

LECTOR RFID RT400a

Contexto:

RFID (*Radio Frequency Identification*): Es un sistema de comunicación inalámbrica, que tiene como propósito transmitir la identidad de un objeto (similar a un número de serie único) [1]. Además, permite el almacenamiento y recuperación de datos mediante dispositivos denominados etiquetas, tarjetas o transpondedores RFID y un subsistema de procesamiento de datos o *middleware* RFID. Estos dispositivos pueden ser adheridos o incorporados a un producto o individuo. Su funcionamiento es simple, la etiqueta que contiene los datos o información de identificación, genera una señal de radiofrecuencia con dichos datos. Esta señal es captada por un lector RFID, el cual se encarga de leer la información o extraerla y pasarla en formato digital al subsistema de procesamiento de datos o la aplicación que usa la tecnología.

En la actualidad, esta tecnología se está usando sobre todo para evitar el robo, ya que es una forma económica de localizar objetos. Está presente en prendas de ropa, termos, chips para mascotas, manillas de ingreso, tarjetas de identificación y bancarias. En cuanto a aplicaciones potenciales, RFID podría ser un sustituto ideal de los códigos de barra, o estar presentes en la señalización de carreteras con balizas para indicar a los coches los límites de velocidad y también en interiores como instituciones o empresas para mantener informados a los usuarios respecto a las actividades de las mismas o tener un control dentro de estos lugares por parte de los administradores [2].

Un ejemplo real es el de la compañía de Disney quienes están implementando en sus parques estas tecnologías en objetos como termos o pulseras inteligentes llamadas MagicBands, con el objetivo de mejorar el servicio a sus clientes, optimizar las visitas y liberar al cliente de toda carga. En el caso de las manillas, estas se conectan inalámbricamente a los accesos de las atracciones, haciendo que el usuario de esta pulsera no necesite dinero y/o entradas, y únicamente necesite acercar el dispositivo a los lectores, incluso para pedir una comida o bebida. Esto demuestra que esta tecnología facilita la prestación de servicios y la interacción de usuarios.

Basado en lo anterior, en la presente práctica utilizamos un lector RFID, el cual se conecta vía USB a nuestro dispositivo IoT (Raspberry Pi 3B). Como el lector RFID trabaja por medio de comandos COM, se hace necesario programarlo con una serie de comandos hexadecimales dados por el fabricante [3]. Además, tendremos un termo con una etiqueta RFID de la compañía Disney, el cual tiene un identificador. Con dichos componentes configurados, se lleva a cabo la lectura del termo a través del dispositivo IoT, el cual mediante un script desplegará la información sobre la identificación, contenida en el.

Requisitos:

Antes de empezar la realización de la práctica es necesario tener, descargar e instalar las siguientes herramientas y dependencias:

1. Computador y conexión a internet.

- Con el sistema operativo Windows, acceso a la red de área local (conexión a internet).

2. Elementos RFID.

- Lector RFID RT400a

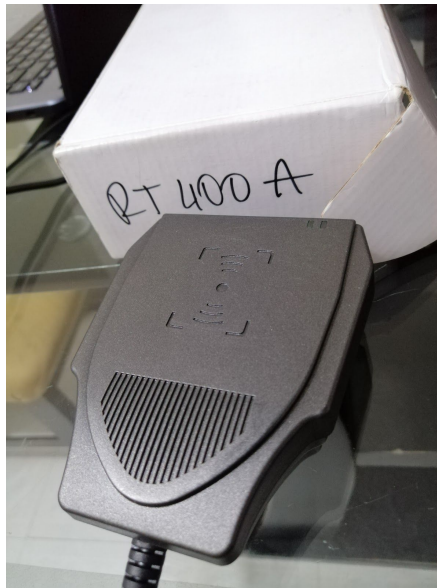


Figura 1. Lector RFID

- Etiqueta RFID o termo con tecnología RFID (elementos usados en parques de Disney)



Figura 2 y 3. Tags RFID (UHF) y termo RFID (UHF)

3. Placa Raspberry Pi modelo 3B.

- la presente práctica se desarrollo con este modelo de raspberry, sin embargo no existen inconveniente al usar otras.

4. Imagen recomendada:

- Raspbian Jessie (enlace de descarga):
<http://downloads.raspberrypi.org/raspbian/images/raspbian-2017-07-05>
- Tutorial:
<https://www.raspberrypi.org/documentation/installation/installingimages/README.md>

5. Python 2.7

- Python 2.7 está preinstalada en la imagen del requisito 4 .
- En caso de que el sistema operativo no cuente con esta versión de Python, puede descargarse con el siguiente link:
<https://www.python.org/downloads/release/python-2715>

6. PuTTY

- Herramienta para conexión SSH.
- Únicamente para Windows.
- Puedes descargar la aplicación en el siguiente enlace: <https://www.putty.org>

Arquitectura:



Figura 4. Arquitectura

Desarrollo:

1. Conexión SSH con la tarjeta Raspberry Pi:

- Conecta tu el dispositivo IoT (Raspberry Pi) a internet utilizando la interfaz Ethernet o WiFi (Si has configurado previamente las credenciales), para esto es necesario usar una pantalla, un mouse y un teclado, conectados a los periféricos de la raspberry que facilite las configuraciones del entorno del dispositivo.

(Windows)

- Abre la aplicación PuTTY, ingresa la IP local, ingresa el puerto 22 y selecciona la conexión como SSH

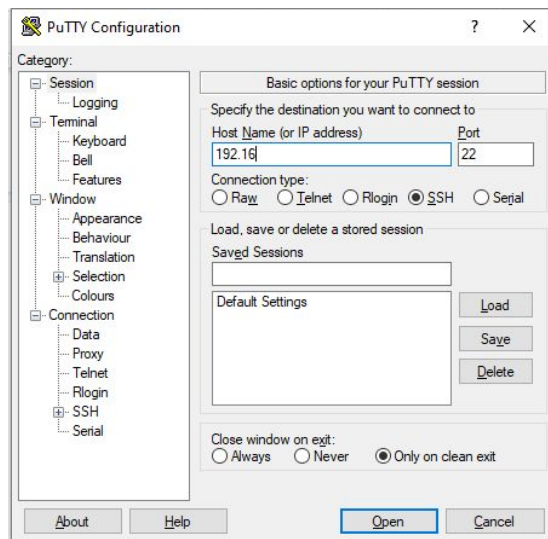


Figura 5. Inicio con Putty

- Después de este proceso se abre una ventana en donde debes ingresar el usuario y contraseña de tu dispositivo. Por defecto el usuario es "pi" y la contraseña es vacía o en otros caso raspberry.
- Ahora estás dentro de tu dispositivo Raspberry Pi.

(Linux)

- Abre un terminal y escribe `ssh pi@direccion_ip_raspberry` (si el usuario que se maneja no está por defecto cambia "pi" por tu usuario).
- Se te pedira un password ingresa la contraseña por defecto (raspberry) o ingresa la que has configurado.
- Ahora estás dentro de tu dispositivo Raspberry Pi.

2. Configuración e instalación de herramientas extra:

- Instalar herramientas de desarrollo de Python (pyserial):

```
sudo apt-get update
sudo apt-get upgrade
sudo pip install pyserial
```

3. Crear Script Python para lector RFID:

- Cree un nuevo script con el nombre comRFIDSingle.py:
- Pega el código dentro del editor que está al final del documento.
- Ingresa los siguientes comandos para guardar los cambios:

```
CTRL + x
Yes
```

ENTER

- Una vez creado el script anterior, vuelve a crear un nuevo script con el nombre `comRFIDMultiple.py`:

```
sudo nano comRFIDMultiple.py
```

- Pega el código dentro del editor que está al final del documento.
- Ingresa los siguientes comandos para guardar los cambios:

CTRL + x

Yes

ENTER

4. Probar el primer programa en dispositivo

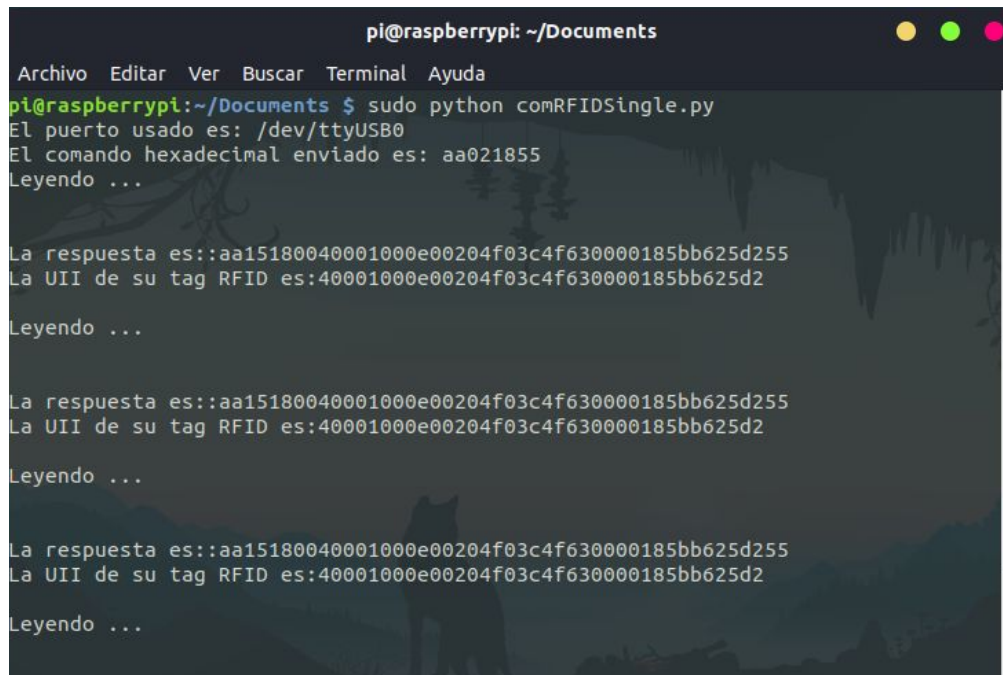
- Conecta el lector RFID a un puerto USB
- Comprobar en qué puerto se encuentra (es recomendable solo tener un dispositivo usb conectado para no confundir los puertos), para esto ingresa el siguiente comando:

```
ls /dev/*USB*
```

- El comando anterior arrojará el puerto que está utilizando el lector RFID, si su puerto es diferente a `'/dev/ttyUSB0'` cambie el puerto en la línea 4 del código `comRFIDSingle.py`
- Ingresa el siguiente código para iniciar el programa:

```
sudo python comRFIDSingle.py
```

- Pon un tag RFID a una distancia no cercana al lector RFID (15 cm aproximadamente), en el caso del termo se debe poner justo encima del lector RFID.
- En el terminal debe aparecer algo como lo mostrado en la figura 6.



```
pi@raspberrypi: ~/Documents
Archivo Editar Ver Buscar Terminal Ayuda
pi@raspberrypi:~/Documents $ sudo python comRFIDSingle.py
El puerto usado es: /dev/ttyUSB0
El comando hexadecimal enviado es: aa021855
Leyendo ...

La respuesta es::aa15180040001000e00204f03c4f630000185bb625d255
La UII de su tag RFID es:40001000e00204f03c4f630000185bb625d2

Leyendo ...

La respuesta es::aa15180040001000e00204f03c4f630000185bb625d255
La UII de su tag RFID es:40001000e00204f03c4f630000185bb625d2

Leyendo ...

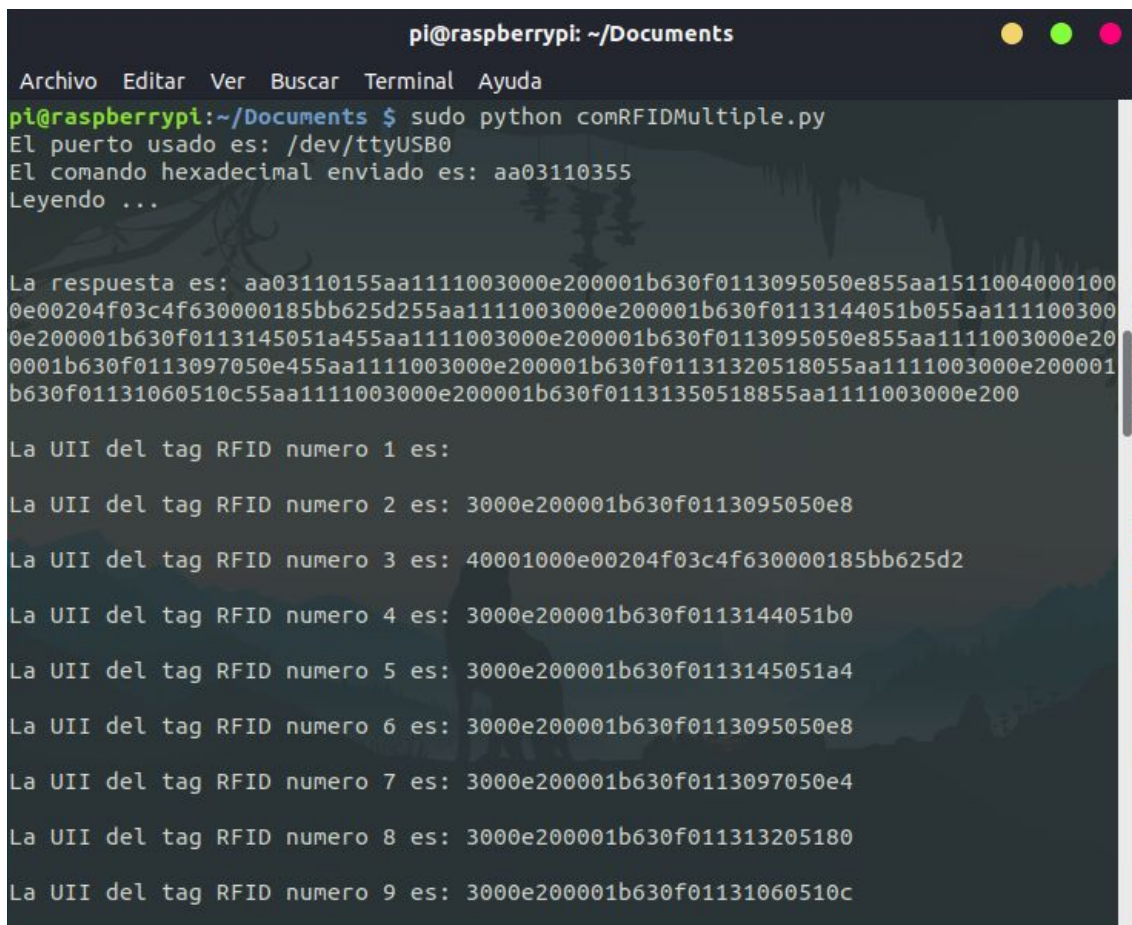
La respuesta es::aa15180040001000e00204f03c4f630000185bb625d255
La UII de su tag RFID es:40001000e00204f03c4f630000185bb625d2

Leyendo ...
```

Figura 6. Ejecución script `comRFIDSingle.py`

5. Probar el segundo programa en dispositivo

- Conecta el lector RFID a un puerto USB
- Comprobar en qué puerto se encuentra (es recomendable solo tener un dispositivo usb conectado para no confundir los puertos), para esto ingresa el siguiente comando:
`ls /dev/*USB*`
- El comando anterior arrojará el puerto que está utilizando el lector RFID, si su puerto es diferente a '/dev/ttyUSB0' cambie el puerto en la línea 4 en el código `comRFIDMultiple.py`
- Ingresa el siguiente código para iniciar el programa:
`sudo python comRFIDMultiple.py`
- Pon varios tags RFID a una distancia no cercana al lector RFID (15 cm aproximadamente), en el caso del termo se debe poner justo encima del lector RFID.
- En el terminal debe aparecer algo como lo mostrado en la figura 7.



```
pi@raspberrypi: ~/Documents
Archivo Editar Ver Buscar Terminal Ayuda
pi@raspberrypi:~/Documents $ sudo python comRFIDMultiple.py
El puerto usado es: /dev/ttyUSB0
El comando hexadecimal enviado es: aa03110355
Leyendo ...

La respuesta es: aa03110155aa1111003000e200001b630f0113095050e855aa1511004000100
0e00204f03c4f630000185bb625d255aa1111003000e200001b630f0113144051b055aa111100300
0e200001b630f0113145051a455aa1111003000e200001b630f0113095050e855aa1111003000e20
0001b630f0113097050e455aa1111003000e200001b630f01131320518055aa1111003000e200001
b630f01131060510c55aa1111003000e200001b630f01131350518855aa1111003000e200

La UII del tag RFID numero 1 es:

La UII del tag RFID numero 2 es: 3000e200001b630f0113095050e8
La UII del tag RFID numero 3 es: 40001000e00204f03c4f630000185bb625d2
La UII del tag RFID numero 4 es: 3000e200001b630f0113144051b0
La UII del tag RFID numero 5 es: 3000e200001b630f0113145051a4
La UII del tag RFID numero 6 es: 3000e200001b630f0113095050e8
La UII del tag RFID numero 7 es: 3000e200001b630f0113097050e4
La UII del tag RFID numero 8 es: 3000e200001b630f011313205180
La UII del tag RFID numero 9 es: 3000e200001b630f01131060510c
```

Figura 7. Ejecución script `comRFIDMultiple.py`

Código comRFIDSingle.py:

```
import serial
import time

port = '/dev/ttyUSB0'
ser = serial.Serial(port, 57600, timeout=1) #the USB port is ttyUSB0 this can
change, the recommended baud rate is 57600

#these hexadecimal code can be used with ser.write(). we use commandInventoryS, for
more functions consult RT400_CommunicationsProtocol.pdf
commandInventoryS = "\xAA\x02\x18\x55" #Hex to query a single tag in
single mode
commandToSend = commandInventoryS

print("El puerto usado es: "+str(ser.name))
ser.write(commandToSend)
print("El comando hexadecimal enviado es: "+str(commandToSend.encode('hex')))

i = 0
while i < 5:
    print("Leyendo ...")

    rfidRead = ser.read(50)
    rfidReadSTR = str(rfidRead.encode('hex'))
    print("\n")
    print("La respuesta es: "+rfidReadSTR)
    uii = rfidReadSTR.replace("55", " ")
    print("La UII de su tag RFID es: "+uii[8:]+ "\n")

    ser.close()
    time.sleep(3)
    ser = serial.Serial(port, 57600, timeout=1)
    ser.write(commandToSend)

    i += 1

ser.close()
print("Programa finalizado")
```

Código comRFIDMultiple.py:

```
import serial
import time

port = '/dev/ttyUSB0'
ser = serial.Serial(port, 57600, timeout=1) #the USB port is ttyUSB0 this can
change, the recommended baud rate is 57600

#these hexadecimal code can be used with ser.write(). we use commandInventoryM, for
more functions consult RT400_CommunicationsProtocol.pdf
commandInventoryM = "\xAA\x03\x11\x03\x55" #Hex to query a single tag in loop
mod
commandToSend = commandInventoryM #select the command to use

print("El puerto usado es: "+str(ser.name))
ser.write(commandToSend)
print("El comando hexadecimal enviado es: "+str(commandToSend.encode('hex')))

i = 0
numCont = 1
while i < 5:
    print("Leyendo ...")
    print("\n")
```

```

rfidRead = ser.read(188)
rfidReadSTR = str(rfidRead.encode('hex'))

print("La respuesta es: "+rfidReadSTR+"\n")
uii = rfidReadSTR.replace("55", "\n")
uii = uii.split()
for x in uii:
    print("La UII del tag RFID numero "+str(numCont)+" es: "+x[8:]+\n")
    numCont += 1

ser.close()
time.sleep(3)
ser = serial.Serial(port, 57600, timeout=1)
ser.write(commandToSend)
i += 1
numCont = 1

ser.close()
print("Programa finalizado")

```

Problemas Comunes:

1. Configuración de puertos puede cambiar repentinamente.
 - Ingresar el comando `ls /dev/*USB*` y cambiar la línea 4 por el puerto que se muestra por la ejecución del comando.
 - Otra solución más simple es desconectar y reconectar el lector RFID, aunque es necesario comprobar si el puerto cambió.
2. El script no se ejecuta.
 - Una de las razones por la que el script no se podría ejecutar, es debido a que un tag RFID está muy cerca del lector, esto impide el correcto funcionamiento de este.

Referencias:

- [1] «RFID - Wikipedia, la enciclopedia libre». [En línea]. Disponible en: <https://es.wikipedia.org/wiki/RFID>.
- [2] «Rfid: qué es y cómo funciona | VIU». [En línea]. Disponible en: <https://www.universidadviu.com/rfid-que-es/>.
- [3] «RT400_CommunicationsProtocol». [En línea]. Disponible en: http://www.stronglink-rfid.com/download/RT400_CommunicationsProtocol.pdf

Información complementaria:

Visita el sitio web del fabricante: <http://www.stronglink-rfid.com/es/rfid-readers/rt400a.html>