macOS 13 VENTURA beta

Beitrag von „GoodBye“ vom 18. August 2022, 09:45

Arkturus

wenn SecureBootModel konfiguriert ist, kann es durch eingabe im Terminal geprüft werden mit:

nvram 94b73556-2197-4702-82a8-3e1337dafbfb:AppleSecureBootPolicy

If the variable is found, it can be one of the following:

- %02 - Full Security Mode
- %01 - Medium Security Mode
- %00 - No Security Mode

Unter SecureBootModel in der OC Plist den dem SmBios entsprechenden Wert eintragen:

https://dortania.github.io/Ope…what-is-apple-secure-boot

Den wert zum eintragen in ApEcid ermittelt man mit:

Code

```
python3 -c 'import secrets; print(secrets.randbits(64))'
```

Beim Neustart wird dann oder du wählst es aus in das Recovery gebootet, dort dann im Terminal ausführen (HD mit dem Namen deiner HD ersetzen) :

- bless --folder "/Volumes/HD/System/Library/CoreServices" --bootefi --personalize

Dann ist SecureBootModel konfiguriert.

SecureBootModel is used set the Apple Secure Boot hardware model and policy, allowing us to enable Apple's Secure Boot with any SMBIOS even if the original SMBIOS did not support it(ie. no T2 present on pre-2017 SMBIOS). Enabling SecureBootModel is the equivalent of "Medium Security"

(opens new window), for Full Security please see ApECID

Currently the following options for `Misc -> Security -> SecureBootModel` are supported:

Value	SMBIOS	Minimum macOS Version
Disabled	No model, Secure Boot will be disabled.	N/A
Default	Currently set to x86legacy	11.0.1 (20B29)
j137	iMacPro1,1 (December 2017)	10.13.2 (17C2111)
j680	MacBookPro15,1 (July 2018)	10.13.6 (17G2112)
j132	MacBookPro15,2 (July 2018)	10.13.6 (17G2112)
j174	Macmini8,1 (October 2018)	10.14 (18A2063)
j140k	MacBookAir8,1 (October 2018)	10.14.1 (18B2084)
j780	MacBookPro15,3 (May 2019)	10.14.5 (18F132)
j213	MacBookPro15,4 (July 2019)	10.14.5 (18F2058)
j140a	MacBookAir8,2 (July 2019)	10.14.5 (18F2058)
j152f	MacBookPro16,1 (November 2019)	10.15.1 (19B2093)
j160	MacPro7,1 (December 2019)	10.15.1 (19B88)
j230k	MacBookAir9,1 (March 2020)	10.15.3 (19D2064)
j214k	MacBookPro16,2 (May 2020)	10.15.4 (19E2269)
j223	MacBookPro16,3 (May 2020)	10.15.4 (19E2265)
j215	MacBookPro16,4 (June 2020)	10.15.5 (19F96)
j185	iMac20,1 (August 2020)	10.15.6 (19G2005)
j185f	iMac20,2 (August 2020)	10.15.6 (19G2005)
x86legacy	Non-T2 Macs in 11.0(Recommended for VMs)	11.0.1 (20B29)

#Special Notes with SecureBootModel

- The `Default` value is not recommended as if you plan to use this with ApECID for full security, we recommend setting a proper value (i.e. closest to your SMBIOS or versions of macOS you plan to boot) since the `Default` value is likely to be updated in the future.
  - In addition, `Default` is set to `x86legacy` which will breaking booting High Sierra through Catalina.
  - `x86legacy` is not required for normal Mac models without T2's, any of the above

values are supported.

- The list of cached drivers may be different, resulting in the need to change the list of Added or Forced kernel drivers.
    - ie. IO80211Family cannot be injected in this case, as it is already present in the kernelcache
- Unsigned and several signed kernel drivers cannot be used
    - This includes Nvidia's Web Drivers in 10.13
- System volume alterations on operating systems with sealing, like macOS 11, may result in the operating system being unbootable.
    - If you plan to disable macOS's APFS snapshots, please remember to disable SecureBootModel as well
- Certain boot errors are more likely to be triggered with Secure Boot enabled that were previously not required
    - Commonly seen with certain APTIO IV systems where they may not require IgnoreInvalidFlexRatio and HashServices initially however Secure Boot does.
- On older CPUs (ie. before Sandy Bridge) enabling Apple Secure Boot might cause slightly slower loading by up to 1 second
- Operating systems released before Apple Secure Boot landed (ie. macOS 10.12 or earlier) will still boot until UEFI Secure Boot is enabled. This is so,
    - This is due to Apple Secure Boot assuming they are incompatible and will be handled by the firmware just like Microsoft Windows is
- Virtual Machines will want to use `x86legacy` for Secure Boot support
    - Note using any other model will require `ForceSecureBootScheme` enabled