

Bluetooth (LE) Hardware und Dual/Triple Boot

Beitrag von „Pillenflitzer“ vom 7. März 2024, 20:13



Wer, wie ich, eine Bluetooth Tastatur/Maus in Verbindung mit mehreren Betriebssystemen nutzt hat das Problem dass jedes Gerät mit jedem OS separat synchronisiert werden muss.

Mit diesem kleinen Tutorial müsst ihr das ganze nur einmalig unter jedem Betriebssystem machen und müsst anschließend nie wieder umstellen.

Wichtig!

Es ist zwingend eine Tastatur oder Maus mit BT HID Proxy notwendig wo die Verbindung über einen Neustart hinausgeht.

Das heißt ihr müsst unabhängig von der ganzen Geschichte jetzt schon im BootManager (OpenCore z.B.) hin und her wechseln können mittels Pfeiltasten.

Soweit mir bekannt schließt das ältere Logitech Tastaturen aus. Wie es aktuell ist kann ich leider nicht sagen.

Ich benutze durchweg das Wort **Unique ID**. Richtigerweise handelt es sich dabei aber tatsächlich um eine Mac-Adresse.

In dieser Anleitung wird auf ältere Bluetooth Hardware (Bluetooth - Legacy) und neuerer Bluetooth Hardware (BLE) eingegangen.

Bei der Bluetooth - Legacy Hardware zieht sich die **Unique ID** durch jegliche Betriebssysteme. Nur der **LinkKey** ist immer anders.

Das ist bei BLE - Hardware nicht der Fall!

Dort wird für jedes Betriebssystem eine **Random Unique ID** festgelegt und muss immer mit geändert werden!

Anmerkung!

Es wird für jedes Betriebssystem Administrator/root Zugriff benötigt!

Schritt 1:



Ihr müsst euch zwingend für eine Taste eurer Tastatur und Maus entscheiden und bei allen Betriebssystemen beibehalten.

Ein nachträglicher Wechsel ist nur bedingt möglich!

Die sinnvollste Vorgehensweise ist das ihr eure Bluetooth Geräte komplett aus jedem Betriebssystem entfernt.

Reines entkoppeln/trennen reicht nicht aus!

Die Geräte müssen richtig entfernt werden.

In diesem Tutorial nutze ich macOS als Vorgabesystem für die Keys somit ist es das letzte

Betriebssystem wo die Geräte synchronisiert werden.

Ob ihr jetzt mir Linux oder Windows anfangt bleibt euch überlassen.

Ich fange mit Linux an da ich in Windows parallel einen weiteren, nötigen Schritt vollziehe.

Nach dem wir alle Geräte aus allen Betriebssystemen entfernt haben und uns für eine Taste entschieden haben kommen wir zu Schritt 2.

Schritt 2:

Linux booten und Maus und Tastatur neu Verbinden/Synchronisieren.

Schritt 3:

Windows starten und Maus und Tastatur neu Verbinden/Synchronisieren.

Wenn das erledigt ist laden wir uns die PsTools von der Microsoft Seite.

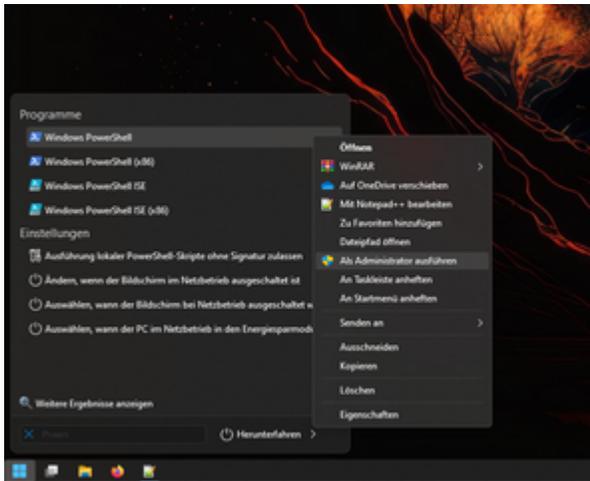
Diese werden benötigt um Administratorzugriff auf die Registry zu haben.

[PsTools](#)

Wir benötigen aus dem ganzen Paket ausschließlich die **PsExec.exe** welche nach **C:\Windows\system32>** kopiert werden muss.

Ihr habt natürlich auch die Möglichkeit mit dem passenden Befehl **psexec** aus dem Download-Ordner aus zu starten.

Jetzt öffnen wir die **PowerShell** als Administrator.



und geben folgenden Befehl ein:

```
psexec -i -d -s regedit.exe
```

Das ganze sollte dann so aussehen.



Wir navigieren nach **Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\BTHPORT\Parameters** und sollten dort unter Devices 2 Geräte sehen. (in unserem Fall die Maus und Tastatur)

Wie auf dem Bild zu sehen habe ich beide Geräte zur Veranschaulichung schon benannt.

Beim Device 1 handelt es sich um ein Legacy Bluetooth Gerät.

Das heisst die **Unique ID 202000c5ee6** zieht sich durch alle Betriebssysteme und sollte unter Linux und macOS leicht wiederzufinden sein.

Beim Device 2 handelt es sich um ein BLE Gerät. Dort wird jedem OS eine **Unique ID** in abgewandelter Form zur Verfügung gestellt.

Die gute Nachricht es betrifft nur die letzte Zahl bzw. letzten Buchstaben.

Das heisst auch unter Linux und macOS wird das Gerät so aufgeführt (**c7ac3be65f8x**)

x = Random

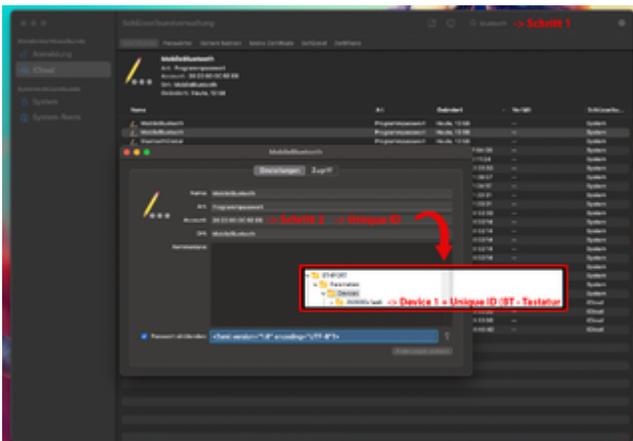
Ich empfehle jedem sich hiervon einen Screenshot anzulegen (oder aufzuschreiben) um sich das ganze unter macOS zu vereinfachen.

Anschließend kann das ganze ohne irgendwelche Änderungen geschlossen werden und macOS gestartet werden.

Schritt 4:

Als letztes **Synchronisieren/Verbinden** wir unsere Geräte mit macOS.

Ist das erledigt öffnen wir die Schlüsselbundverwaltung und geben oben rechts (**Schritt 1**) **bluetooth** ein.



macOS macht es uns hier ziemlich einfach da man sich leicht am Änderungsdatum (**Geändert**) orientieren kann.

Anschließend ein **Doppelklick** auf **MobileBluetooth**.

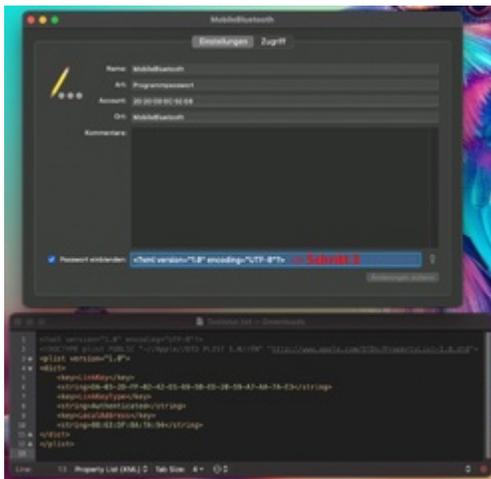
Mit welchem Gerät ihr Anfang ist natürlich euch überlassen ich fange hier mit der **202000c5ee6** an.

Ihr müsst euch hier zwingend an der unter **Account** angegebenen **Unique ID** orientieren!

Als nächsten aktivieren wir ganz unten **Passwort einblenden**.

Hier kommt der oben erwähnte nötige **root**.

Den ganzen Inhalt kopieren wir in eine **Text-Datei**.

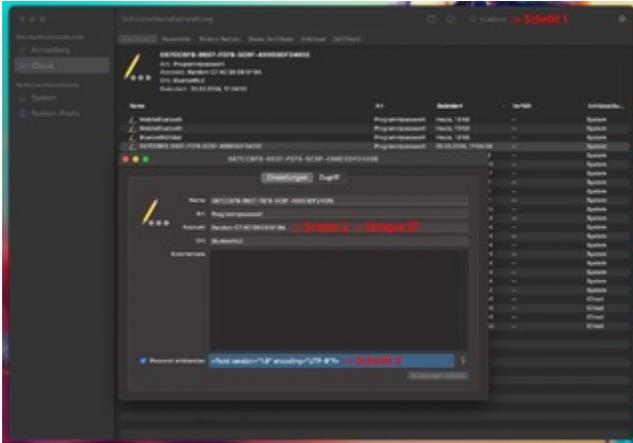


Das wars schon für Bluetooth Legacy Hardware. Ihr benötigt nur den **LinkKey**.

Für **BLE** Geräte ist die Sache etwas umständlicher und benötigt ein paar zusätzlichen Schritte.

Wählt jetzt in der Schlüsselbundverwaltung den nächsten **MobileBluetooth** Device aus.

Achtet bitte wieder auf die richtige **Unique ID** unter Account.



macOS sagt uns hier schon das die **Unique ID** Random ist. Sollte aber wie oben erwähnt bis auf die letzte Stelle soweit übereinstimmen.

Bei mir stimmt die **Unique ID** natürlich schon komplett überein da ich das ganze schon gemacht habe.

Jetzt aktivieren wir wieder **Passwort einblenden** und kopieren das ganze wieder in eine Text-Datei.

Bei **BLE - Geräten** ist es wichtig sich die von macOS vorgegebene **Unique ID** mit zu notieren um sie anschließend in Windows und Linux zu übertragen.

Ansonsten funktioniert das ganze nicht.



Des Weiteren wird bei **BLE - Geräten** kein reiner **LinkKey** genommen sondern läuft über einen **Long-Term-Key (LTK)** und einen **Remote Identity-Resolving-Key (IRK)**.

Diese werden hier als Base64 ausgegeben und können problemlos mittels **Hackintool (Calc)** oder ähnlichen umgewandelt werden.

Long-Term-Key (LTK)



Remote Identity-Resolving-Key (IRK)



Soweit mir bekannt war es in früheren macOS Versionen nötig den **LinkKey**, **LTK** und **IRK** per

reversed in andere Betriebssysteme zu integrieren.

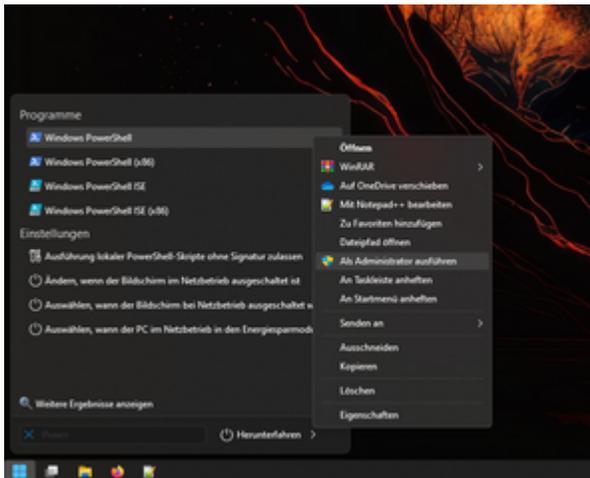
Das ist seit Monterey (vllt auch Big Sur) nicht mehr nötig.

Das ganze bitte wieder in einer **Text-Datei** notieren und so bereitstellen das wir sie unter Windows und Linux nutzen können.

Das sinnvollste wäre ein **Fat32** oder **exFat** USB Stick.

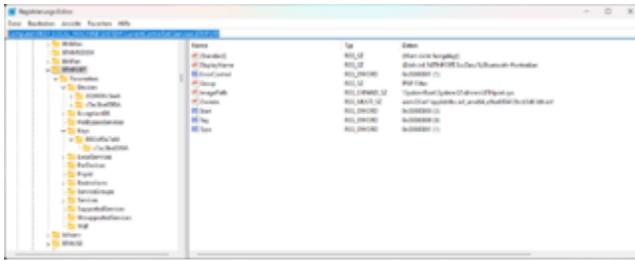
Schritt 5:

Jetzt gehen wir wieder zurück in Windows und wiederholen die in **Schritt 3** beschriebene Vorgehensweise die **Registry** wieder als Admin zu öffnen.



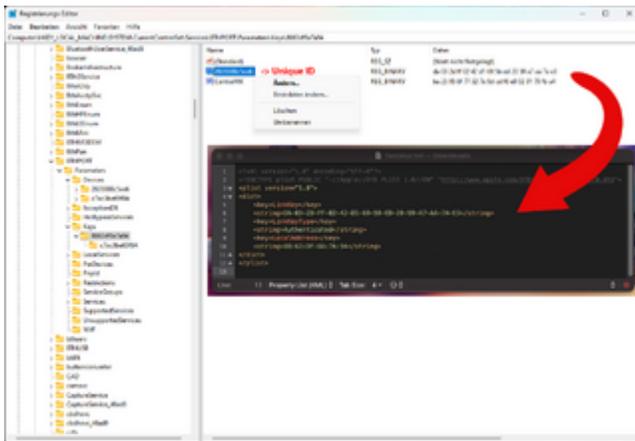
```
psexec -i -d -s regedit.exe
```

und **Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\BTHPORT\Parameters** navigieren wieder nach



Um die **LinkKeys** von **Bluetooth-Legacy-Hardware** zu übertragen reicht es einfach im Bereich **Keys** den aus macOS notierten Wert 1:1 zu übertragen.

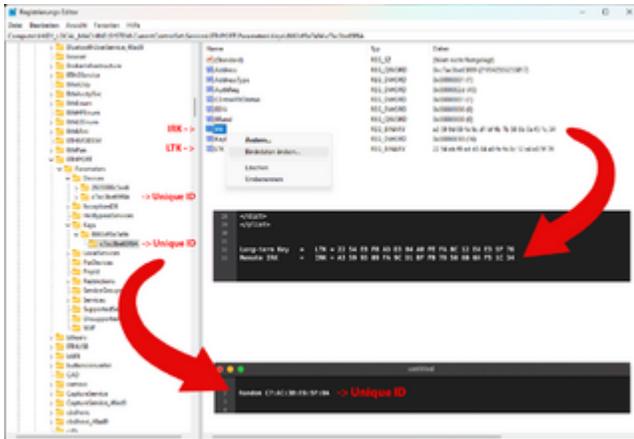
Das geht mittels **Rechtsklick** auf eure **Unique ID** und **Binärdaten ändern**.



Als nächstes müssen wir noch die notierten Werte des **BLE - Gerätes** übertragen.

Es ist wichtig das das reine einfügen des **IRK** und **LTK** nicht reichen um das Gerät zum funktionieren zu bewegen.

Es muss zwingend die von macOS erhalten **Random Unique ID** mitgenommen werden und in Windows und Linux angepasst werden.



Bitte drauf achten nicht den **IRK** mit dem **LTK** zu wechseln.

In meinem Fall spielt es keine Rolle wie rum ich mir das notiere.

Windows möchte zuerst den **IRK** und dann den **LTK**, Linux hingegen erst den **LTK** und dann den **IRK**.

Das ganze nochmal in Ruhe vergleichen und wenn soweit alles passt kann 1x neugestartet werden und sollte nun gehen.

Schritt 6:

Jetzt widmen wir uns Linux.

Ich gehe mal davon aus das grundlegende Sachen bei Linux-Nutzern vorhanden sind und werde auf die Vorgehensweise nur grob eingehen.

Nach dem wir Linux gestartet haben begeben wir uns in die Konsole (Terminal) und melden uns mittels **su -l** als **root** an.

Anschließend werfen wir einen kurzen Blick in den Bluetooth-Ordner.

```
ls /var/lib/bluetooth/
```



gefolgt von einem

```
cd /var/lib/bluetooth/[Unique ID]
```

[Unique ID] wäre in dem Fall **88:63:DF:8A:7A:94**

Mit **ls** lassen wir uns nochmal den Inhalt anzeigen.



Wie auf dem Bild zu sehen finden wir wieder unsere **Legacy Unique ID** und unsere **BLE Unique ID** wieder.

Die **BLE Unique ID** muss wieder an an die **Random Unique ID** seitens macOS angepasst werden.

```
mv ?7\:??\:??\:??\:5F\:8x ?7\:??\:??\:??\:5F\:8A
```

x = Random, A = von macOS

Jetzt befinden sich in **/var/lib/bluetooth/88:63:DF:8A:7A:94/** unseren beiden Geräte als Ordner mit den richtigen **Unique ID**.

20:20:00:0C:5E:E6 = Tastatur (BT - Legacy)

C7:AC:3B:E6:5F:8A = Maus (BLE)

In beiden eben erwähnten Ordnern befindet sich nochmal eine info Datei wo die Werte geändert werden müssen.

Das könnt ihr mir dem Editor eure Wahl mache ich nutze nano

```
nano /20\:20\:00\:0C\:5E\:E6/info
```



und

```
nano /C7\:AC\:3B\:E6\:5F\:8A/info
```



Das wars. Entweder 1x komplett Neustarten oder nur Bluetooth mittels Konsole Neustarten.

Bei Fragen stehe ich gern zur Verfügung genauso wie für irgendwelche Verbesserungsvorschlägen.



P.S Happy hackint0sh

Beitrag von „al6042“ vom 7. März 2024, 21:29

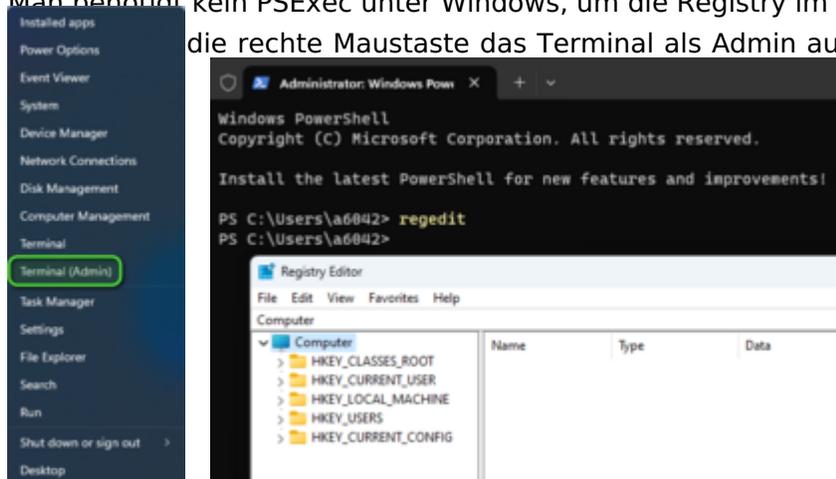
Vielen Dank für die Anleitung... 🤖

Leider kann ich dessen Funktion nicht testen, da meine Logitech Maus/Tastatur-Kombi über den Logitech-Dongle am USB-Port des Monitors alle auf dem PC installierten Betriebssysteme direkt unterstützt... 🤖

Aber hier noch ein kleiner Tipp:

Man benötigt kein PSEXec unter Windows, um die Registry im Admin-Mode zu öffnen.

Man benötigt die rechte Maustaste das Terminal als Admin ausführen und darin die regedit.exe



Sollte den gleichen Effekt haben, wie die PSEXec-Aktion.

Beitrag von „Pillenflitzer“ vom 7. März 2024, 21:37

Danke für die Rückmeldung. Ich habe mich leider etwas zu oberflächlich ausgedrückt um nicht ganz so tief in die Materie zu gehen.

Das öffnen der Windows-Registry ist grundsätzlich auch wie von dir beschrieben möglich.

Allerdings hat man bei dieser Vorgehensweise selbst als Admin kein Zugriff auf die in Windows hinterlegten Bluetooth-Keys.

Ergo können sie nicht geändert werden 😊

Der Umweg über PSEXec um Bluetooth zu realisieren ist leider unausweichlich.

Beitrag von „al6042“ vom 7. März 2024, 21:49

Verstehe... wegen dem Parameter -s (als Systemkonto).

Beitrag von „iPhoneTruth“ vom 23. Juli 2024, 18:23

Hat bei mir bestens funktioniert. Besten Dank für die sehr gute Anleitung.

Ich mußte lediglich in Windows 11 für den Befehl "`psexec -i -d -s regedit.exe`" die einfache "Eingabeaufforderung" als Administrator ausführen. Mit der "Windows PowerShell" als Administrator wollte es nicht klappen. Das als Hinweis für andere, die da vielleicht auch hängenbleiben.