# Bruce Schneier

## Schneier on Security

A weblog covering security and security technology.

« Bioterrorism | Main | Movie Theaters Want to Jam Cell Phones »

March 16, 2006

RFID Chips and Viruses

Of course RFID chips can carry viruses. They're just little computers.

More info here. The coverage is more than a tad sensationalist, though.

EDITED TO ADD (3/16): I thought the attack vector was interesting: a Trojan RFID attacks the central database, rather than attacking other RFID chips directly. Metaphorically, it's a lot closer to biological viruses, because it actually requires the more powerful host being subverted, and there's no way an infected tag could propagate directly to another tag.

Posted on March 16, 2006 at 06:55 AM

### Trackback Pings

TrackBack URL for this entry:
http://www.schneier.com/cgi-bin/mt/mt-tb.cgi/775

### Comments

I guess by now that it's no little supprise that any micro based system can carry viruse or other malware code.

What is still surprising is the "technology first" rush about getting a product to market place without taking due precaution to prevent malware getting into a product with a programable micro built in...

The thought occurs about SmartCard / RFID Pasports and other ID card systems, after all whats to stop a virus getting in that makes everybody "Joe Doe" of "1600 Penselvania Ave"...

Posted by: Clive Robinson at March 16, 2006 07:19 AM

---

I am not aware of anything that will stop the destruction or modification of information on the RFID chip in a passport.

The obvious fall back method of having an immigration officer perform a visual inspection of the document has been working for decades.

Posted by: MathFox at March 16, 2006 07:55 AM

---

Why would an RFID chip need to be writable? It should be programmed once, and then only output its information thereafter. By making it writable (or having the ability to be changed), you make it worthless for its intended purpose.

Posted by: kashmarek at March 16, 2006 08:14 AM

---

I think that "little computers" is a red herring. An RFID tag can carry a virus in the same way that a floppy disk can carry a virus. Some tags have persistent read/write storage, and this can carry data. Poorly written apps (NOT running on the RFID tag itself) could process this data incorrectly in a way that causes that app to execute it as code.

Of course, a paper 2D bar code could trigger the same virus, but this is too retro for newsworthy kneejerking.

Posted by: Dave Engberg at March 16, 2006 08:18 AM

---

"Of course, a paper 2D bar code could trigger the same virus, [...]"

RFID does make it easier; I can build an RFID transmitter with a high gain antenna and inject the virus from a distance. You can't do that with bar code. The proposed British RFID number plate system comes to mind as a prime target.

Posted by: stacy at March 16, 2006 08:28 AM

---

@Clive:

Why do you find that surprising? We're in a free-enterprise system, where the main goal (some would say *only* goal) is to make money. You can't make money if someone else gets their product out first.

The dominance of marketing over technology has proven that point again and again.

Posted by: Jim Hyslop at March 16, 2006 08:38 AM

---

@Stacy

Actually the Transport For London Oyster card strikes me as a realy good target. As far as I am aware it is still the worlds largest MIFare card system in actual deployment (uses the 13.56 ISM band RFID's speced by SchlumbergerSema etc).

As it is used for buying and using transport tickets the RFID has to be modifiable at all times (otherwise the off line hand held units would not work) TfL are also pushing it for Newsagent and other Kiosk type sales.

Now if you put a time delayed virus in your card and went through every major London Railway Station in one day (there's around 14 you would need to visit) then just about every commuter with an Oyster card would be infected over the next day or to...

It could make life in London hell for the 6-7 million journes made each day

Posted by: Clive Robinson at March 16, 2006 08:58 AM

---

@Jim Hyslop

For exactly the reason given by you.

This behaviour is so endemic and has been repeatedly questioned in the ordinary press as criminally stupid, that I cannot see any company standing up in a court and saying they where not aware of the problem. So they are not performing due diligence, and their product is likewise not fit to market.

Only problem is how do you get them into court and get a good result (for the consumer).

Maybe Insurance companies should refuse to give them product liability cover which should make quite a few companies think twice.

Posted by: Clive Robinson at March 16, 2006 09:05 AM

---

Geoff Marshall holds the official world record for shortest time to visit every station on the Tube network -18 hours and 35 minutes. Just infect his RFID card :-)

Posted by: Kees Huyser at March 16, 2006 09:08 AM

---

@kashmarek:

I concur. While it may be convenient to alter the info on an RFID, it would be better to just replace it. Class 0 (Read Only) is for pre-programmed tags, and Rewritable Class 0 (Class 0+) is for those who want to program the tags themselves (from http://www.impinj.com/page.cfm?ID=rewritableClass0). Kinda reminds me of all the insanity surrounding Y2k and embedded systems.

Posted by: 1915bond at March 16, 2006 09:14 AM

---

If I read the article correctly, they wrote specific middleware that is vulnerable to SQL Injection attacks, and then put simple ASCII text on the tags, and showed that their badly-designed middleware is susceptible to these attacks. Then they proceeded with the logical leap that this means that RFID systems in general can carry viruses. To me that is setting up a straw-man argument, especially considering the fact that they are not able to point to a single real-world implementation that would be vulnerable to this attack.

This is exactly like saying that "URLs can carry viruses", since there is a remote possibility that someone writes bad middleware that just treats the query string as something to be directly inserted into an SQL sentence. There is nothing RFID-specific in this particular attack.

Of course there are stupid, bad and dangerous implementations out there. But saying that it's RFID's fault is exaggeration, and just riding on the "RFID is evil" -wave.

There are real issues involved in deploying RFID, and this is not helping in figuring those out...

Posted by: Janne Jalkanen at March 16, 2006 09:17 AM

---

I would add that IMO the real threat is as stated: injecting malicious code into middleware via a rogue tag.

Posted by: 1915bond at March 16, 2006 09:19 AM

---

@Clive Robinson

"As it is used for buying and using transport tickets the RFID has to be modifiable at all times (otherwise the off line hand held units would not work)"

Stored value RFID? Combine that with the information that came out about Kinko's stored value card (http://www.schneier.com/blog/archives/2006/03/fedex_kinkos_pa.html) and what do you get? Free transit. I sure hope they have some fraud detection built into the online part of the system.

Posted by: stacy at March 16, 2006 09:21 AM

---

"You can't make money if someone else gets their product out first."

Are you serious? That's a bit counter-intuitive since you might not make the same amount of money in the same time-frame, but as someone who improves on something you can certainly make some money.

Some might say you can even make more money if you release your product later, but with better marketing and less overhead (e.g. fewer flaws and retrofits). The iPod, for example, certainly was not the first personal MP3 player to hit the market, and Apple would have made even more money had they resolved the quality issues prior to launch.

Posted by: Davi Ottenheimer at March 16, 2006 10:20 AM

---

I agree with the previous posters that rewrittable RFID is ridiculous. Its no wonder we have insecure systems, the people designing them are idiots!

RFID tags are used as "throw away" (write once read many) devices in all the implementations I have seen (i.e. inventory, passports, etc.). There is no reason the RFID would ever need to be rewritten after it is originally programmed. In fact, in all these applications, being able to rewrite an RFID after it is initially programmed would compromise the integrity of the data the RFID is representing. In these applications, why would you ever need the RFID to be rewrittable. It should be write once read many, period.

This is especially true for applications like a passport. Having a rewrittable RFID in a passport is like having the ID information on the paper part of the passport written in pencil and the picture as a removable sticker. Just like the paper part, the RFID must be unalterable after it is programmed. If changes are needed, a new paper passport document with new write-once-read-many RFID must be issued.

Now, on the original post, write-once-read-many RFIDs won't protect against poorly written middleware from getting infected from malicious RFIDs, but it would prevent other RFIDs from getting modified and infected from corrupted middleware.

Seems simple.

Posted by: SayWhat at March 16, 2006 10:25 AM

---

"You can't make money if someone else gets their product out first."

The early bird may get the worm, but the second mouse gets the cheese.

Just because you are first in doesn't mean you will reap the reward!

Posted by: Bob at March 16, 2006 10:30 AM

---

"Why would an RFID chip need to be writable?"

Cost. You can save money if you can alter the data instead of replacing the chip, although this obviously has to be weighed against data integrity control issues.

Functionality. RFID can be a very convenient way to keep records with an item such as checkin/out, maintenance, status, etc. and potentially less prone to error than human/manual entry.

Incidentally, someone once told me a story about the first US military-issue chip-based badges that were issued. They had a write-once strategy but included a user email address on the chip. Unfortunately this meant a new badge had to be issued every time an email address changed, which turned out to be far more often than anyone expected and so budgets were thrown way off the mark as the write-once cards were in a constant state of change/destruction.

Posted by: Davi Ottenheimer at March 16, 2006 10:37 AM

---

Odd that you don't link to the paper:

"Is Your Cat Infected with a Computer Virus?"

http://www.rfidvirus.org/papers/percom.06.pdf

Posted by: Fred Page at March 16, 2006 11:02 AM

---

@ Fred

Interesting paper, but Bruce discusses squid, not cats.

I noted that their recommendations were similar (if not identical) to the usual suspects in app security:

1) boundary checking
2) input validation

3) turn-off unnecessary services
4) least-privilege / role-based access
5) parameter binding (see #2)
6) isolate the servers (see #4)
7) code-review

Nothing surprising, really. And I did not see anything that said RFID should be restricted to WORM (pun not intended).

Posted by: Davi Ottenheimer at March 16, 2006 11:19 AM

---

@Davi

Sorry; I'll ask Rieback, Crispo and Tanenbuam to keep that in mind for their next paper next time I chat with them :-)

Posted by: Fred Page at March 16, 2006 11:40 AM

---

If anybody is interested in the Transport For London Oyster Card and how it came to be have a look at,

http://rfid.idtechex.com/knowledgebase/en/casestudy.asp?freefromsection=122

The last part about Stored Value Card (SVC) should make ripe pickings for fraudsters if the system is allowed to be used "off line" as it would appear the only security for lost/stolen cards is to send out a revocation list of stolen IDs (a bit like the old Credit Card Hot list).

Posted by: Clive Robinson at March 16, 2006 11:50 AM

---

@Davi

IF done properly then there is no reason for the cards to be Read Only.

However it then falls down to flexability of the system. If you asume a central write authority with secure machines then any authentication system (within reason) should be sufficient.

However if you have either OffLine or machines where the security is not gaurenteed then you do have problems. And that's when it starts getting expensive to implement properly, and cost cutting starts taking over, usually without a Risk/Benifit analysis...

Ho Hum I guess the accountants will inherit the earth ;)

Posted by: Clive Robinson at March 16, 2006 11:56 AM

---

in the coming era of automated checkout when shoppers will take goods past an rfid reader for an automatic debit to their bank card, a crook will be able to take a reader into a store and infect those tags so that every one carried past the store's reader results in an automatic credit to the bank card instead. this will put a whole new face on shoplifting.

Posted by: another_bruce at March 16, 2006 12:11 PM

---

@Clive

Not that it is pertinent to the discussion, but I find it amusing that the source photo associated with the Transport for London Oyster card UK is clearly from an American roadway.

Posted by: 1915bond at March 16, 2006 12:29 PM

---

When they made RFID cards writtable, they failed to put into place the protections that should go with such change. This puts all existing use at risk, including our soon to be federal ID card, passports, and incantations of RFID in credit/debit cards. Cost cutting measure...ha! Wasted investment. Who is going to fix this, or worse yet, pay for the mistake?

Posted by: kashmarek at March 16, 2006 03:33 PM

---

When they made RFID cards writtable, they failed to put into place the protections that should go with such change. This puts all existing use at risk, including our soon to be federal ID card, passports, and incantations of RFID in credit/debit cards. Cost cutting measure...ha! Wasted investment. Who is going to fix this, or worse yet, pay for the mistake?

Posted by: kashmarek at March 16, 2006 03:34 PM

---

Post a comment

Name:

Real names aren't required, but please give us something to call you. Conversations among several people called "Anonymous" get too confusing.

Email Address:

E-mail is optional and will not be displayed on the site.

URL:

Remember Me? Yes No

Comments:

Preview Post

Powered by Movable Type 3.2. Photo at top by Steve Woit.