

Update von 059 zu 060 doch komplizierter als gedacht...

Beitrag von „Arkturus“ vom 13. September 2020, 14:53

da ich mit der OC v0.6.2 nicht ein Stück weitergekommen bin, habe ich mit der Anleitung von 5T33Z0 die config.plist und die sample.plist verglichen.

Ergebnis:

```
#####
# OC Config Compare #
#####

Checking for values missing from User.plist:
config.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x10,0x0)
config.plist -> Kernel -> Missing Key: Block
config.plist -> Kernel -> Add -> Array - Missing Key: Arch
config.plist -> Kernel -> Add -> Array - Missing Key: Comment
config.plist -> Kernel -> Add -> Array - Missing Key: MaxKernel
config.plist -> Kernel -> Add -> Array - Missing Key: MinKernel
config.plist -> NVRAM -> Add -> TC430110-AB2A-4000-AB00-FC41995CF82 - Missing Key: SystemAudioVolume
config.plist -> NVRAM -> Add -> TC430110-AB2A-4000-AB00-FC41995CF82 - Missing Key: run-efi-updater

Checking for values missing from Sample:
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x0,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x14,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x16,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x19,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1A,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1B,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1C,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1C,0x0)/PCI(0x0,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1C,0x1)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1C,0x1)/PCI(0x0,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1C,0x2)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1D,0x0)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1F,0x2)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x1F,0x3)
Sample.plist -> DeviceProperties -> Add - Missing Key: PCIRoot(0x0)/PCI(0x2,0x0) - 2
Sample.plist -> Kernel -> Missing Key: Delete

Press [enter] to return...]
```

warum Deviceproperties nicht stimmen, weiß ich nicht. Ich habe aus der laufenden config.plist den arch Add übernommen. Ich habe den Eindruck, das PlistEdit Pro nicht ganz zuverlässig arbeitet

Wie ich aber die fehlenden Angaben zur Lilu.kext ergänzen soll, ist mir ein Rätsel. Die Configuration.plist gibt da z.B. für MaxKernel folgende Anweisung aus:

6. MaxKernel

Type: plist string

Payload: Empty string

Description: Adds kernel driver on specified macOS version or older.

Kernel version can be obtained with `uname -r` command, and should look like 3 numbers separated by dots, for example 18.7.0 is the kernel version for 10.14.6. Kernel version interpretation is implemented as follows:

$$\text{ParseDarwinVersion}(c, \lambda, \mu) = c \cdot 10000 \quad \text{Where } c \in (0, 99) \text{ is kernel version major} \\ + \lambda \cdot 100 \quad \text{Where } \lambda \in (0, 99) \text{ is kernel version minor} \\ + \mu \quad \text{Where } \mu \in (0, 99) \text{ is kernel version patch}$$

Kernel version comparison is implemented as follows:

$$\alpha = \begin{cases} \text{ParseDarwinVersion}(\text{MinKernel}), & \text{If MinKernel is valid} \\ 0 & \text{Otherwise} \end{cases} \\ \beta = \begin{cases} \text{ParseDarwinVersion}(\text{MaxKernel}), & \text{If MaxKernel is valid} \\ \infty & \text{Otherwise} \end{cases} \\ \gamma = \begin{cases} \text{ParseDarwinVersion}(\text{FindDarwinVersion}()), & \text{If valid "Darwin Kernel Version" is found} \\ \infty & \text{Otherwise} \end{cases}$$
$$f(\alpha, \beta, \gamma) = \alpha \leq \gamma \leq \beta$$

Here `ParseDarwinVersion` argument is assumed to be 3 integers obtained by splitting Darwin kernel version string from left to right by the `.` symbol. `FindDarwinVersion` function looks up Darwin kernel version by locating "Darwin Kernel Version *c*.*λ*.*μ*" string in the kernel image.

Das übersteigt meine Kapazitäten und ist für mich nicht Ziel der Übung.

Ich verlasse erstmal die OCNighthly und gehe zu 0.6.1 zurück.

Danke an Alle, die geholfen haben.