How to generate DFU packet (nRF52832)

Created by Jona Cappelle

Step A: Generate keys

We need a pair of Public and Private Key to encrypt the signature and sign the DFU image using ECDSA_P256_SHA256.

- Install **nrfutil**:

  ```
  pip install nrfutil
  pip install constants
  ```

  Requirements: python version == 3.8.6 with pip version == 20.1.1

- Generate your own private and public key

  ```
  nrfutil.exe keys generate private.key
  nrfutil keys display --key pk --format code private.key --out_file public_key.c
  ```

- Replace the file in *"NRF_SDK/examples/dfu/dfu_public_key.c"* with the generated **"dfu_public_key.c"**

Step B: Build the bootloader

(Only if first time building on computer)

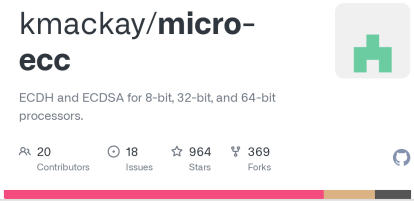
- Compile the **uECC library** (uECC library is needed for the bootloader to decrypt the signature)

  Download uECC from GitHub and extract into *"SDKFolder\external\micro-ecc\micro-ecc"*

kmackay/micro-ecc

A small and fast ECDH and ECDSA implementation for 8-bit, 32-bit, and 64-bit processors. The static version of micro-ecc (ie, where the curve was selected at compile-time) can be found in the "static"

https://github.com/kmackay/micro-ecc



- In *"SDKFolder\external\micro-ecc\nrf52hf_armgcc\armgcc"* **run make**

```
make
% Example:
C:/Users/JonaCappelle/Box/NOMADe/SOFTWARE/nRF5_SDK_17.0.2_d674dde/xpack-windows-build-tools-4.2.1-2/bin/make
```

- **Build bootloader** from project and program to device *"SDKFolder\examples\dfu\secure_bootloader\pca10040_s132_ble\armgcc"*

Step C(1): Generate DFU .zip packet (APP + SD)

- Prepare **application "nrf52832_xxaa.hex"** file in *"WorkspaceFolder\pca10040\s132\armgcc\nrf52832_xxaa.hex"*
- Prepare **softdevice**: **"s132_nrf52_7.2.0_softdevice.hex"** copy to same folder (default location: *"SDKFolder\components\softdevice\s132\hex\s132_nrf52_7.2.0_softdevice.hex")*
- Copy **"private.key"** generated in step A to same folder

```
nrfutil pkg generate --hw-version 52 --application-version 5 --application nrf52832_xxaa.hex
--sd-req 0x0101 --sd-id 0x0101 --softdevice s132_nrf52_7.2.0_softdevice.hex --key-file priva
te.key app_dfu_sd_app_1.zip
```

-- *sd-req 0x0101* → specific for s132 7.2.0, can be found in *"s132_nrf52_7.2.0_release-notes-update-2.pdf"*

-- *sd-is 0x0101* → same as req

"app_dfu_sd_app_1.zip" → name of zip DFU packet

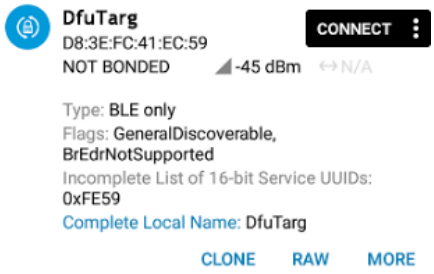
Step C(2): Generate DFU .zip packet (BOOTLOADER + APP + SD)

- Prepare **application "nrf52832_xxaa.hex"** file in *"WorkspaceFolder\pca10040\s132\armgcc\nrf52832_xxaa.hex"*
- Prepare **softdevice**: **"s132_nrf52_7.2.0_softdevice.hex"** copy to same folder (default location: *"SDKFolder\components\softdevice\s132\hex\s132_nrf52_7.2.0_softdevice.hex")*

- Copy **"private.key"** generated in step A to same folder
- Run “*04_bootloader_settings_merge_flash*”, this generates a “*merged.hex*”
- Upload the “*merged.hex*” to the nRF52

Step D: Performing DFU

- Copy **".zip" DFU packet** to smartphone with **nRF Connect** App



- Connect and upload ZIP