

Erledigt **RTCMemoryFixup.kext**

Beitrag von „kuckkuck“ vom 15. Februar 2019, 13:30

Grundsätzlich existieren für uns unter CMOS Offsets bis Byte 256. Die bekannten Patches limitieren das ganze auf 128 Byte, was im Normalfall das Problem bereits behebt.

Sollte keiner der bekannten Patches (wie DSDT oder config Fix) eingebaut sein und man benutzt RTCMemoryFixUp, wird wieder die volle Bandbreite benutzt (256 Byte) und dementsprechend kann es wieder zu CMOS resets etc kommen (Inkompatibilität von AppleRTC und Firmware).

Es gilt jetzt also die problematischen Offsets zu finden und zu definieren. Daraufhin werden die entsprechenden Offsets blockiert und können keinen reset mehr erzeugen, RTCMemoryFixUp simuliert jedoch weiterhin Read/Write auf diese Offsets.

Mir ist kein Weg bekannt die problematischen Offsets aus einem Log o.ä zu lesen, das heißt es gilt testen. Ist RTCMemoryFixUp installiert und der Hacky resettet sich nach Sleep/Reboot, kann es losgehen.

Die Offsets 00-0D (0-**13**) sind unkritisch, die brauchen wir nicht testen. Die Idee ist jetzt erst einmal die problematische Memory Bank zu finden (die Erste geht von 00-7F/0-127 und die Zweite von 80-FF/128-256) (im Worst Case haben beide ein Problem).

Dafür excluden wir per Bootarg erstmal 13-127, also **rtcfx_exclude=0D-7F** und testen durch Sleep, Wake und Reboot. Sollte das Problem behoben sein, müssen wir zwischen 0D-7F genauer die problematischen Offsets finden.

Sollte das Problem nicht behoben sein, testen wir die zweite Bank, also **rtcfx_exclude=80-FF**.

Sagen wir mal, das Problem ist jetzt erst behoben, nun können wir im Bereich 80-FF (128-256) weiter testen.

Eine Taktik um nicht jeden Bereich einzeln testen zu müssen, ist den Bereich immer zu halbieren, um zu erfahren in welcher Hälfte das Problem liegt. Hierfür bietet sich jetzt also zB folgendes an: **rtcfx_exclude=80-C0**. Existiert das Problem mit diesen Excludes ebenfalls nicht, können wir im Bereich 80-C0 (128-192) weiter testen (zB wieder die Hälfte). Existiert das Problem hingegen wieder, können wir uns den Bereich C0-FF weiter anschauen um hier den

problematischen Offset(-Bereich) zu finden. Und so weiter bis man keinen Bock mehr hat...

Schon an diesem Punkt ist die Methode cleaner als das limitieren auf 128 Bytes only (RehabMan Methode), da nun bereits über 128 Bytes verfügbar sind - eine Verbesserung.