

APPLIED HACKING

Subscribe for updates on my latest research, access to unpublished videos, and learn how to keep yourself safer, online and off.

Subscribe

MagSpoof - "wireless" credit card/magstripe spoofer

- Allows you to store all of your credit cards and magstripes in one device
- Works on traditional magstripe readers **wirelessly** (no NFC/RFID required)
- Can **disable** Chip-and-PIN (code not included)
- Correctly **predicts** Amex credit card numbers + expirations from previous card number (code not included)
- Supports all three magnetic stripe tracks, and even supports Track 1+2 simultaneously
- Easy to build using Arduino or other common parts

By [@SamyKamkar](#)

MagSpoof is a device that can spoof/emulate any magnetic stripe or credit card. It can work "wirelessly", even on **standard magstripe/credit card readers**, by generating a strong electromagnetic field that emulates a traditional magnetic stripe card.

Note: MagSpoof does **not** enable you to use credit cards that you are not legally authorized to use. The Chip-and-PIN and Amex information is **not** implemented and using MagSpoof requires you to have/own the magstripes that you wish to emulate. Simply having a credit card number and expiration is not enough to perform transactions. MagSpoof **does** allow you to perform research in other areas of magstripes, microcontrollers, and electromagnetism, as well as learn about and create your own devices similar to other existing, commercial technologies such as [Samsung MST](#) and [Coin](#).



MagSpoof can be used as a traditional credit card and simply store all of your credit cards (and with modification, can technically disable chip requirements) in various impressive and exciting form factors, or can be used for security research in any area that would traditionally require a magstripe, such as readers for credit cards, drivers licenses, hotel room keys, automated parking lot tickets, etc.

Live demonstration and more details available in the video:

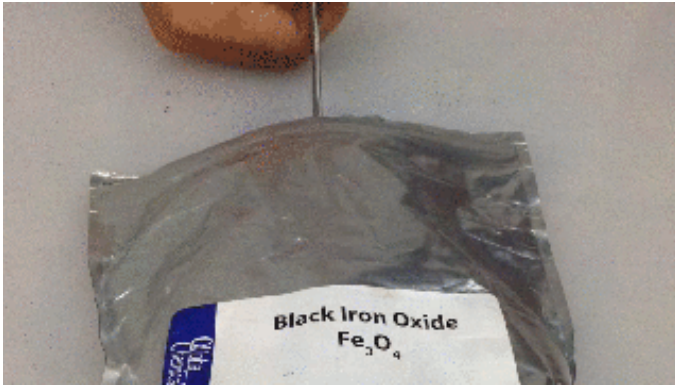


Point of Contact: [@SamyKamkar](https://twitter.com/SamyKamkar) // code@samy.pl // <http://samy.pl>

Released: November 24, 2015

Source code / schematic: <https://github.com/samyk/magspoof>

How magnetic stripes work

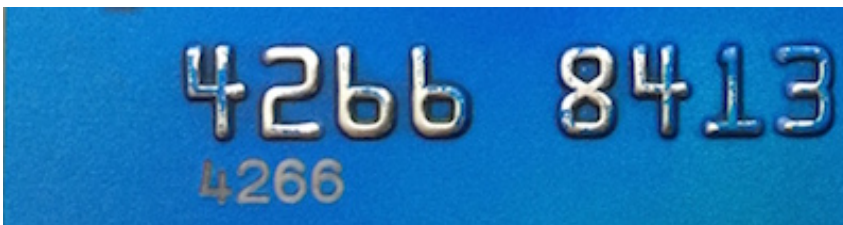


Magnetic stripes (magstrips) are in fact magnetic. What's so cool about magstrips is that while the magnetic strips inside are weak, they're still strong enough to attract small ferrous particles and wide enough that we can **fully extract all data from a magstripe or credit card with the naked eye**.

In the video, I actually dip my credit card into a bag of iron oxide. The magnetic elements in the magstripe attract the iron oxide and after pulling it out, you can actually see each bit in the tracks. Magnetic strips can have up to three tracks, however credit cards only have two tracks (Tracks 1 and 2). Track 2 has less bits per inch so it's easier to see. Let's take a look:



Front of card:



As you can see on my card, I've written out the bits above the strips. Two solid stripes is a 1 and a stripe followed by a space is a 0. First, let's flip the card 180 degrees (looking at it upside down), and read it as least significant bit first. We ignore all the initial 0's until the very first 1.

Track 2 is 5 bits per character, least significant bit first, and the 5th bit is an odd parity bit. By taking each 5 bits and reading the first 4 bits, we can now begin extracting the credit card number (and other data, such as expiration date, discretionary data, CVV/CVC/CSC/CAV depending on card type).

Note that the CVV is different than the CVV2. The CVV is stored on the magstripe, while the CVV2 is printed on a card. (CVV is specific to Visa, and each issuer has their own name such as CSC, CVC, etc.)



How MagSpoof Works

MagSpoof emulates a magnetic stripe by quickly changing the polarization of an electromagnet, producing a magnetic field similar to that of a normal magnetic stripe as if it's being swiped. What's incredible is that the magstripe reader requires no form of wireless receiver, NFC, or RFID -- MagSpoof works wirelessly, even with **standard magstripe readers**. The stronger the electromagnet, the further away you can use it (a few inches in its current iteration).

MagSpoof also uses inexpensive, off the shelf parts (described in the Hardware section), and can be built with almost nothing more than an Arduino, wire and a battery! I use a motor driver to provide a reasonable amount of power.

Normally electromagnets have an iron core, however we lose the core for the sake of space and portability. Also, while the iron core does make the electromagnet more efficient, we still produce more than enough power to work.

MagSpoof improves on new cards such as Coin. I'm a customer of Coin, and while I love their

app and the card, the card actually works a very small percentage of the time. After looking over Coin's FCC docs, I noticed they use two coils to produce a (very small) electromagnetic field, however it's severely deficient and the card works less than 50% of the time for me, sadly.

I found that by emulating a card with MagSpoof, if I send Track 1 one way, and then send Track 2 reversed, every card reader will assume I simply swiped a card back and forth, use the data from both tracks and my strong electromagnet, and properly read all of the data. This is extremely effective, uses only a single coil, and works for both tracks simultaneously. This also allows MagSpoof to work on Track 3.

Additionally, if you're using a Chip card with Coin, you **still** need to bring your actual credit card to dip, however because MagSpoof can disable Chip-and-PIN (see below), it does **not** require you to bring your card with you.

I've removed the Chip-and-PIN disabling functionality from MagSpoof.

Security Issues

One of the primary issues I've found is that some of the new forms of security (well, new in the US) are set in the "service code" portion of the magstripe, most specifically **Chip-and-PIN**.

The service code within a credit card magstripe defines several attributes of the card, including whether the card can dispense cash, where it can work (nationally, internationally), and most interestingly, whether the card has a built in IC (Chip) and if it has a pin (Chip-and-PIN / EMV).

If your card has a chip inside and you go to a retailer that supports Chip but swipe just your magstripe, the point of sale (PoS) system will ask you to dip your card/chip for additional security if it supports it.

However, the bits stating the card has Chip-and-PIN can be turned off from the magstripe.

This means if you take a card to a retailer that would normally request you to dip, you can actually get away with not dipping your chip at all while performing a successful transaction, **evading the security measures altogether**.

7.5 Service code assignments

Table 3 — Service code assignments

Value	Position 1		Position 2 Authorization processing	Position 3	
	Interchange	Technology		Allowed services	PIN requirements
0	—	—	Normal ^e	No restrictions	PIN required
1	International ^a	—	—	No restrictions	—
2	International ^a	Integrated circuit card ^b	By issuer ^f	Goods and services only	—
3	—	—	—	ATM only	PIN required
4	—	—	By issuer ^f unless explicit bilateral agreement applies	Cash only	—
5	National ^c	—	—	Goods and services only	PIN required
6	National ^c	Integrated circuit card ^b	—	No restrictions	Prompt for PIN if PED present ^g
7	Private ^d	—	—	Goods and services only	Prompt for PIN if PED present ^g
8	—	—	—	—	—
9	Test	—	—	—	—

American Express Card Number Prediction

What initially led me to investigate magnetic stripes was my Amex card. After losing a card and Amex quickly sending me a replacement, I noticed many of the digits were similar. I pulled up the numbers to several other Amex cards I had, and then compared against more than 20 other Amex cards and replacements and found **a global pattern that allows me to accurately predict American Express card numbers by knowing a full card number, even if already reported lost or stolen.**

This means if I were to obtain your Amex card and you called it in as lost or stolen, the moment you get a new card, I know your new credit card number.

I also know the new expiration date as the expiration date is fixed based on when the new card was requested, and you can determine if the new card has been requested by performing an auth on the existing card.

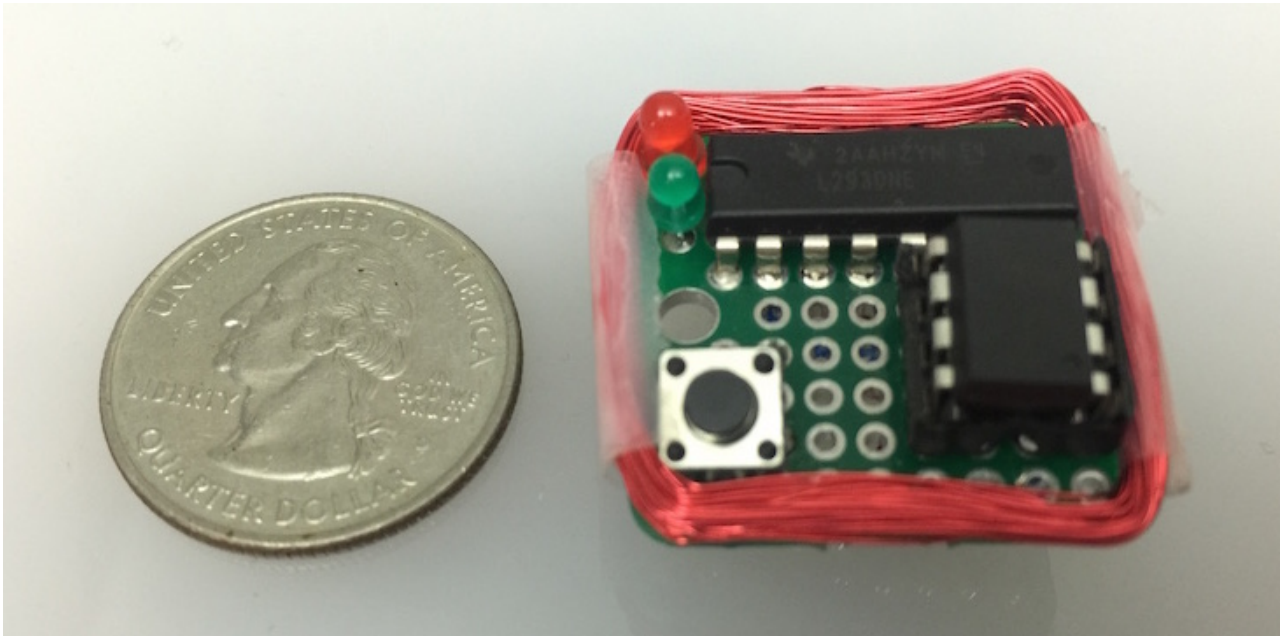
The CID (aka CVV2 on Visa) printed on the card is protected by a secret 3DES key that encrypts the PAN (Primary Account Number, aka credit card number), service code (see above), and expiration. The service code can be easily determined as most cards will contain the same service code.

I also determined that the CSC (essentially behaves like a CID or CVV2 on the magstripe) for a lost or stolen card **continues to work** for a newer, predicted card. An attacker would be able to use a stolen card's CSC with the predicted card number and expiration to make actual purchases.

To actually perform the transaction without arousing suspicion, an attacker would be able to use a magstripe writer (e.g., the well known MSR605), or a device like MagSpoof, to "load" the newly devised card information onto a card like Coin. Coin itself does not actually verify the CID (CVV2), thus allowing an attacker to load data, and then use the Coin card in person without knowing the CID and exploiting these various issues, as well as disabling Chip-and-PIN.

I notified American Express of the issue and I will not be releasing their algorithm.

Hardware



[Atmel ATtiny85](#) (microcontroller)

An Atmel ATtiny85 is the microcontroller to drive the entire system. It stores all of the magnetic stripe / credit card data. In a thinner, credit-card sized (0.8mm thick!) version, I use an [ATtiny10].

[L293D H-Bridge](#) (motor driver)

I use an L293D H-bridge to drive the electromagnet. The L293D is a motor driver, but motors are actually driven by the electromagnet(s) and magnets inside of them. Any standard driver should work here. Technically the L293D doesn't work down at 3.7V (voltage of the LiPo battery), but it works surprisingly well. In the credit-card size version, I suggest using a the TI DRV8835 or [TI DRV8833](#).

[24AWG Magnet Wire](#) (coil)

I use somewhere around ~24AWG magnet wire to act as the coil to produce the electromagnetic field. This piece of wire incredibly produces an electromagnetic field that makes the card reader believe a card is being swiped. Incredible. By rapidly controlling the polarization of this field, the magstripe reader believes the flipped bits of a real card are being swiped through the reader.

100mAh 3.7V LiPo battery (the powah)

A small 100mAh 3.7V lipo battery powers our contraption. For the credit card size version (not shown here), I use a battery from [PowerStream](#).

100uF Capacitor

Keep enough energy in this capacitor to provide the electromagnet with power when we need it, otherwise it will pull too much current and reset the microcontroller. This is the capacitor kit I use as it has all the standard values I'd need.

LED

To signal to us when we transmit information. I use this LED kit as it has a nice variety of LEDs.

100Ohm Resistor

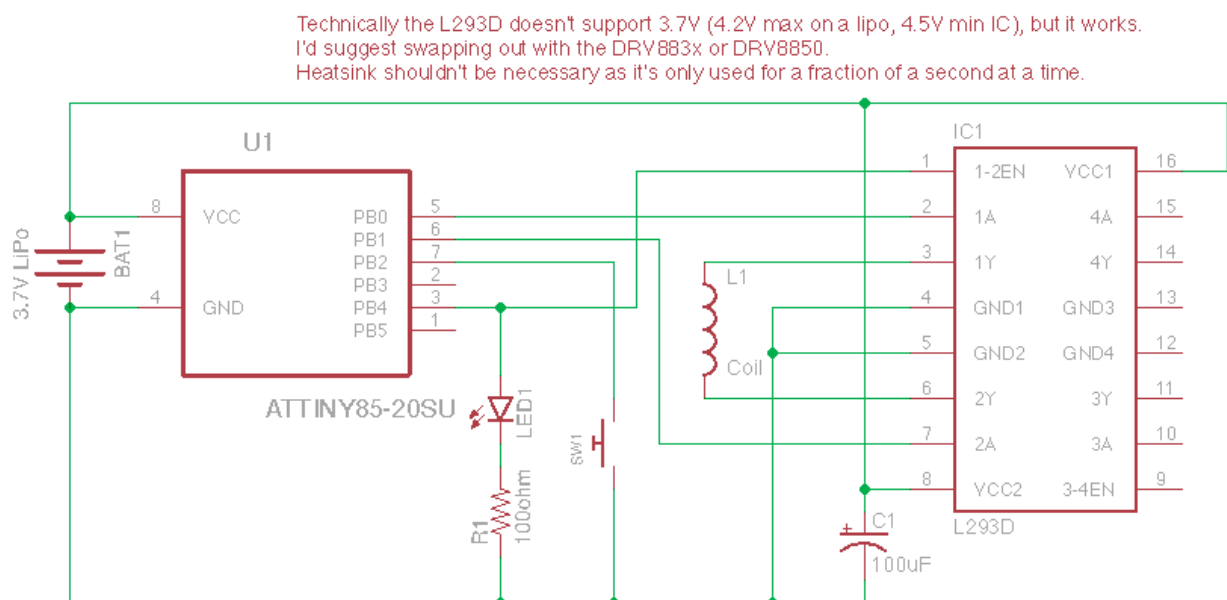
Don't burn out the LED.

Momentary Switch

Initiate the electromagnet.

Mini-Protoboard

For soldering everything together.



Firmware

MagSpoof

MagSpoof's source code and schematic can be obtained in entirety from my github:

<https://github.com/samyk/magspoof>

MagSpoof is compatible with the Arduino framework and can work on traditional Arduinos as well as ATtiny chips.

Resources

You can learn about magnetic stripes and credit cards from a few places, including:

- [My video on MagSpoof](#)
 - [ISO/IEC 7810](#)
 - [ISO/IEC 7811](#)
 - [ISO/IEC 7812](#)
 - [ISO/IEC 7813](#)
 - [ISO 8583](#)
 - [ISO/IEC 4909](#)
 - [MagTek Magnetic Stripe Standards](#) (pdf)
 - [Magnetic Stripe Card on Wikipedia](#)
 - [Amex's Web Services Plural Interface](#)
-

Contact

Point of Contact: [@SamyKamkar](#)

You can see more of my projects at <http://samy.pl> or contact me at code@samy.pl.
