

TINU oder besser gesagt "TINU Is Not Uni*****"

Beitrag von „al6042“ vom 9. April 2018, 17:53

Aus der Hand des Entwicklers [@ITzTravellnTime](#) kommt dieses kleine aber feine Tool zum Erstellen von macOS Install-Datenträgern.



Hier vorliegend in der offiziellen Version 1.1:

Features:

- Einfache und selbsterklärende GUI zum Starten des macOS Install Media Creation Prozesses
- Kann mit jeder macOS install.app genutzt werden, die das "createinstallmedia"-executable im "Resources"-Ordner beinhaltet (Auch für Betas und komplett neu Versionen)
- Das Ziel kann ein USB-Stick oder auch eine Partition auf einer Festplatte sein, muss aber mindestens 7 GB Platz haben.
- Funktioniert auch im macOS Recovery Modus, somit kann ein neuer Datenträger auch während eines laufenden macOS Install Vorgangs oder von der "Recovery HD" erzeugt werden.
- Das Ergebnis entspricht einem unbearbeiteten [Install Stick](#) für Original-Macs und ist somit "Vanilla", wie es für die Command Line Methode "createinstallmedia" üblich ist.
- Das Produkt ist Open source, somit kann jederzeit nachvollzogen werden, was das Programm auf deinem Rechner tut. Ebenso kannst du dir eine eigene Version des Tools durch Anpassungen im Source Code zusammenstellen
- Einfach das Programm ausführen und den Anweisungen folgen.
- Kein Bedarf den Datenträger im Vorfeld mit dem Festplattendienstprogramm zu bearbeiten. TINU formatiert das Ziel vor ausführen des eigentlichen Erstell-Vorgangs.

- Benutzt aktuelle und moderne APIs, SDKs und die Swift 3 Programmiersprache
- Transparenter Grafik Modus möglich (Alt + S)

Was das Tool in Zukunft noch alles machen soll:

- Erweiterte Optionen zum Anpassen des macOS Install Datenträgers
- Installer Anpassungen: z.B. "Kernel cache", "Prelinked Kernel" und Austauschen von Boot Dateien (kann für ältere Macs oder Beta Versionen interessant werden, ebenso auch bei AMD-Systemen)
- Die Installation und Konfiguration von Clover (inkl. EFI-Treiber und Custom-DSDTs)
- Das Einbinden von Kexten in den Kext-Ordern von Clover
- integration einer config.plist Vorlagen-Datenbank aus einem zentralen, aber offenen Repository
- Unterstützung für weitere Sprachen

Requirements:

- Ein Computer mit mindesten Mac OS X Yosemite oder neuer (Mac OS X El Capitan wird mindestens benötigt um TINU im macOS Recovery Modus oder dem Install Vorgang ausführen zu können.)
- Ein USB-Stick oder eine freie Partition von mindestens 7 GB Größe
- Eine macOS/Mac OS X install.app (Mavericks oder neuere Versionen werden unterstützt) im Ordner /Programme (/Applications) im Hauptverzeichnis eines jeglichen Datenträgers im Computer (außer natürlich dem Datenträger, der als Ziel genutzt werden soll 😊)

Ein dazugehöriges Wiki in Entwicklung findet ihr unter <https://github.com/ITzTravellnTime/TINU/wiki>

Den FAQ-Bereich in englischer Sprache findet ihr unter <https://github.com/ITzTravellnTime/TINU/wiki/FAQs>

Den Source Code für Version 1.1 gibt es somit natürlich unter <https://github.com/ITzTravellnTime/TINU>

Credits: [\[url='https://www.hackintosh-forum.de/index.php/User/48927-ITzTravellnTime/'\]](https://www.hackintosh-forum.de/index.php/User/48927-ITzTravellnTime/)
[ITzTravellnTime](https://www.hackintosh-forum.de/index.php/User/48927-ITzTravellnTime/)

Beitrag von „LuckyOldMan“ vom 31. Mai 2018, 17:48

Gibt es da eine ganz besondere Vorgehensweise mit diesem netten Tool, der ich nicht gefolgt bin oder warum mag es das PW meines Administrators nicht?



Immer und immer wieder werde ich aufgefordert, das PW einzugeben, weil kein leeres Feld als PW akzeptiert wird. Gebe ich es ein, kommt erneut der Hinweis wie auf Bild 2, als wäre es nicht vorhanden.

Wie hätte es das Tool denn gerne?

Gruß
LOM

Beitrag von „al6042“ vom 31. Mai 2018, 18:01

Das hatte ich die Tage auch schon mal gesehen, wenn das Passwort extrem umfangreich war...

Versuche es mal mit dem aktuelle Release Candidate für Version 2.0 ->
<https://dl.dropboxusercontent.com/s/nvzss51qcieynve/TINU.zip>

Beitrag von „LuckyOldMan“ vom 31. Mai 2018, 18:08

[@al6042](#)

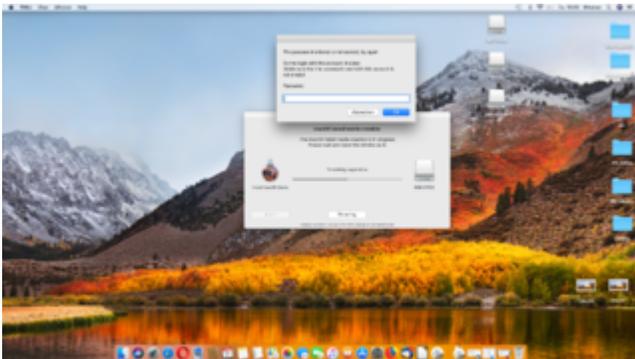
Nach Jahren habe ich mich doch entschlossen, mein bisheriges PW "1234" aufzugeben, um es etwas schwieriger zu machen. 😊

Danke für den Link!

Gruß
LOM

Nachtrag:

Nöööhh ...



Doch wieder 1234 ?

Beitrag von „al6042“ vom 31. Mai 2018, 18:13

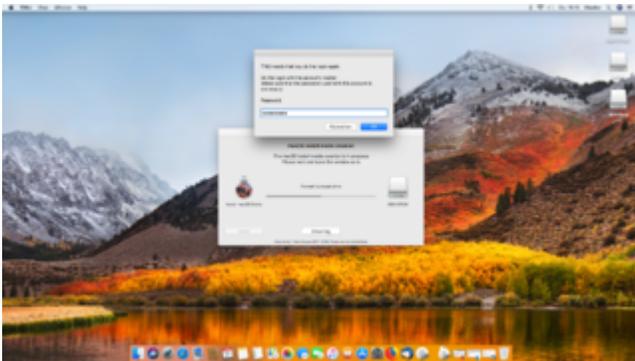
Dann melde dich mal als Admin (nicht BNutzer) an und führe das Tool damit aus.

Beitrag von „LuckyOldMan“ vom 31. Mai 2018, 18:26

[@al6042](#)

Selbst mein Admin hat nicht die absolute Macht! 😊

Offensichtlich hat das Tool ein kleines Problem mit den PWs, denn auch hier wurde wieder diese sinnbefreie Meldung präsentiert.



Erst nach erneuter Eingabe des Admin-PWs ging es dann schlußendlich los.



Sollte irgendwie/-wo vermerkt werden, dass es nur im Admin-Modus mit doppelter PW-Eingabe möglich ist.

Ergibt das jetzt eine ein- oder eine zwei-phasige-Installation?

Gruß
LOM

Beitrag von „al6042“ vom 31. Mai 2018, 18:36

Ich bin mir gar nicht mehr sicher ob es ein oder zwei Phasen waren...

Während der HackCon habe wir damit die Büchsen installiert und es ging ziemlich flott... 😊

Beitrag von „LuckyOldMan“ vom 31. Mai 2018, 18:57

Dann werde ich jetzt mal eine Simulation machen.

Ein UEFI-CloverStick liegt bereit - der Install-Stick ist gerade fertig geworden.

Wenn ich mich bis morgen nicht mehr melde, hat eine winzige Kleinigkeit nicht gepasst
(obwohl ich ja immer Alles so mache wie mir angesagt wird).



Gruß
LOM

Nachtrag:

Ist eine 2-phasige Installation mit Zwischenstopp ähnlich dem Recovery-Aufruf.

Beitrag von „al6042“ vom 4. Juni 2018, 07:02

Der [@ITzTravellnTime](#) hat letzte Nacht eine neue Version von diesem überaus umfangreichen Tool bereitgestellt.



Hier nun die offizielle Version 2.0:

Hier die Neuigkeiten:

- Neue graphische Oberfläche für einfachere Handhabung
- Verbesserte Engine für mehr Einfluss und Fehlerkorrektur

- Neue erweiterte Optionen, mit mehr Features und Verbesserung der Install-Medium-Customizing-Möglichkeiten (z.B. ein Clover EFI Ordner Installer)
- Die macOS-installer.app muss nun nicht mehr unter /Programme (/Applications) liegen, sondern kann über die Auswahl auch aus anderen Ordnern heraus geöffnet werden.
- Neue Menü für den direkten Zugriff auf die macOS Installer im AppStore (Menü-Zeile -> Help -> Get a macOS installer app)
- Besser Diagnose Infos
- Die Möglichkeit macOS System Dateien innerhalb des Install-Mediums zu ersetzen
- Neuer Modus für die Nutzung TINUs zur Install-Medium-Erstellung in innerhalb eines macOS-Installers oder der Recovery

Den Source Code für Version 2.0 gibt es somit natürlich wieder unter <https://github.com/ITzTravellnTime/TINU>

Credits: [@ITzTravellnTime](#)

Beitrag von „LuckyOldMan“ vom 4. Juni 2018, 09:03

[@al6042](#)

Leider ist es immer noch nicht möglich, wie bei vielen anderen Programmen auf Standardbenutzer-Ebene zu arbeiten.

Hier würde es wie beim Clover Configurator genügen, durch Admin-Name & -PW-Abfrage die "Veränderungen" zu authentifizieren. Statt dessen kommt erst die Admin-Authentifizierung, um dann etwas später noch mal zusätzlich nach dem Standardbenutzer-PW zu fragen und das inkl. der erneuten Abfrage mit Hinweis auf "Leer...", damit das Ganze dann "unerwartet ..." zum Abbruch führt. 😞

Meiner Meinung nach wenig bedienfreundlich und ich vermute, dass das auch nicht so gewollt ist. Wenn doch, wäre ich auf die Erklärung gespannt.

Gruß
LOM

Beitrag von „al6042“ vom 4. Juni 2018, 09:40

Dann würde ich dich bitten direkt mit dem Entwickler in Verbindung zu treten...

ich kann hier als Mittelsmann nicht fungieren, da ist ch dieses Problem nicht bei mir nachstellen kann.

Beitrag von „al6042“ vom 6. Juni 2018, 18:59

Und da ist schon die neue Version 2.0.1, mit folgenden Änderungen:

- Kleine Bugs beseitigt und Support für macOS Mojave 10.14 beta hinzugefügt

Beitrag von „al6042“ vom 24. Juli 2018, 11:20

Und hier mal noch die aktualisierte Version 2.0.2 mit folgenden, kleinen Änderungen:

- Bug beseitigt, in Bezug auf die Erstellung von Install-Medien mit älteren macOS Installer.apps
- Beheben von Programm-Abbrüchen die beim Testen mit Debug-Variablen auftauchten.

Beitrag von „al6042“ vom 9. Dezember 2018, 18:53

Und heute ganz frisch dazugekommen, die aktuelle Version 2.1 mit folgenden Neuerungen:

- Jede Menge Änderungen und Neuigkeiten unter der Haube
- Umstieg auf XCode 10
- Mojave Dark Mode Unterstützung
- Neue EFI Partition Mounter eingebaut
- Log-Fenster wurde redesigned
- verbesserte Übersetzung (italienisch zu englisch) für bessere Verständlichkeit
- verbessertes Layout und GUI
- neue Passwort-Abfragen... sicherer und zuverlässiger
- Umfangreiche Bug-Fixes (Dank den Benutzern und Beta-Testern)
- weitere Optimierungen
- Reorganisation des Source Codes
- Weitere Personen in die Credits aufgenommen (PPC & TT)

Beitrag von „al6042“ vom 24. Mai 2020, 14:42

Hier mal noch die letzte Beta der Version 3.0, mit der auch Catalina arbeiten kann.

Beitrag von „al6042“ vom 27. Juni 2020, 12:48

Und dank [Lincoln](#) 's Hinweis und der darauf hin erfolgten Prüfung auf GitHub, gibt es auch eine neue Beta, mit folgenden Änderungen:

- Fixed Catalina support
 - Added Big Sur support
 - Added [SIP](#) detection
 - Reworked Installer app download window
 - The app will now start with a password prompt, either graphical or via terminal
 - Added diagnostics mode with administrator privileges
 - Password prompt is no longer mandatory for the diagnostics mode
 - Improvements to the installer app detection system and added support to Big Sur
 - Improved drive detection system
 - Added root users detection and optimization (when the app is running as root there will be some optimization to the way scripts are handled)
 - Various code optimization and improvements
 - Commented the boot-files replacement code which will be removed in the future
 - The readme which is created on installer drives has been rewritten
 - Bug Fix
-

Beitrag von „al6042“ vom 11. Oktober 2020, 20:11

Hier noch die offizielle Beta4 von TINU 3.0, ohne das Terminal-Fenster.

Version 3.0 BETA 4:

- Added multiple language system to let the app to be translated in different languages in future versions of the tool
- Added a BETA Italian language translation
- Rewritten or revisioned most of the UI's code
- Huge code polish and revision of the way most systems of the app works
- Rewritten the diagnostics mode code (the app will no longer always ask you to use the

- diagnostics mode and the "Error code 1" problem is fixed)
- Installer app size detection to let you avoid using installer apps too big for your disk
 - Improved EFI partition mounter (reworked some ui code and added support for opening the config.plist with multiple apps)
 - Revisioned disk detection system
 - Revisioned disk icon detection system
 - Reworked progress-bar system
 - Removal of several hardcoded contents which were moved into some .json files which are embedded into the app
 - Solved several UI bugs
 - Improved support for Big Sur
 - Reworked notifications
 - Other "under the hood" work which is not really worth mentioning because this list is already quite long

Release notes:

- The source code can be compiled just with Xcode 10.1 since it still uses swift 3 (the project will be moved to swift 4 or 5 when release 3.0 will be finished) and since it uses some Xcode 10's specific features.
- Building of EFI partition mounter as a standalone tool is broken in the source code included with this release, since a lot of changes in how TINU works, aren't applied yet to the Xcode project for the standalone EFI partition mounter app. Keep an eye on the development branch of this repo for an updated source code which will solve this problem.
- This version is (at the time of release) the recommended version if you want to use Catalina or Big Sur

In case of any bugs or issues is highly recommended to create an issue on the [TINU Github repo](#)