Output:
gx_alarm_ccr_i_avg_threshold => 200
gx_alarm_ccr_t_avg_threshold => 200
gx_alarm_ccr_u_avg_threshold => 200
```

2. Approach for CPS hosted in VMWare.

Step 1 . Log into ClusterManager and modify the value of the necessary parameter in **/var/qps/config/deploy/csv/Configuration.csv.** Consider these sample parameters here with a current value of 20.

```
[root@installer ~]# cat /var/qps/config/deploy/csv/Configuration.csv | grep ccr
gx_alarm_ccr_i_avg_threshold,20,
gx_alarm_ccr_t_avg_threshold,20,
gx_alarm_ccr_u_avg_threshold,20,
[root@installer ~]#
```

Run this command and modify the value of sample parameters in the **Configuration.csv** file.

```
[root@installer ~]#vi /var/qps/config/deploy/csv/Configuration.csv
```

Run this command to check if values are changed in the **Configuration.csv** file.

```
[root@installer ~]# cat /var/qps/config/deploy/csv/Configuration.csv|grep ccr
gx_alarm_ccr_i_avg_threshold,200,
gx_alarm_ccr_t_avg_threshold,200,
gx_alarm_ccr_u_avg_threshold,200,
[root@installer ~]#
```

Step 2. Run this command to import the new configuration to ClusterManager.

```
[root@installer ~]# /var/qps/install/current/scripts/import/import_deploy.sh
```

Step 3. Run this command to verify the change in ClustManager facter values.

```
[root@installer ~]# facter | grep -i ccr
gx_alarm_ccr_i_avg_threshold => 200
gx_alarm_ccr_t_avg_threshold => 200
gx_alarm_ccr_u_avg_threshold => 200
[root@installer ~]#
```

Step 4. Run this command to rebuild the CPS package.

```
[root@installer ~]# /var/qps/install/current/scripts/build_all.sh
```

Step 5. Run this command to download all the Puppet scripts, CPS software, **/etc/hosts** files and update each VM with the new software from Cluster Manager.

```
[root@installer ~]# /var/qps/install/current/scripts/upgrade/reinit.sh
```

Step 6. Login to pcrfclient and run this command to verify the changes in facter values.

```
[root@dc1-pcrfclient01 ~]# facter | grep ccr
gx_alarm_ccr_i_avg_threshold => 200
gx_alarm_ccr_t_avg_threshold => 200
gx_alarm_ccr_u_avg_threshold => 200
[root@dc1-pcrfclient01 ~]#
```