FLIPPER ZERO - U2F

CONTEXTE & THÉORIE (INTRODUCTION)

La Hiérarchie de l'Authentification (MFA)

La sécurité des accès repose sur la multiplication des facteurs. Dans l'échelle de robustesse, le Flipper Zero permet de passer du niveau "Standard" au niveau "Militaire".

Niveau	Méthode	Facteur	Résistance au Phishing	Analogie de Charles
Faible	SMS / Email	Possession (Simulée)	**Nulle** (SIM Swap / Interception)	**WarGames** : C'est comme parler sur une fréquence radio ouverte. N'importe qui avec un scanner peut écouter la fréquence et intercepter le code.
Moyen	**TOTP** (Authenticator)	Possession (Soft)	**Moyenne** (Si l'utilisateur donne le code au pirate)	**Mission Impossible** : "Ce message s'autodétruira dans 30 secondes." Si vous n'êtes pas là au moment précis pour attraper le code, il disparaît.
Fort	**U2F / FIDO** (USB)	Possession (Hard)	**Totale** (L'échange cryptographique est invisible)	**GoldenEye** : La "Clé de Tir Nucléaire". Connaître le code de lancement (Mot de passe) ne suffit pas. Si vous n'insérez pas physiquement la clé dans le pupitre pour la tourner, le missile ne part pas.

Le Protocole FIDO/U2F

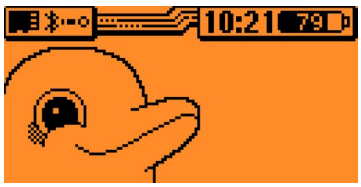
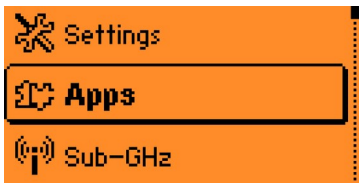
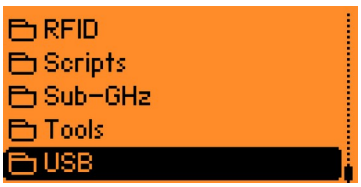
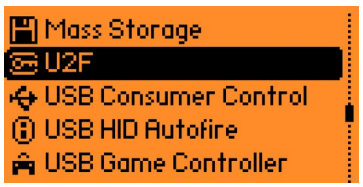
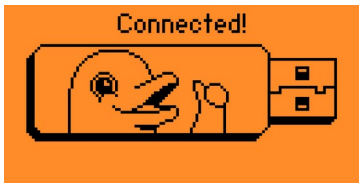
Le standard **U2F (Universal 2nd Factor)**, géré par l'Alliance FIDO, repose sur un défi cryptographique (Challenge-Response). Contrairement au TOTP où l'humain copie un code (faillible), ici le serveur envoie une énigme mathématique que seule la puce sécurisée du Flipper peut résoudre en signant avec sa clé privée. Si le site est un faux (Phishing), la signature échoue. C'est imparable.

Pourquoi une configuration spécifique sous Linux ?

Contrairement à Windows ou macOS qui disposent de "Couches d'Abstraction Matérielle" (HAL) propriétaires pour gérer les clés de sécurité, **Linux / Debian** laisse le contrôle total à l'utilisateur. Par défaut, le noyau (Kernel) isole les périphériques USB pour la sécurité. Ce document détaille comment créer le "pont" (Règles Udev) pour autoriser votre navigateur à dialoguer avec le Flipper Zero via le protocole HID (Human Interface Device).

MODULE EXTENSION U2F (LINUX/DEBIAN)

Contexte : Utilisation du Flipper Zero comme clé de sécurité physique (FIDO/U2F) sur un poste Linux.

<table>
<tr>
<td> 
</td>
<td> 
</td>
<td> 
</td>
<td> 
</td>
</tr>
<tr>
<td> 
</td>
<td> 
</td>
<td></td>
<td></td>
</tr>
</table>

Vérification de la Reconnaissance (Kernel)

Débranchez le Flipper. Lancez l'app **U2F** sur le Flipper. Rebranchez-le. Exécutez immédiatement :

```
sudo dmesg | tail -n 10
```

```
[13351.079725] usb 1-1: Product: U2F Token
[13351.079727] usb 1-1: Manufacturer: Flipper Devices Inc.
[13351.098482] hid-generic 0003:0493:5741.000D: hiddev3,hidraw7: USB HID v1.00 Device [Flipper Devices Inc. U2F
Token] on usb-0000:08:00.3-1/input0
[13619.499110] usb 1-1: USB disconnect, device number 16
[13623.928369] usb 1-1: new full-speed USB device number 17 using xhci_hcd
[13624.090308] usb 1-1: New USB device found, idVendor=0483, idProduct=5741, bcdDevice= 1.00
[13624.090314] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[13624.090317] usb 1-1: Product: U2F Token
[13624.090318] usb 1-1: Manufacturer: Flipper Devices Inc.
[13624.109229] hid-generic 0003:0493:5741.000E: hiddev3,hidraw7: USB HID v1.00 Device [Flipper Devices Inc. U2F
Token] on usb-0000:08:00.3-1/input0
```

Cherchez une ligne parlant de hid-generic ou hidraw. Si vous voyez des erreurs USB, le câble ou le port est suspect.

Le Flipper change d'identité selon son mode :

- **Mode Idle :** PID 5740
- **Mode App U2F :** PID 5741 (C'est le point critique souvent oublié).

Vérification des Droits (Permissions)

C'est le test critique. Voyons qui a le droit de parler au Flipper. Exécutez :

```
ls -l /dev/hidraw*
```

```
crw-------  1 root root 244, 0 Jan  4 09:42 /dev/hidraw0
crw-------  1 root root 244, 1 Jan  4 09:42 /dev/hidraw1
crw-rw----+ 1 root root 244, 2 Jan  4 09:42 /dev/hidraw2
crw-rw----+ 1 root root 244, 3 Jan  4 09:42 /dev/hidraw3
crw-rw----+ 1 root root 244, 4 Jan  4 09:42 /dev/hidraw4
crw-rw----+ 1 root root 244, 5 Jan  4 09:42 /dev/hidraw5
crw-------  1 root root 244, 6 Jan  4 09:42 /dev/hidraw6
crw-rw----+ 1 root root 244, 7 Jan  4 09:42 /dev/hidraw7
```

Vous devriez voir une liste. Repérez le périphérique le plus récent (créé à l'instant).

- **Résultat attendu :** crw-rw----+ 1 root root ... (Notez le + à la fin, qui indique que l'ACL uaccess a fonctionné et vous a donné les droits).
- **Résultat échec :** crw------- 1 root root ... (Seul root a accès, la règle udev a échoué).

Création de la règle Udev

```
sudo nano /etc/udev/rules.d/69-flipper-zero.rules
```

```
# Flipper Zero Serial / CDC ACM (Mode Normal)
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="5740", TAG+="uaccess"
KERNEL=="hidraw*", SUBSYSTEM=="hidraw", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="5740", TAG+="uaccess"

# Flipper Zero U2F App (Le fameux PID 5741 que nous avons vu dans les logs)
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="5741", TAG+="uaccess"
KERNEL=="hidraw*", SUBSYSTEM=="hidraw", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="5741", TAG+="uaccess"

# Flipper Zero DFU (Recovery)
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="df11", TAG+="uaccess"
```

Sauvegardez (Ctrl+O, Enter) et quittez (Ctrl+X).

Application du Changement

```
sudo udevadm control --reload-rules && sudo udevadm trigger
```

Vérification avec fido2

```
sudo apt install fido2-tools
```

```
Installing:
  fido2-tools

Summary:
  Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 24
  Download size: 44.2 kB
  Space needed: 181 kB / 901 GB available
```

```
fido2-token -L
```

Scénario A : La commande retourne une ligne type /dev/hidrawX: vendor=0x0483, product=0x5741 (Flipper Devices Inc. U2F Token).

- **Verdict :** Votre OS est parfaitement configuré. **C'est Firefox le coupable.**

Scénario B : La commande ne retourne rien.

- **Verdict :** C'est encore un problème de droits/udev ou de librairie manquante (libfido2).

CONFIGURATION NAVIGATEUR (Firefox)

```
which firefox
```

Si ça répond /usr/bin/firefox : C'est bien (version APT native).

Si ça ne répond rien ou si vous lancez via une icône qui pointe vers /var/lib/flatpak/... ou /snap/... : **C'est la cause racine.**

Pourquoi ? Les navigateurs en **Flatpak** ou **Snap** sont dans une prison (Sandbox). Ils n'ont *pas le droit* de voir les clés USB U2F, même avec les règles udev correctes, sauf si on leur donne une permission spécifique.

La Configuration Interne de Firefox (about:config)

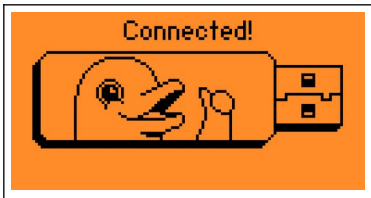
Si vous êtes bien sur un Firefox natif (/usr/bin/firefox) et que fido2-token -L voit la clé, alors Firefox fait du zèle.

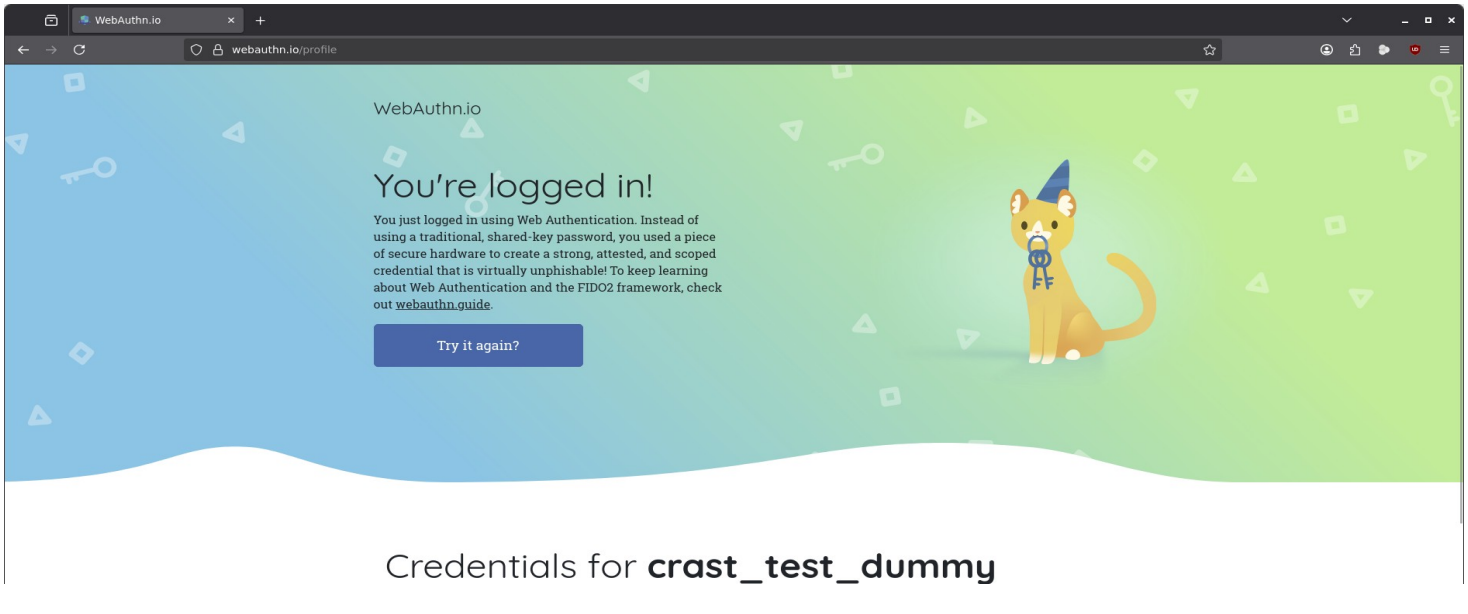
1. Ouvrez un nouvel onglet : about:config
2. Acceptez le risque (comme toujours).
3. Cherchez : security.webauthn
4. Vérifiez ces valeurs :
    - security.webauthn.u2f = **true**
        - **si .u2f n’existe pas : Forcer le support U2F (Legacy)**
            1. Dans la barre de recherche, tapez : security.webauthn.u2f
            2. Regardez la barre en bas qui propose : Boolean | Number | String.
            3. Laissez sur Boolean (le rond bleu).
            4. Cliquez sur le bouton [+] (Plus) à droite.
            5. Assurez-vous que la valeur est à true.
    - security.webauthn.ctap2 = **true**
5. **Forcez le backend (Optionnel mais radical) :**
    - Cherchez security.webauthn.rust_service. S'il est à true, passez-le à false (pour revenir à l'ancien backend C++, parfois plus tolérant avec les émulateurs comme le Flipper). *Note : Sur les Firefox très récents, cette option peut ne plus exister.*

VALIDATION & LIMITES

Utiliser **WebAuthn.io** pour valider la chaîne technique avant de blâmer un service tiers. **Paramètres obligatoires pour le Flipper :**

- **User Verification :** Discouraged
  - *Pourquoi ?* Le Flipper n'a pas de lecteur d'empreinte ni de clavier pour un code PIN complexe. Si le site exige ("Required") une vérification, le Flipper échouera.
- **Attachment :** Cross-Platform
  - *Pourquoi ?* Cela force le navigateur à chercher une clé externe (USB/NFC) et ignorer les "Passkeys" internes du PC.
- **Discoverable Credential :** Discouraged
  - *Pourquoi ?* Le Flipper en mode U2F ne stocke pas les "Resident Keys" (crédentiels résidents) comme une YubiKey série 5. Il génère les clés à la volée.
- **Attestation :** None
  - *Pourquoi ?* Le Flipper n'a pas de certificat racine signé par une autorité de certification (CA) reconnue par l'alliance FIDO. Si vous mettez "Direct" ou "Indirect", le test échouera car le certificat du Flipper est "fait maison".





Base d'Erreurs Connues (Known Errors)

Symptôme : Le Flipper réagit correctement (LED/Vibration + "Press Button") mais le site web affiche une erreur générique après l'interaction physique.

Diagnostic : Politique de sécurité trop stricte du Service Provider (ex: Infomaniak, Banques).

Analyse : Le service exige probablement une attestation certifiée (Hardware Certification) ou un standard FIDO2 pur que l'émulation logicielle du Flipper ne peut fournir sans clé privée constructeur.

Contournement : Utiliser le protocole **TOTP** (App Authenticator) pour ces services spécifiques.