

Primer Puzzle en Ruby: Elechouse PN532 NFC

Igor Yuziv

01/10/2022

1 Introducció

Al puzzle 1 jo he fet servir la placa PN532 d'elechouse amb la raspberry 2pi. He descarregat el sistema operatiu de la raspberry de la pàgina oficial i al moment de configurar he activat el ssh per poder després accedir des del terminal del portàtil. Per connectar internet a la raspberry he fet servir l'opció de compartir internet des del meu mòbil a través del portàtil.

A continuació explicaré tot el procés que he fet servir per configurar les biblioteques i el serial port de la raspberry.

2 Configuració

2.1 Instal·lar biblioteques

```
$ sudo apt-get install libusb-dev libpcsclite-dev libtool
```

```
$ sudo apt-get install automake autoconf
```

2.2 Clonar libnfc

```
$ git clone https://github.com/nfc-tools/libnfc.git
```

2.3 Compilar libnfc

```
$ cd libnfc
$ autoreconf -vis
$ ./configure --prefix=/usr --sysconfdir=/etc
$ make
$ sudo make install
```

2.4 Configurar Serial Port

```
$sudo mkdir /etc/nfc/ #Crear el directori
$sudo cp libnfc/libnfc.conf.sample /etc/nfc/libnfc.conf
$sudo nano /etc/nfc/libnfc.conf
Modificar la linia device.connstring = "pn532_uart:/dev/ttyUSB0"
i canviar per device.connstring = "pn532_uart:/dev/ttyAMA0"
```

En aquest punt s'ha de remarcar el problema que he tingut a l'hora de configurar. Com que jo faig servir la Raspberry pi2 només tinc el port AMA0 per serial port, en canvi, a partir de la versió 3 es fa servir ttyS0 i com que tots els tutorials a internet fan servir la versió de la raspberry a partir de la 3, he perdut molt de temps a esbrinar que el problema era aquest a l'hora de fer servir el comando nfc-list.

En aquest punt de la configuració amb el comando nfc-list des del terminal obrirem el port pel nostre sensor i a continuació amb nfc-poll comprovarem que el sensor funciona amb la targeta nfc.

2.5 Configurar la Biblioteca ruby-nfc

```
$gem install ruby-nfc
```

3 Codi en Ruby

```
require 'ruby-nfc'
require 'logger'
```

```

require 'colorize'

class Rfid

  #return uid in hexa str

  def read_uid
    reader = NFC::Reader.all
    reader[0].poll(Mifare::Classic::Tag) do |tag|
      begin
        uid = tag.uid_hex.upcase
        return uid
      end
    end
  end

end

end

if __FILE__ == $0
  rf = Rfid.new
  puts "ESPERANT_LECTURA".bold.red
  uid = rf.read_uid
  puts ""+uid.bold.cyan
end

```