

--- E3 FLASHER -- NOR ONLY ---

REQUIRED TOOLS:

=====

* A HEXEDITOR ... FOR EXAMPLE:

HXD -- <http://mh-nexus.de/en/hxd/>

* FLOWREBUILDER 4.2.2.0

* WINDOWS CALCULATOR - SET TO "PROGAMMER MODE" : TO CALCULATE METLDR/BOOTLOADER SIZE IN BYTES FROM HEX VALUE

FOLLOW THIS EASY GUIDE AND U CAN VERIFY A NOR DUMP..EASY....FOR NAND...NEVER USED THEM SO CANNOT COMMENT ON IT...IT IS BASICALLY THE SAME...SO IF U CAN FOLLOW THIS...I'M POSITIVE U CAN VERIFY NAND ALSO....STILL NOT SURE? JOIN PS3HAX.NET AND SEND UR DUMP TO THE PROVIDED "DUMP-CHECK" EMAIL SO TRUSTED VOLANTEERS WILL DO IT FOR U AS A SECOND-OPINIUM.....

=====

DO NOT RESTORE/FLASH ANYTHING BEFORE THE DUMP/BACKUP OF THE NORFLASH IS VERIFIED!!

=====

NOW FOLLOW THE VIDEO HOWTO AND DISCOVER IT ISN'T THAT HARD AFTERALL :)

TO PREVENT WASTING MORE TIME AS REQUIRED ON THE TOTAL GUIDE, THERE ARE A COUPLE SIMPLE TESTS ONE CAN DO AFTER DUMPING TO KNOW ATLEAST THAT THE MDE DUMP IS "INVALID"

Note!! >>

before starting u have to know the differents between dumps made by either:

A) Progskeet/Teensy (those hardware flasher dumps the flash already in "human read-able language =

IF.I header at offset 200 in the dump

B) E3 flasher dumps -- this hardware flasher dumps/write the flash in PS3 read-able language =

FI.I header at offset 200 in the dump ... in layman's terms progskeet/teensy dumps it "unscrambled" and e3 dumps it byte reversed "scrambled" >> IFI = unscrambled and FI.I = scrambled

=====

to verify a dump it needs to be bytereversed..aka human readable aka unscrambled = IF.I header @ 200

=====

okay..lets do 2 quik tests to check if the dump is even close to valid/complete:

zip/rar tip from euss and the guy knows his stuff for sure ;)...learned alot from it..but some have only 1/2 word needed to understand and another...well just don't get..no problemo..follow and notice it's pretty easy anyway...BUT O SO IMPORTANT

BECAUSE FLASHING/RESTORING A CORRUPT/INVALID FLASH (PER CONSOLE KEYS) WITHOUT HAVING A VALID ONE TO BEGIN WITH WILL 100% PERMA-BRICK UR ps3 AND YEAH PERM BRICK AINT NO FUN..ALTHOUGH THEY TEND TO WORK PERFECT AS A DOOR-STOPPER or so i heard.. :P

```
RIGHT...the 2 quik tests..whop >> OPEN FLOWREBUILDER 4.2.2.0 and select option:  
* extract a byte reversed NOR dump or an interleaved..... for Progskeet/Teensy  
dumps
```

and for E3 flasher style select this option:

```
* byte reverse and extract a NOR dump..... (final output = bkpps3.swap.bin = IF.I
header @ 200 -- (only for verifying the dump) >>>>
```

to flash back a dump made with E3 flasher u previously verified it needs ofcourse the original "bkpps3.bin" (FI.I header) and when one uses a "patched 355/430 coresos E3 flasher dump, make sure to use the correct patches for the E3 flasher!! not sure yet again? then use Rogero's NOR patcher for 355 and the auto patcher with the three musketeer patches, those patchers detects the "input" = either FI.I or IF.I on the fly and patches them correctly without manually messing around :)

u still here? at this point...we assume u know the differences between the TWO $\wedge\wedge$ = FI.I vs IF.I

2 quik verify test 1:

1a - quik test 1:

OPEN FLOWREBUILDER 4.2.2.0 and try extracting the dump...u get ERRORS while extracting? STOP now and re-check ur wires/e3 clip connection..it is bad = missing important files!

1b - quik verify test 2:

```
zip/rar that dump up and NOTE the size of the created "ZIP/RAR"..lower then
9/10MB? YES? STOP now and go re-check ur wires/e3 clip connection = it is bad =
missing important files..
```

perm brick for sure if u flash them back to the PS3 !!!

after u did the 2 quik tests and the dump extracted without any error on
Flowrebuilder and the ZIP/RAR is like 9/10MB's.....it's on the good track....now follow
the "complete" verify guide to make totally sure IT IS VALID...u can't be to sure...what a waste
otherwise..correct?!

now for the real verify...follow these steps below together with the bookmarked "ps3devwiki Validating flash dumps

[http://www.ps3devwiki.com/wiki/Validating flash dumps](http://www.ps3devwiki.com/wiki/Validating_flash_dumps)

now it looks skipping some steps..it isn't....the bytereverse part we explained earlier = FII or IFI

```
=====
=====
00 00 00 00 AC 0F FF E0 00 00 00 00 AD DE EF BE (HEX) and saceru_eoldare
(Text) = E3 flasher original dump
```

or

```
00 00 00 00 0F AC E0 FF 00 00 00 00 DE AD BE EF (HEX) and asecure_loader
(Text) = Progskeet/Teensy original dumps
```

If all the below steps are followed and everything is in order u can be 100% sure the dump is complete and valid :)

2.1 Statistical analysis

2.6 malformed headers / filenames / regionnames

2.7 absent files / regions >>>

```
=====
example: BOOTLOADER_0 (per console stuff)
OFFSET 00FC0000 for NOR = cannot be FF's or 00's..there must be data!
```

```
=====
2.8 malformed files / regions:
```

```
=====
example: BOOTLOADER_0
OFFSET 00FC0000 for NOR >>>
28 B4 4F D2 F9 3F BC 43 28 B4 4F D2 F9 3F BC 43
corrupted bootldr in NOR ^ (it should always start with 00 00 there)
=====
```

2.9 Repetitions (meaning this value can only be there 1 time only...not duplicated = bad wires/bad clipping) :

```
=====
example = DEADBEEF magic header
OFFSET 000000010 for NOR >>
00 00 00 00 0F AC E0 FF 00 00 00 00 DE AD BE EF
=====
```

when it finds "duplicates" of this value(bootloader_0) as also another "unique" value :

617365637572655F6C6F616465720000 (asecure_loader in hex)

if it finds duplicates of those two ^ then u can use part 2.9.2 of the ps3devwiki verify guide to pinpoint the "bad wire/cold solder etc..) ==
2.9.2 OK, you've found the pattern, now how do we deduct from that the faulty line?

if it happens with E3 flasher than simply re-clip the E3 flasher (check out playerkp420 HOWTO on PS3HAX.net to sand down the E3clip properly to FIT on all ps3 consoles...DO NOT USE A WOODEN GRINDER IT WILL RAPE THE E3CLIP FOR SURE..JUST NEEDS A BIT SANDED AWAY..UNSURE? CHECK HIS howto)

AND NOW FOR THE FINAL STEP BUT O SO IMPORTANT!! =
=====

2.10 Check metldr+bootldr sizes

for this step we use the ordinary windows calulator in "programmer mode"

/Target_ID IDPS

OFFSET 002F077 (NOR)

metldr offsets (size and type)

1) size metldr:

OFFSET 0081E (NOR) = the value = SIZE in HEX

2) type metldr:

OFFSET 00842 (NOR) = the value = TYPE metldr (each model have different values, some models share the same values >>> like CECHLxx/CECH20xx) check it all!!

bootloader offsets (1x size and 2x type = 2x identical value for type!!)

1) size bootldr in HEX (use the windows calculator to convert to bytes)

2A) type bootldr:

OFFSET 1 = 00FC0002 (NOR)

OFFSET 2 = 0FC0012 (NOR)

all verified..it takes longer to type all this crap then doing it hehe...

Done.....go have fun...this dump is valid/complete and if its metldr1 whit minverchk 3.56 it can also be downgraded at this time using the new three musketeer patches...ONLY if the console/dump after checking = melldr2! ur stuck...in this example its a win :) go use the 430coresos patches and patch dump with it and downgrade using Rogero 4.30 2.05 via the recovery

All credits to ps3devwiki for the info and all users!!

have fun!

end