README.md

pracdev-playbook-automation

The Playbook Automation service

This tool allows you to create actions and playbooks based on rest api action type.

Installation

Download the respective executable

For Mac:

```
playbook_automation
```

For Win:

```
playbook_automation.exe
```

Usage

```
playbook_automation -pc 10.48.29.178 -2 -a -n dev87840
```

Provide the Prism Central IPs and ServiceNOW, the credentials for the same will be obtained during Runtime.

```
usage: main.py [-h] -pc PRISM_CENTRAL [-c CONFIG] [-t {1,2,3,4,5}] -n
               SNOW_INSTANCE_NAME (-i | -d | -s | -a) (-1 | -2)

optional arguments:
  -h, --help            show this help message and exit
  -pc PRISM_CENTRAL, --prism_central PRISM_CENTRAL
                        Prism Central IPs separated by comma
  -c CONFIG, --config CONFIG
                        Action and Playbook Configuration file path
  -t {1,2,3,4,5}, --no_of_threads {1,2,3,4,5}
                        Number of worker threads to be created
  -n SNOW_INSTANCE_NAME, --snow_instance_name SNOW_INSTANCE_NAME
                        ServiceNOW Instance name
  -i, --ignore          Ignore existing playbooks and create new
  -d, --delete          Delete existing playbooks and create new
  -s, --skip            Skip creation for existing playbooks
  -a, --ask             Interactively ask for each existing playbook
  -1, --critical_21     TOP 21 critical alerts
  -2, --critical_all    ALL critical alerts
```

```
playbook_automation -pc 10.48.29.178 -2 -a -n dev87840
```

This tool is configured by default to create playbooks for the 21 critical alerts or All critical alerts . Pass -1 for 21 alerts and -2 for all critical alerts.

The tool can be used to create various alerts using different configuration files in the following yml format.

```
action_templates:
- id: 1
```

```
  action_name: "Send to ServiceNow"
  action_type_name: "service_now"
  desc: "Push alerts to ServiceNOW"
action_rules:
  1:
    - action_rule_name: “$playbook_name”
      trigger_type: “alert_trigger”
      alert_id: “$”
      alert_severity: “$”
      actions:
      - action_template_id: 1
```

The values with $ can be replaced and customised. Pass the YML file in the -c parameter and run the tool.

Example Config file

```
action_templates:
- id: 1
  action_name: "Send to ServiceNow"
  action_type_name: "service_now"
  desc: "Push alerts to ServiceNOW"
  max_retries: 2
action_rules:
  1:
  - action_rule_name: "SNOW Alert MetadataDiskMountedCheck"
    trigger_type: "alert_trigger"
    alert_id: "A101055"
    alert_severity: "[\\\"critical\\\"]"
    actions:
    - action_template_id: 1
```