

Documentation for script to integrate tool through json

A python script has been developed which can be used to add a new tool to deakin detonator toolkit.

To run the script you have to use the command:

```
python create_script.py --name {tool_name} --json_file {path_to_json_file} --description {description of tool}
```

The json file is an integral part of the script and would be used to specify the different flag options, their values and types. The json file would contain multiple records with each record corresponding to a flag for the tool. A sample record looks like:

```
"arg1": {  
    "name": "depth",  
    "req_flag": true,  
    "flag_val": "d",  
    "have_preset": false,  
    "preset_vals": "",  
    "label": "Max depth",  
    "is_required": true  
},
```

Where,

name: specifies the name of the flag, like ip, port, speed

req_flag: specifies if a flag value requires a value to specify the flag value or not. This is required as some tools have flags which require a value for specifying they have been initialised. For example, in nmap to specify a port, we need to precede the port value with “-p” value (nmap -p 2000) so here req_flag will be set as true for the port flag. Some commands do not need such values, like in ping, the ip you need to echo need not be preceded by any flag (ping google.com) so for this req_flag will be False

flag_val: The value of the flag that has to precede the argument value. In case req_flag is false, it can be set to None. Using the last example, for port flag of nmap, the flag_val will be p while for ping it will be None.

Have_preset: Some flags of commands have a set of predefined values and they only accept one of those values as input. For example, in nmap tool, the speed flag only accepts values in ["T0", "T1", "T2", "T3", "T4", "T5"]. Thus, for this flag, have_preset would be set as true. For the same command, it has a flag ip, which accepts any value and thus the value for “have_preset” will be False for that flag

preset_vals: This will contain the list of values if a flag can only accept a finite set of values. It can be set to None if Have_preset is False. Following the previous example, the speed flag will have the values ["T0", "T1", "T2", "T3", "T4", "T5"] in this field.

label: This will be the label that will be visible for the flag on the toolkit, and thus is just for user understanding.

Is_required: This field can be used to specify if the given flag is mandatory for the successful execution of the command. For example, numTopPorts is not a mandatory flag for nmap while ip is a mandatory flag whose value must be given.

Using this guide, json can be created for the tools which have to be integrated into the toolkit. Some json files are already created in the test folder of scriptBuilder component which can be used as reference for building json Files.