t=0x8624c60 [0,0]
Contents: [Dobrica PavlinuÅ¡iÄ 's random unstructured stuff]

- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (RFID tag)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (blank tag)
    - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (3M Manufacturing Blank)
    - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Generic blank)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (programmed tag)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (security)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (disable tag)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (RFID reader device)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Related blog posts)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (More information)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO standard)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 14443, Proximity cards)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 14443-1 Physical characteristics)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 14443-2 Radio frequency power and signal interface)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 14443-2/AMD2 Amendment 2: Bit rates of fc/64, fc/32 and fc/16)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 14443-3 Initialization and anticollision)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 14443-3 Amendment 1: Bit rates of fc/64, fc/32 and fc/16)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 14443-4 Transmission protocol)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 15693, Vicinity cards)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 15693-1 Physical characteristics)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 15693-2 Air interface and initialisation)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 15693-3 Anticollision and transmission protocol)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 10373-6, 10373-7, Test methods for the contactless integrated circuit(s) cards)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 10373-6 Proximity cards)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 10373-6/AMD1 Amendment 1: Additional PICC test methods)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 10373-6/AMD2 Amendment 2: Improved RF test methods)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 10373-7 Vicinity cards)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 10536, Close-coupled cards)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (ISO/IEC 10536-1 Physical characteristics)

There are many on-line resources about RFID. However, most of them are not well suited for beginners. So, if you just got RFID reader (3M in this case) and want high-level overview of what it is and what it can do, you are out of luck.

Until now, that is. This guide should help you decide if RFID is right thing for your library and when you make choice to implement it, how to do it.

This document will describe my experience with 3M 810 reader using RFID 501: RFID Standards for Libraries RFID_501.pdf

RFID tag

Best way to think about RFID tags is like contact-less readable barcode.

Our particular tags come in two form: RFID stickers (to be placed on books) and plastic credit-card like cards (we use them for patrons).

Have in mind that established practice is to have different RFID systems for books and patrons (we are use same one). When we asked about using same system for books and patrons, we got reply: "we don't have experience with it".
In practice, we have problem with 3M selfcheck software in some special conditions where book reader have patron card in range it gets network connection error with SIP server.
Since normal configuration of selfcheck stations is to have two different systems for patrons and books this problem doesn't show up for other users.

Each tag has unique serial number (SID) assigned by manufacturer and used in RFID collision detection protocol. It looks like hexadecimal number starting with letter `E0` like this: `E00401003123AA26`.

It's best to think of SID as unique identifier of physical tag.
Your information system will have it's own ID (barcode?) for some item.
There are cases in which you might want to change physical tag sticker on book because it's damaged. In that case, you will change SID of that item, but not barcode (which is data programmed on tag itself).

Our initial idea was to use data programmed on chip for everything, and just ignore SIDs, but we found out that there is class of RFID devices which can read ONLY SID from chip (in our case it's photocopying system).
3M software does record SIDs to log file when programming chips, but that's all. It essential ignores it for all practical intends and purposes.

Chips have 7 blocks of user data on it, each block with 4 bytes which enables us to store 28 bytes of user specified data on each tag.

blank tag

3M Manufacturing Blank

Easiest case is blank tag, in which all data on chip is 0x55

```
0       55 55 55 55             blank tag
1       55 55 55 55
2       55 55 55 55
3       55 55 55 55
4       55 55 55 55
5       55 55 55 55
6       00 00 00 00
```

Generic blank

Generic blank seems to erase only first three blocks with zeros:

```
00      00 00 00 00
01      00 00 00 00
02      00 00 00 00
```

while rest of tag will be unchanged **including rest of data on tag**

programmed tag

Tags programmed with 3M software have following data layout on them:

```
0       04 is 00 tt             i [4 bit] = number of item in set       [1 .. i .. s]
                                        s [4 bit] = total items in set
                                        tt [8 bit] = item type

1       dd dd dd dd             dd [16 bytes] = barcode data
2       dd dd dd dd
3       dd dd dd dd
4       dd dd dd dd

5       bb bl ll ll             b [12 bit] = branch [unsigned]
                                l [20 bit] = library [unsigned]
6       cc cc cc cc             c [32 bit] = custom signed integer
```

This basically means that your barcode or identifier of item or patron can have up to 16 characters (by default numeric, but you can extend that to handle alphanumeric and special character if you need that) and three integer values: branch `0 .. 4095`, library `0 .. 1048575` and custom data `-2147483648 .. 2147483647`.

You might want to use those values to uniquely identify your library and branch so that your RFID tags in books won't collide with other libraries.
If you leave decision just to providers of equipment, you might end up with 300000 tags which have plain and simple 0 in those fields. Guess which value will have tags of next library which that provider will have? My guess would be 0 also.

Writing correct numbers in that fields is not enough. If you want to use 3M software, you will also have to setup it to ignore all other tags which doesn't match your library and branch.

security

There is also single byte called AFI or security which can be changed without accessing content of chip. This byte is also readable by more primitive RFID devices like doors to check if book have been checked out from library.

3M is using `0xD7` (215) value for secured items (door will beep) and `0xDA` (218) as unsecured. It seems that all other values are ignored.
(I would guess that other manufacturers are using different values)

As I mentioned before, since we don't have any special values in branch, library or custom field, we have situations in which patron cards get secured when patron walks by checkout counter and 3M software is left in checkout mode.
This triggers door to ring when patron passes which is not ideal.

disable tag

3M software have option to disable tags. Initial examination showed that it's simply programming of tag with following content:

```
00      ff 00 00 00
01      00 00 00 00
02      00 00 00 00
03      00 00 00 00
04      00 00 00 00
05      00 00 00 00
06      00 00 00 00
```

and security set to `d7` (this might be value from tag before disabling it, I'll have to re-check this)

While 3M software will ignore tags programmed with this content, there is **not permanent disabling of tag** since it can be programmed using other software.

RFID reader device

Reader consists of several part:

- black pad - reader antenna
- reader - small box with micro controller and usb port
- software

Reader is recognized as USB serial device with it's own protocol on serial port. We are mostly interested in it's protocol and our ability to use reader and tags with our custom software.

At first, I assumed that protocol with RFID readers is some kind of standard.
After extensive search on Internet I wasn't able to find any documentation about this particular protocol (I even tried to compare it with existing open source implementations just to be sure).

So, only solution was to do clean-room reverse engineering, and using that technique I developed perl module which can talk with RFID reader which is available at http://svn.rot13.org/index.cgi/RFID

After initial reverse engineering of protocol I rewrote support for 3M and CPR reader which is available at https://github.com/dpavlin/Biblio-RFID

Related blog posts

fetchrss:
http://mjesec.ffzg.hr/~dpavlin/blog/mt/mt-search.cgi?tag=RFID&Template=feed&IncludeBlogs=1

- There was an error: 500 Internal Server Error

More information

If this was too geeky for you here is some additional materials:

- RFID for libraries FAQ
- RFID - overview of protocols, librfid implementation and passive sniffing

ISO standard

- ISO 15962.2004 - object identifier structure
- ISO 15693 - RFID (layer 2)
- ISO 18000 Part 3 Mode 1 - 13.56MHz
- ISO/IEC JTC1/SC17/WG8

ISO/IEC 14443, Proximity cards

The Standard series ISO/IEC 14443 consists of 4 parts, which are:

ISO/IEC 14443-1 Physical characteristics

17n1363t.doc 17n1363b.doc

ISO/IEC 14443-2 Radio frequency power and signal interface

17n1522t.pdf 17n1522c.doc

ISO/IEC 14443-2/AMD2 Amendment 2: Bit rates of fc/64, fc/32 and fc/16

17n2343T.pdf 17n2343F.doc

ISO/IEC 14443-3 Initialization and anticollision

17n1531t.pdf 17n1531c.doc

ISO/IEC 14443-3 Amendment 1: Bit rates of fc/64, fc/32 and fc/16

17n2342T.pdf 17n2342F.doc

ISO/IEC 14443-4 Transmission protocol

17N1689T.pdf 17n1689c.doc

ISO/IEC 15693, Vicinity cards

ISO/IEC 15693-1 Physical characteristics

17n1355t.doc 17n1355b .doc

ISO/IEC 15693-2 Air interface and initialisation

17n1486.pdf 17n1486c.doc

ISO/IEC 15693-3 Anticollision and transmission protocol

17n1692t.pdf 17n1692c.doc

ISO/IEC 10373-6, 10373-7, Test methods for the contactless integrated circuit(s) cards

ISO/IEC 10373-6 Proximity cards

17n1695t.pdf 17n1695c.doc

ISO/IEC 10373-6/AMD1 Amendment 1: Additional PICC test methods

17n2258t.pdf 17n2258t.doc 17n2258C.doc

ISO/IEC 10373-6/AMD2 Amendment 2: Improved RF test methods

17n2225t.pdf 17n2225F.doc

ISO/IEC 10373-7 Vicinity cards

17n1697t.pdf 17n1697c.doc

ISO/IEC 10536, Close-coupled cards

ISO/IEC 10536-1 Physical characteristics

17n1480t.PDF 17n1480c.doc