

# VAQA Quickstart

## Table of Contents

1. What is VAQA?
2. How to CVS checkout VAQA?
3. VAQA hierarchy explained.
4. VAQA test run phases.
5. How to run BASELINE test using VAQA?
6. How to run VAQA encryption test case?

### 1. What is VAQA?

VAQA is test automation harness written in python.

It's written to simplify the task of running a test or set of tests multiple times with different combinations of configuration parameters.

### 2. How to CVS checkout VAQA?

```
# cvs login
# cvs co tools/harness/vaqa
```

### 3. VAQA hierarchy explained.

VAQA

- test-cases        :- This is vaqa test case directory. User is supposed to add new test cases in this directory.
- config            :- This contains global configuration file. i.e. GLOBAL.conf
- test-results      :- Test case result log files are saved at this location.
- Engine            :- VAQA harness library is written in this directory.

### 4. VAQA test run phases.

#	Test Run Cycle	comments
1.	Init Phase	create FS
		mount FS
		add GP
2.	Run Phase	execute test case
3.	Cleanup Phase	delete GP
		umount FS

- i. Init Phase        :-
  - Read config files (GLOBAL + LOCAL ) and generate merged configuration . [ LOCAL takes precedence over

*GLOBAL ]*

*i.e. If GLOBAL.conf has POLICIES=aes256 and test specific local configuration file has POLICIES=encrypt\_all then test case will be run with*

*POLICIES set to 'encrypt\_all'*

- Create FS ( optional )
- Mount FS ( optional )
- Add GP ( optional )
- Export all dictionary values as environment variables.

ii. Run

- Execute the case using python subprocess module and create test log files in 'test-results' directory.

iii. Cleanup Phase

- Delete GP ( optional )
- Unmount FS ( optional )

## 5. How to run baseline test ?

- Add 'SINGLE\_RUN=1' in test specific LOCAL configuration file.

Example:

A. Create a test case in 'test-cases' dir.

```
# mkdir VAQA/test-cases/test
```

```
# cat VAQA/test-cases/test/test1.sh
```

```
echo "This is test1.sh script"
```

```
echo "USER_VAR = $USER_VAR"
```

```
echo "Run 'hostname' command."
```

```
hostname
```

B. Create a test case specific LOCAL config. file.

```
# cat VAQA/test-cases/test/test1.sh.conf
```

```
[FIXED]
```

```
MNTPT=/gp
```

```
SINGLE_RUN=1
```

```
USER_VAR=TEST
```

C. Run the test case.

```
# cd VAQA/Engine/bin
```

```
# ./run test/test1.sh
```

Note:

- If user want to take care of create FS, mount FS, adding GP and deleting GP as per need then she can run the test case with **SINGLE\_RUN=1** set in test specific local configuration file along with other configuration parameter used in the test case.
- VAQA has one more configuration parameter "NO\_ENCRYPTION" which can be used to run BASELINE test cases multiple times with different configuration parameters.  
Here, remove **SINGLE\_RUN=1** from config. file and set **NO\_ENCRYPTION=1**. Now if we run the VAQA test case with some KEY in ITERABLES section with two values, then the test case will run twice.  
**NO\_ENCRYPTION=1**, enables all three phases of test run cycle.  
i.e. It enables INIT, RUN and CLEANUP steps but it skips ADD GP and DELETE GP operation from INIT and CLEANUP.

## 6. How to run VAQA encryption test case?

Example:

A. Create a GLOBAL configuration file and add all frequently used configurations.

```
# cat VAQA/config/GLOBAL.conf
```

```
[FIXED]
```

```
MNTPT=/gp
```

```
FSTYPE = ext4
```

```
# Device file
```

```
SECFS_DEVICE = /dev/sdb1
```

```
[ITERABLES]
```

```
POLICIES = aes128, aes256
```

```
[SEC_SERVER]
```

```
# Add DSM related config parameters.
```

```
username=voradmin
```

```
password=Ssl12345#
```

```
domain=domain1
```

```
server=asrm-ss6-oxf.i.vormetric.com
```

```
[BUILD]
```

```
BUILD_SERVER=10.3.10.45:/nfs-builds/nightly/agent
```

```
BUILD_MNTPT=/build_mnt
```

```
BUILD_GREP=vor_5.2.3_20
```

```
AGENT_TYPE=fs
```

```
BUILD_VERSION=5.2.3
```

B. Create a test case in 'test-cases' dir.

```
# mkdir VAQA/test-cases/test
```

```
# cat test2.sh
```

```
echo "This is test2.sh script"
```

```
echo "Current encryption policy = $POLICIES"
```

```
echo "USER_VAR = $USER_VAR"
```

C. Create a test case specific LOCAL config. file.

```
# cat VAQA/test-cases/test/test2.sh.conf
```

```
[FIXED]
```

```
USER_VAR=TEST
```

D. Run the test case.

```
# cd VAQA/Engine/bin
```

```
# ./run test/test2.sh
```

Here, test2.sh will execute twice as we have defines POLICIES in [ITERABLES] section with two values.

i.e. Test will run twice, with aes128 and then with aes256

VAQA test2.sh run steps:-

- i. Update environment variables with merged dict. ( GLOBAL + LOCAL and POLICIES=aes128 )
- ii. Create 'ext4' filesystem on SECFS\_DEVICE '/dev/sdb1' device.
- iii. Mount SECFS\_DEVICE (/dev/sdb1) on MNTPT (/gp)
- iv. Add GP to GP0 ( /gp/gp0 ) with encryption policy aes128
- v. Run test2.py script
- vi. Delete GP.
- vii. Unmount SECFS\_DEVICE.
- viii. Go to step ( i ) and run the same test again with config parameters ( GLOBAL + LOCAL and POLICIES = aes256

*Note: For detailed explanation of VAQA please check : [VAQA Readme](#)*