

Primer Puzle en Ruby i Configuració RPi

Lucas Mira

Configuració de la Raspberry Pi4

Primerament el que vaig fer va ser instal·lar la última versió del sistema operatiu, descarregant-lo de la pàgina de Raspberry Pi i gravant-lo en una targeta de memòria micro SD utilitzant el programa balenaEtcher. Per a configurar el sistema operatiu i crear el meu usuari dins de la Raspberry ho vaig fer mitjançant una terminal gràfica ja que a casa disposava d'una pantalla amb connexió HDMI, amb un adaptador a microHDMI.

Una vegada configurat el que vaig fer va ser activar l'SSH des de la configuració de la Raspberry i crear un arxiu SSH a la micro SD. Per a treballar amb el terminal de la Raspberry sense utilitzar pantalla el que vaig fer es instal·lar una aplicació que es diu PuTTY que permet fer la connexió SSH de manera molt ràpida, només cal connectar el cable Ethernet des de l'ordinador a la placa i marcar la casella SSH a la aplicació i a host name escriure raspberrypi.local.

Per a compartir internet a la placa vaig compartir dades des del punt d'accés personal que disposa el iPhone, seguidament vaig accedir a la RPi mitjançant el monitor i a l'apartat de xarxes vaig buscar la meua posant-hi la contrasenya que prèviament havia posat des de l'iPhone i així vaig obtenir la WIFI, mirant la IP que assigna el meu iPhone a la RPi però poder accedir-hi des del puTTY en cas de no tenir el cable Ethernet.

Una vegada configurada la WIFI el que vaig fer va ser instal·lar el Ruby a la Raspberry a partir de la instrucció:

```
$ sudo snap install ruby -classic.
```

Posteriorment vaig activar el I2C a la raspi-config a través del terminal i vaig instal·lar la llibreria libnfc amb l'ordre

```
$ sudo apt install libnfc5 libnfc-bin libnfc-examples.
```

I li vaig fer saber la adreça del lector a/etc/nfc/libnfc.conf escrivint:

```
device.name = "pn532_i2c"
```

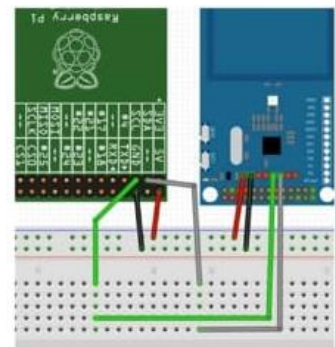
```
device.connstring="pn532_i2c:/dev/i2c-1"
```

També vaig instal·lar la gema ruby-nfc amb

```
$ sudo gem install ruby-nfc.
```

Una vegada fet això, vaig connectar la placa amb el sensor seguint un esquema de la pàgina web:

<https://itead.cc/nextion/raspberry-pi-drives-itead-pn532-nfc-module-with-libnfc/>.



Per comprovar que tot funcionava correctament i que el sensor podia detectar les RFID el que vaig fer es executar al terminal per a veure la RFID.

```
$ nfc-poll
```

```
lucas@Lucas:~$ nfc-poll
nfc-poll uses libnfc 1.8.0
NFC reader: pn532_i2c opened
NFC device will poll during 36000 ms (20 pollings of 300 ms for 6 modulations)
ISO/IEC 14443A (106 kbps) target:
  ATQA (SENS_RES): 00 04
  UID (NFCID1): 13 47 af 0f
  SAK (SEL_RES): 08
Waiting for card removing...nfc_initiator_target_is_present: Target Released
done.
lucas@Lucas:~$
```

Creació i execució del Puzzle1

El primer que vaig fer va ser instal·lar la llibreria ruby-nfc per a que funcioni el codi amb el NFC-pn532, vaig utilitzar la següent comanda:

```
$sudo gem install ruby-nfc.
```

El codi utilitzat per al funcionament del Puzzle es el següent:

```
1  require 'ruby-nfc'
2
3  class Rfid
4      @@readers = NFC::Reader.all
5
6      def read_uid
7          @@readers[0].poll(Mifare::Classic::Tag) do |tag|
8              begin
9                  uid = tag.uid_hex.upcase
10                 return uid
11             end
12         end
13     end
14 end
15
16 if __FILE__ == $0
17     rf = Rfid.new
18     puts "Si us plau, apropieu la targeta al lector"
19     uid = rf.read_uid
20     puts "El uid en hexadecimal es: ", uid
21 end
```

La sortida del terminal es la següent:

```
lucas@Lucas:~ $ cd Desktop
lucas@Lucas:~/Desktop $ ruby p1.rb
Si us plau, apropieu la targeta al lector
El uid en hexadecimal es:
1347AF0F
lucas@Lucas:~/Desktop $
```