# Stability test with HW reset (power loss)

## Test environment

Actually, this is the latest test environment. From the test env in section 2.1, there is a Ethernet connection between target and PC. The reason i remove the Ethernet connection is the driver of usb Ethernet converter has bug(relate to chipidea) and it already be reported to mentor, but i didn’t get any information that the bug was fixed.

BUT It is necessary for the target communicating with PC(power controller). Because the power loss should only happened during writing operations. But before start writing operations, the test need check files’ checksum which the files was created in last test loop. Because no one know how long time it will take to do the files’ checksum. The solution is send signal to PC after finish checking the checksum and before start creating new files. This can make sure the power loss happened during writing.

Because the usb Ethernet converter driver issue. It is impossible to use the Ethernet connection. Current test env is using UART to send the signal to PC. To do simple modification of the source of putty. On the target, the test script send the signal to PC via echo some special string to UART/console. When the putty receive these special string, it will trigger a script in windows which will power off the target in random time.

PC:

1. PUTTY(special edition)

2. A App for HW reboot controller

Power controller \*1\*

Target : IMX6 eMMC

power line

Parallel port

UART

\*1\* : This power controller is the box in softTec. If you want to know the detail of it, please contact *Dahlhoff Achim (CM-AI/PJ-CF32)* .

## SW flow chart

System startup

rcS: bring up test script \*1\*

Script : mount test partitions and Check mount result

Check test files‘ checksum

Print special string on console to ask putty to bring up a script on PC. \*1\*

Power loss in random time

Delete test files

Checksum error

Disable auto run the test script in rcS. Exit current test.

Creating/writing files

\*1\* : this script will trigger power loss in random time during the writing operations.

## stability-1031.out

Typical usage:

**Create files:**

***stability-1031.out -d /mnt/p1/test -a ws -t 81920 -p 1 -c 10 -b 1 -i $count -e 1***

-a : ro/wo/ws/wf ro: do files checksum wo: create file ws: open file with O\_SYNC flag wf: call fsync after close

-d : specify the directory of the files for option –a

-v: enable dump log to file. specify the directory to save log file. It will also enable the backup function for the error files. You can ignore it.

-t: how many files will be created

-p: honeyPot head, partition number

-c: how many blocks of one file.

-b: use chunk or not. Set 0 use small fixed size buffer. Set 1 alloc memory size same as the file.

-i: honeyPot head, the current loop counter

-e: honeyPot head, add the system time or not

--

-r: specify the directory for rename operation~~(need –x operation at same time)~~

~~-x: don’t add random string into honeypot header and filename~~

**Check Files’ checksum for specific directory**

***stability-1031.out -d $TESTDIR -a ro***

**check single file’s checksum**

***stability-1031.out -f FILENAME -a ro***

**For the transaction safe test:**

Check file’s checksum: ***stability-1031.out* -d $P6DST -a ro**

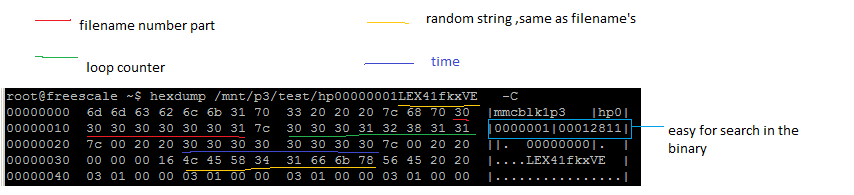
Create files: ***stability-1031.out* -d $P5TMP -a wf -t $NUMFILES -p 5 -c 10 -b 1 -i $count -e 1 -r $P5DST**

Note:

Because any FS has the limitation about the numbers of files. So if the –t is too big and –c is too small. It will cause abnormal to the FS.

About the honeyPot format, please get more information from the file: *Concept-Outline FS-Honeypot\_updated.doc*

Below is a example of the honeyPot head:



### emergency\_restart .out

This command is used in the stability test with sw reset.

It force system reset by enable system WDT and set WDT timeout value.

Typical usage, restart system in 1s:

emergency\_restart.out –t 1

-t : timeout time in seconds