IOWA STATE UNIVERSITY

RFID Spoofing Device

Cpr E 537 Project

Josh Jordan
Matt Smith

Abstract

RFID is a growing technology that is finding its way into a variety of applications. In addition to access cards, these devices are being placed in credit cards, passports, and product shipments. Because compromise of several of these items can be destructive to the well being of companies and individuals, it is necessary that these devices be secure against spoofing and data theft. Unfortunately, there has been a lot of debate about how vulnerable these devices are to spoofing and data compromise.

Our project was to research and attempt to spoof the RFID access cards used by the university using minimal resources. The university uses a variety of HID Proximity devices [http://www.hidglobal.com/technology.php?tech_cat=1&subcat_id=10] and the corresponding access cards. We had used several approaches to coerce the card for data and to determine its response. We will discuss the findings each of these approaches and how we carried out our experiments.

Research

Background

The first steps that we took were to research the devices, existing exploits, and the concepts behind RFID. There are two primary frequencies that RFID access tags operate at. These two frequencies are 125 kHz [ISO 18000-2A] and 13.5 MHz [ISO 14443]. One of the card readers commonly used by the University is the ThinLine II. This device operates at the 125 kHz frequency standard. This information can be found by reading the datasheet. The university uses other devices that are compatible with the 13.5 MHz standard but we thought that the cards used by students and faculty would comply with the 125 kHz standard. To get our hands dirty, we tried a little bit of "aggressive research" on the proximity cards. We examined some of the "inner workings" of the cards to determine if there was any helpful information that could be obtained from within the devices. This gave us some hints and these will be discussed in the next section.

To attempt to find more information, we started looking for evidence in US patents. To find this information, we visited the United States Patent Office online patent database at http://patft.uspto.gov. Doing an advanced search of "AN/HID AND ABST/RFID" (assignee name = HID and abstract = RFID), gave us a set of 4 results. Of these 4 patents, 2 seemed to be related to our project. The first is a method of signal stripping using a set of clamping diodes. The second is a method of detecting access cards using a beacon so that power can be conserved. In neither of patents is there a mention an encryption, challenge response, or secret key scheme. This led us to believe that the HID cards might be subject to a simple replay attack.

There are several people that have discussed the insecurities of RFID. One individual that has gotten a great deal of attention is Chris Paget. In 2007, at the Black Hat Conference, Paget was scheduled to demo the ability to hack RFID using less than

$20 in parts purchased from eBay[http://www.securityfocus.com/news/11444] [http://www.youtube.com/watch?v=fDimlEdeGjM&feature=related]. His presentation was met with resistance from HID Corporation. Paget intended to show schematics and source code to his RFID spoofer but HID threatened him with legal action. After a long discussion, the presentation was altered to remove the schematics and source code.

Boing Boing TV [tv.boingboing.net] posted a video in 2008 discussing a device that Pablos Holman created using a part he purchased on eBay for $8 [http://tv.boingboing.net/2008/03/19/how-to-hack-an-rfide.html]. This device can be used to read the RFID chip embedded in a number of the newer credit cards when placed in close proximity. The information that it retrieves contains the card holder's name and all relevant credit card information and displays it for easy viewing to the hacker. Again, this was done using an $8 reader that he was able to obtain on eBay without any issues. If the information on credit cards can be so easily and cheaply retrieved, they can be considered anything but secure.

Jonathan Westhues [http://cq.cx/index.pl] was able to clone a Verichip and Flexpass in his free time using basically no equipment. He discusses how the typical proximity card authentication system uses a 125 kHz carrier and that, using this knowledge, it is possible to spoof them. He began by creating an AM reader consisting of a peak detector, a passive low-pass filter, an AC couple, an active low-pass filter, and a comparator. This connected to an Arduino and was used to successfully duplicate a Flexpass. Because we initially believed that the HID system would probably not be identical, we decided that we wanted to start from the beginning and conduct all research and experimentation ourselves.

Aggressive Research

We wanted to get a closer look at the inner workings of the access cards. The device consisted of a single antenna and a small circuit. This indicated to us that the card uses a single frequency to communicate. This was important because we thought that the card might possibly get power on one frequency and transmit on another. An example of what we found can be seen below in Figure 1.

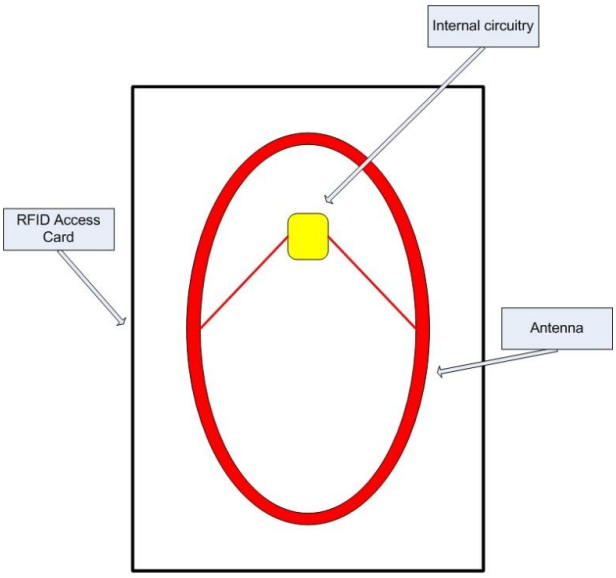

Figure 1 Example of RFID Card Innards

Antenna Design

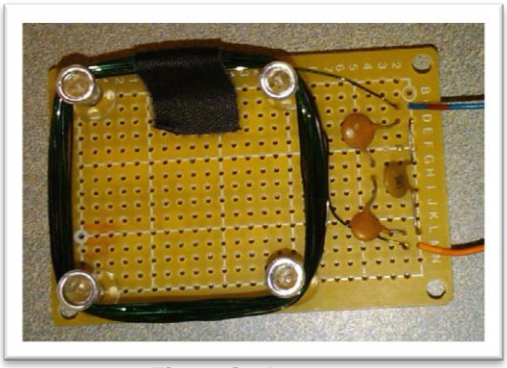

Figure 2 - Antenna

To continue our research, we started by fabricating an antenna designed to work at our expected frequency, 125 kHz. One of the antennas can be found in Figure 2. To obtain optimum data, antennas need to be designed around a center frequency. There are a variety of antennas that can be used depending on the characteristics. The first antenna we discussed was the dipole antenna. We found that these devices are usually built at a quarter wavelength. Unfortunately, this requires a large antenna length. This is shown in the following equation.

$$\frac{C}{f} = \frac{3 * 10^8\ m/s}{125000\ 1/s} * \frac{1}{4} = 600\ m$$

Because of this, we decided that a loop antenna would provide better characteristics for our purposes. A loop antenna usually consists of an inductor and capacitor that is resonant at the design frequency. This is called a tank and is governed by the following equation.

$$\omega = 2 * \pi * f = \frac{1}{\sqrt{L * C}}$$

In the above, L is the inductance, C is the capacitance, and f is the design frequency. We began by selecting a capacitance of .1 μF and designed an inductor for 1.62 μH. The design of the inductor was done using the following equation.

$$L = N^2 * \frac{2 * W * u_0 u_r}{\pi} * \left[\ln\left(\frac{w}{a}\right) - .77401\right]$$

Where N is the number of turns, W the side length, and a is the wire radius. After creating the antenna, we had to tune the device. This is because our equation does not account for small W, which was about 5 cm. To tune, the parallel capacitance was lowered and the number of turns in the inductor were lessened.

Signal Capture

With the antenna built, we began by taking one of the oscilloscopes in Coover's labs down to the card reader on the west side of the building. With the antenna hooked directly to the input of the oscilloscope, we analyzed the transmission from the reader in an attempt to determine the beacon that the reader used to begin to speak with the card. Initially, we found that the signal was a sine wave of the expected frequency with a varying peak to peak voltage. This looked like a series of sine waves laid on top of each other. The image looked similar to Figure 3.

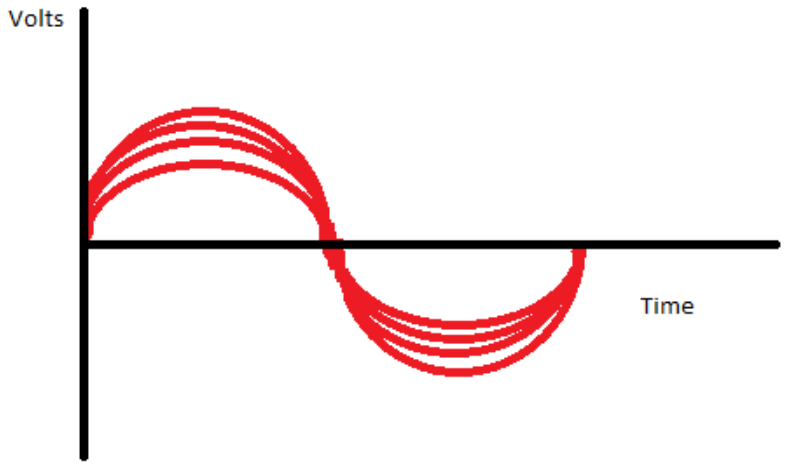

Figure 3 – Sample output

Upon closer inspection, we found that the beacon signal was actually slightly more complex. A more accurate representation of the signal can be seen in Figure 4 below. This agrees with the information about the beacon that was discovered in the patent information. In this figure, we see that there are two sets types of signals present. Initially, we believed that this was important but found that smaller card readers do not output this signal. Figure 5 is an image of the beacon signal at a smaller card reader.

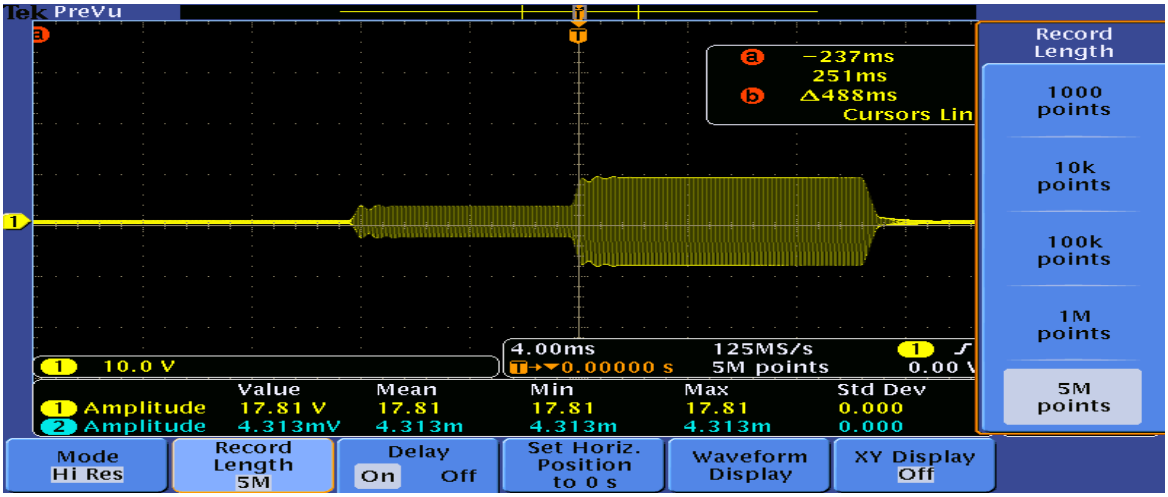

Figure 4 Beacon of Large Reader

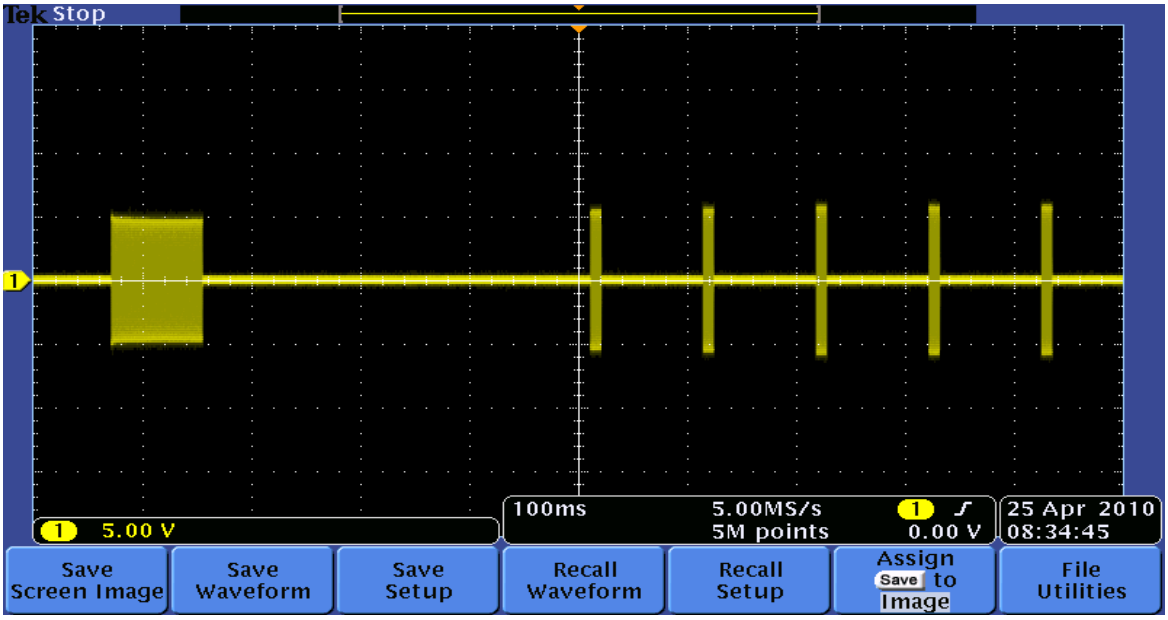

Figure 5 Beacon of Small Card Reader

In this version, the latter half is the beacon signal. You will notice that there is no small voltage value present before the 10 Vpp signal. Indicating that this small signal is not required to activate the card (which is shown in the first half).

Because we were unable to determine any information directly from the beacon, we started attempting to draw information from the excited value of the card. There are a variety of modulation schema that could be used to encode the RFID signal so we started researching methods of decoding. At the suggestion of our professor, we tried to to decode using both amplitude and translational modulation. This seemed like a likely posibility because RFID devices are cheap and this is a cheap encoding schema. We found that the most common method of AM

demodulation is a device called an envelope detector. An example of this can be seen below in figure 6.

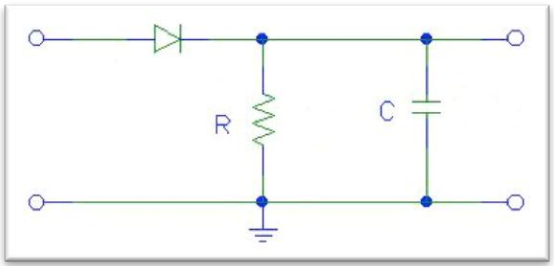

Figure 6 Envelope Detector Circuit

To begin, however, we inserted a large capacitor into the envelope detector and measured the reader's output again. Unfortunately, we found that there is a region that must be followed to ensure that the AM decay is not too large but also not too small. One common configuration that we used was R = 2.2k and C = 10 nF. This ensured that we could capture a modulated signal at fm < 1/(2 *π * R * C) or 7.2 kHz. We picked this frequency because we knew that the modulated signal must be much lower than the carrier for AM to work. These attempts to demodulate the card's transmitted value were ineffective. We found that variation in the output of the signal was less than 1 Volt. We thought this is not what should happen in an AM signal. Additionally, there was no indication of a frequency/phase shift keying because the demodulated signal held a very consistent high value and the direct measurement of the responses had no change in phase or frequency. The direct measurement of the signal (no demodulation) can be seen below in Figure 7.

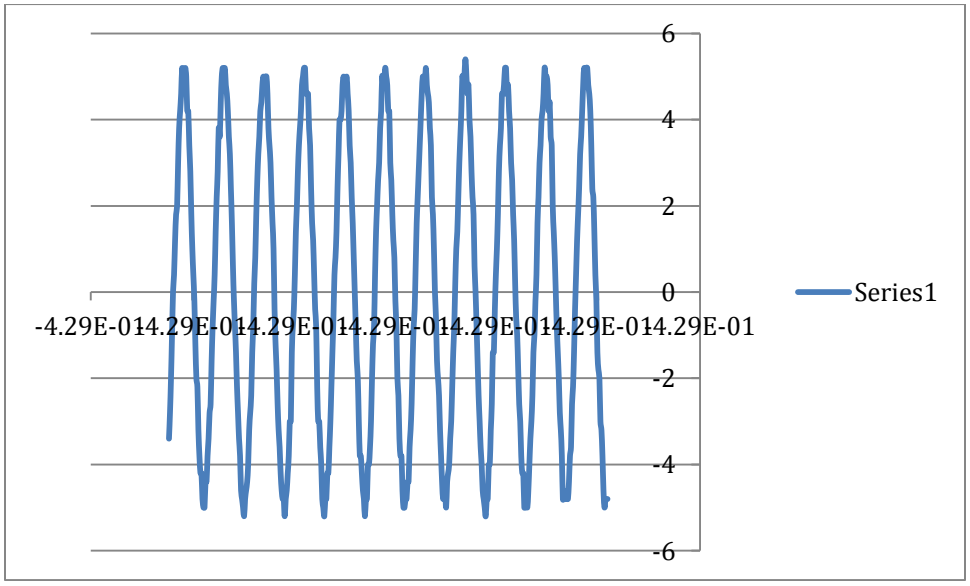

Figure 7 Zoomed Image of Device Response

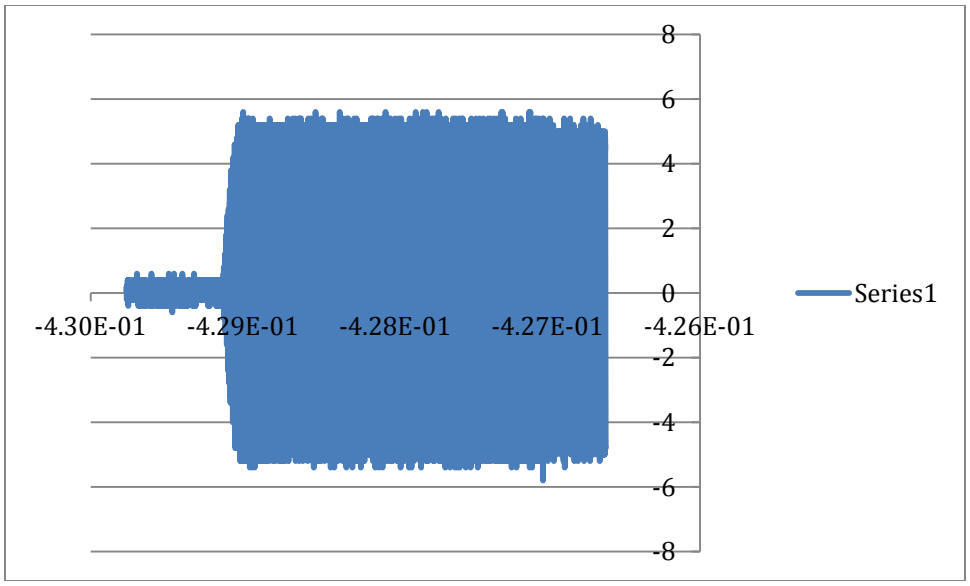

Figure 8 Enlarged Device Response

We went back to the drawing board thinking maybe there was something we missed in the signal. The capture of the beacon showed that, when active, sent out 1 volt for 10ms, then 5 volts for another 10 ms, before returning to 0. This process repeats infinitely until a card reacts to the beacon. In order to determine what the card was doing when reacting to the beacon, we worked to program a microchip that would duplicate the functionality of this beacon. We thought that doing this would allow us to read the card's signal then replay it at a slower rate. We could then replay this to signal express and export as a CSV or Excel document. This slower rate would make it easier for us to determine the card's response to a beacon and also allow us to apply some signal analysis in Matlab. It was not until later, after reading the manual for information about FFT, we found that the scope could export CSV data directly.

After setting up an ADC and fast enough clock to remove any aliasing, we realized that our plan to capture a very large signal (several seconds at 16 kHz sampling frequency) would be very difficult. The Microchip 18F4520 only contains 4 kb of memory. Capturing .25 seconds would have been sufficient to but we needed to first find the trigger to our card.

Our next idea was to read the demodulated signal using an audio program. We believed that this was acceptable substitute for our Microchip given that we were already limited to 8 kHz from the demodulator circuit. The program that we selected is called Audacity. We inserted an audio plug over the envelope detector so that it would output the value and plugged it into a computer's microphone input. Not only did this allow us to record and store values for an extended length, it enabled us to be more mobile.

With the antenna and envelope detector connected to a computer running Audacity, we went back to the reader on the west side of Coover. Our first run resulted in the waveform pictured in Figure 9. This shows the beacon as well, but exhibited some rather strange behavior given the components in the system.

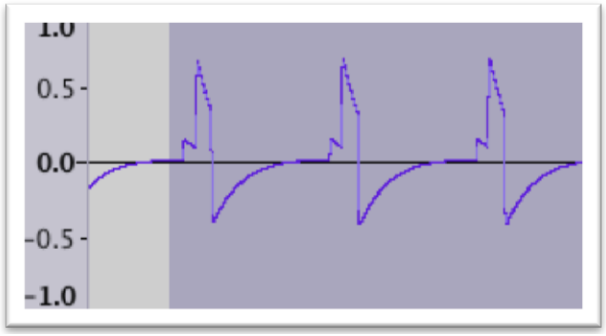

Figure 9 Audacity Measurement

As we mentioned, there is a diode in the envelope detector. This means that the voltage should be non-negative at all times. Unfortunately, we see above that there is a negative voltage. To check if this was problem with our circuit or Audacity, we checked the beacon against an oscilloscope measurement with the same envelope detector. This yielded a figure similar to Figure 8 except with all positive voltages and minor fluctuations in the high value. We could not determine the cause of these problems except that either the audio port on our computers or Audacity was introducing these decays and discharges. Our closest explanation is the presence of blocking capacitors in the microphone port. As a DC would store up energy on our side, a negative charge would build on the computer's side of the capacitor. Once the signal disappeared, a flow of current would equalize this negative charge and create a negative current.

Modulation Attempts

Next, we attempted to create a number of AM demodulation circuits. While most of them were pretty much non-functional in this case and either did nothing at all or completely distorted the waveform, we did eventually found an amplifier system from a lab experiment at another school. This contained no values but had a nice example of the AM modulated output. To do this, we needed the base current in both transistors to be equal at DC bias. Unfortunately, we were unable to recall the math required to force the bias to the correct value. The schematic can be seen below in Figure 10.

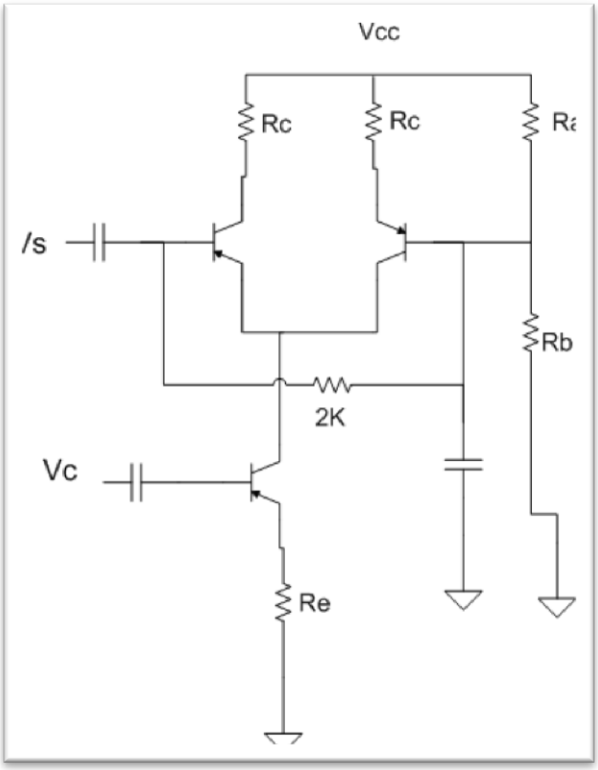

Figure 10 Amplitude Modulation Circuit

Audacity Analysis

Not letting our failures spend too much of our time, we decided to try our new approach for reading cards and examining the output. We hoped this would give a clearer picture of what was going on in the card reader even though there were problems with the beacon detection. We also thought that we could use a computer to read values then replay them using a Python script or C program and forcing them through an AM modulator. To test against a device we knew worked, we took the circuit and computer up to Frederiksen Court where one of us had access. We began by taking a couple of readings of the beacon to verify that it was same to the one at Coover and then triggered the reader with the card. The resulting waveform contained a rather large series of bits encoded using a variation of amplitude modulation, we believe (Figures 11 & 12). This response coincided precisely with the insertion of the card into the system multiple times and contained a signal too perfectly modulated to have just been noise.

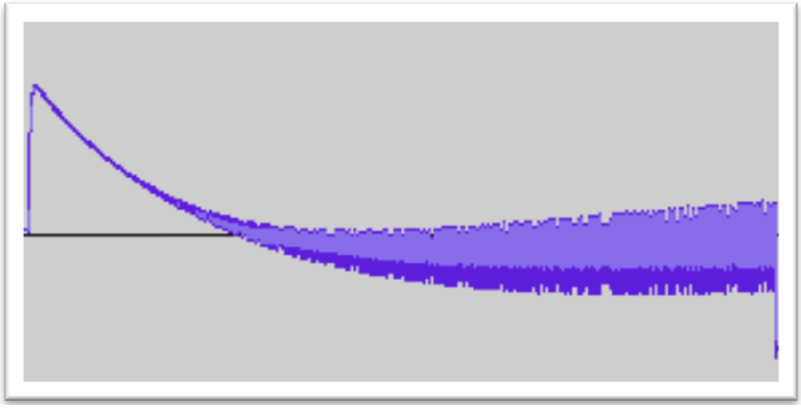

Figure 11 Demodulated RFID Response

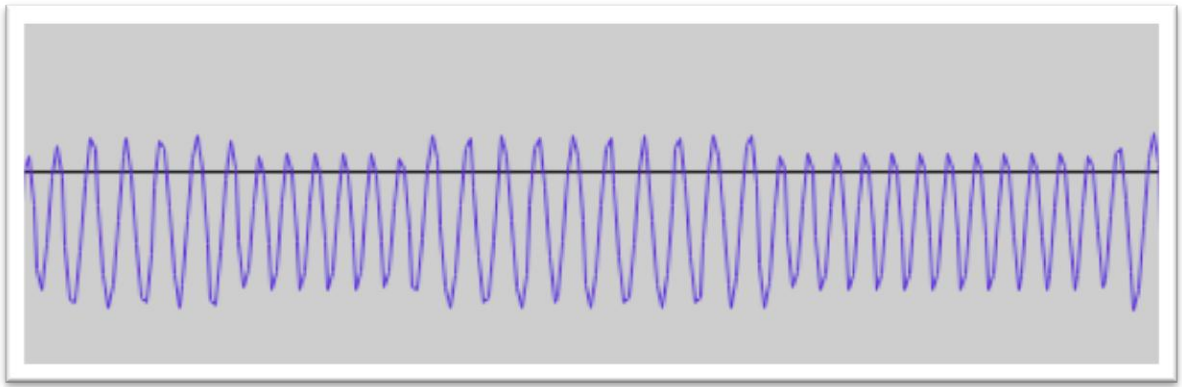

Figure 12 Zoomed Demodulated RFID Response

We began to look into different methods of retrieving the bit sequence that had been encoded. The first and most obvious method we tried was to manually read the signal and determine the sequence by hand. We encountered a number of issues with this, however. The first, and most troublesome, was that when the signal initially began to transmit, the voltage level on it was too low to be able to determine the amplitude difference between them. While these results seemed promising, we decided that these were inconclusive. Because of the increase of signal strength in the response (Figure 11) we were unable to determine a sharp starting point. Additionally, the difference between a high and low wave was somewhat arbitrary. This meant that we had to make an educated guess at the number of bits in each sequence. As can be see in Figure 12, this isn't a large problem most of the time, but when two, three, or four bits of the same value are next to each other, it becomes hard to determine how many there are. We even tried adding an amplifier with a low pass filter. Unfortunately, this did not make the signals clearer.

We used the smallest time period that we saw as the clock. From that we looked at the changes in the signal output and assumed that a sudden jump in voltage was a 1 and sudden drop was a 0. You can see a portion of our interpretation of the cards output in Figure 13. The reason this is only a portion is because the data is difficult to read from the small variations at the beginning of the signal. Instead, we began reading the data from the end of the measured signal.

CAAAB4AAAAD52B34CA	3555665555A55556A959A658D5559955569555
8AAAAD5158D58D32C1	D5558C5555A55558A959A658D5559955569555
A95554AAAAAA566996	3555665555A55556A959A658D5559955569555

Figure 13 Partial Measurement of Signal

As you can see, these results are far from conclusive but do seem to indicate that the HID system has a very large portion that is reused. In fact, there are 19 bytes that line up almost exactly. We believe that this is too large to simply be header information. As for the remaining portion, we believe that there may be some kind of additional information sent with the device. This could be in the form of a time stamp or possibly a nonce. This would indicate that the device is not susceptible to a simple replay attack.

Matlab Simulation

Because results were somewhat inconclusive, we decide to take another shot at interpreting the values. This time, we ended up discovering how to dump the waveforms out to a CSV file. We then entered these values into Matlab so that we could use its AM demodulation function.

We began by just importing the data into MATLAB and plotting the waveform using the functions inside of the program. This did successfully reproduce the waveform (Figures 14 & 15).

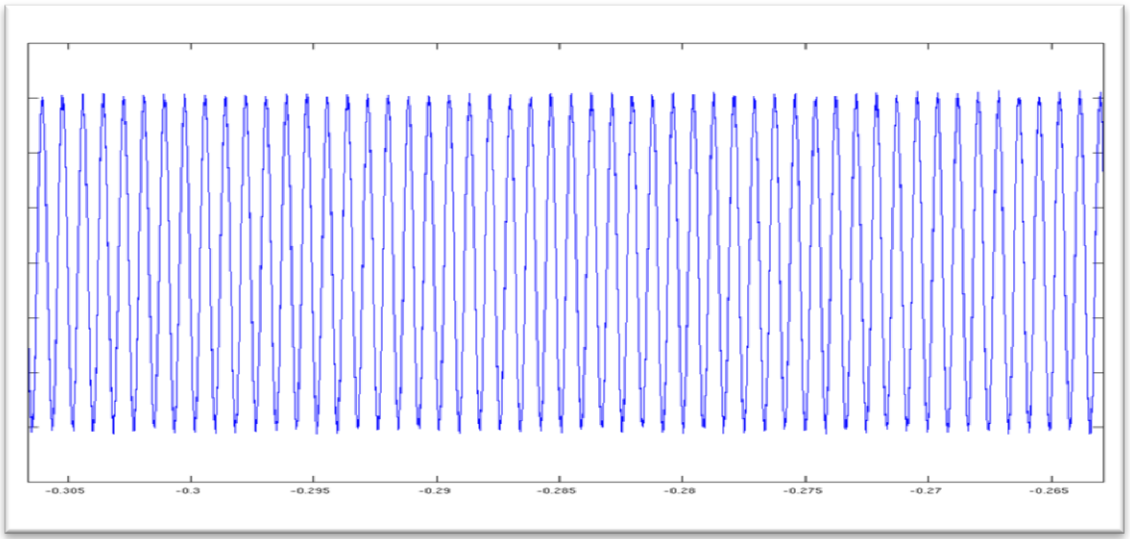

Figure 14 Imported CSV Waveform

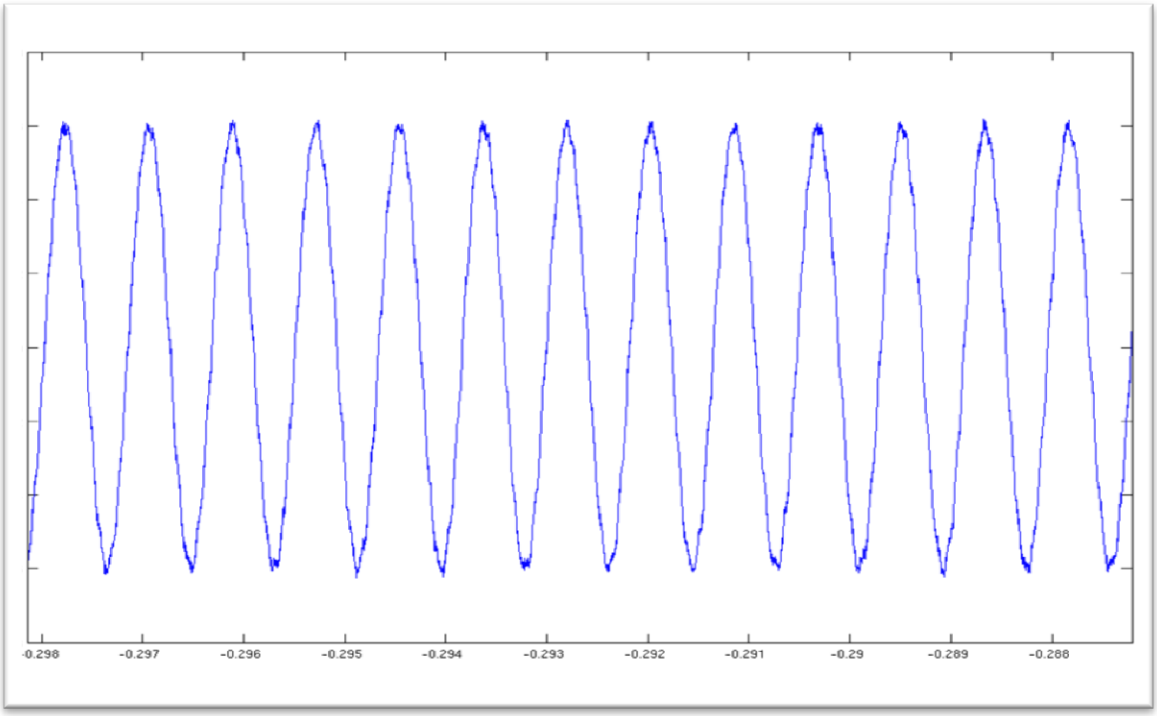

Figure 15 Signal After AM Demodulation

While it is hard to see in the figures, the signal was still embedded onto the 125 kHz carrier frequency. This makes sense as the signal was captured straight from the device and not run through the envelope detector as before. We then attempted to demodulate using MATLAB's built in function, but it failed to alter the waveform in any noticeable way. The commands used are listed below.

```
plot(TIME, V) # Used to view the original waveform
V2 = amdemod(V, 125000, 5000000) # Demodulate the waveform
plot(TIME, V2) # View the demodulated waveform
```

Verichip Hack

Because there were many reports of the HID system being breakable, we decided to take another look at Jonathan Westhues's attempts [http://cq.cx/prox.pl] and the HID patent. The patent mentions a diode clamping mechanism. In Westhues's system, we see that there is our envelope detector, followed by a band pass filter, an amplifier, and comparator-like circuit. The reason this was interesting was because of the comparator. We believed that the comparator could be acting as the diode clamping circuit that was talked about in the HID patent. Westhues's circuit can be seen below, in Figure 16.

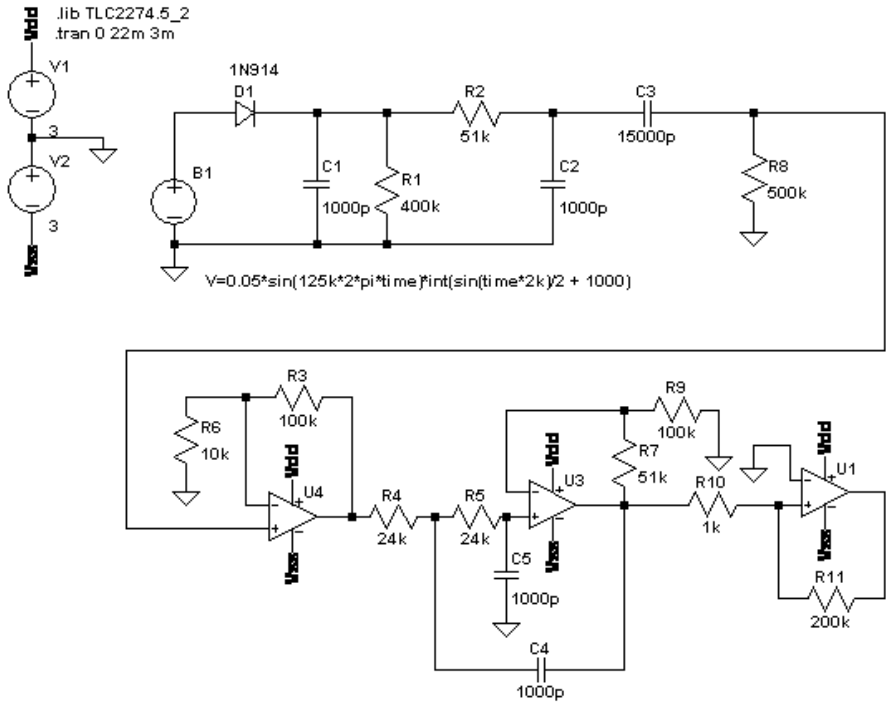

Figure 16 Verichip RFID Hack

We constructed this circuit and made one more attempt at making something from the reader's response to the card. We were unable to draw any meaning from the captured waveforms. They can be seen in Figures 17 & 18.

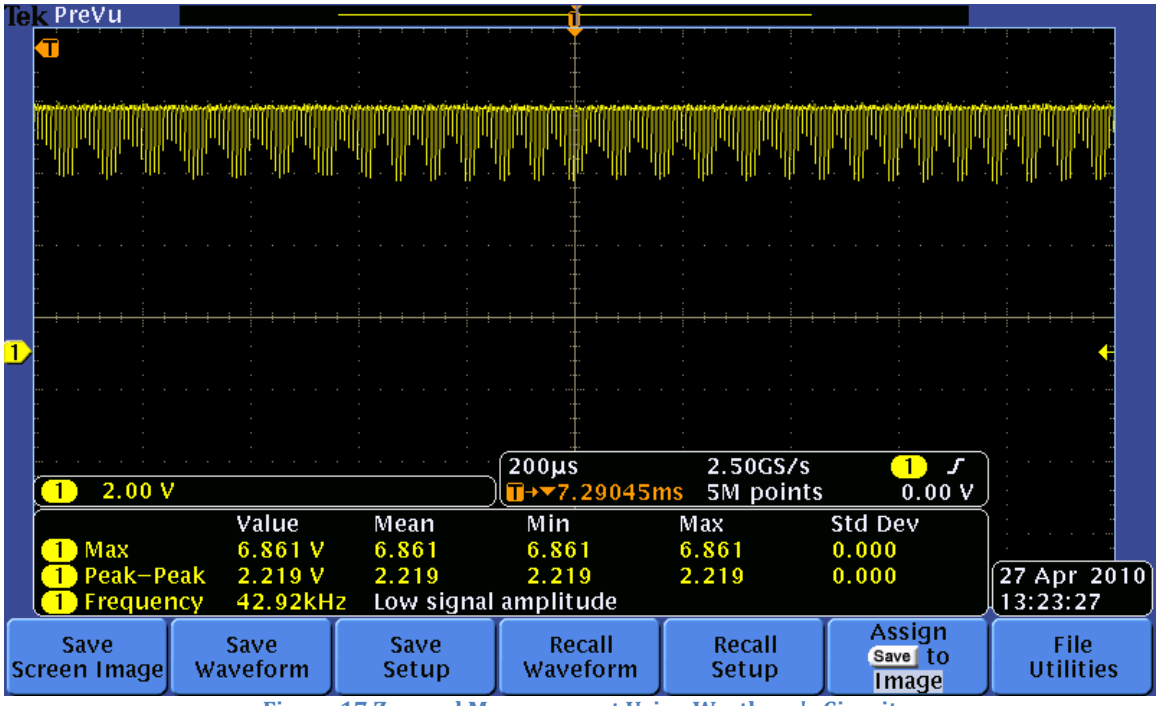

Figure 17 Zoomed Measurement Using Westhues's Circuit

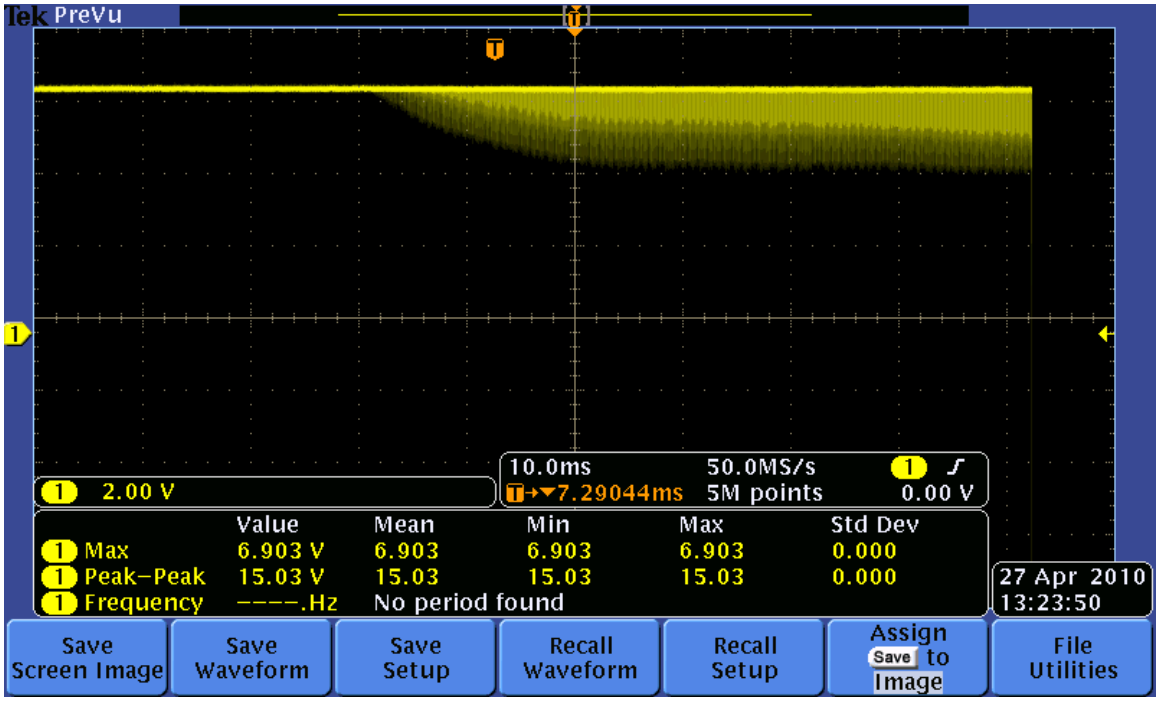

Figure 18 Measurement Using Westhues's Circuit

Conclusion

In our research, we found several ways of gathering RFID data. We also dealt with a great deal of electronics to try and obtain that data. Our results seem to indicate that the HID RFID system is not completely susceptible to a simply replay attack. However, they do seem to indicate that there is a fairly rudimentary scheme for protecting the system. Our belief is that more data and a better understanding of the HID patents would yield a good idea of how to exploit this system. Because of this, we recommend that groups that use, or believe that they will use, the HID system in a secure setting, should think about employing a secondary form of authentication. This could include a set of biometrics or key code access. There are several groups that offer these kinds of systems, including HID.