

RFID-Authentifikation

P. Bischofberger, J. Birnbreier, C. Ludwig
Fachinformatiker Systemintegration

Abstract

Es wurde ein Programm unter Python entwickelt, welches die Authentifizierung eines Users per RFID-Chip ermöglicht. Die ID des Chips wird kontaktlos ausgelesen und mit einer SQL-Datenbank abgeglichen. Ist der User hinterlegt wird ein Relais geschaltet, welches den Türöffner aktiviert. Die User-Datenbank kann über das Programm administriert werden. Zudem werden alle erfolgreichen Zutritte gelogget.

RFID

steht für Radio-Frequency-Identification. Ein Sender-Empfänger System für berührungsloses Identifizieren.

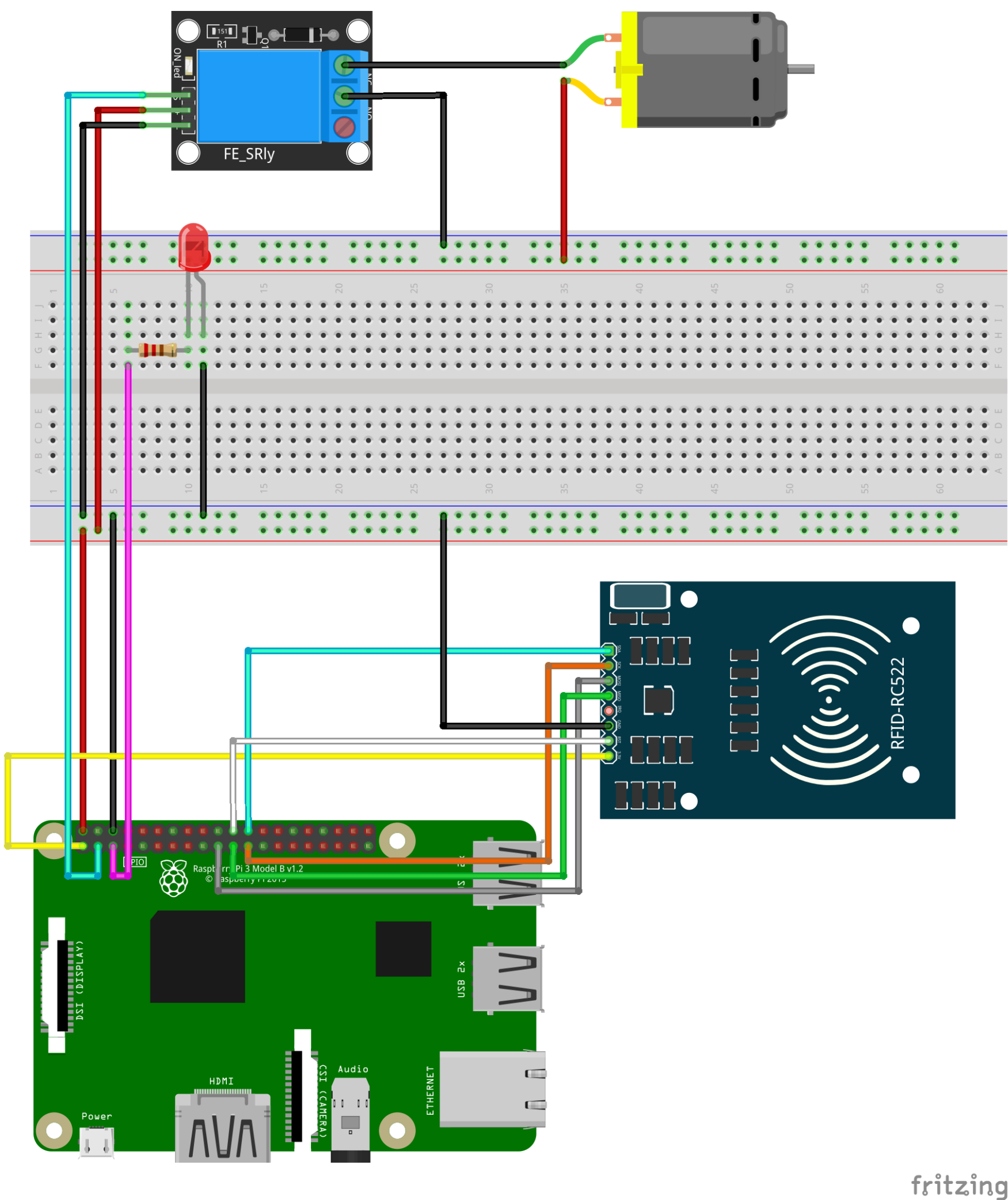
Das Lesegerät erzeugt ein elektromagnetisches Wechselfeld. Der Transponder (oder RFID-Tag) empfängt und absorbiert diese Wellen, was dann als Stromversorgung für das Tag fungiert. Der Mikrochip im Tag decodiert das empfangene Signal und codiert daraufhin die Antwort - im einfachsten Fall die Tag-ID. (1)

Komponenten/Software

Raspberry-Pi 3b+	Debian 11
5v Relais	MariaDB
RFID RC522 Read/Writer	Python

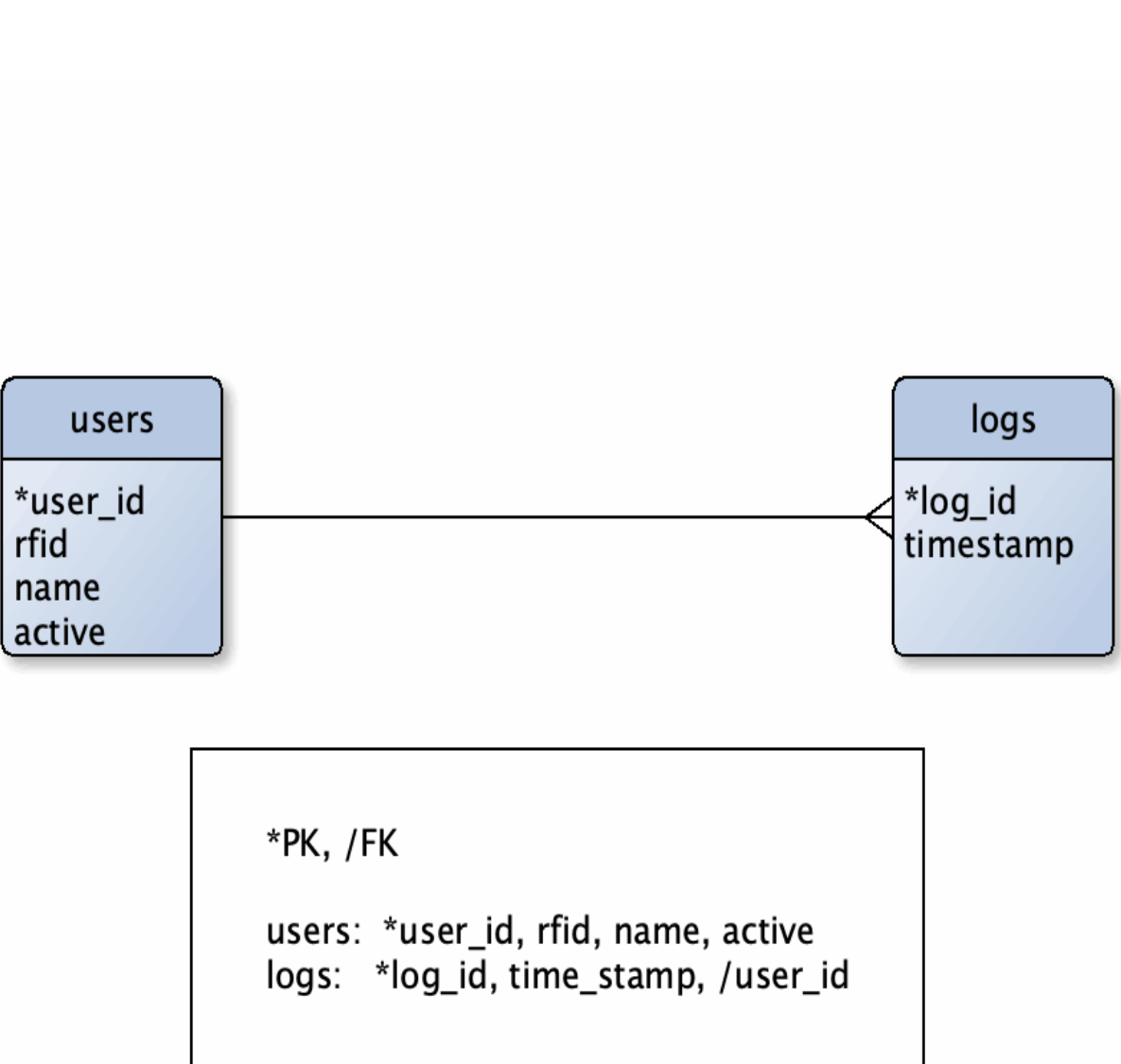
Schaltplan

Das RC522 Modul wird mit den Raspberry GPIOs verbunden. Über jeweils einen digital Out wird das Relay und die LED angesteuert. Das Relay schliesst den Türöffner-Stromkreis. Die LED dient als optisches Feedback.



Datenbank

Auf Basis von MariaDB wurde eine simple Datenbank aufgesetzt, in welcher die User verwaltet werden und zugehörige Logs bei erfolgreicher Authentifizierung erfolgen.



```
CREATE TABLE users (  
  user_id INT PRIMARY KEY AUTO_INCREMENT,  
  rfid BIGINT UNIQUE,  
  name VARCHAR(80),  
  active TINYINT(1) DEFAULT 1  
);  
  
CREATE TABLE logs (  
  log_id INT PRIMARY KEY AUTO_INCREMENT,  
  time_stamp TIMESTAMP,  
  user_id INT REFERENCES users(user_id),  
);
```

Referenzen

- (1) Wikipedia - RFID
- (2) Hardware & Security, Tobias Scheible
- (3) Alibaba
- (4) <https://proxmark.github.io/proxmark3/>

Sicherheits-Aspekte

Wird die Tag-ID im Klartext übertragen (wie in unserem Fall), besteht ein großes Sicherheits-Risiko. Ein Angreifer kann mit einem Lesegerät unbemerkt die ID des Tags auslesen (Antenne im Rucksack/Hose) und dadurch den Tag klonen. Bei fortgeschrittenen Varianten ist ein kleiner Mikrocontroller mit kryptografischen Funktionen verbaut. Hierfür müssen Reader und Tag miteinander kompatibel sein. Weit verbreitet ist der MIFARE-Classic Standard, der jedoch als unsicher gilt. (2)

Angriffsvektoren

Ziel eines Angreifers ist es, den RFID Tag auszulesen, um eine Kopie zu erstellen und dadurch die Zutrittskontrolle zu umgehen.

- 1. RF Field Detector (für Reconnaissance)
- 2. RFID-Tag-Cloner
- 3. Proxmark 3 RDV4.01
- 4. NFC-Kill

Detektoren sind nichts anderes als kleine Antennen mit einer LED, um RFID Felder aufzuspüren. Typische Frequenzen sind 13,56 MHz, 125 kHz und 134 kHz. Mit **Clonern** lassen sich einfache Standard-Tags kopieren auf beschreibbare Tags. Der **Proxmark 3** ist das «schweizer Taschenmesser» unter den RFID-Tools. Mit ihm lassen sich sehr viele Standards simulieren und man kann den Mifare Classic Schutz umgehen mit der gleichnamigen Software unter Kali-Linux. Mit einem **NFC-Kill Tool**, welches ein sehr starkes elektromagnetisches Feld erzeugt, lässt sich ein RFID-Tag sogar zerstören.



Abb.3: RFID-Tag-Cloner
(3)



Abb.4: Proxmark 3 RDV4.01
(4)

Sicherheits-Fazit

- > Sicherheit des eingesetzten Standards muss gegeben sein - wurde er schon gebrochen?
- > RFID-Blocker benutzen um Auslesen zu verhindern
- > Falls technisch möglich einen 2. Faktor benutzen
- > Nutzer Security-Awareness-Schulungen