**StackStorm**

**version 0.0.2**

**Note :**

**reference code : https://github.com/dilshan23/stackstorm\_Slack**

**Task : Getting messages from slack and filter for sending email and further filtering with a Rule (only yahoo.com) and send email and respond back to slack(Please see the attached video)**

First Steps:

1.Install StackStorm(st2) and run st2 locally and set a Slack app

use docker

first git clone st2 docker repository and run $sudo docker-compose up -d

and after $sudo docker-compose exec st2client bash

You can go to [http://localhost](http://localhost/) and see the st2 UI ..u can login with admin password and can change later in st2 cli

2.Design and install the project pack

**Pack**: is a code repository (usually resides in remote repository like github) ..It consists of folders of actions,sensors and rules..also pack.yaml file which include info about pack

you can install in st2 cli : st2 pack install https://github.com/dilshan23/stackstorm\_Slack

and edit config yaml file and move to config folder in st2

ex:

$nano packs/slack\_dilshan/slack\_dilshan.yaml

$cp packs/slack\_dilshan/slack\_dilshan.yaml /opt/stackstorm/configs/slack\_dilshan.yaml

note : install nano also $ apt-get update and apt-get install nano -y

its good practise to reload st2 after some change

3.Designing the Pack

in this scenario we write a sensor,py file which polls slack messages in some interval in this case I set to 10 seconds ..and use sensor\_service.dispatch() method to inject a **Trigger with a payload designed to pass to an action..**This is coordinated by slack.yaml file with appropriate pack name and trigger name.

Then we define rule.yaml file inside we specify which **Action it should trigger**

**In Action** script we have main method run() where we specify the slack sending logic..This is inherited form st2 base class Action

**Rule**

name: "slack\_respond\_to\_email\_send"

pack: "slack\_dilshan"

description: "This rule scans the Slack for email send instructions ,send an email and respond back to channel"

enabled: true

trigger:

type: "slack\_dilshan.new\_update"

parameters: {}

criteria:

trigger.text:

type: "contains"

pattern: "yahoo" ### Rule critieria

action:

ref: "slack\_dilshan.send\_message"

parameters:

text: "{{trigger.text}}"

4. Design methods for sensors

We can use webhooks in case we can create some endpoints in st2 server with api tokens..Or we can use above Polling method

**Logs**

**Triggers**

In ui we can see this in triggers Tab

{

"trigger": "slack\_dilshan.new\_update",

"occurrence\_time": "2023-07-20T09:40:36.939000Z",

"status": "processed",

"id": "64b90114fc61a76e9eec05d6",

"payload": {

"text": ":dddd@test.com|dddd@test.com"

}

}

status is "processed" indicates that the action triggered by "slack\_dilshan.new\_update" has been successfully executed.

References:

https://github.com/surfer190/fixes/blob/master/docs/stackstorm/stackstorm.md