System and Security settings

hbdna-edit will require full access to the HB DNAfx-GIT device through USB in read/write mode.
As USB devices are from sensible to critical (ie. mass storage, possible network adapters, …) such access requires to be as secure and safe as possible. Currently this doc only covers Linux systems. Mac and Windows are planned to be covered later.

On the Linux side :

This will be easily done by creating a dedicated user and group for hbdna-edit usage (hbdna).
This user group will be granted read/write access to usb devices, but only to access the specific DNAfx-git device.

Recommended best practice : Those settings should apply both in production and during development to protect the system from any malware while granting users a full access to the DNAfx_git device. The only risk from those users, would then be a insidious malware forcing a firmware update, but that would really be focused on that specific device so is highly improbable to be deserving a specific investigation and development. Therefore risk may be considered as acceptable.

This will be enforced by restricting access to the device, which ids have been identified previously :

- idVendor = 0483
- idProduct = 5703

Then, to be granted such access a single user (prod or dev) just requires to be added to the hbdna group.

You will need to be root to do those settings (or use sudo if you prefer).
Here is a step by step procedure for doing so :

1. **create the hbdna user :**

```
[root@izabel ~]# useradd hbdna
```

2. **create an usb udev rule :**

   Create a new file as **/lib/udev/rules.d/40-usb-hbdna.rule**.
   I created a dedicated rule file, to avoid possible overwriting by system or other software during upgrade/patching that could reset their existing rules.That would remove the hbdna setting.

I choose the name “40-usb-hbdna.rule” because those rules are read sequentially and the prefix “40-” is used for declaring most USB devices access on my systems (Fedora), (ie. “40-usb-media-players.rules” declaring possible usb multimedia devices).
However you are free to choose your own prefix and name.

udev rules are applied depending on criterias of selection to attributes that identify the device on its USB bus. The attributes to define the rule we want are SUBSYSTEMS, ATTRS{idVendor} and ATTRS{idProduct}. Those have been identified previously. You should not need more criterias.

Enter your preferred editor (ie. vi), enter the following and save your rule.

```
#  Harley Benton DNAfx-git devices access rule
# User groups granted with read/write access

ACTION=="add", SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483",
ATTRS{idProduct}=="5703", MODE="0660", GROUP="hbdna"
```

Then reload the rules :

```
[root@izabel ~]# udevadm control –reload
```

replug the device and look at your regenerated dev file :

```
[root@izabel ~]# ls -l /dev/hidraw5
crw-rw----. 1 root hbdna 241, 5  3 oct.  07:22 /dev/hidraw5
```

hbdna group members now can write to that device (and that device only), other /dev/hidraw devices are still for root access only in 600 mode.

3. Add your user to the hbdna group

Assuming your user is named “myuser”, execute the following command :

```
[root@izabel ~]# usermod -a -G hbdna myuser
```

4. In case of issue…

From all tests done,with this specific rule, I did not encounter any issue.
However, as playing with udev rules for some time, I know those may be tricky sometimes to say the least.

So, if your udev rule does not apply (rights on device file are not changed), here are some hints :

- Be sure to have replugged the device to force device file recreation after you reloaded the rules.
- Be sure no other udev rule overwrite your settings by being executed after your own. Eventually rename it with a prefix ensuring that cannot occur (ie. 99-ZZZ-usb-hbdna.rule - it should be executed as the last one.)
- eventually reboot your platform. This should not be necessary as once reloaded udev rules apply at each device file creation.
- If issue is still there, then it means that there are some specificities in your USB settings that makes your rule non applicable. Typically it may be that some extra parameters are required in the rule. Use the qualification output from udevadm info to check if some other parameters could help.