

BKM for Script 2

The provided code allows creation and updating the test result in **test cycle #n** for each test case present in a given **test cycle #n-1**, with the same status and reason values in **test cycle #n-1**. The input is a JSON file that can be easily modified to suit the user's needs and the script can be run using either Basic AUTH or Kerberos AUTH authentication APIs.

- The script updates the "status.reason" field for each TR in **test cycle #n** with the corresponding value in **test cycle #n-1**. If a Test Result with same configuration is not present for a particular test case in **test cycle #n**, then it will create a new Test Result with **test cycle #n-1** values.

i.e. status.reason = copied from previous cycle

- In both the cases (update/create) of a Test Result in **test cycle #n**, it will set the "updated_reason" field to "leveraged#test cycle name #n-1"

i.e. updated_reason = leveraged#test_cycle_name #n-1

JSON File Inputs:

- a) **CSV_PATH_PREV**: It is the path to csv file which contains all the TC_IDs and their respective TC_Title, configuration, test_cycle_name and status.reason value for test cycle #n-1.

NOTE: Order should be maintained in the csv file [TC_ID, TC_Title, Configuration, Test_Cycle_Name, Status.Reason] and first row should contain the Title of each column.

- b) **Test Cycle Prev**: It is the name of previous test cycle (test cycle name #n-1).
- c) **Test Cycle Curr**: It is the name of current test cycle (test cycle name #n). The same provided value will be written to test_cycle field of newly created test results.
- d) **Link-Type**: It should be "parent-child" in case of updating and creating the TR's.
- e) **Release_Aff**: It will be the program release name to which TC refers, example: bios.alderlake
- f) **Owner**: User-id of person making the changes, example "namanaga"
- g) **Reason_other**: This field is required when status_reason is set to "blocked.other"
- h) **Status_Reason**: It is a list of status.reason values which will be considered to fetch Test Case Ids from provided csv file. It can also be treated as a list of actual result values.

There are two Python Scripts included in the folder: "script2_preprod.py" for testing on the pre-prod HSDS server, and "script2_prod.py" for use on the HSDS production server. These two files fetch input from JSON file.

It should be noted that values provided in the JSON file will be applied to all the test results created or updated. Additionally, if using Basic AUTH, valid credentials can be obtained using provided link <https://hsdes.intel.com/appstore/token/>.

CSV File Example:

	A	B	C	D	E	F
1	TC_ID	TC_Title	configuration	test_cycle	status.reason	
2	1309611366	[Code inspect][bios.cpu_pm][Division_By_Zero]Verify through Code Inspection_bios.l bios.lunarlake-host_based_static_code_analyzer_env		bios.a21.A0 PO Exit.Client-BIO complete.fail		
3	1309611784	[Code inspect][bios.cpu_pm][Bad_Shift]Verify through Code Inspection_bios.lunarlake-host_based_static_code_analyzer_env		bios.a21.A0 PO Exit.Client-BIO complete.fail		
4	1309611785	[Code inspect][bios.cpu_pm][Division_By_Zero]Verify through Code Inspection_bios.lunarlake-host_based_static_code_analyzer_env		bios.a21.A0 PO Exit.Client-BIO blocked.si_bug		
5						
6						
7						
8						
9						

Key Points:

- In the test_cycle_name column in csv file, only single test cycle name should be present.
- The order of columns as mentioned above should be strictly followed.
- Test Case should have valid **Owner-Team** value.
- JSON and Python file should be in same folder and JSON file name should be **"script2_config.json"**.
- No change is required in Python file, user needs to only change the JSON input file as per the need.

Observations:

- In a single API call to update or create, only 25 test results can be created or updated.
- For a csv containing 700-800 TC IDs, the script may take up to 20 mins to process and execute.