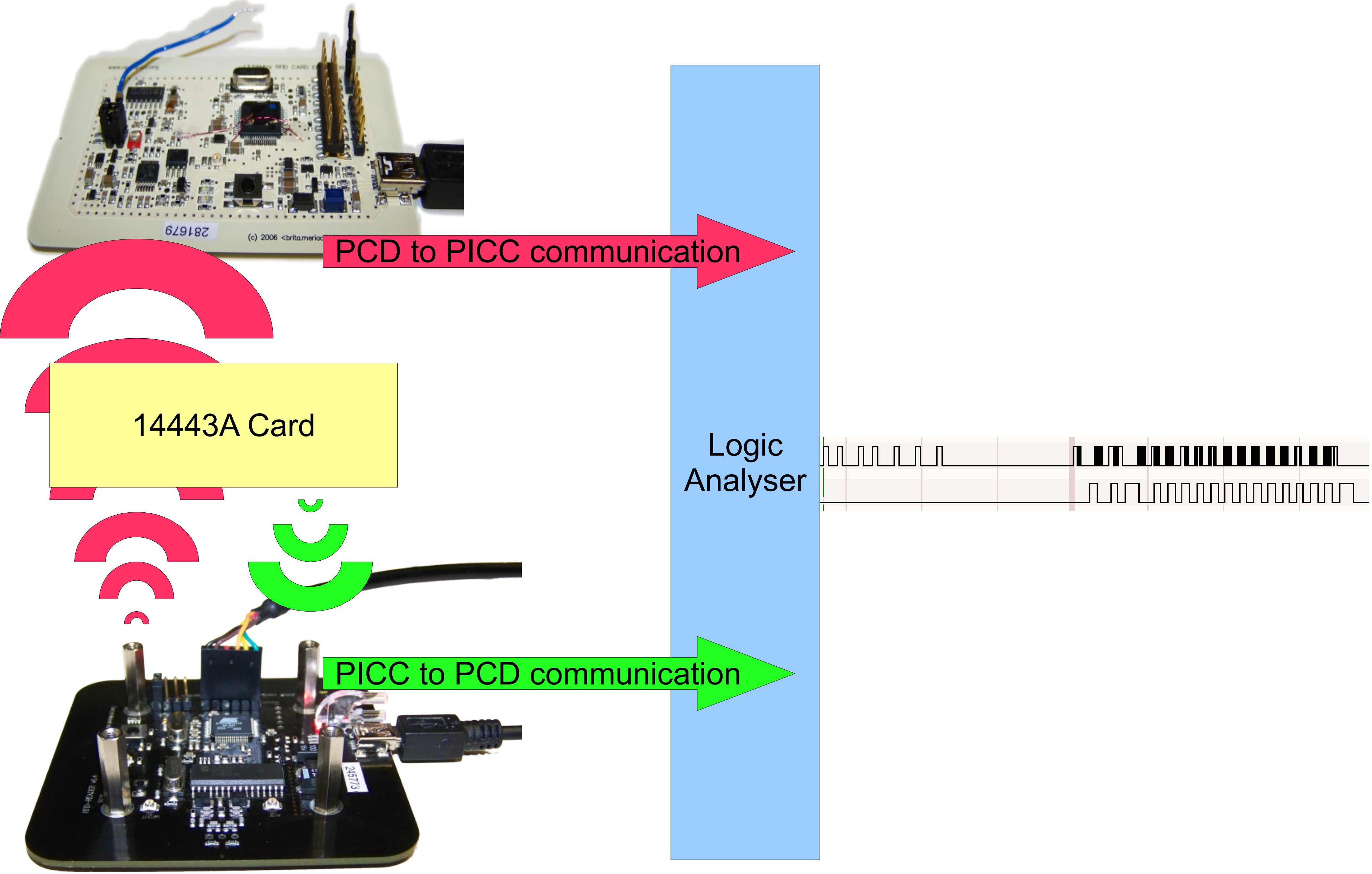
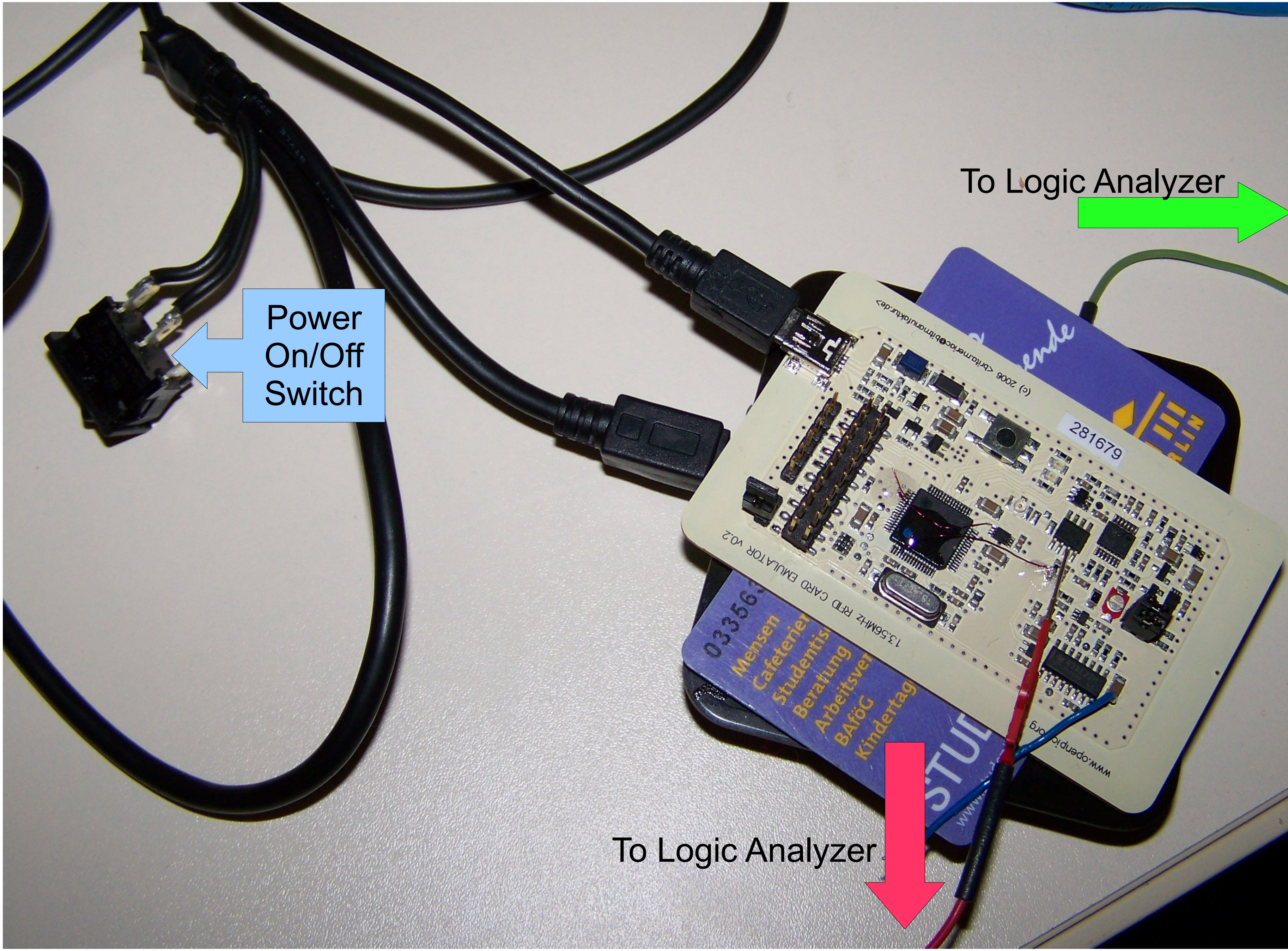

PCD to PICC communication
14443A Card
Logic Analyser
PICC to PCD communication


To Logic Analyzer
Power On/Off Switch
281679
13.56MHz RFID CARD EMULATOR v0.2
To Logic Analyzer

Strategy:

1) Turn everything off
2) Start capture on logic analyzer
3) Switch on PCD, automatically boots firmware and performs authentication
4) Stop capture and save data
5) Put data through Manchester/Miller decoder (and filter out authentication frames)
6) Restart from step 1

First few mutual authentication exchanges observed:

```
PICC:  FF 1  CF 1  80 0  E3 0
 PCD:  38 1! C5 1  B5 1! 45 0  84 0! D5 1! 04 0  7F 1!
PICC:  DF 0  58 1! 61 1! B3 0

PICC:  7D 1  DA 0  7E 1  41 1
 PCD:  1E 0! 98 1! 43 1! FB 1! D6 0  CD 1! 65 0! E5 1!
PICC:  A6 1  23 1! 0A 1  9C 1

PICC:  7D 1  DA 0  7E 1  41 1
 PCD:  53 1  03 1  8F 1! 3A 1  66 0! 85 1! D5 1! 48 0!
PICC:  87 0! 8E 0! 75 0  D3 1!

PICC:  7D 1  DA 0  7E 1  41 1
 PCD:  1E 0! 98 1! 43 1! FB 1! D6 0  CD 1! 65 0! E5 1!
PICC:  A6 1  23 1! 0A 1  9C 1
```

Statistics for 27 trials

Count	PICC->PCD				PCD->PICC					
1	3C 1	1E 1	85 0	D2 1	AE 1!	29 0	3E 1!	97 0	8D 1	...
1	4D 1	23 0	ED 1	A6 1	57 0	3F 1	5E 1!	F2 0	B5 0	...
1	4D 1	23 0	ED 1	A6 1	76 1!	DF 1!	E3 0	1C 1!	CD 1!	...
3	77 1	3F 1	BF 0	EE 1	10 1!	B9 0	B0 0	14 1	37 0	...
1	77 1	3F 1	BF 0	EE 1	34 1!	13 1!	9B 1!	9B 1!	F2 0	...
6	7D 1	DA 0	7E 1	41 1	1E 0!	98 1!	43 1!	FB 1!	D6 0	...
4	7D 1	DA 0	7E 1	41 1	52 0	68 0	D8 1	63 0!	BB 1	...
1	7D 1	DA 0	7E 1	41 1	53 1	03 1	8F 1!	3A 1	66 0!	...
1	7D 1	DA 0	7E 1	41 1	C3 1	40 1!	90 0!	30 0!	6D 1!	...
1	B2 1	E8 1	CD 0	40 0	23 0	66 0!	5A 0!	C3 0!	46 1!	...
1	B2 1	E8 1	CD 0	40 0	8E 1	05 0!	58 0	26 1!	15 1!	...
1	BB 1	9F 1	5F 1	77 1	45 0	80 1!	7B 1	0B 0	92 1!	...
1	BB 1	9F 1	5F 1	77 1	49 1!	11 0!	98 1!	B1 1	67 0	...
2	E4 1	56 1	36 1	BB 1	7A 1!	AB 1!	A3 1	D9 0	A2 0	...
1	E4 1	56 1	36 1	BB 1	F6 1	23 0	70 1!	F9 1	A9 1	...
1	FF 1	CF 1	80 0	E3 0	38 1!	C5 1	B5 1!	45 0	84 0!	...

The initial state of the cipher must be derived from UID and key, e.g. by xoring UID and key (or similar function).

Idea: Flipping a bit of the key and flipping the corresponding bit in the UID (on the PCD side) should yield the same initial state.

Results:

Bit flipped in UID	Bit flipped in key	
0	0	success
1	1	success
2	2	success
3	3	success
4	4	success
5	5	no success

Next idea: Flipping one bit in the key might need flipping multiple bits in the UID to reach the same state

<table>
<tr><th>Bit flipped in key</th><th>Bits flipped in UID</th><th></th><th>in hex</th><th>equals</th></tr>
<tr><td>0</td><td>0</td><td>success</td><td>0x1</td><td></td></tr>
<tr><td>1</td><td>1</td><td>success</td><td>0x2</td><td>0x1 << 1</td></tr>
<tr><td>2</td><td>2</td><td>success</td><td>0x4</td><td>0x2 << 1</td></tr>
<tr><td>3</td><td>3</td><td>success</td><td>0x8</td><td>0x4 << 1</td></tr>
<tr><td>4</td><td>4</td><td>success</td><td>0x10</td><td>0x8 << 1</td></tr>
<tr><td>5</td><td>0 5</td><td>success</td><td>0x21</td><td>(0x10 << 1) | 1</td></tr>
<tr><td>6</td><td>1 6</td><td>success</td><td>0x42</td><td>0x21 << 1</td></tr>
<tr><td>7</td><td>2 7</td><td>success</td><td>0x84</td><td>0x42 << 1</td></tr>
<tr><td>8</td><td>3 8</td><td>success</td><td>0x108</td><td>0x84 << 1</td></tr>
<tr><td>9</td><td>4 9</td><td>no success</td><td>0x210</td><td>0x108 << 1</td></tr>
<tr><td>9</td><td>0 4 9</td><td>success</td><td>0x211</td><td>(0x108 << 1) | 1</td></tr>
<tr><td>10</td><td>1 5 10</td><td>no success</td><td>0x422</td><td>0x211 << 1</td></tr>
<tr><td>10</td><td>0 1 5 10</td><td>success</td><td>0x423</td><td>(0x211 << 1) | 1</td></tr>
<tr><td>11</td><td>1 2 6 11</td><td>success</td><td>0x846</td><td>0x423 << 1</td></tr>
<tr><td>…</td><td></td><td></td><td></td><td></td></tr>
<tr><td>31</td><td>2 4 6 7 12 14 16 17 19 21 22 26 31</td><td>success</td><td>0x846b50d4</td><td></td></tr>
</table>

Consequence: Corresponding key/uid pairs can be generated that yield the same initial cipher state.

Card X

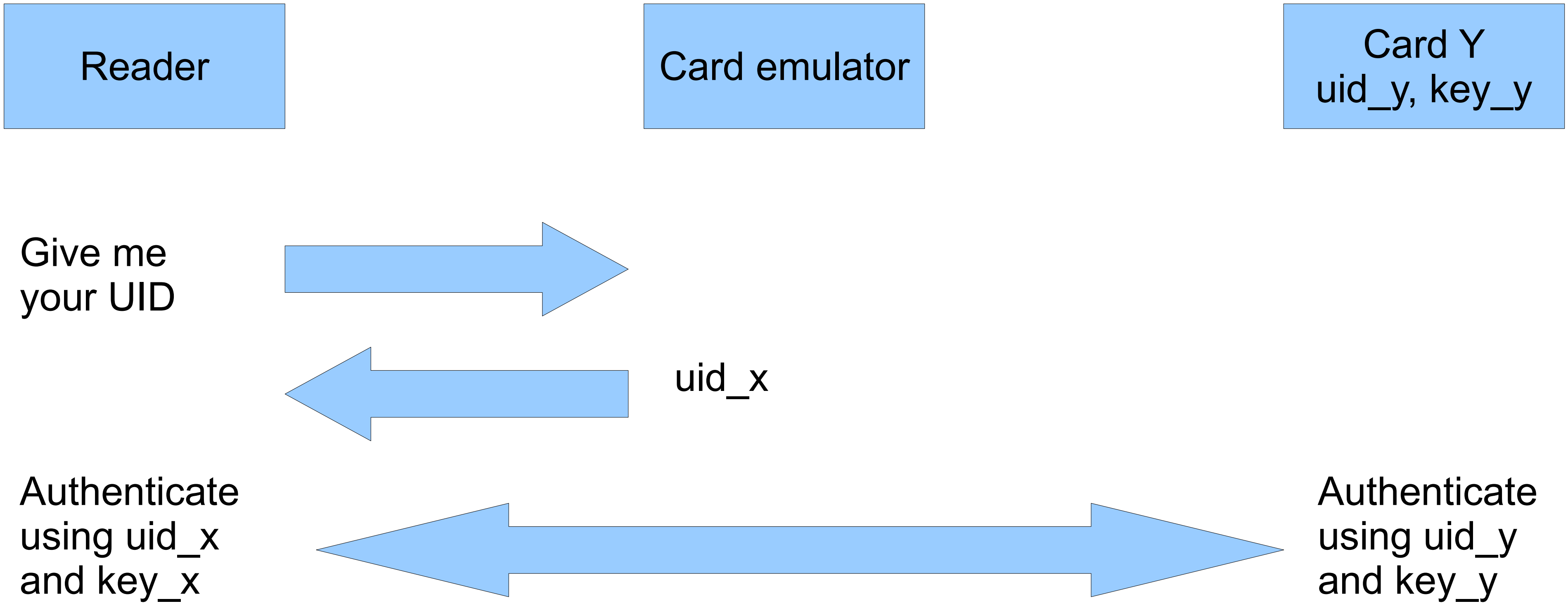
uid_x
key_x

Card Y

uid_y
key_y

Given uid_x, key_x and uid_y we can generate key_y
Enables UID/card spoofing when the key is known **without knowledge of the algorithm**

Example usage:



Success!
The reader is now talking to card Y
but thinks it's talking to card X