

# RFID Thief v2.0

## Table of Contents

- Overview
  - Proxmark 3
  - Long Range Readers
  - Wiegotcha
- Raspberry Pi Setup
- Wiring
  - Raspberry Pi
  - HID iClass R90
  - HID Indala ASR620
  - HID MaxiProx 5375
  - Controller (Optional)
- Tutorial
  - iClass R90
  - Indala ASR620
  - MaxiProx 5375
- Components
- References

## Overview

This post will outline how to build and use long range RFID readers to clone iClass, Indala & Prox cards used for Access Control.

## Proxmark 3

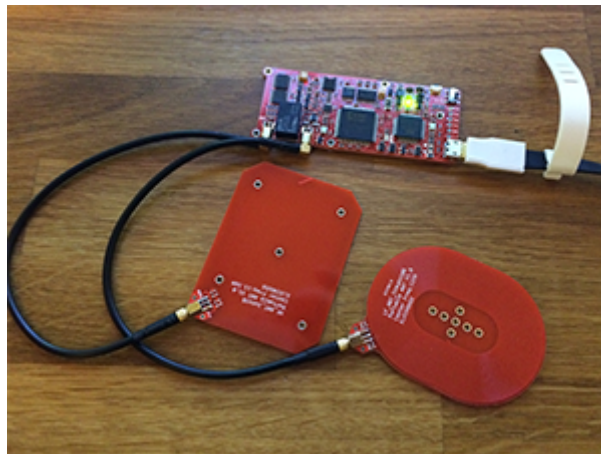
If you are unfamiliar with the Proxmark 3, it is a general purpose RFID Cloning tool, equipped with a high and low frequency antenna to snoop, listen, clone and emulate RFID cards. There are currently 5 versions of the Proxmark 3, all use the same firmware and software however some have more/less hardware features.

Version	Picture
---------	---------

Original



RDV1



RDV2



RDV3




RDV4



## Long Range Readers

There are 3 main types of long range readers HID sell, the R90, ASR-620 and MaxiProx 5375. Each reader supports a different type of card:

Reader	Card Type	Picture
--------	-----------	---------

<p>HID iClass R90</p>	<p>iClass Legacy (13.56 MHz)</p>	 A black, rectangular HID iClass R90 device is shown on a white surface. It has a red hexagonal light on the right side. In the background, a laptop and some cables are visible.
<p>HID Indala ASR-620</p>	<p>Indala 26bit (125 kHz)</p>	 A black, rectangular HID Indala ASR-620 device is shown on a wooden surface. It has a small, square, black light on the left side. The brand name "INDALA" is visible at the bottom.

HID MaxiProx 5375	ProxCard II (125 kHz)	
-------------------	--------------------------	---

## Wiegotcha

Wiegotcha is the awesome software for the Raspberry Pi developed by [Mike Kelly](#) that improves upon the Tastic RFID Thief in the following areas:

- Acts as a wireless AP with a simple web page to display captured credentials.
- Automatically calculates the iClass Block 7 data for cloning.
- Uses a hardware clock for accurate timestamps.
- AIO solution, eliminates the need for custom PCB's and multiple breakout boards.
- Utilizes an external rechargeable battery.

## Raspberry Pi Setup

This build will make use of the Raspberry Pi 3 to receive the raw Wiegand data from the long range readers and provide an access point to view/save the collected data.

### MicroSD Card Setup

1. Download and extract [Raspbian Stretch](#).
2. Download [ethcher](#) or any disk writer you prefer.
3. Write the Raspbian Stretch .img file to the MicroSD card using a USB adapter.
4. Unplug and replug the USB adapter to see 'boot' drive.

5. Edit `cmdline.txt` and add `modules-load=dwc2,g_ether` after the word `rootwait` so that so that it looks like this:

6. Edit `config.txt` and append `dtoverlay=dwc2` to the end of the file.

7. Create a blank file within the 'boot' directory called `ssh`.

## Raspberry Pi Configuration

1. Connect the RPi to your local network and ssh to it using the default password `raspberrypi`.

2. Run `sudo su` to become the root user.

3. Clone the Wiegotcha repository to the `/root` directory.

4. Run the install script in the `Wiegotcha` directory.

5. Follow the prompts as requested, the RPi will reboot once completed. Be patient, this process can take some time.

6. After reboot, reconnect to the RPi using ssh and enter the following:

7. RPi will reboot and the installation is completed.

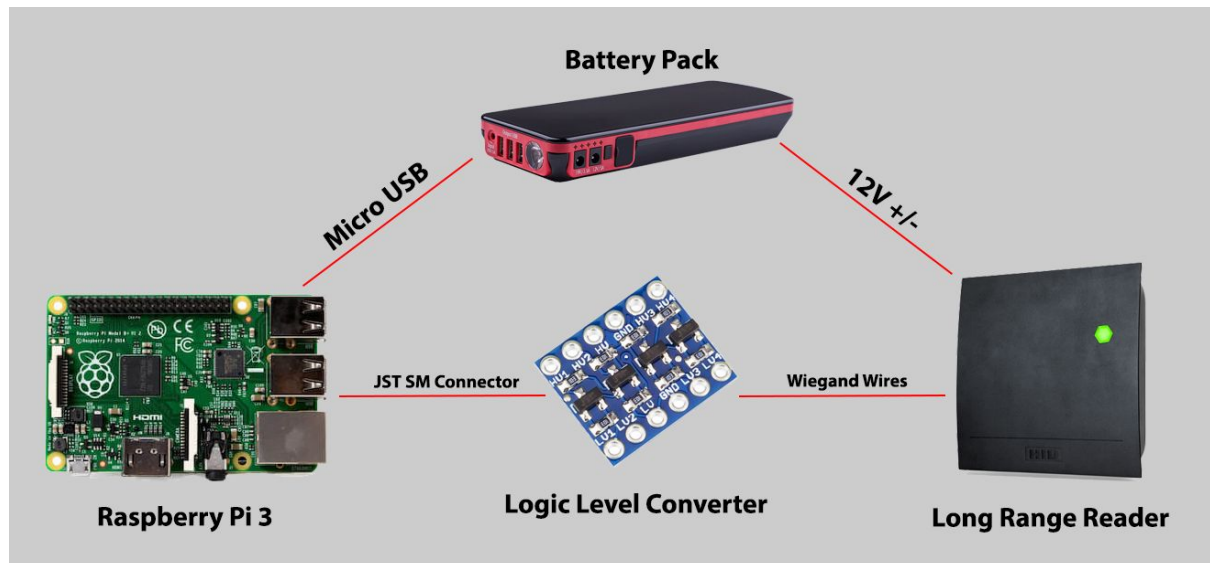
The RPi will now broadcast with the ESSID: Wiegotcha, you can connect to it using the passphrase `Wiegotcha`. Wiegotcha assigns the static IP `192.168.150.1` to the RPi.

## Wiring

Each reader will require a Bi-Directional Logic Level Converter, this is used to convert the 5v Wiegand output from the readers to the 3.3v RPi GPIOs. For quality of life, I have added JST SM connectors allowing quick interchangeability between the different long range readers.

You may choose to add another external controller with switches to power the readers on/off, enable/disable sound or vibration, however this is optional.

The following is a general overview of how the components are connected together:



## RPi

- GPIO Pins 1,3,5,7,9 -> Hardware RTC

## RPi to Logic Level Converter

- GPIO Pin 4 -> LLC HV
- GPIO Pin 6 -> LLC LV GND
- GPIO Pin 11 -> LLC LV 1
- GPIO Pin 12 -> LLC LV 4
- GPIO Pin 17 -> LLC LV

## Long Range Reader to Logic Level Converter

- LRR DATA 0 (Green) -> LLC HV 1
- LRR DATA 1 (White) -> LLC HV 4
- LRR SHIELD -> LLC HV GND

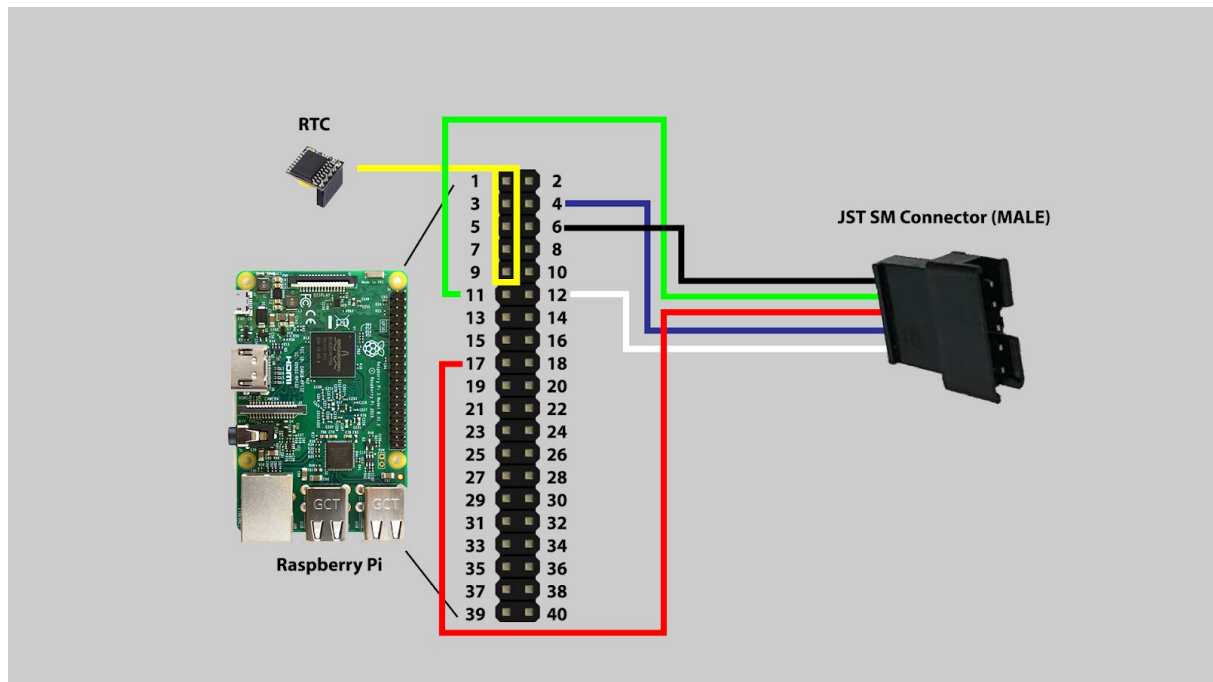
# Raspberry Pi

1. Connect the Hardware RTC to GPIO pins 1,3,5,7,9.

2. Solder female jumper wires to a male JST SM connector according to the table below and connect to the RPi.

RPi	JST SM Connector
GPIO Pin 4	Blue

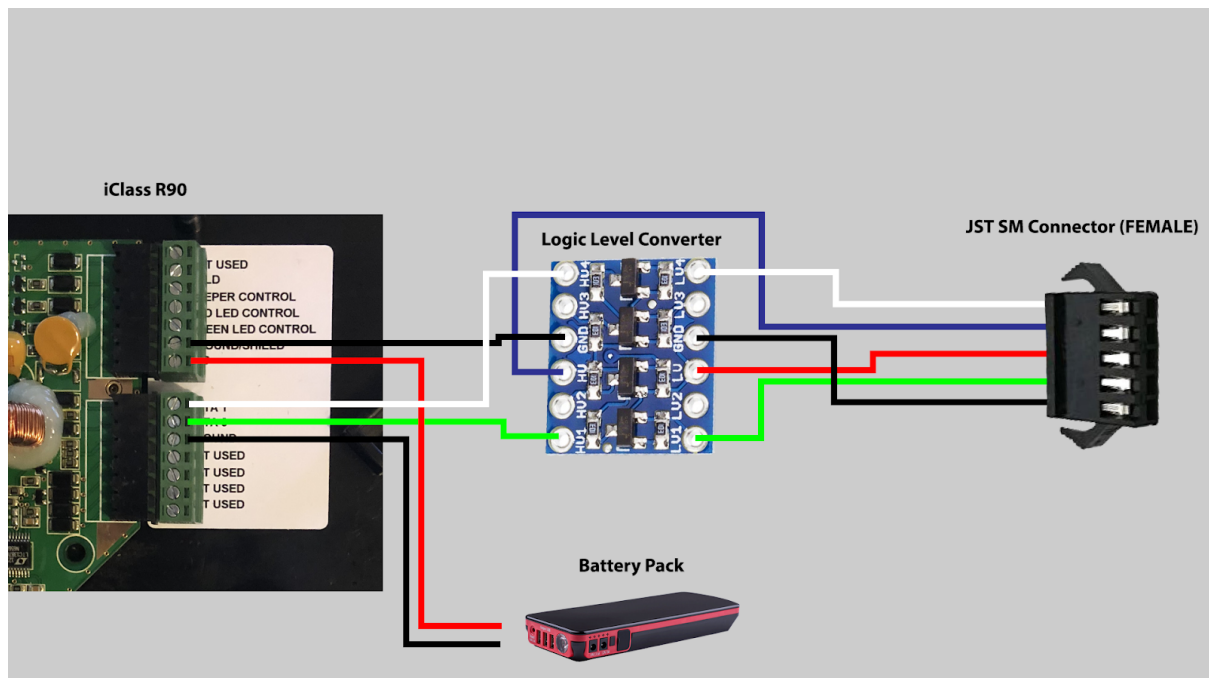
GPIO Pin 6	Black
GPIO Pin 11	Green
GPIO Pin 12	White
GPIO Pin 17	Red







## HID iClass R90



1. Join wires from the HID R90 to the logic level converter according to the table below.

<b>HID R90</b>	<b>Logic Level Converter</b>
P1-6 (DATA 0)	HV 1
P1-7 (DATA 1)	HV 4
P2-2 (GROUND/SHIELD)	HV GND

2. Solder female jumper wires from the logic level converter to a female JST SM connector according to the table below.

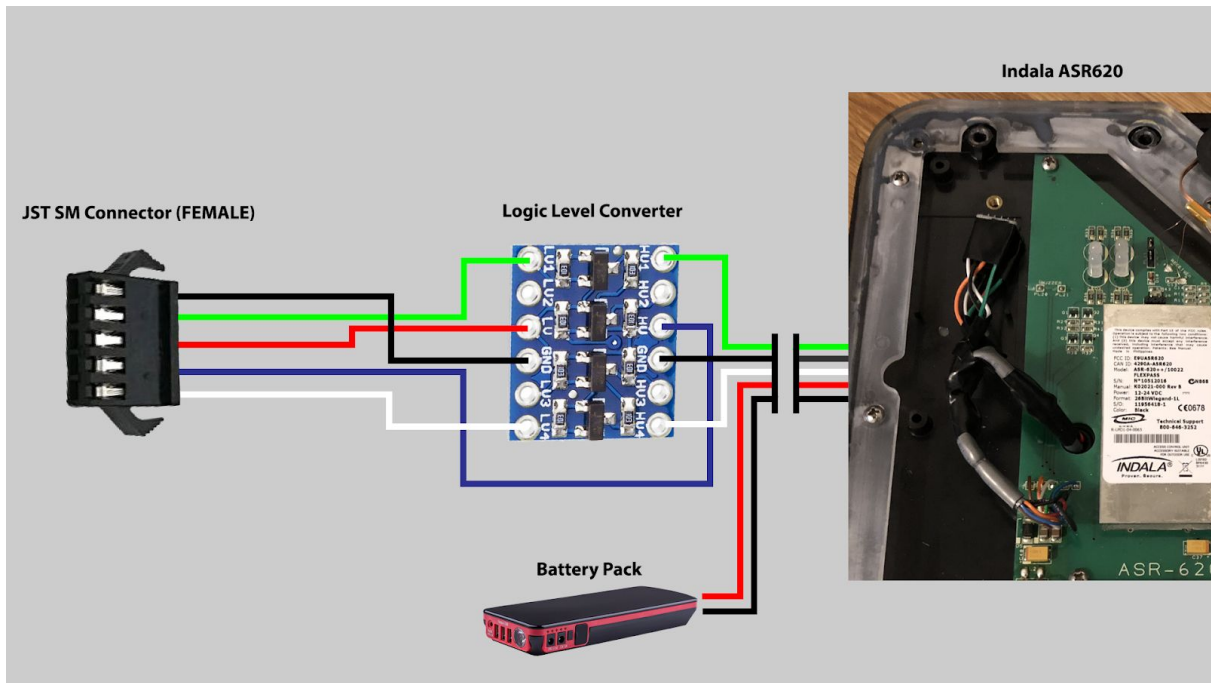
<b>Logic Level Converter</b>	<b>JST SM Connector</b>
LV	Red
LV GND	Black
LV 1	Green
LV 4	White
HV	Blue

3. Join Positive and Negative cables from the HID R90 to a DC connector/adaptor.

<b>HID R90</b>	<b>DC Connector/Adapter</b>
P2-1	Positive (+)
P1-5	Negative (-)

## HID Indala ASR620

The Indala ASR620 will have a wiring harness from factory that you can utilize, the shield wire is within the harness itself so you need to slice a portion of the harness to expose.



1. Splice and solder wires from the Indala ASR620 to the logic level converter according to the table below.

Indala ASR620	Logic Level Converter
Green (DATA 0)	HV 1
White (DATA 1)	HV 4
Shield	HV GND

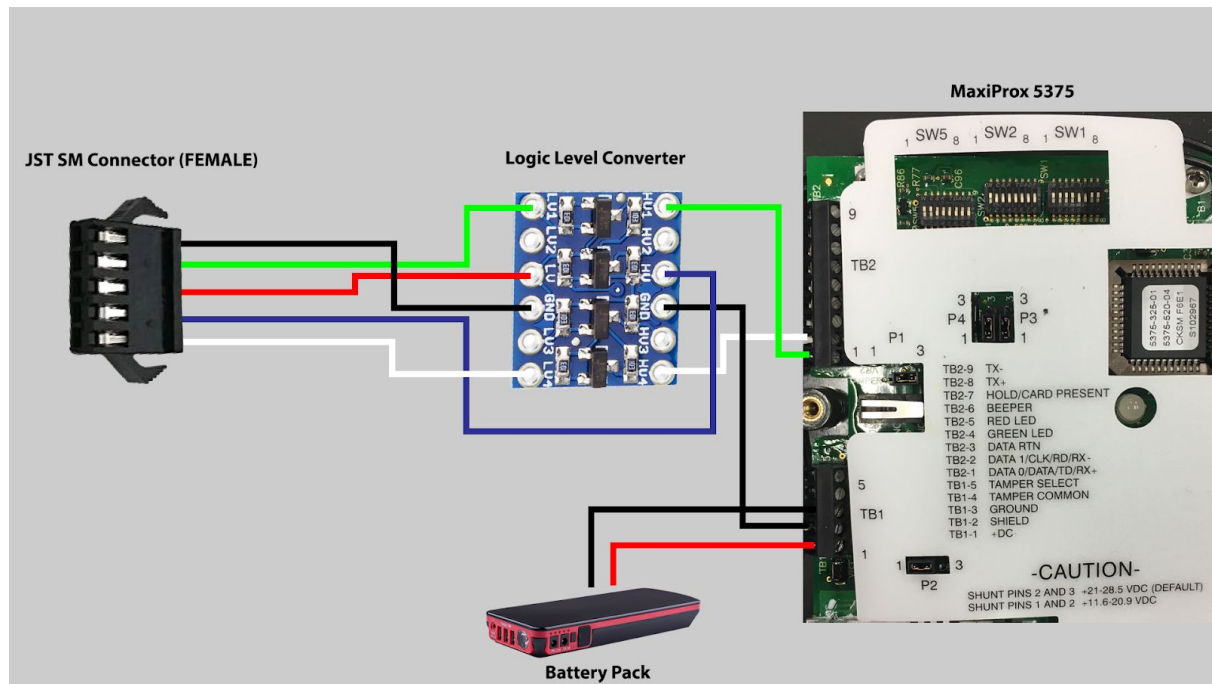
2. Solder female jumper wires from the logic level converter to a female JST SM connector according to the table below.

Logic Level Converter	JST SM Connector
LV	Red
LV GND	Black
LV 1	Green
LV 4	White
HV	Blue

3. Join Positive and Negative cables from the Indala ASR620 to a DC connector/adaptor.

<b>Indala ASR620</b>	<b>DC Connector/Adapter</b>
Red	Positive (+)
Black	Negative (-)

## HID MaxiProx 5375



1. Join wires from MaxiProx 5375 to the logic level converter according to the table below.

MaxiProx 5375	Logic Level Converter
TB2-1(DATA 0)	HV 1
TB2-2 (DATA 1)	HV 4
TB1-2 (SHIELD)	HV GND

2. Solder female jumper wires from the logic level converter to a female JST SM connector according to the table below.

Logic Level Converter	JST SM Connector
LV	Red
LV GND	Black

LV 1	Green
LV 4	White
HV	Blue

**3.** Join Positive and Negative cables from the MaxiProx 5375 to a DC connector/adaptor.

<b>MaxiProx 5375</b>	<b>DC Connector/Adapter</b>
TB1-1	Positive (+)
TB1-3	Negative (-)

## Controller

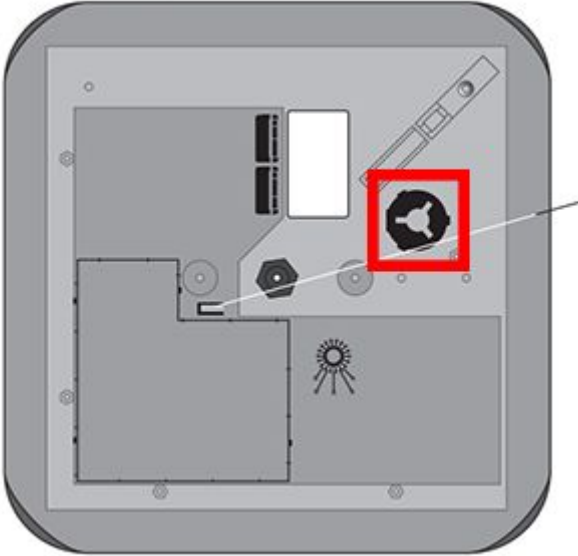
Hearing a loud beep from your backpack when you intercept a card is probably not good, to avoid this, I made a makeshift controller, to easily power on/off and switch between sound or vibration or both.

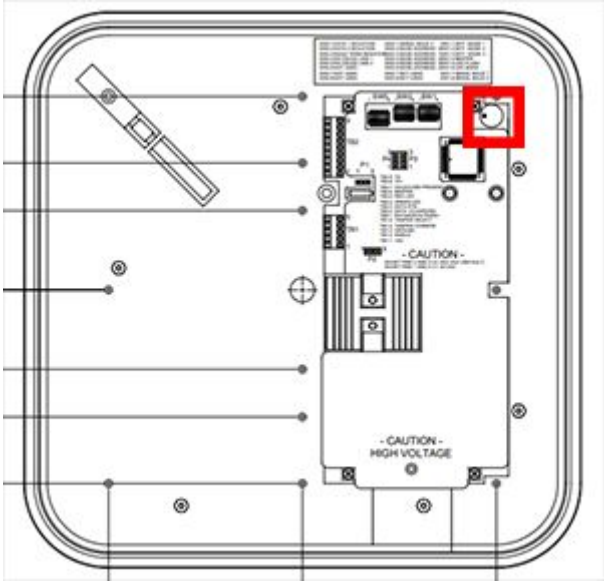
Each long range reader contains a sound buzzer either soldered or wired to the board, you can de-solder and replace this with extended wires to the controller.

Within the makeshift controller you can splice/solder a sound buzzer (reuse the readers), vibrating mini motor disc, switches and a voltage display.





Reader	Sound buzzer Location
HID iClass R90	

HID MaxiProx 5375	
HID Indala ASR-620	N/A - External

## Tutorial

This section will show you how to clone the intercepted cards from the long range readers using the Proxmark 3.

## iClass R90

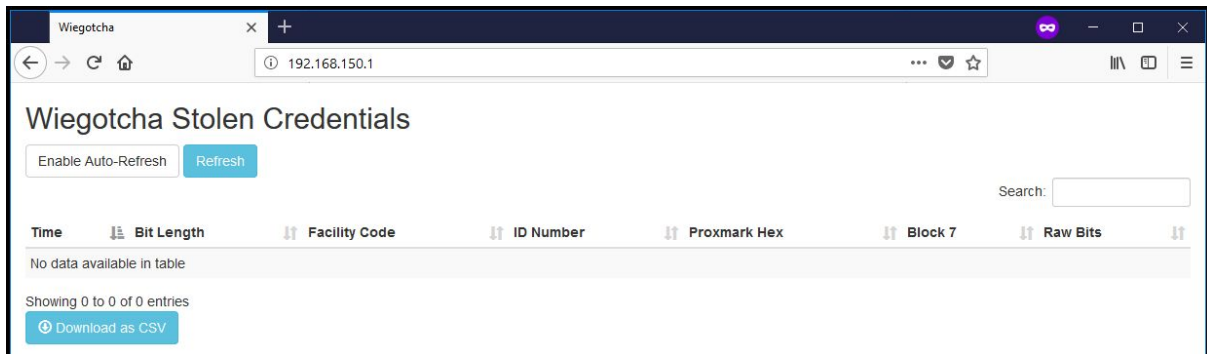
iClass legacy cards are encrypted using a master authentication key and TDES keys. The master authentication key allows you to read and write the encrypted blocks of the card however you will require the TDES keys to encrypt or decrypt each block.

You can find the master authentication key in my [Proxmark 3 Cheat Sheet](#) post & step 6 of this tutorial. The TDES keys are not publicly available, you will have to source them yourself using the [Heart of Darkness](#) paper.

The R90 will read the card, decrypt it and send the Wiegand data to Wiegotcha.

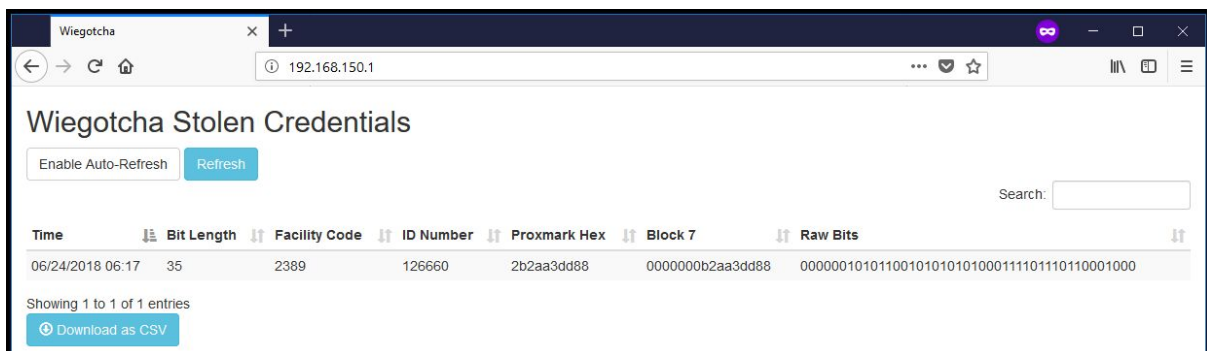
**1.** Assemble/Power on the components and connect to the RPi Access Point [Wiegotcha](#).

**2.** Navigate to <http://192.168.150.1> via browser.

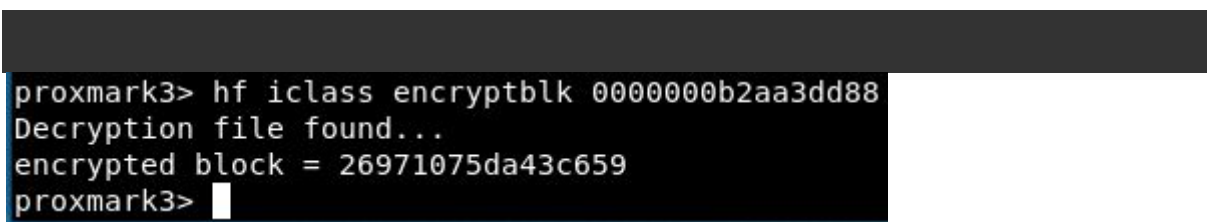


3. Place/Intercept a iClass Legacy card on the long range reader.

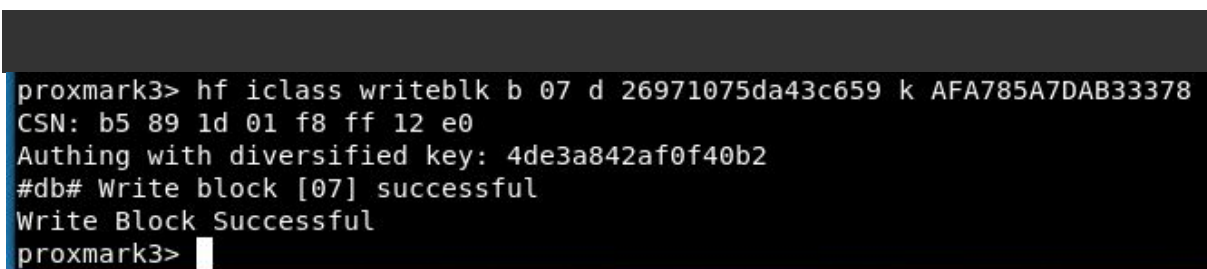
4. Copy the data from the **Block 7** column into clipboard.



5. Encrypt the **Block 7** data using the Proxmark 3.



6. Write the encrypted **Block 7** data to a writable iClass card.



7. Done! if it all worked correctly, your cloned card will have the same **Block 7** data as the original. You can confirm with the following:



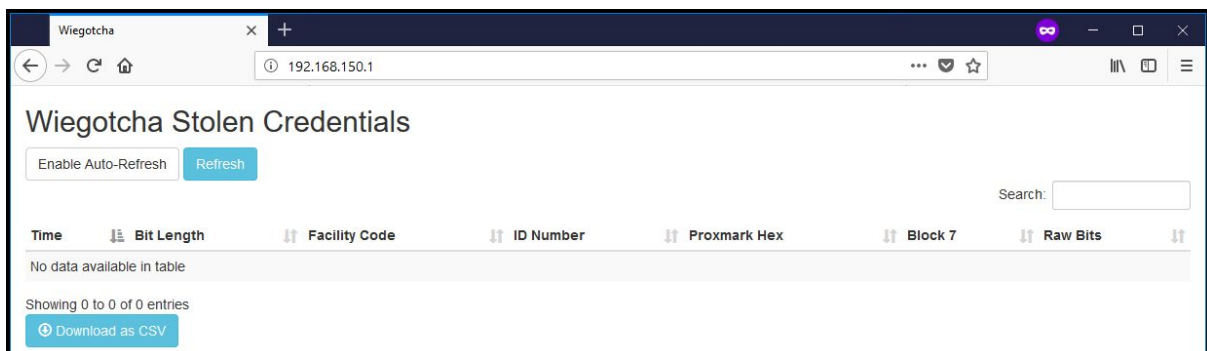
```
proxmark3> hf iclass dump k AFA785A7DAB33378
-----+-----+-----+-----+
CSN    |00| b5 89 1d 01 f8 ff 12 e0 |
-----+-----+-----+-----+
Block  |01| 12 ff ff ff 7f 1f ff 3c |
Block  |02| d8 ff ff ff ff ff ff ff |
Block  |03| 4d e3 a8 42 af 0f 40 b2 |
Block  |04| ff ff ff ff ff ff ff ff |
Block  |05| ff ff ff ff ff ff ff ff |
Block  |06| 03 03 03 03 00 03 e0 17 |
Block  |07| 26 97 10 75 da 43 c6 59 |
Block  |08| 2a d4 c8 21 1f 99 68 71 |
Block  |09| 2a d4 c8 21 1f 99 68 71 |
Block  |0A| ff ff ff ff ff ff ff ff |
Block  |0B| ff ff ff ff ff ff ff ff |
Block  |0C| ff ff ff ff ff ff ff ff |
Block  |0D| ff ff ff ff ff ff ff ff |
Block  |0E| ff ff ff ff ff ff ff ff |
Block  |0F| ff ff ff ff ff ff ff ff |
Block  |10| ff ff ff ff ff ff ff ff |
Block  |11| ff ff ff ff ff ff ff ff |
Block  |12| ff ff ff ff ff ff ff ff |
Block  |13| 22 20 19 e3 4e 7f 00 00 |
-----+-----+-----+-----+
Saving dump file - 19 blocks read
Saved data to 'iclass_tagdump-b5891d01f8ff12e0-4.bin'
proxmark3> 
```

## Indala ASR620

1. Assemble/Power on the components and connect to the RPi Access Point

Wiegotcha.

2. Navigate to <http://192.168.150.1> via browser.

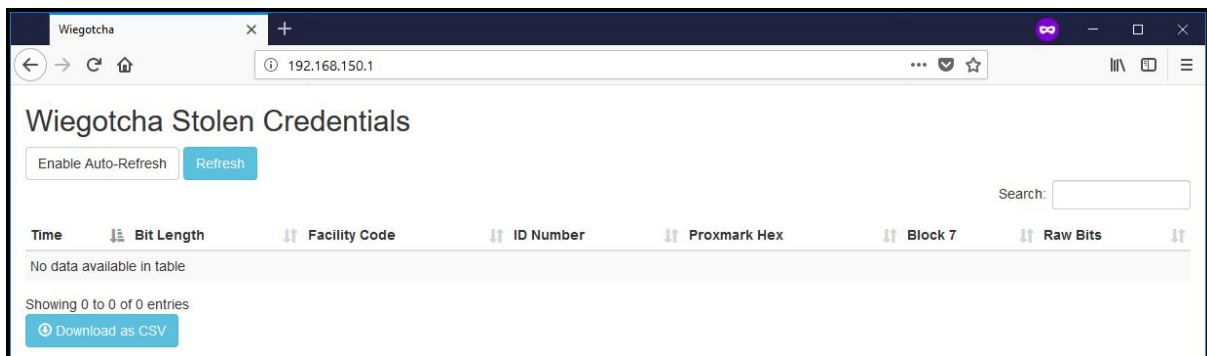


3. Place/Intercept a Indala card on the long range reader.



## MaxiProx 5375

1. Assemble/Power on the components and connect to the RPi Access Point **Wiegotcha**.
2. Navigate to <http://192.168.150.1> via browser.



3. Place/Intercept a ProxCard II card on the long range reader.
4. Copy the data from the **Proxmark Hex** column into clipboard.



5. Clone the **Proxmark Hex** data to a T5577 card using the Proxmark 3.



```
proxmark3> lf hid clone 2004060a73
Cloning tag with ID 2004060a73
#db# DONE!
proxmark3> █
```

7. Done! if it all worked correctly, your cloned card will have the same **Proxmark Hex, FC & SC** data as the original. You can confirm with the following:

```
proxmark3> lf search
NOTE: some demods output possible binary
      if it finds something that looks like a tag
False Positives ARE possible

Checking for known tags:

HID Prox TAG ID: 2004060a73 (1337) - Format Len: 26bit - FC: 3 - Card: 1337

Valid HID Prox ID Found!

Valid T55xx Chip Found
Try lf t55xx ... commands

proxmark3> █
```

## Components

Most of the components can be found cheaply on eBay or your local electronics store, the most expensive components are the long range readers and the Proxmark 3.

- Raspberry Pi 3
- Proxmark 3 RDV2
- 12v USB Power Bank
- HID iClass R90
- HID Indala ASR-620
- HID MaxiProx 5375
- Bi-Directional Logic Level Converter
- DS3231 RTC Real Time Clock
- Vibrating Mini Motor Disc
- 32GB MicroSD Card
- JST SM 5 Pin Connectors
- JST SM 4 Pin Connectors

## References

- [Official Proxmark 3 Repository](#)
- [Official Proxmark 3 Forums](#)
- [Mike Kelly's Blog](#)
- [Wiegotcha Github](#)
- [Tastic RFID Thief](#)