

Open Core Debug

Beitrag von „praekon“ vom 15. Oktober 2019, 19:24

Danke!

Ich war mir mit -v nicht sicher, weil ich das bisher nur CLOVER kannte.

Im PDF steht dazu folgendes:

```
4. Target
Type: plist: integer
Fallback: 0
Description: A bitmask (sum) of enabled logging targets. By default all the logging output is hidden, so this
option is required to be set when debugging is necessary.
The following logging targets are supported:
• 0x01 (bit 0) — Enable logging, otherwise all log is discarded.
• 0x02 (bit 1) — Enable basic console (stdout) logging.
• 0x04 (bit 2) — Enable logging to Data Hub.
• 0x08 (bit 3) — Enable serial port logging.
• 0x10 (bit 4) — Enable UEFI variable logging.
• 0x20 (bit 5) — Enable non-volatile UEFI variable logging.
• 0x40 (bit 6) — Enable logging to file.
Console logging prints less than all the other variants. Depending on the build type (RELEASE, DEBUG, or HOOPT)
different amount of logging may be read (from least to most).
Data Hub log will not log kernel and kext patches. To obtain Data Hub log use the following command in macOS:
```

Was nach meinem Verständnis nach, bedeuten würde bei Target 0x02 würde ne Screenausgabe bekommen.

Scheint wohl nicht zu stimmen, oder ich bin Begriffsstutzig

