Attacking and Defending RFID Authentication Systems

Matthew Green

Graduate Cryptographers Unlock Code of 'Thiefproof' Car Key

By JOHN SCHWARTZ

BALTIMORE — Matthew Green starts his 2005 Ford Escape with a duplicate key he had made at Lowe's. Nothing unusual about that, except that the automobile industry has spent millions of dollars to keep him from being able to do it.

Mr. Green, a graduate student at Johns Hopkins University, is part of a team that plans to announce on Jan. 29 that it has cracked the security behind "immobilizer" systems from Texas Instruments Inc. The systems reduce car theft, because vehicles will not start unless the system recognizes a tiny chip in the authorized key. They are used in millions of Fords, Toyotas and Nissans.

All that would be required to steal a car, the researchers said, is a moment next to the car owner to extract data from the key, less than an hour of computing, and a few minutes to break in, feed the key code to the car and hot-wire it.

An executive with the Texas Instruments division that makes the systems did not dispute that the Hopkins team had cracked its code, but said there was much more to stealing a car than that. The devices, said the executive, Tony Sabetti, "have been fraud-free and are likely to remain fraud-free."

The implications of the Hopkins finding go beyond stealing cars.

Variations on the technology used in the chips, known as RFID for radio frequency identification, are widely used. Similar systems deduct highway tolls from drivers' accounts and restrict access to workplaces.

Wal-Mart is using the technology to track inventory, the Food and Drug Administration is considering it to foil drug counterfeiting, and the medical school at the University of California, Los Angeles, plans to implant chips in cadavers to curtail unauthorized sale of body parts.

The Johns Hopkins researchers say that if other radio frequency ID systems are vulnerable, the new

University of California, Berkeley, who reviewed a draft of a paper by the Hopkins team, called it "great research," adding, "I see it as an early warning" for all radio frequency ID systems.

The "immobilizer" technology used in the keys has been an enormous success. Texas Instruments alone has its chips in an estimated 150 million keys. Replacing the key on newer cars can cost hundreds of dollars, but the technology is credited with greatly reducing auto theft. — Early versions of in-key chips were relatively easy to clone, but the Texas Instruments chips are considered to be among the best. Still, the amount of computing the chip can do is restricted by the fact that it has no power of its own; it builds a slight charge from an electromagnetic field from the car's transmitter.

Cracking the system took the graduate students three months, Dr. Rubin said. "There was a lot of trial and error work with, every once in a while, a little 'Aha!' "

The Hopkins researchers got unexpected help from Texas Instruments itself. They were able to buy a tag reader directly from the company, which sells kits for $280 on its Web site. They also found a general diagram on the Internet, from a technical presentation by the company's German division. The researchers wrote in the paper describing their work that the diagram provided "a useful foothold" into the system. (The Hopkins paper, which is online at www.rfidanalysis.org, does not provide information that might allow its work to be duplicated.

The researchers discovered a critically important fact: the encryption algorithm used by the chip to scramble the challenge uses a relatively short code, known as a key. The longer the code key, which is measured in bits, the harder it is to crack any encryption system.

"If you were to tell a cryptographer that this system uses 40-bit

Photographs by Marty Katz for The New York Tim

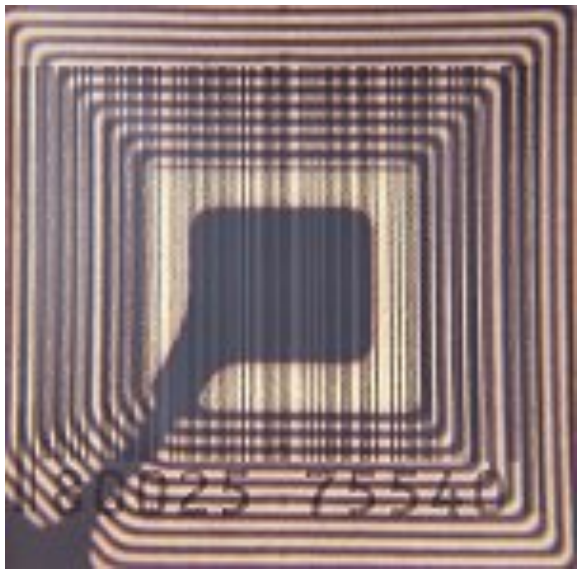
From left, Prof. Aviel D. Rubin, Adam Stubblefiel Matthew Green and Stephen Bono working with card programmed to conduct an assault on a car-key chip, shown magnified at left.

Asking some questions…

- Academics think security is a straightforward problem
  - JHUSI, MSSI program
  - SSH, SSL, etc.
- But in the real world: how does security get deployed?
- A great way to look at this is to examine emerging technologies

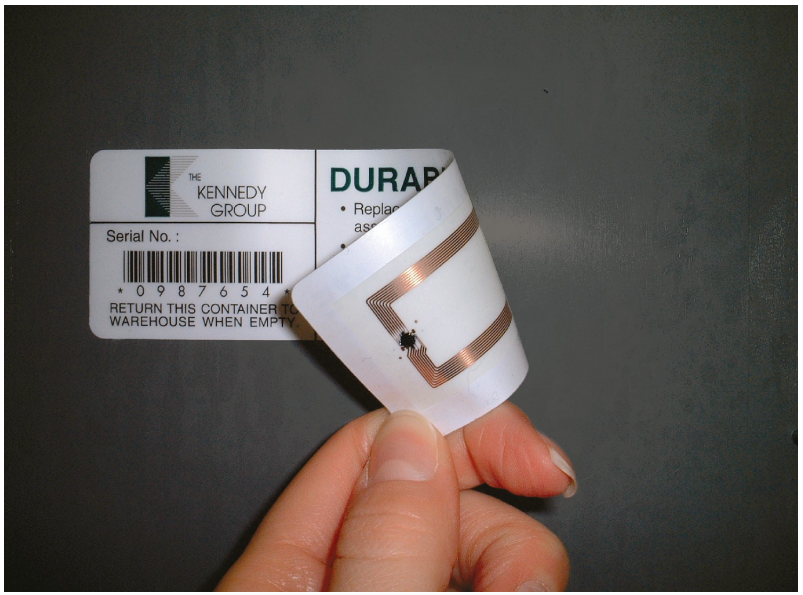
Radio Frequency Identification

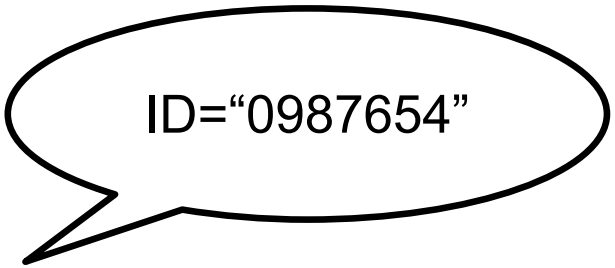
- Limited computing devices
  - Identification, data storage, cryptography
- Contactless: Integrated RF transceiver
  - Communicates with RFID reader via radio protocol



“Walmart” RFID Tags

- Simple RFID just broadcast an ID
  - Simple, short/medium range RF protocol
  - May include collision detection





Johns Hopkins University
Information Security
Institute

“Walmart” RFID Tags

- Severely limited devices
  - Cost is dominating factor
  - $.50 to <$.05 goal cost
  - Dominates Moore’s law
  - No privacy/security protections

Privacy

- Some researchers are worried about the privacy implications
  - RFID tags in your clothes, medicine, passport, inside your body
  - Can you be tracked?

Johns Hopkins University Information Security Institute

But what about Security?

- RFID systems are also being deployed in security-critical applications
  - Payment systems
  - Toll collection
  - Vehicle anti-theft systems

Johns Hopkins University
Information Security
Institute

Example: Vehicle Immobilizer

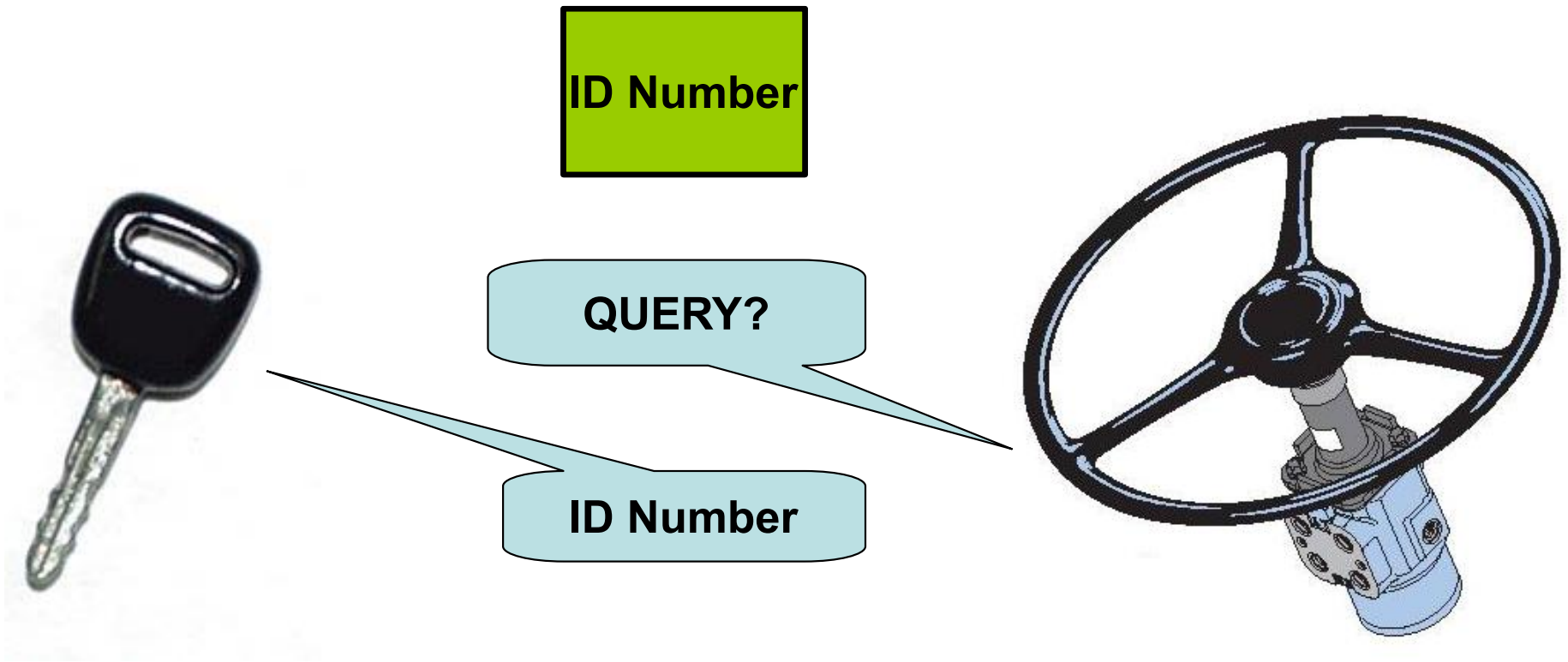
Immobilizer System

An (RFID) chip in your key communicates with the computer in your car

Example: Vehicle Immobilizer

Immobilizer System

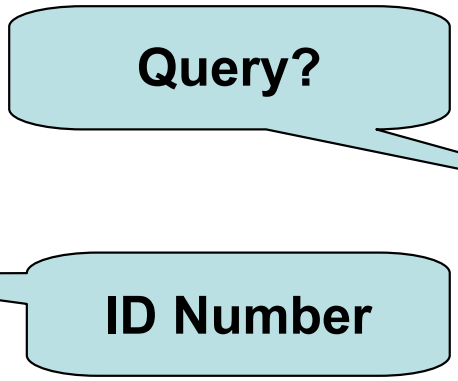
Many (older) designs use simple RFID chips

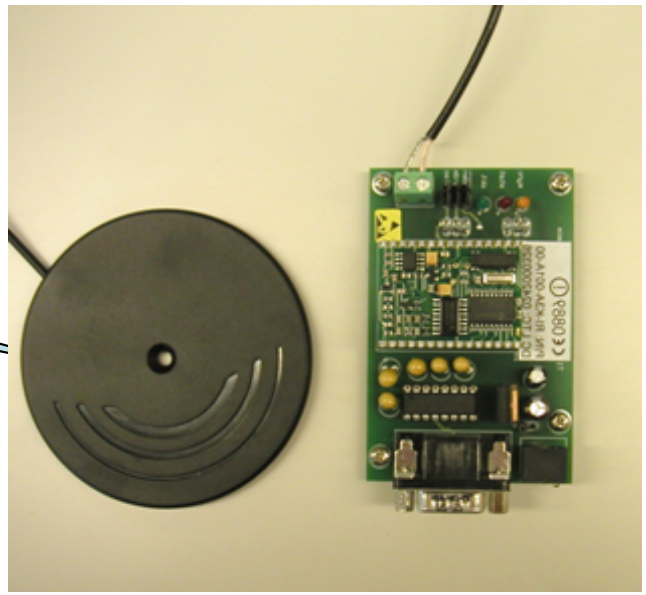


Defeating simple Immobilizers

How to clone a key:

1. Scan target key, get ID number.

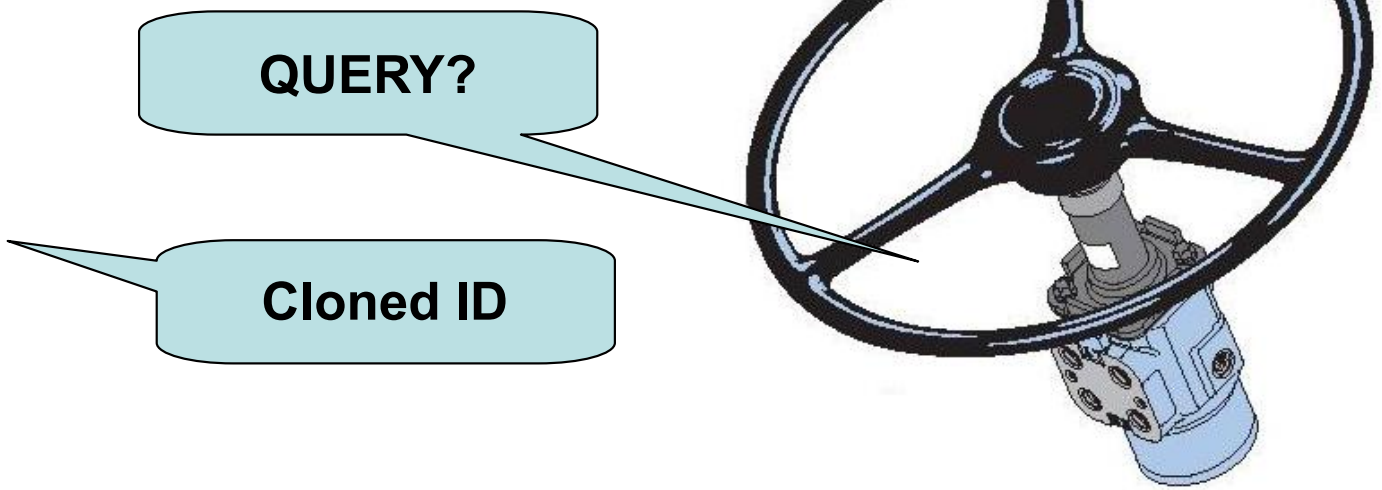




Defeating simple Immobilizers

How to clone a key:

1. Scan target's key, get ID number.
2. Replay response to vehicle.



Our Work

Project Outline

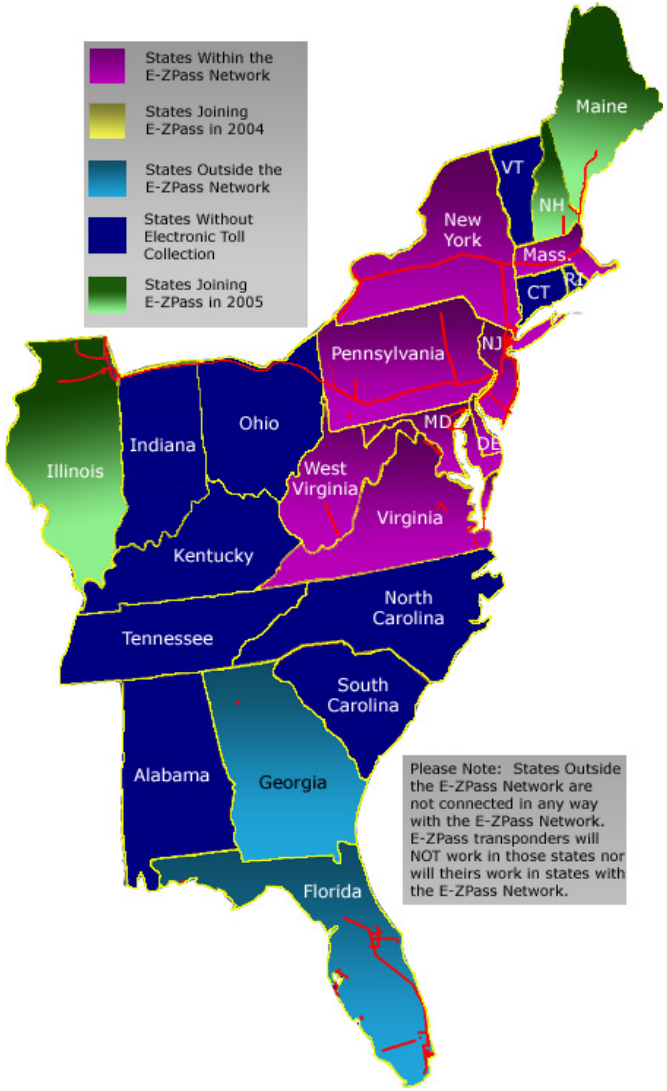
- Examine a few widely-deployed platforms
- Reverse-engineer devices/protocols
  - Overcome physical reader limitations
  - Until we do that, we can't even determine if they have security built in
- Only problem is: we don't know anything about radios…

Our Targets

- EZ-Pass
- ExxonMobil Speedpass (TI DST)

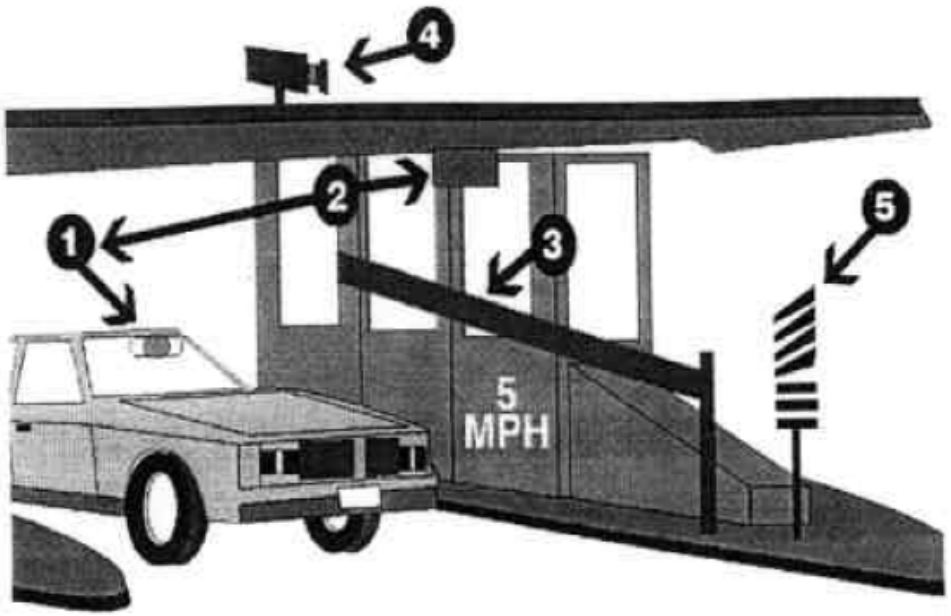
E-ZPass

- High-speed toll collection
  - Widely deployed
  - Real $$
  - **Large read distance**
  - **Reader, protocol not available to the public**



The E-ZPass System

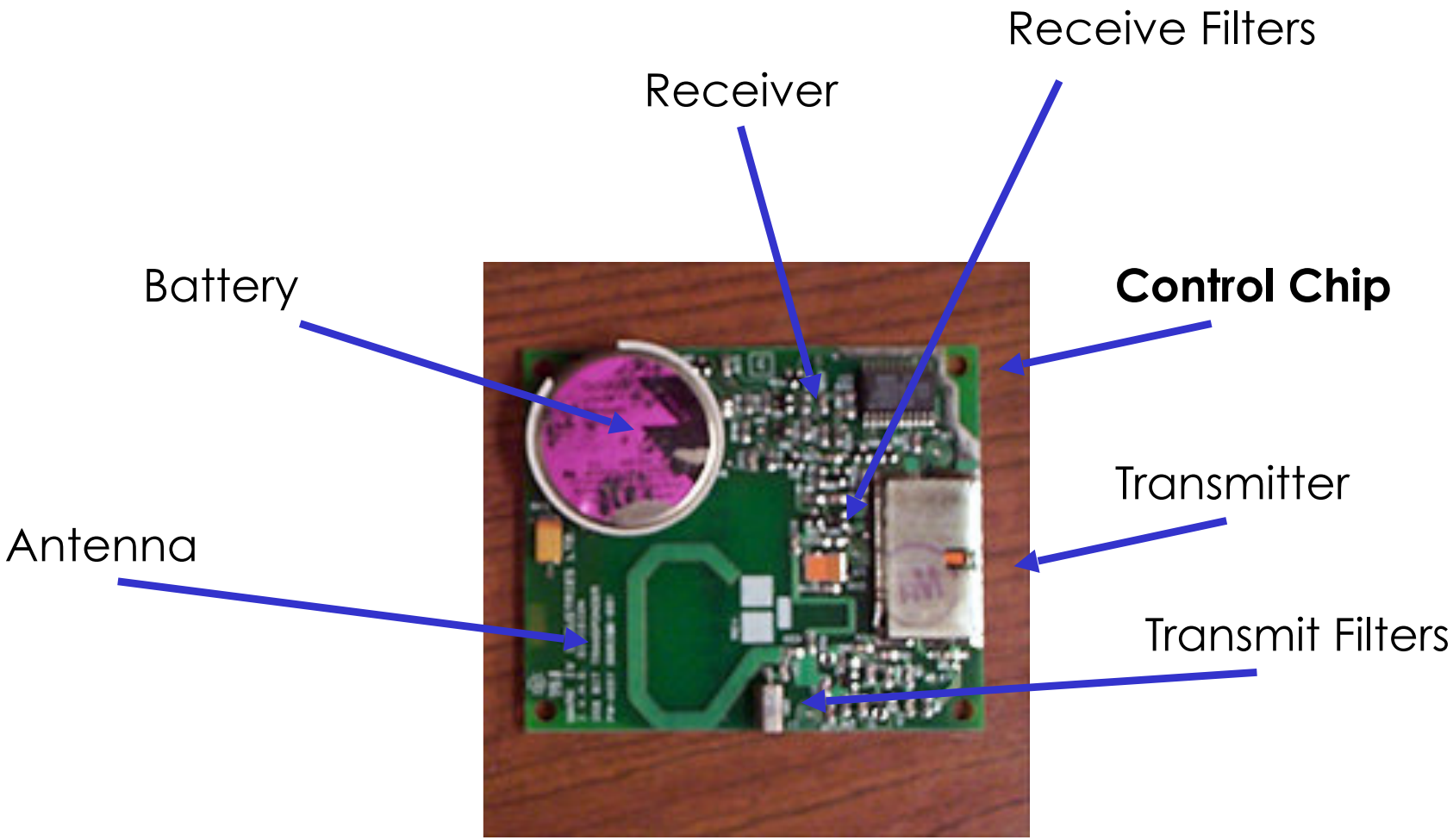
- Tags interrogated by fixed readers
  - Signal read at highway speeds
  - Deliberately limited range within booths, but can (possibly) extend to 100+ ft



E-ZPass Transponder

- Active (powered) transponder
  - Frequency of Operation: 915/914MHz
  - Data transmit rate: 300-500Kbps

Anatomy of an E-ZPass



Determining the Protocol

- **Bad news**: Tags don't do anything until they're activated.
- **Good news**: We have tags, a car, and plenty of toll-booths!

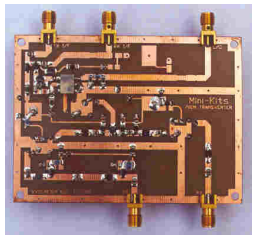
Software Radio Approach

- Snoop the toll-booth protocol using software and commodity PC hardware

Antenna

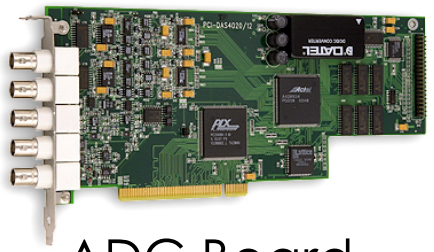
~900MHz RF Transaction

E-ZPass Reader



Transverter
(~900MHz -> ~40MHz)

Software-tunable Radio
(0-60MHz) -> (<20Mhz)



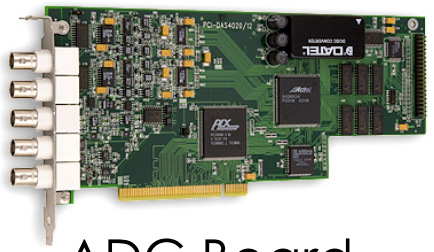
ADC Board

Johns Hopkins University
Information Security
Institute

Franken-Pass Shortcut

- Tag already has antenna and transceiver equipment, so let's use it

E-ZPass Reader

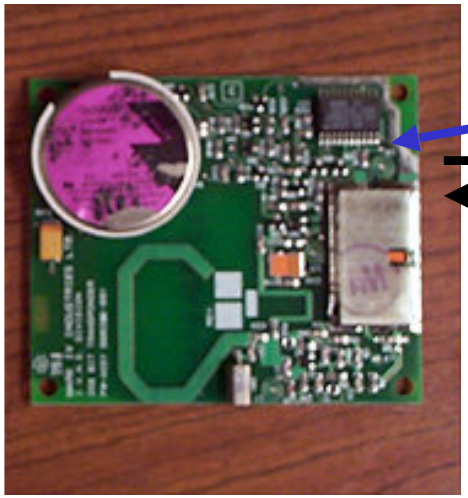


ADC Board

Johns Hopkins University
Information Security
Institute

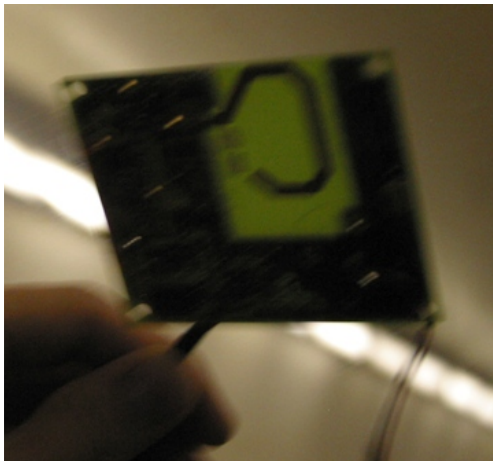
Franken-Pass

E-ZPass Reader



Tx/Rx Lines

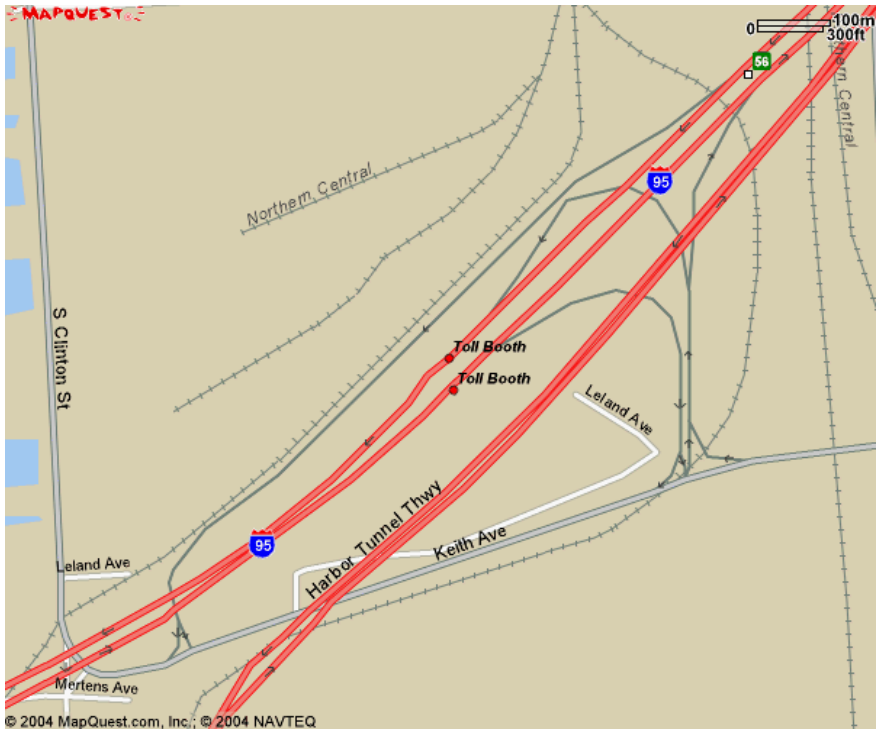
PC



Johns Hopkins University
Information Security
Institute

Field Test #1

- Location: Fort McHenry Tunnel Toll booths, Baltimore Harbor



Johns Hopkins University
Information Security
Institute

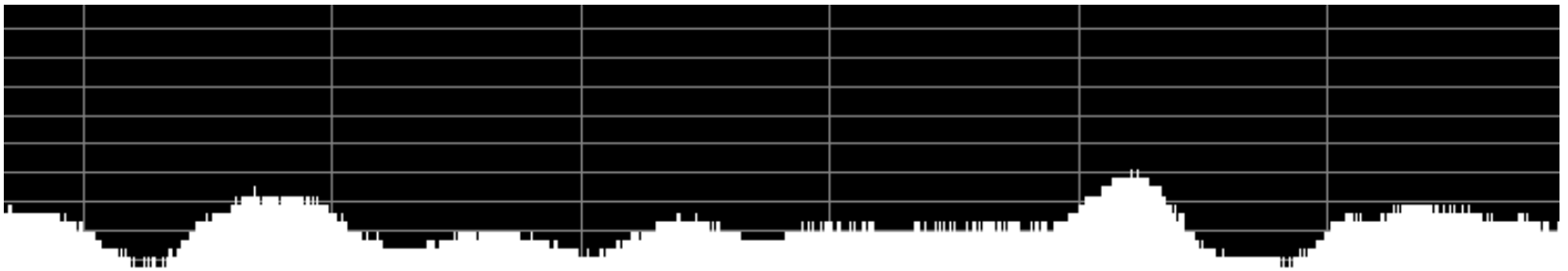
Field Test #1

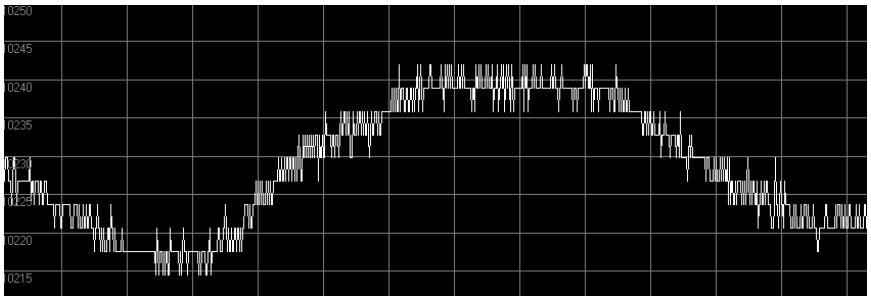
- Equipment list:
  - Modified M-Tag
  - PCI-DAS4020 DAC Card
  - Shuttle XPC SG85

The EZ-Pass Protocol

- Stage 1:

20 µsec activation pulse





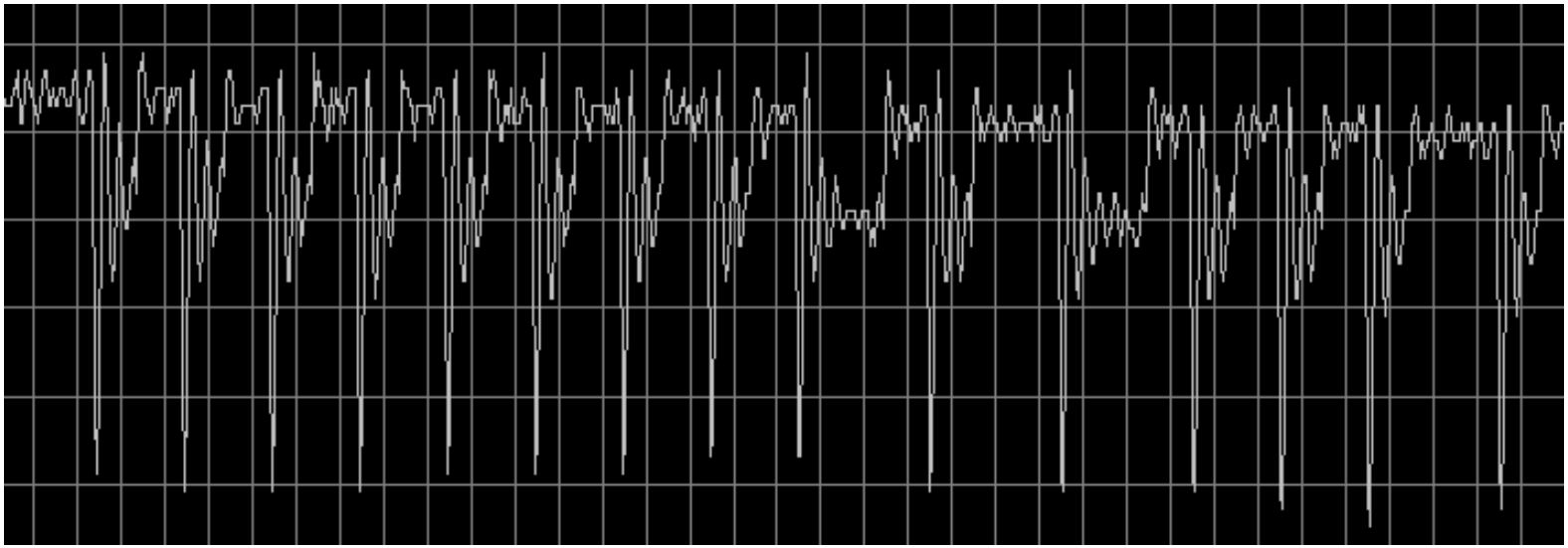
Johns Hopkins University
Information Security
Institute

The EZ-Pass Protocol

- Stage 2:

20 µsec activation pulse

516 µsec response (256 bits + CRC?)
Manchester Encoded



The EZ-Pass Protocol

- Stage 3:

20 µsec activation pulse

516 µsec response (256 bits + CRC)

OPTIONAL 256-bit write phase …?

Attacking EZ-Pass

- Plenty of power, plenty of read range…
  … but no security in the tag
  - Toll booth cameras
- No protection against tracking
  - Anyone can activate the tag, potentially track hundreds of cars
  - Not so useful for us, though: FCC regs limit activation power
  - Doesn't affect "eavesdropping"

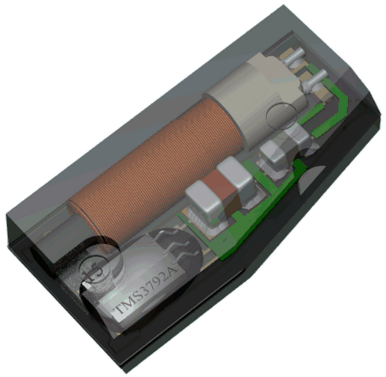
Texas Instruments DST

Vehicle Immobilizers

ExxonMobil Speedpass

Texas Instruments DST

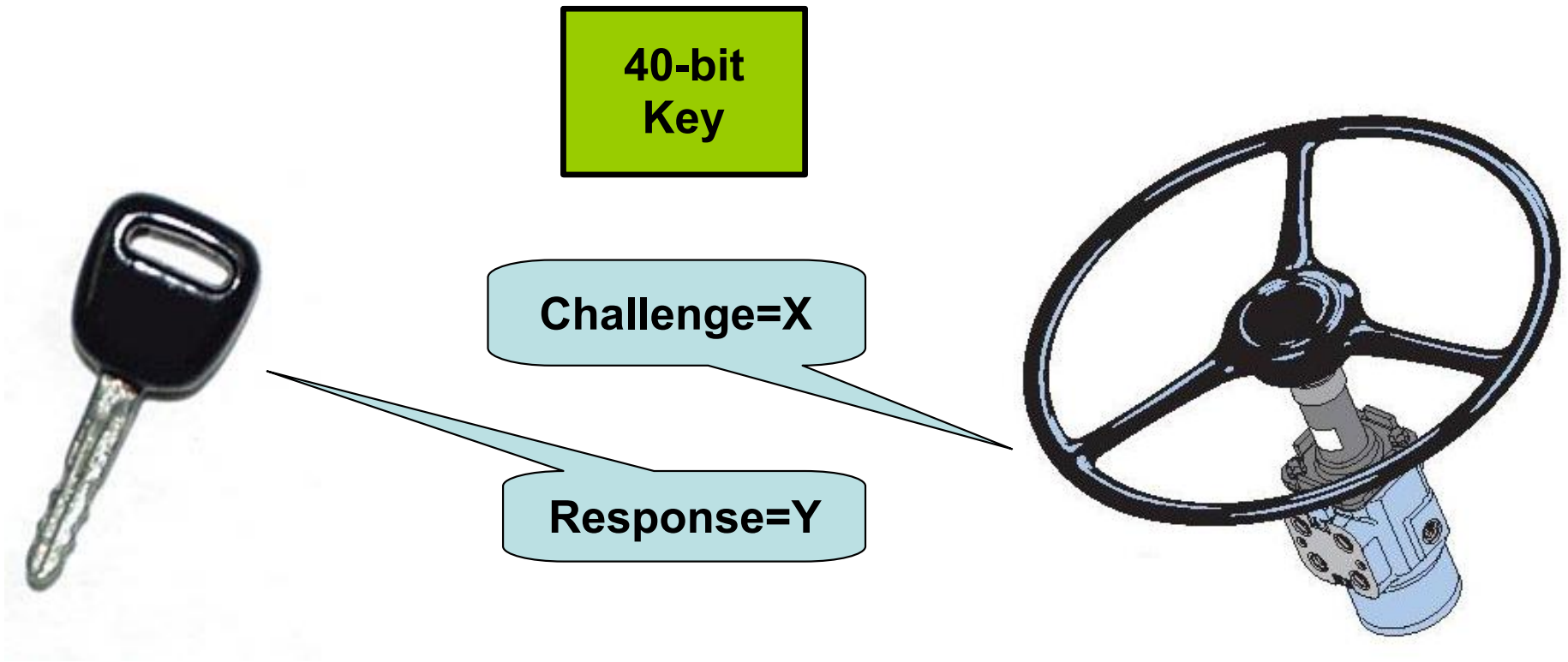
- Operating Frequency: 134 KHz
- Power: Passive
- Read range: ~1ft
- **Security**: **challenge/response protocol**
  - 40-bit challenge, 40-bit key, 24-bit response



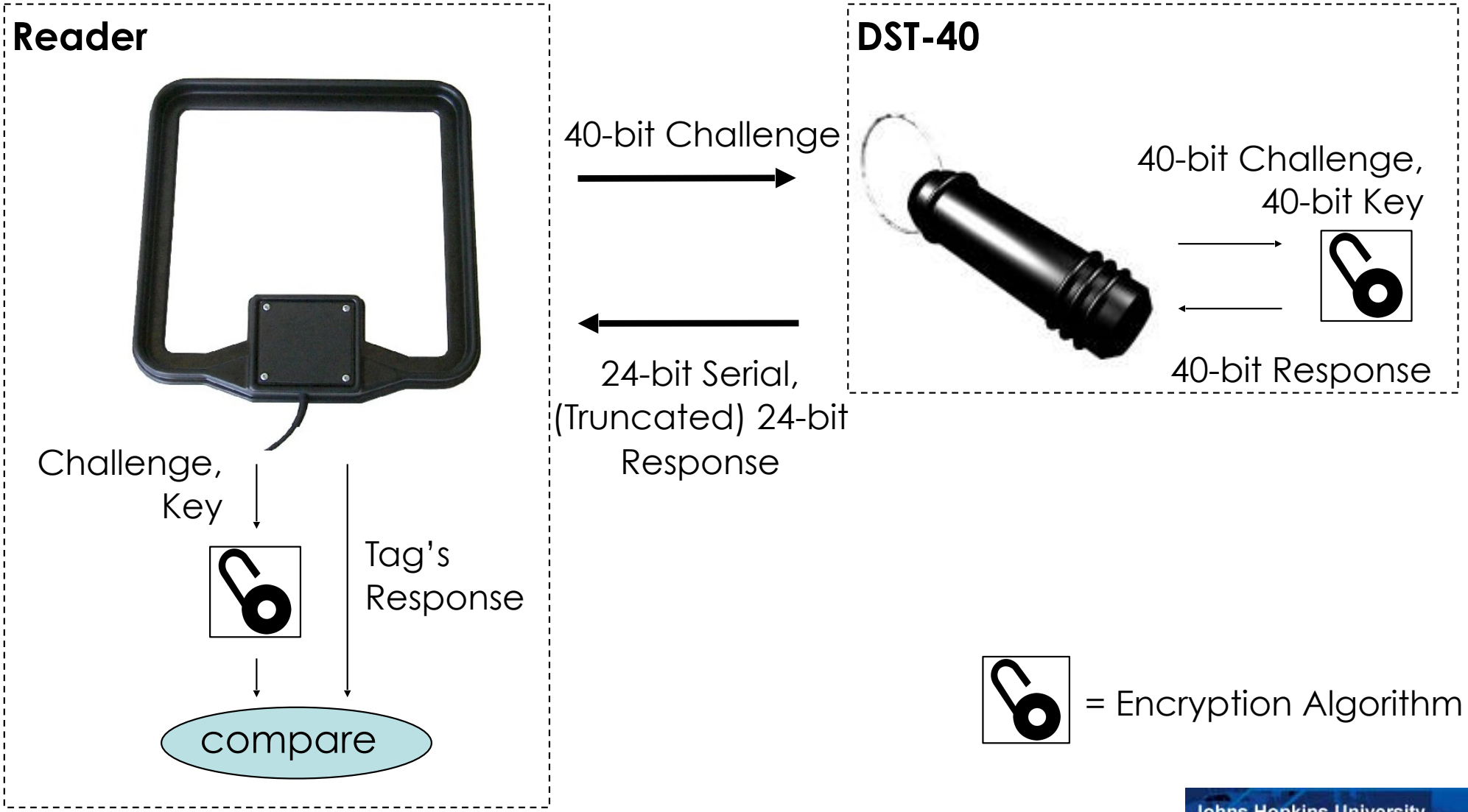
DST Immobilizers

Challenge/Response Immobilizer System

Uses random challenge and cryptography



DST-40 Operation



What is  ?

- TI Proprietary block cipher
  - Available by NDA only
- Even with good cipher, 40-bit key is a major weakness
  - Brute force guessing
  - Full precomputed key table ~5TB
- Problem is: we don't know how the algorithm!

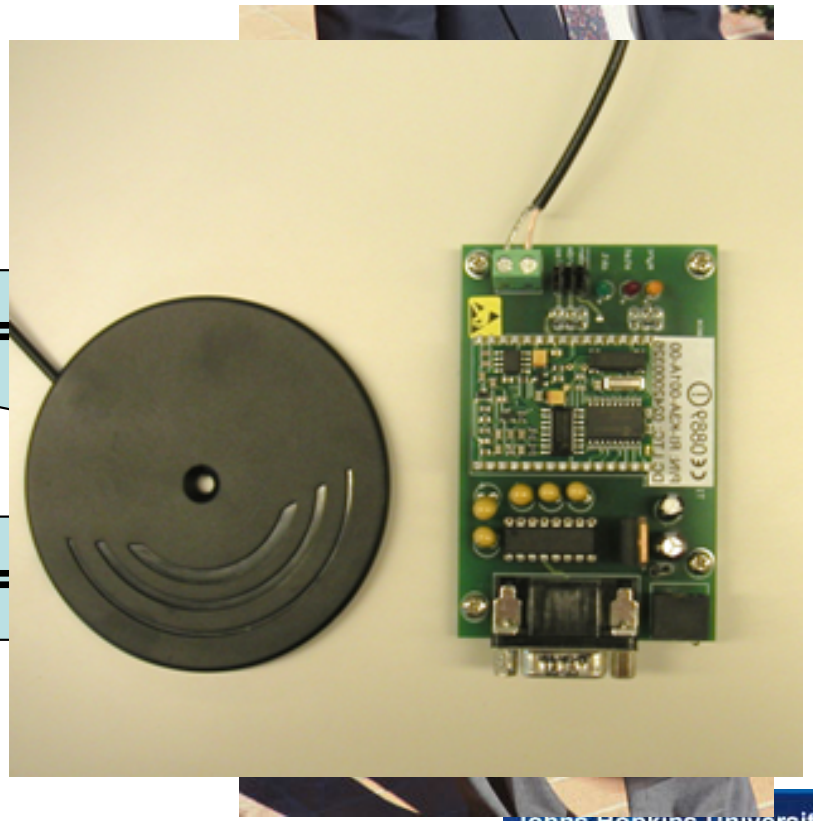
DST Immobilizers

Defeating a DST Immobilizer

1. Get response from original key.

Challenge=

Response=



Finding out what  is

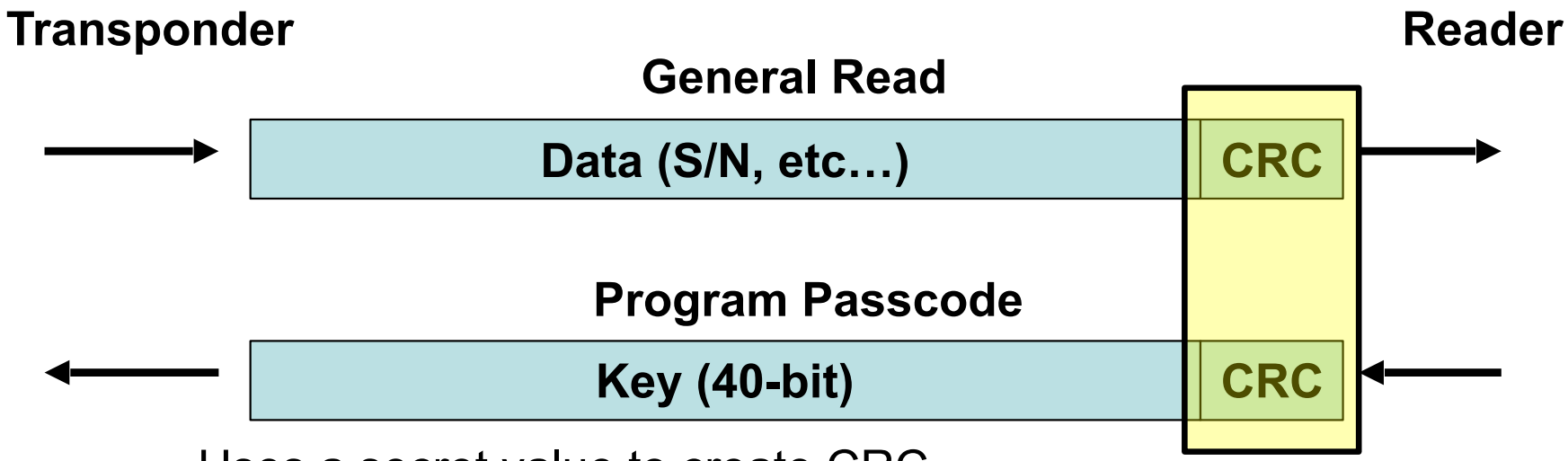
Getting Started

- Purchase TI micro-reader evaluation kit ($400).
  - Reader, antenna.
  - Bunch of DST tags.
  - Lame software.

- Write a better interface.
  - Lets us program in a key, send in challenges.

Security Analysis

Tricky security feature



- Uses a secret value to create CRC.
- All transponders share the same secret value.
- Guessed secret value with computer in < 1 ms.

DST Immobilizers

- Now we can:
  - Program a 40-bit key (“secret code”) into the DST
  - Send it a 40-bit challenge
  - Read back the 24-bit response

Security Analysis

Initial Experiments

Secret code	Challenge	Response
0x0000000000	0x0000000000	0x000000
	0x2222222222	0x222222
	0x5555555555	0x555555
	0x7777777777	0x777777
	0x8888888888	0x888888
	0xAAAAAAAAAA	0xAAAAAA
	0xDDDDDDDDDD	0xDDDDDD
	0xFFFFFFFFFF	0xFFFFFF

Security Analysis

Initial Experiments

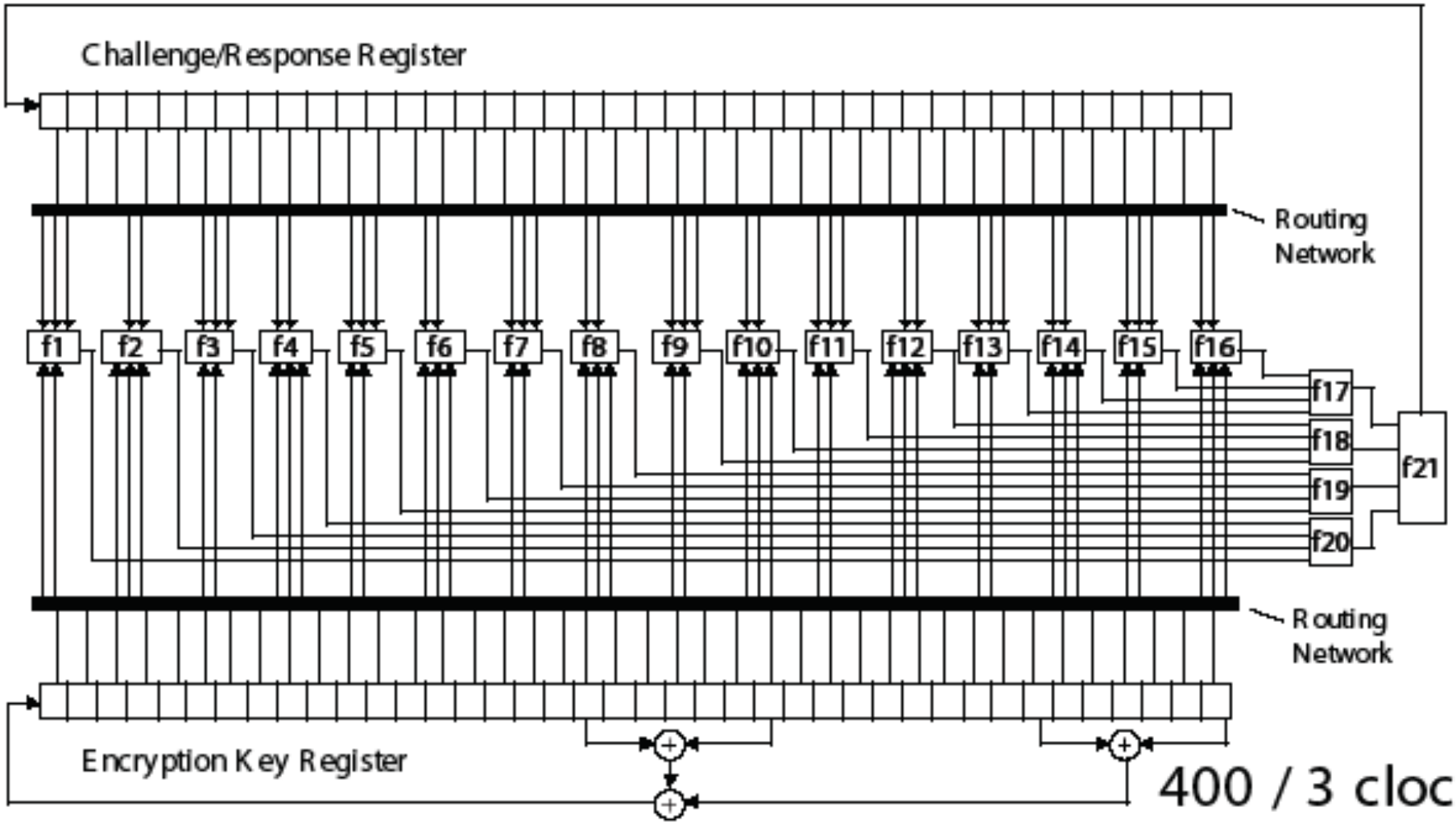
Secret code	Challenge	Response
0x0000000000	0xFFFFFFFFFF	0xFFFFFF
	0xFFFFFFFFFD	0xFFFFFF
	0xFFFFFFFDFF	0xFFFFFF
	0xFFFFFDFFFF	0xFFFFFD
	0xFFFDFFFFFF	0xFFFDFF
	0xFDFFFFFFFF	0xFDFFFF

Security Analysis

Initial Experiments

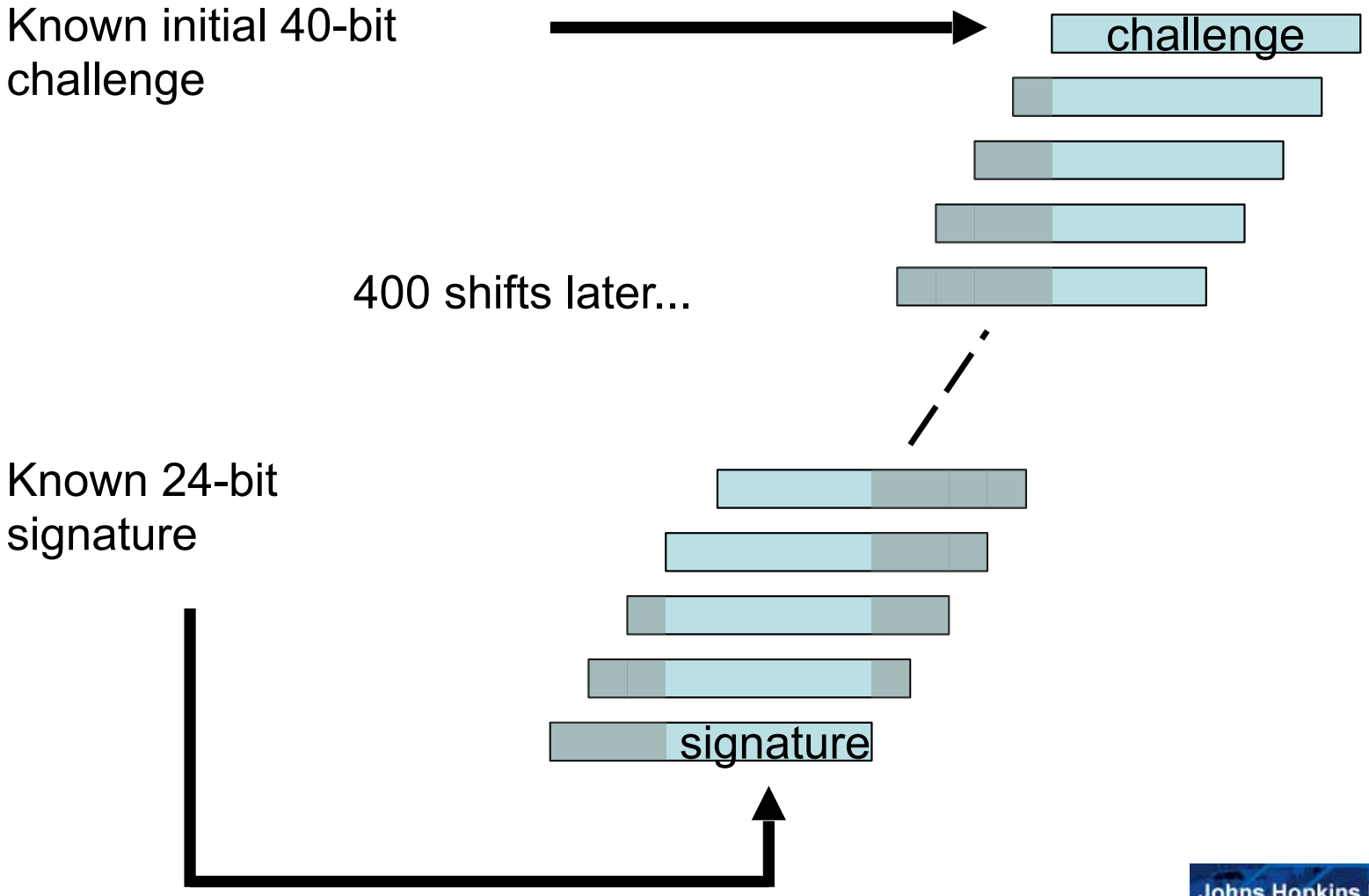
Secret code	Challenge	Response
0x0000000000	0xFFFFFFFFFF	0xFFFFFFFFFF
	0xFFFFFFFFFD	0xFFFFFFFFFD
	0xFFFFFFFDFF	0xFFFFFFFDFF
	0xFFFFFDFFFF	0xFFFFFDFFFF
	0xFFFDFFFFFF	0xFFFDFFFFFF
	0xFDFFFFFFFF	0xFDFFFFFFFF

400 clocks ➔ 10 rounds



Digital Signature DST40 Algorithm implementation

Walking Backwards

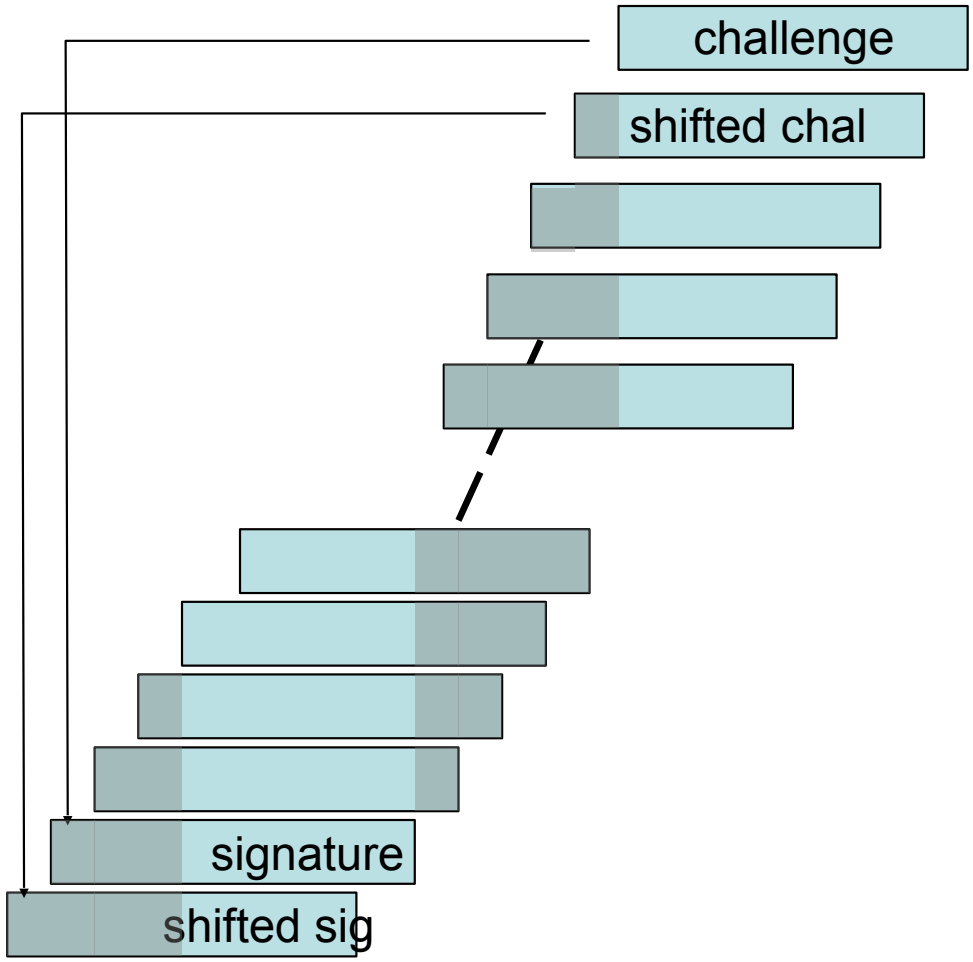


Walking Backwards

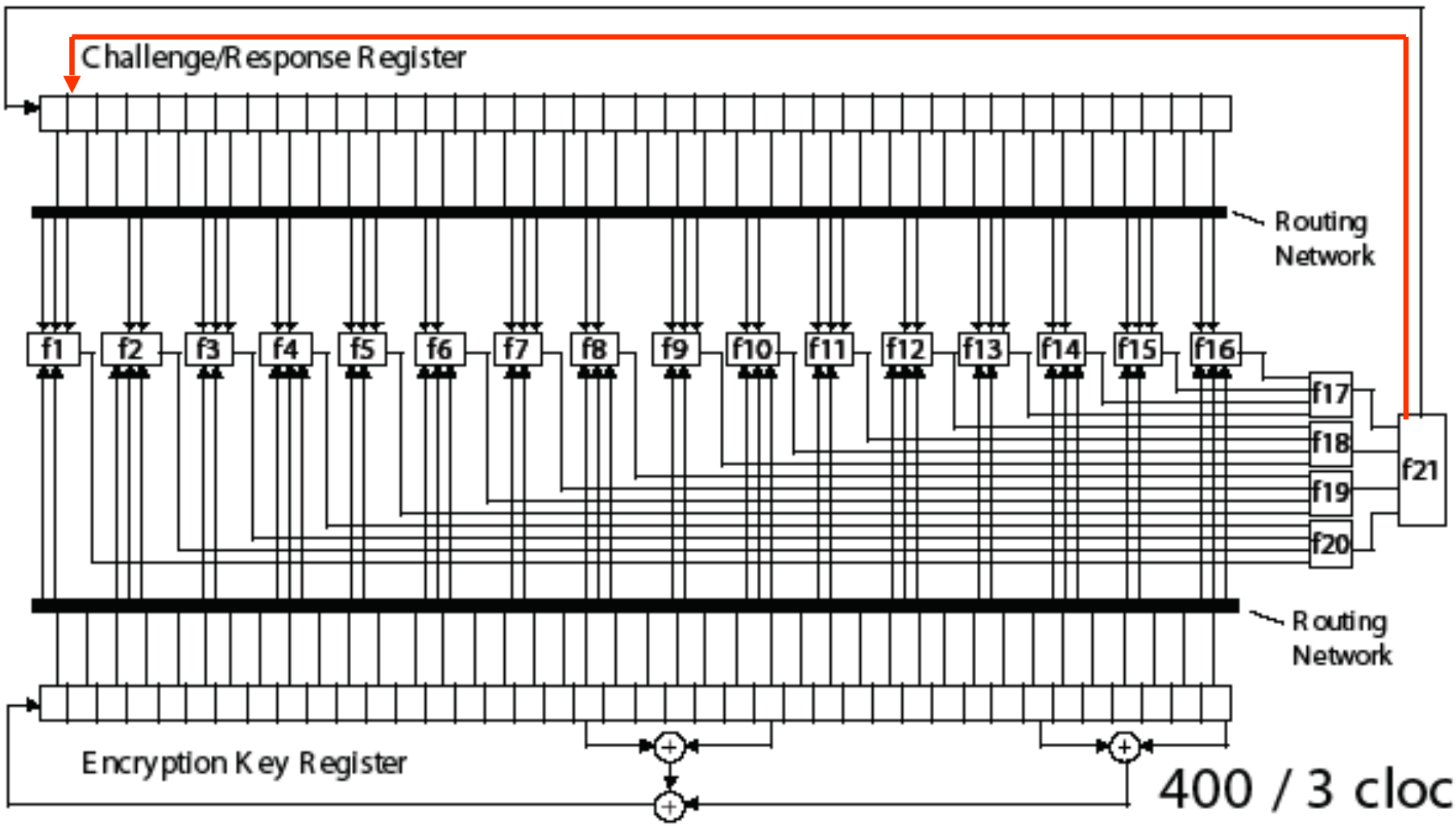
Guess next challenge

See if it yields next signature

Repeat...

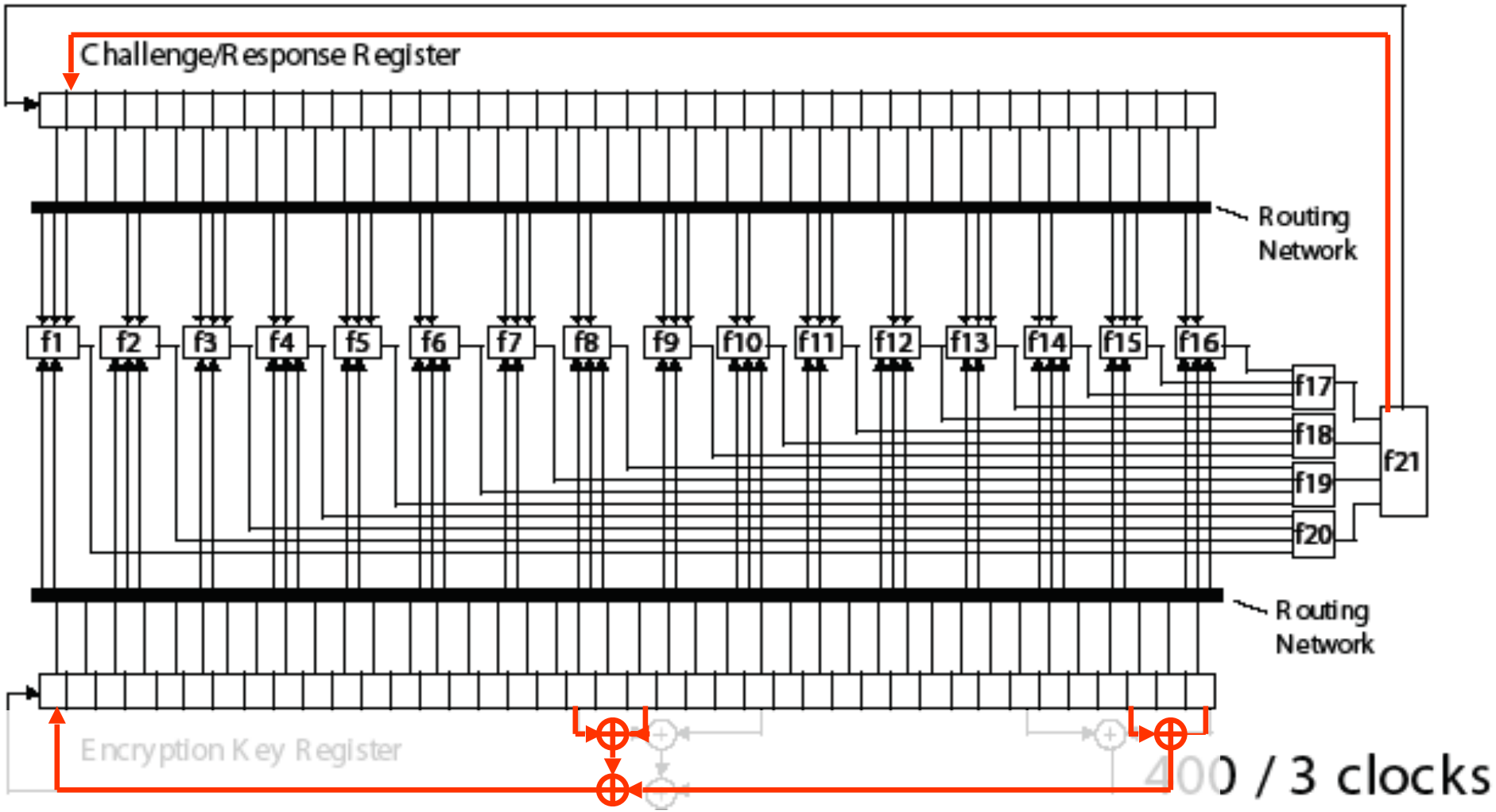


400 clocks ➔ 10 rounds



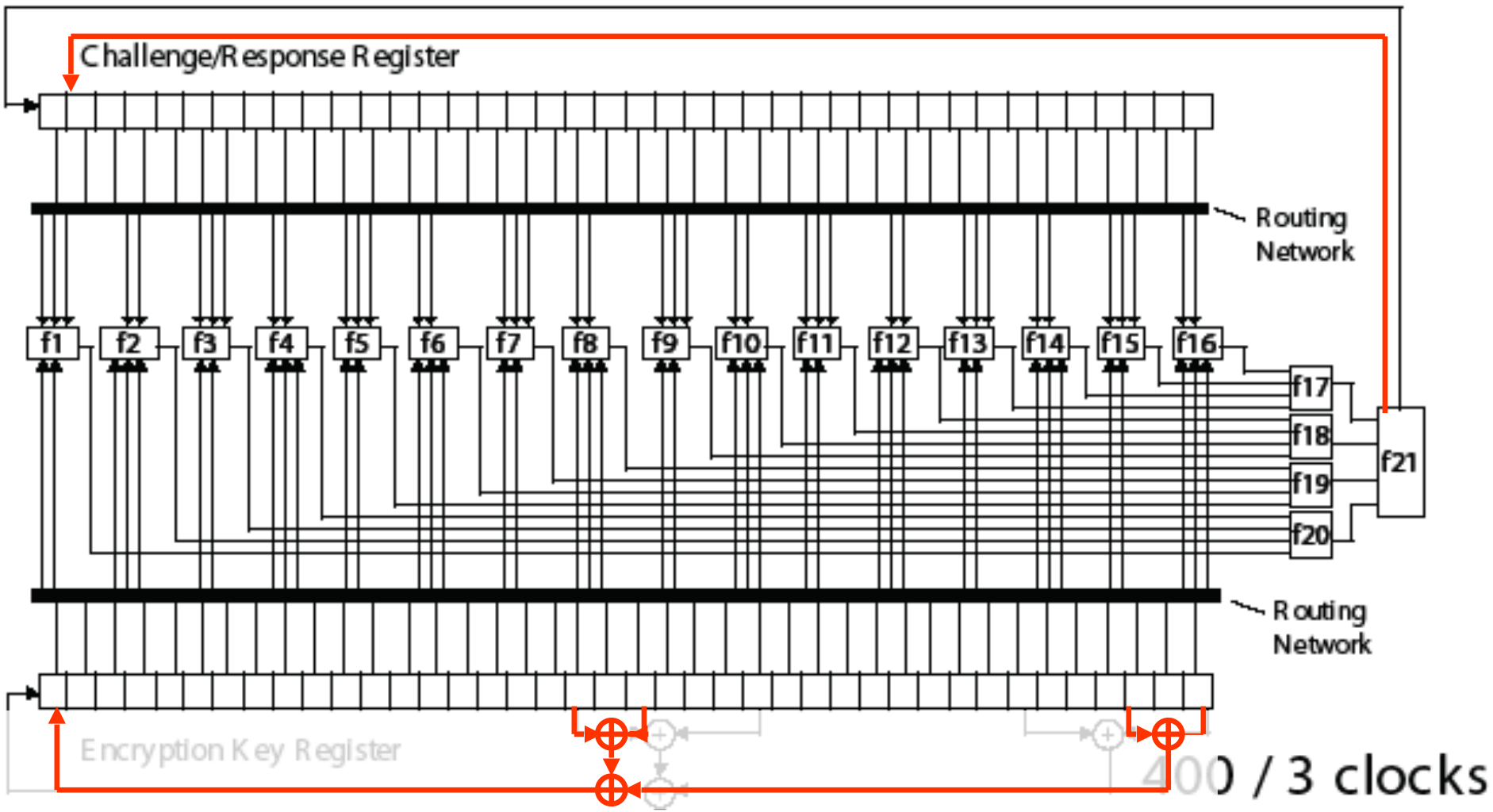
Digital Signature DST40 Algorithm implementation

400 clocks → 10 rounds



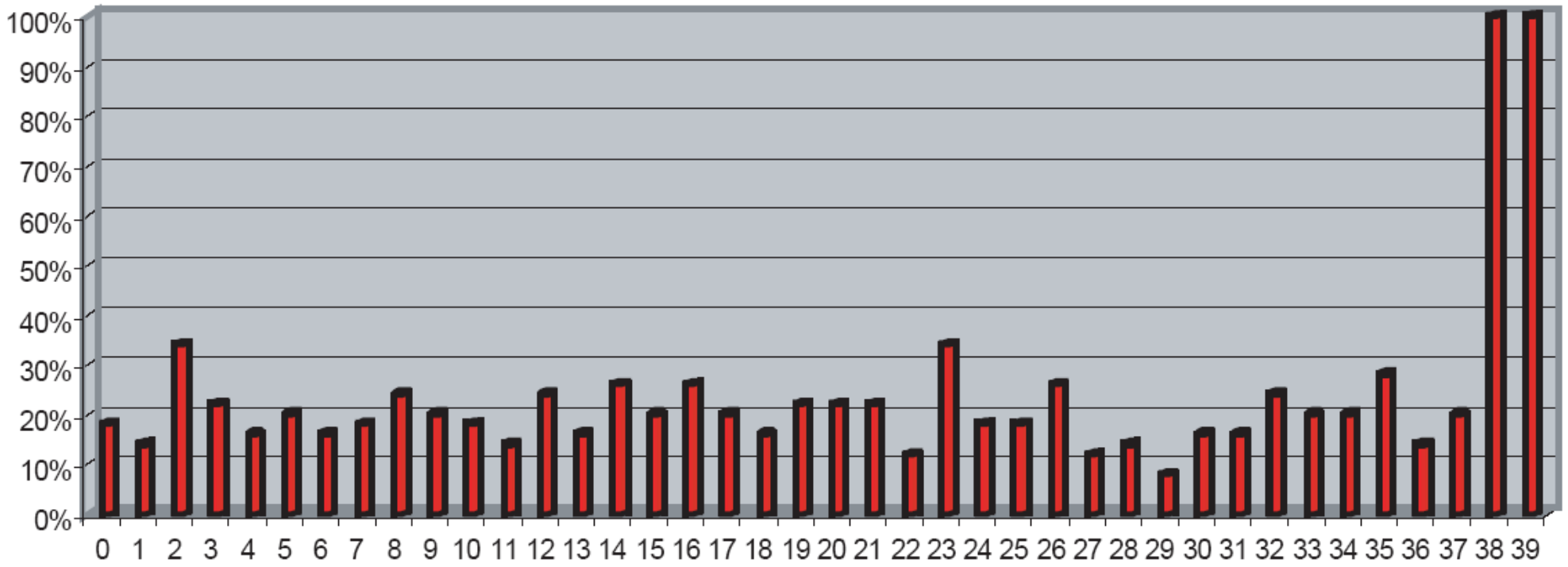
Digital Signature DST40 Algorithm implementation

200 ~~400~~ clocks → 10 rounds

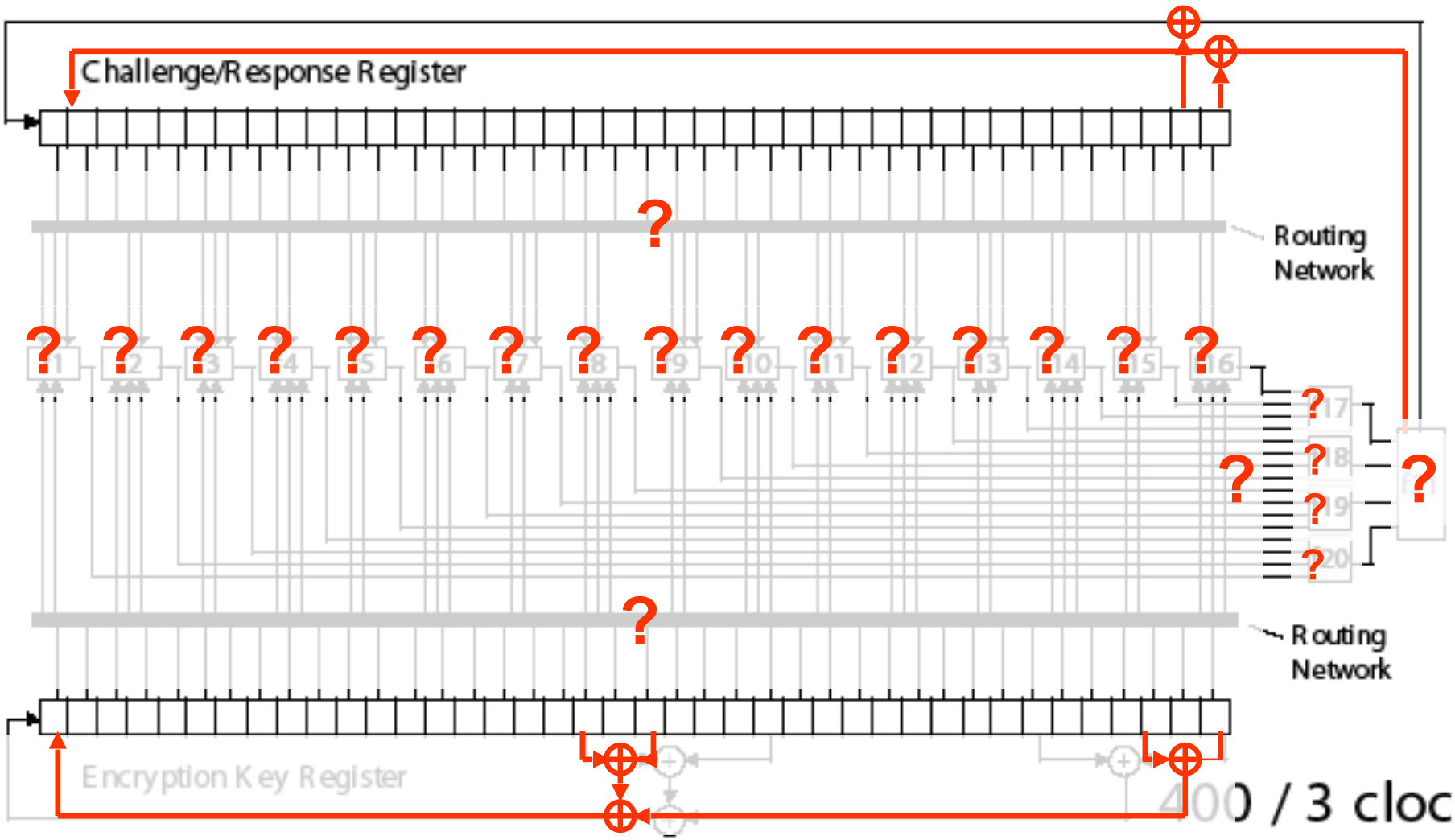


Digital Signature DST40 Algorithm implementation

Single round output tests

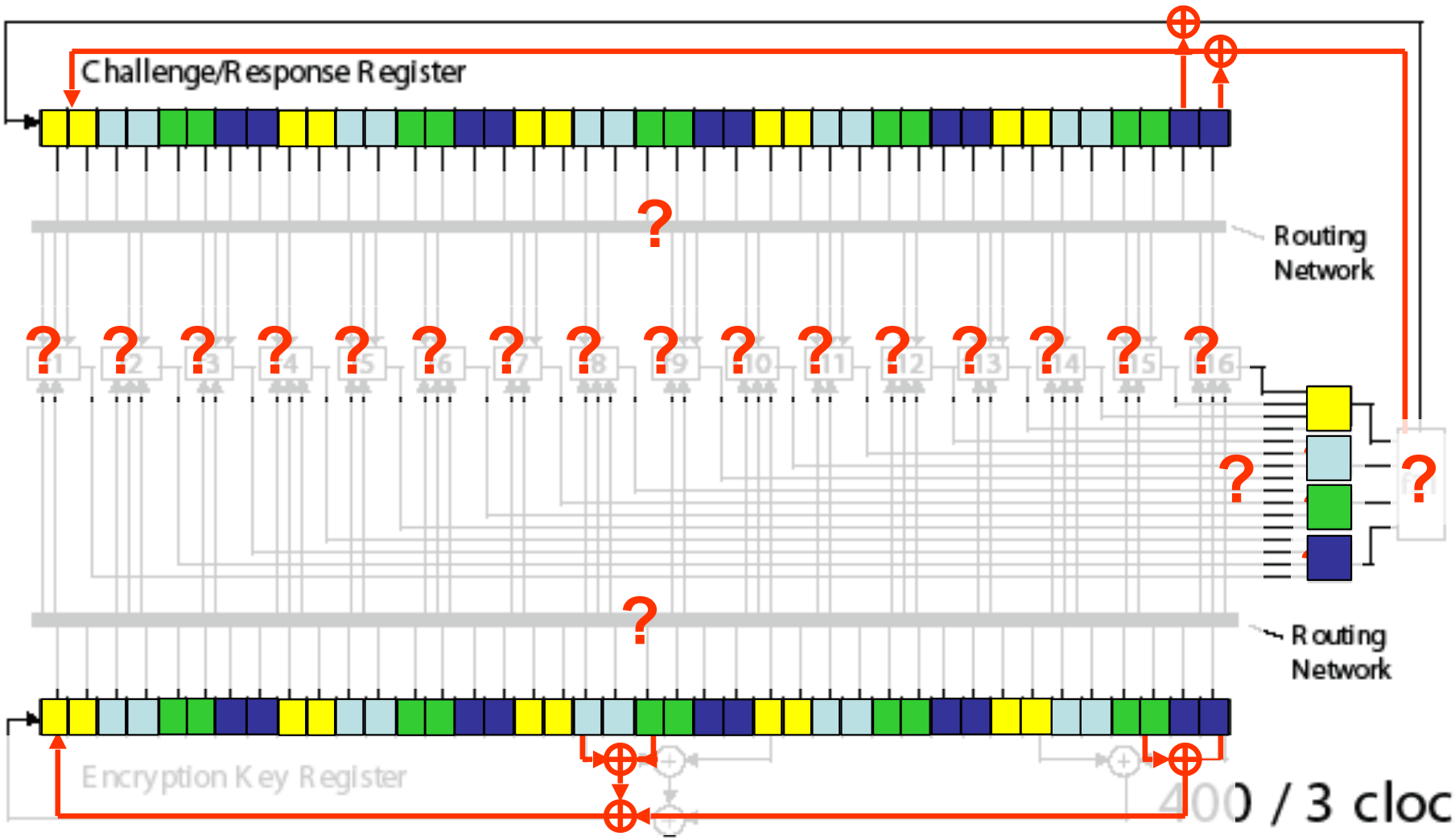


400 clocks ➔ 10 rounds



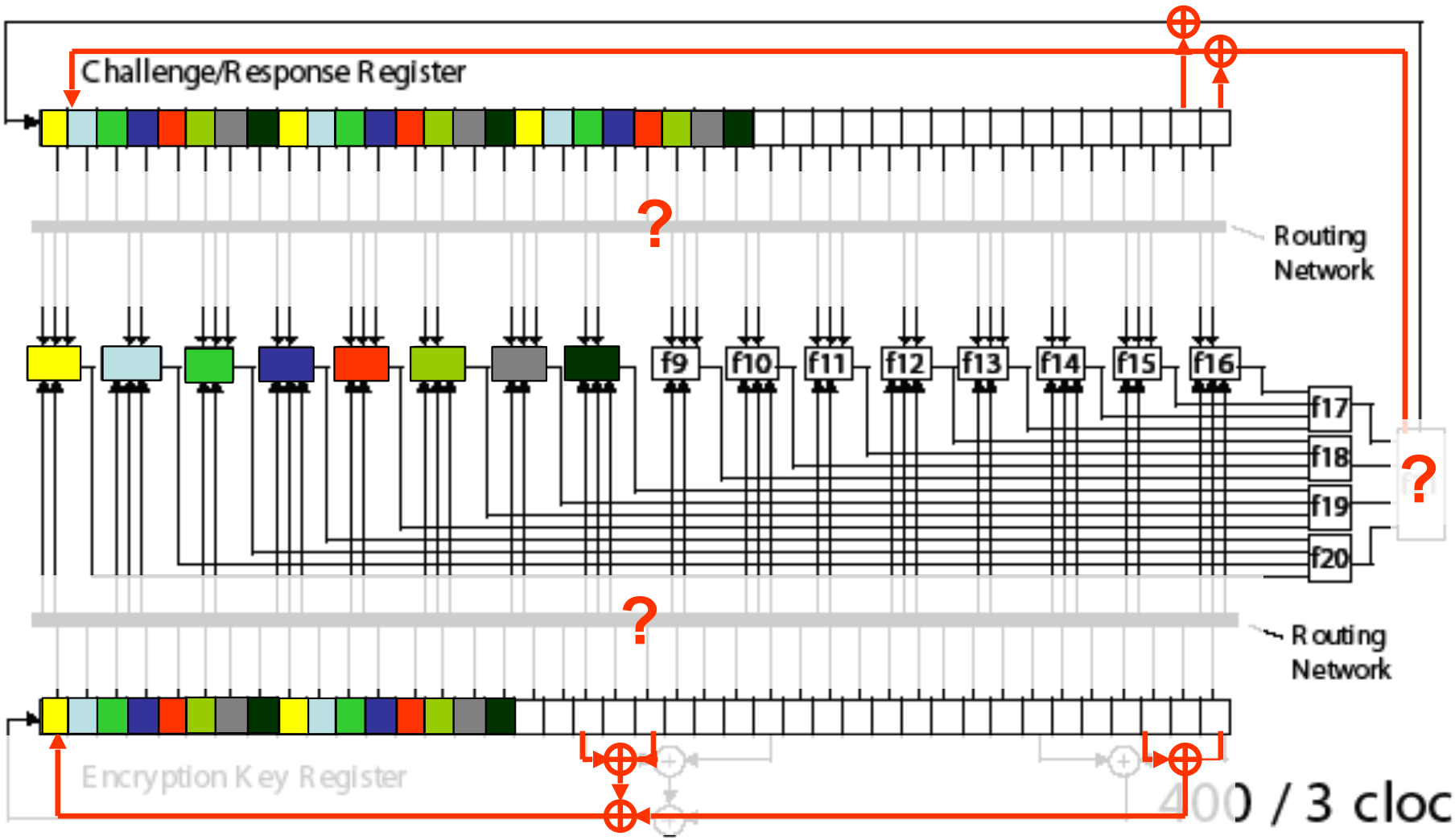
Digital Signature DST40 Algorithm implementation

400 clocks ➔ 10 rounds



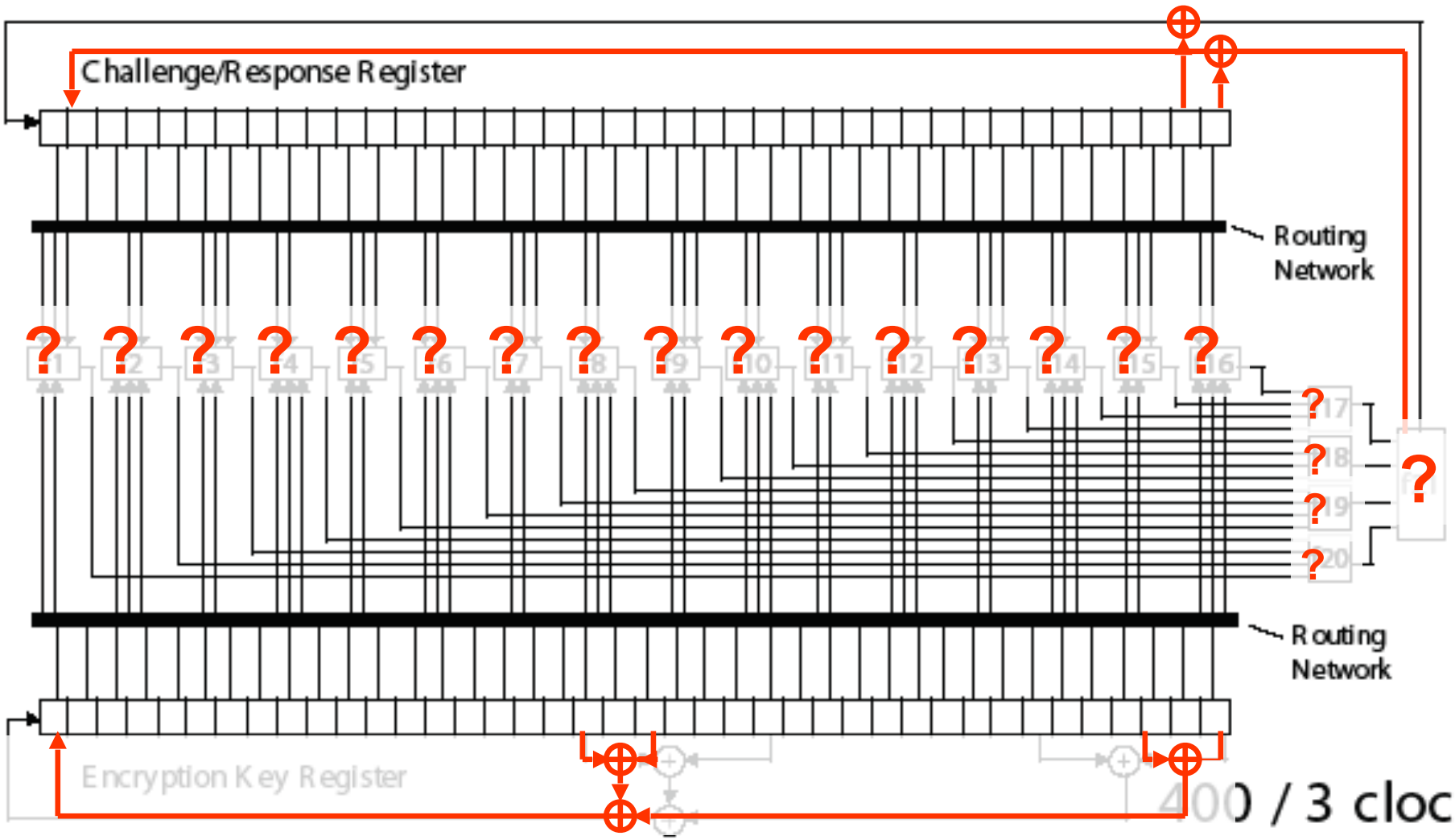
Digital Signature DST40 Algorithm implementation

400 clocks ➔ 10 rounds



Digital Signature DST40 Algorithm implementation

400 clocks ➔ 10 rounds



Digital Signature DST40 Algorithm implementation

Security Analysis

We have the cipher, what now?

- Make a software version.
- Guess a secret key
  - 40-bit secret keys are not very strong
    - 1,099,511,627,776 possible secrets.
  - Brute-force attack
    - Send a Speedpass a challenge, record the response.
    - Encrypt challenge with all possible secrets.
    - Find which one produces the correct response.

Security Analysis

How fast can you guess?

- Software is slow
  - 200,000 encryptions / sec.
  - On average, takes 31 days.

- Hardware is fast
  - Commercially available FPGA ($200).
  - 16 million encryptions / sec.
  - On average, takes 9 hours.

Security Analysis

How fast can you guess?

- More FPGAs
  - 16 x 16 million encryptions / sec.
  - On average, takes 35 minutes.
  - More $$$ = Faster

Security Analysis

How fast can you guess?

- Huge storage table
  - RAID array storage system.
  - 5,000 Gigabytes.
  - Expensive ($10-15k).
  - On average, takes < 1 s.

Security Analysis

How fast can you guess?

- Time/Memory Tradeoff
  - Best of both worlds.
  - Inexpensive (<$1000).
  - On average, takes < 1 minute.
  - Very portable.

+

Real World Testing

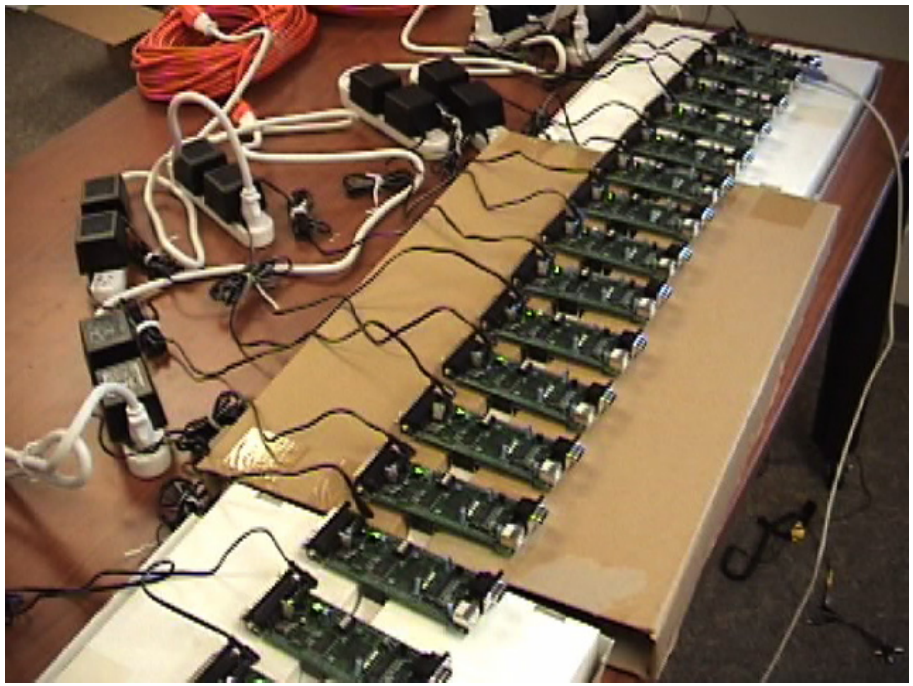
Scanning a Victim (http://rfidanalysis.org)

Equipment: TI evaluation kit, laptop

Real World Testing

Extracting the secret passcodes

- 16 FPGAs, average time 35 min.
- Cracked Speedpasses and Immobilizer chips.



“The Mobilizer”

Emulating a real transponder

Big, Bulky Prototype.

Small PC ($1000).
DAC Board ($1000).
UPS ($300).
Eval kit antenna ($50).
Custom software (Free).

Real World Testing

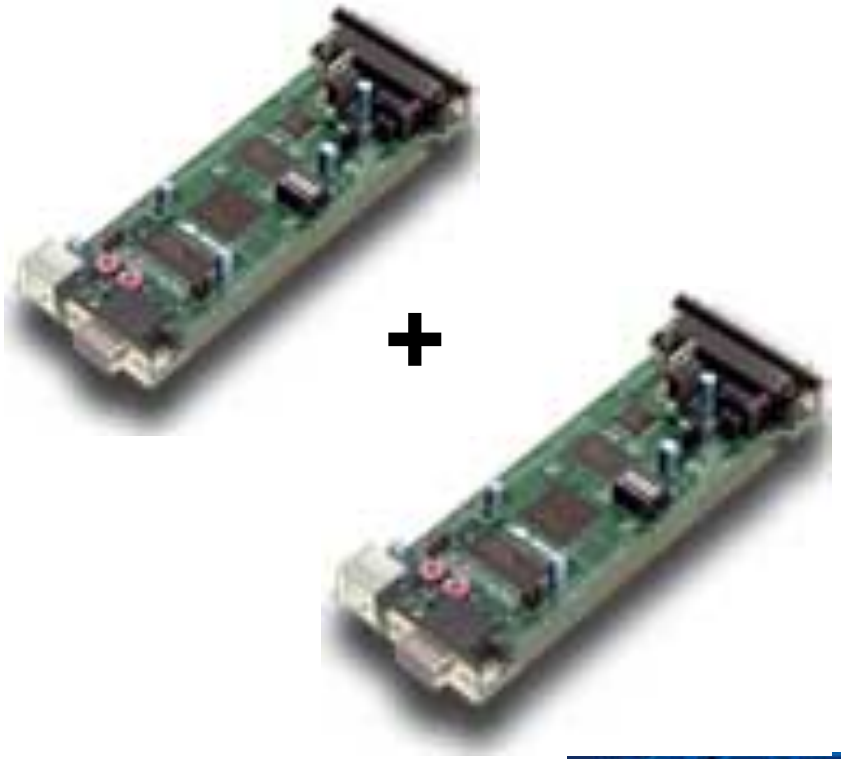
Emulating a real transponder

A 1st Generation Device (not actually built).

FPAA ($200).

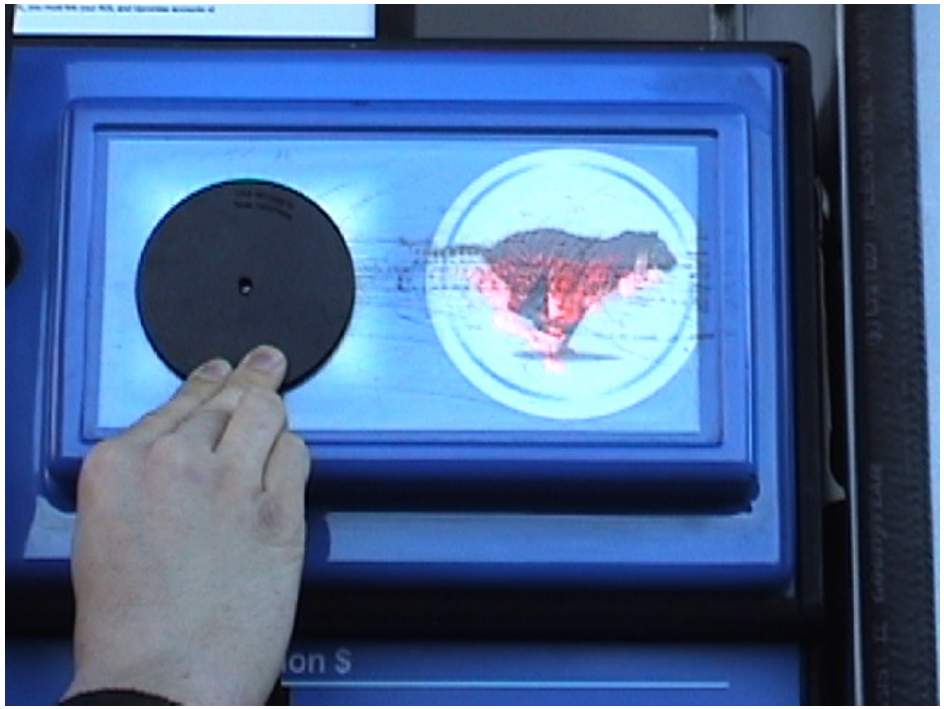
FPGA ($200).

Homemade Antenna ($0).



Real World Testing

Field Tests: (http://rfidanalysis.org)

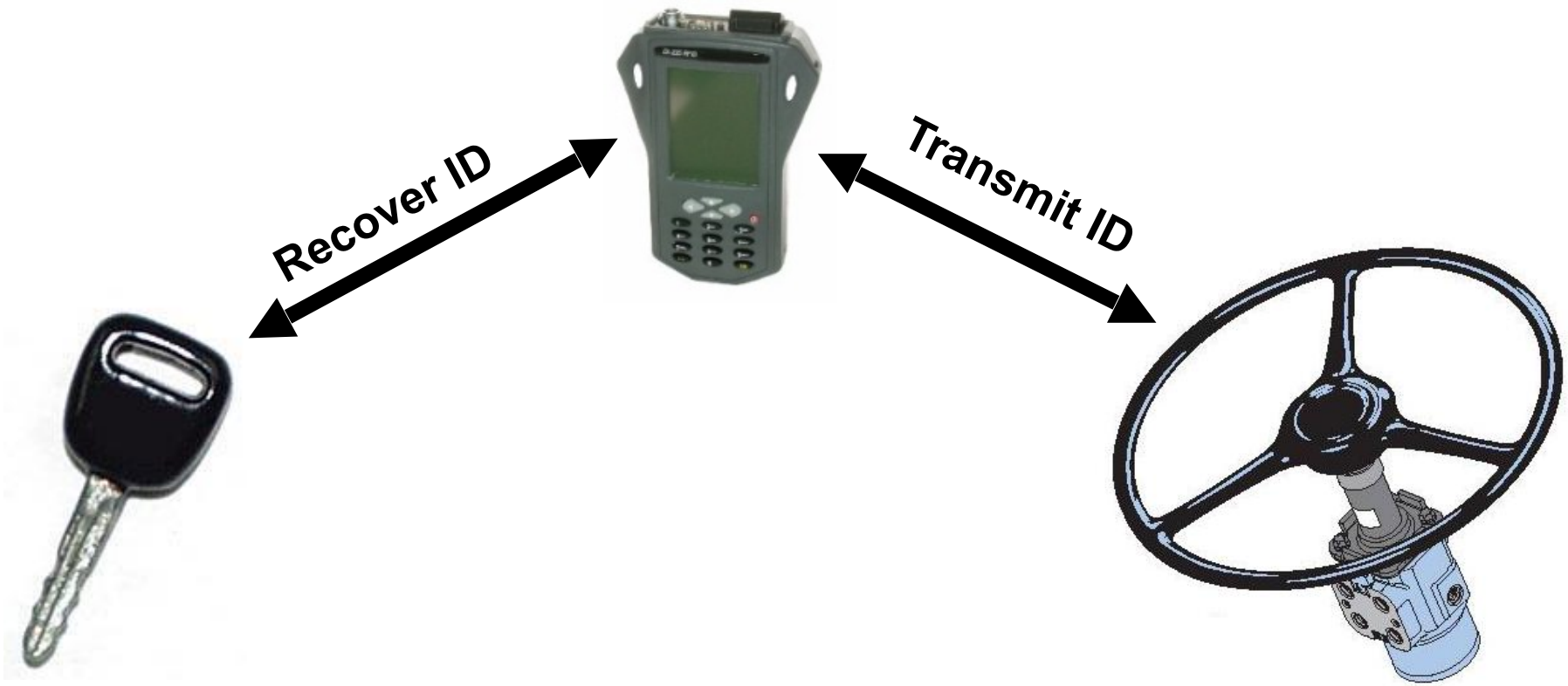



abc 2
INVESTIGATORS
ON THE
Tisha
THOMPSON
abc 2
THE INVESTIGATORS 11:14 52°

More practical attacks

Making this easier.

A device that does everything for you...

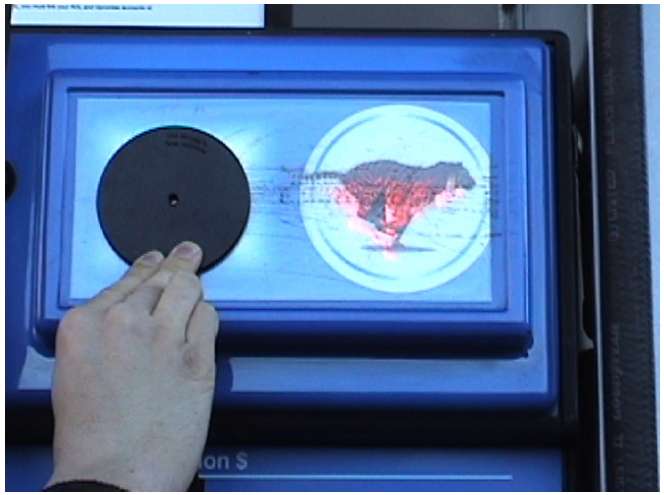


Speedpass

Making this easier.

Copied tag can be used anywhere.

Charges directly to victim's credit card.



Johns Hopkins University
Information Security
Institute

Fixing the problem

Short-Term Fixes

- Very few
- Systems too widely deployed for simple upgrade.
- Tin foil works.
- Diligence on the part of the consumer.

Fixing the problem

Long term Fixes

- **Use standard encryption algorithms.**
  - **AES, HMAC-SHA1, 3DES**
  - **No security through obscurity.**
- No single-tag compromise should compromise the whole system.
  - As with the secret checksum values.
- Use longer key lengths.
  - If that is not possible, understand this limitation!

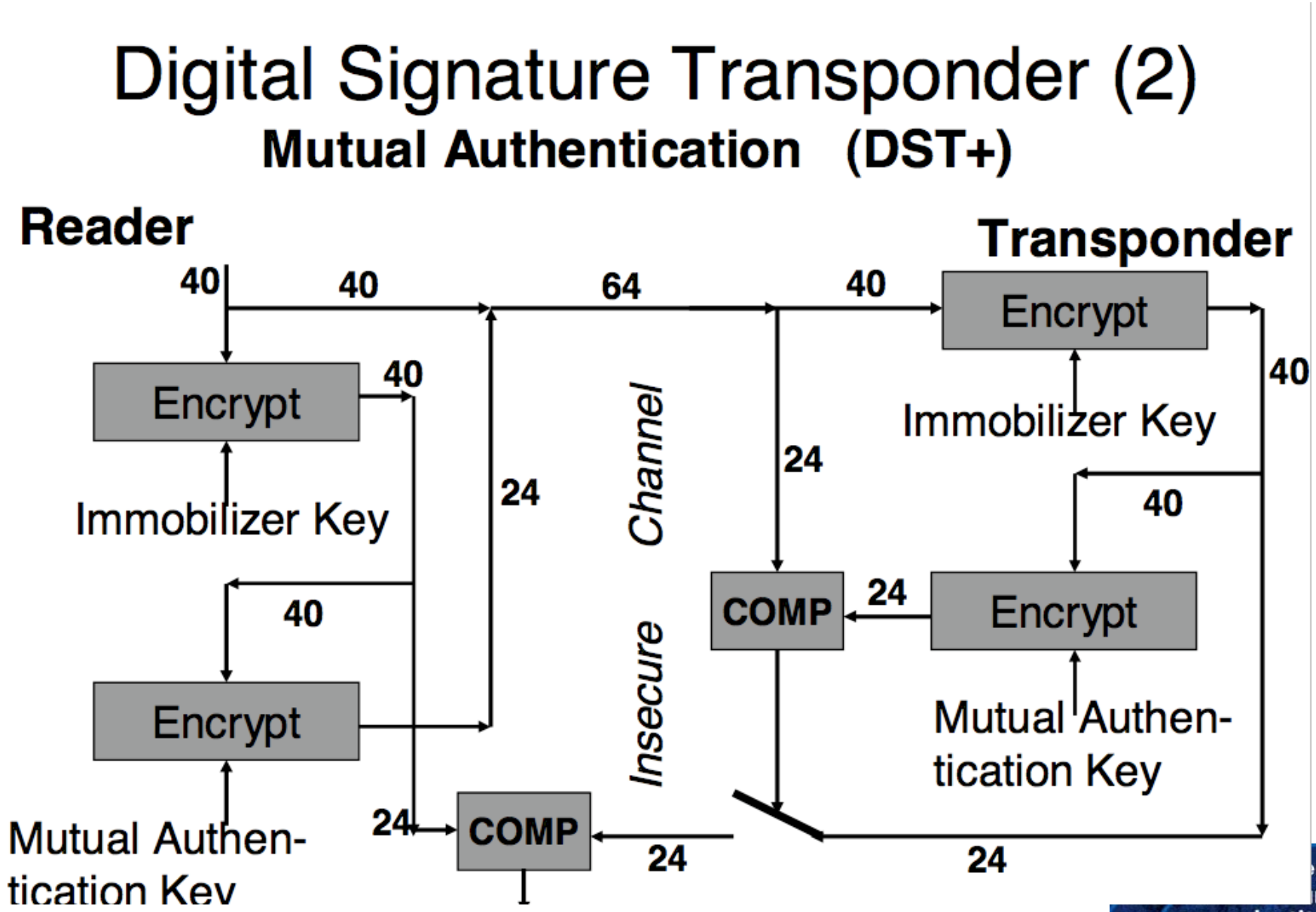
Conclusions

- Widely deployed systems offer no, or limited security
  - Solutions on the way, however
- Privacy protection (tracking) not considered
- Attacks *are* practical-- RF interface can't even stop computer scientists!

The Paper

- S. Bono, M. Green, A. Stubblefield, A. Rubin, A. Juels, M. Szydlo. "Analysis of a Cryptographically-Enabled RFID Device", Usenix Security 2005

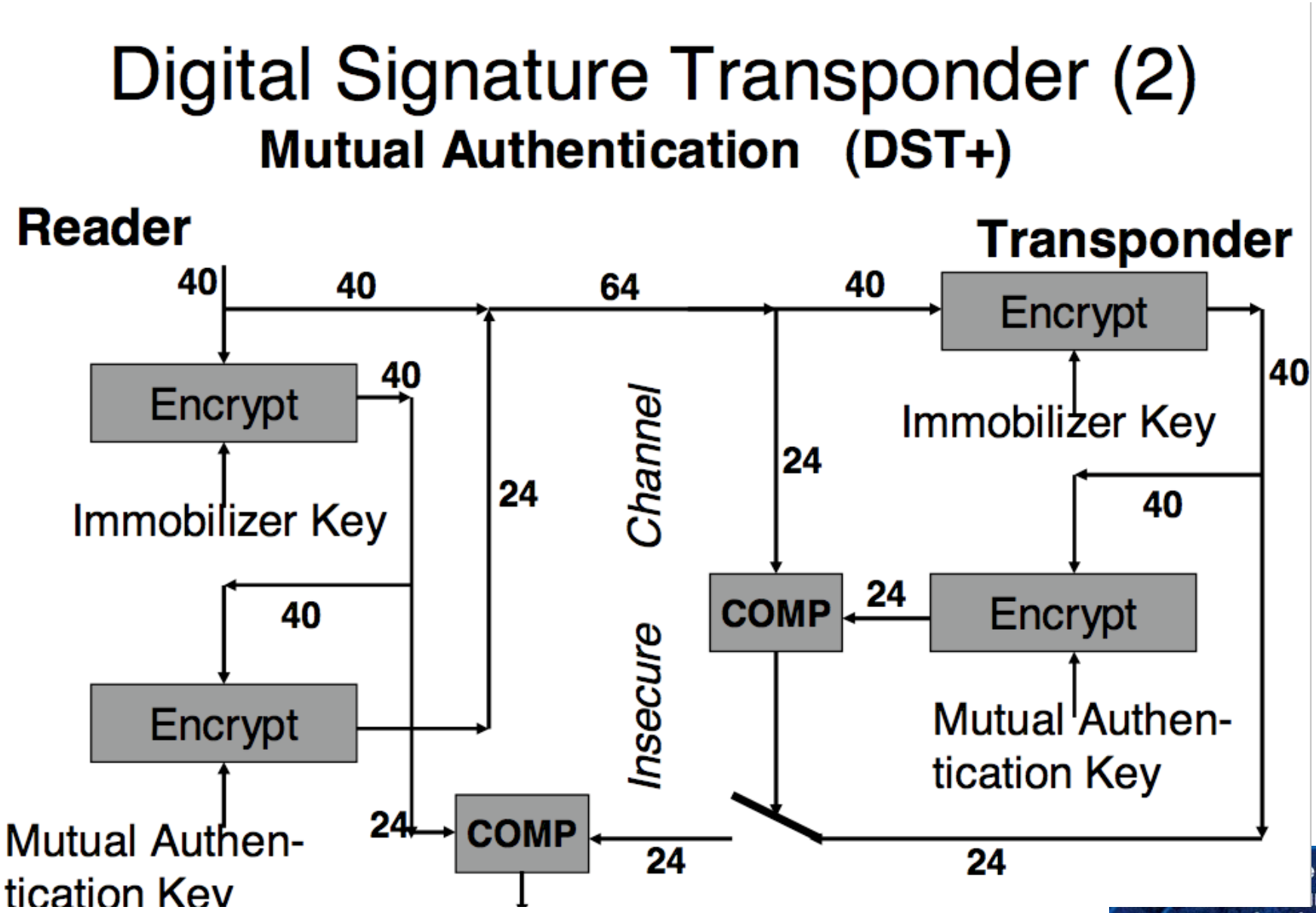
The DST+
(or “How not to fix the problem”)



The DST+

- Mutual Authentication
  - Make the *reader* authenticate itself to the tag
  - Stops attackers from gathering challenge/ responses (in theory)
  - Prevents tracking attacks
- Double Encryption (two separate keys)
  - Encrypting twice must be twice as good

The DST+


Digital Signature Transponder (2)
Mutual Authentication (DST+)
Reader
Transponder
40
40
64
40
Encrypt
40
Immobilizer Key
24
Channel
Insecure
24
Immobilizer Key
40
40
COMP
24
Encrypt
Encrypt
Mutual Authen-
tication Key
Mutual Authen-
tication Key
24
COMP
24
24