Department of Computer Science
National Tsing Hua University

CS 5244: Introduction to Cyber Physical Systems

Unit 22: Security

Instructor: Cheng-Hsin Hsu

Acknowledgement: The instructor thanks Profs. Edward A. Lee & Sanjit A. Seshia at UC Berkeley for sharing their course materials

What is Security?

What's different:

New kinds of functions

Worst-case adversarial conditions

(compare with:

Reliability = the fraction of time that a system performs its specified function for a specified period of time under stated operating conditions)

What is Security?

Secrecy/Privacy

Can secret data be leaked to an attacker?

Integrity

Can the system be modified by the attacker?

Availability

Is the system always able to perform its function?

(Is “denial-of-service” possible?)

About this Lecture

Security is increasingly a major concern for embedded systems designers

→ Voiced by representatives from GM, Boeing, and United Technologies in recent workshop in St. Louis

Need to know about the security pitfalls in design & implementation of embedded systems

Take CS 161 to learn about computer security in general.

Main topic discussed today

Analysis of security properties of an
Implantable Cardioverter Defibrillator (ICD)

An ICD is one kind of IMD
IMD = Implantable Medical Device

IMDs are used to monitor chronic disorders and treat patients with automatic therapies

Reference

This lecture is based on the following article that appeared in May 2008:

"Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses", Daniel Halperin, Thomas S. Heydt-Benjamin, Benjamin Ransford, Shane S. Clark, Benessa Defend, Will Morgan, Kevin Fu, Tadayoshi Kohno, William H. Maisel, IEEE Symposium on Security and Privacy, May 2008.

Warning (adapted from CS 161)

This lecture discusses vulnerabilities in IMDs. This is *not* intended as an invitation to go exploit those vulnerabilities. It is important that we be able to discuss real-world experience candidly, and s*tudents are expected to behave responsibly*.

Berkeley policy is very clear: you may not break into machines that are not your own; you may not attempt to attack or subvert system security. Breaking into other people's systems is inappropriate, and the existence of a security hole is no excuse.

Unethical or inappropriate actions may result in failing the course and being referred for further disciplinary action.

What an ICD does

Pacing

Periodically send a small electrical stimulus to the heart

Defibrillation

Occasionally send a larger shock to restore normal heart rhythm

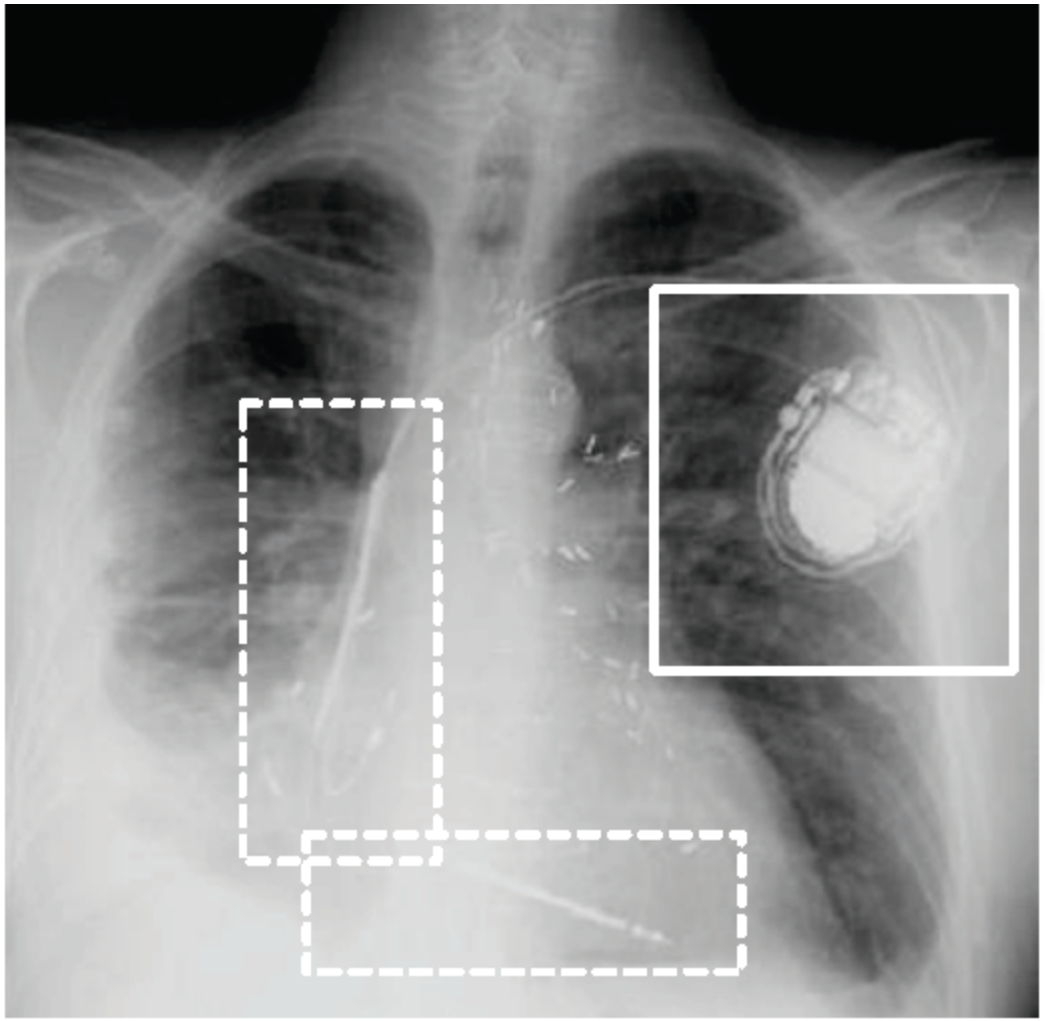


Fig. 1. Chest xray image of an implanted ICD (top right, near shoulder, solid outline) and electrical leads connected to heart chambers (center of rib cage, dotted outline).

The ICD Programmer

The “programmer” is a device intended to be used to:

- perform diagnostics
- read and write private (patient) data
- adjust therapy settings

on the ICD.

Programmer communicates with ICD wirelessly.

- typically 175 kHz short-range communication

Analysis done in the Paper

Considered attacks on ICD security by three classes of attackers:

- Attacker possessing an ICD programmer
- Attacker who simply eavesdrops on communications between an ICD and the programmer, using commodity software radio
- Attacker who eavesdrops as well as generates arbitrary RF traffic to the ICD, possibly spoofing an ICD programmer.

Demonstrated that successful attacks are possible under all three classes!

Results of Experiments by Halperin et al.

	Commercial programmer	Software radio eavesdropper	Software radio programmer	Primary risk
Determine whether patient has an ICD	✔	✔	✔	Privacy
Determine what kind of ICD patient has	✔	✔	✔	Privacy
Determine ID (serial #) of ICD	✔	✔	✔	Privacy
Obtain private telemetry data from ICD	✔	✔	✔	Privacy
Obtain private information about patient history	✔	✔	✔	Privacy
Determine identity (name, etc.) of patient	✔	✔	✔	Privacy
Change device settings	✔		✔	Integrity
Change or disable therapies	✔		✔	Integrity
Deliver command shock	✔		✔	Integrity

TABLE I
RESULTS OF EXPERIMENTAL ATTACKS. A CHECK MARK INDICATES A SUCCESSFUL IN VITRO ATTACK.

Communication between ICD & Programmer

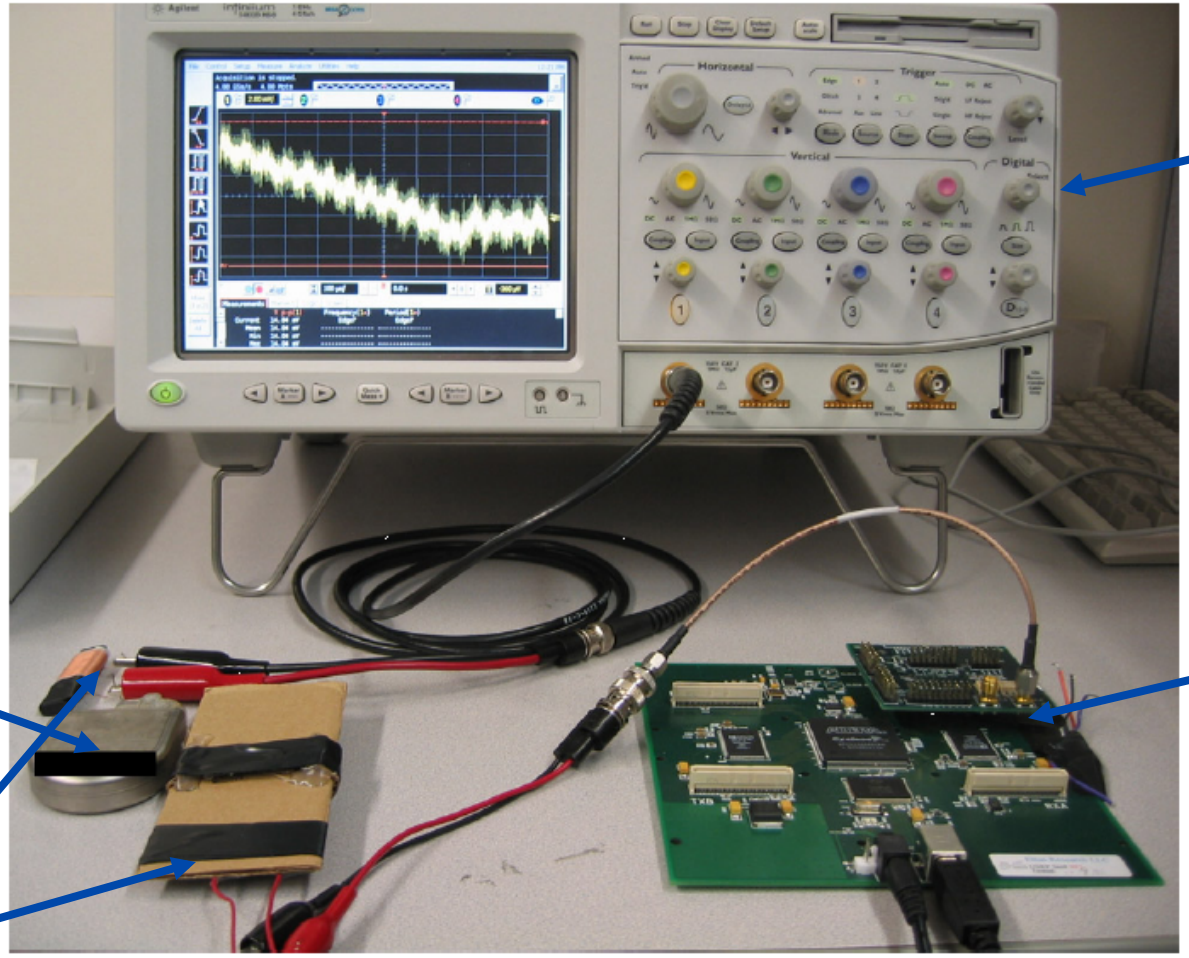
Inside the ICD is a magnetic switch

Programmer “head” has a magnet whose field closes this switch, which causes the ICD to transmit telemetry data, including EKG readings

Main Concepts to Discuss

- Dangers of sending messages in cleartext
  - How protocols can be reverse engineered
- Power matters
  - Use of “Zero-power” protocols
- Security fundamentals
  - Encryption, nonces, Trusted Computing Base (TCB)

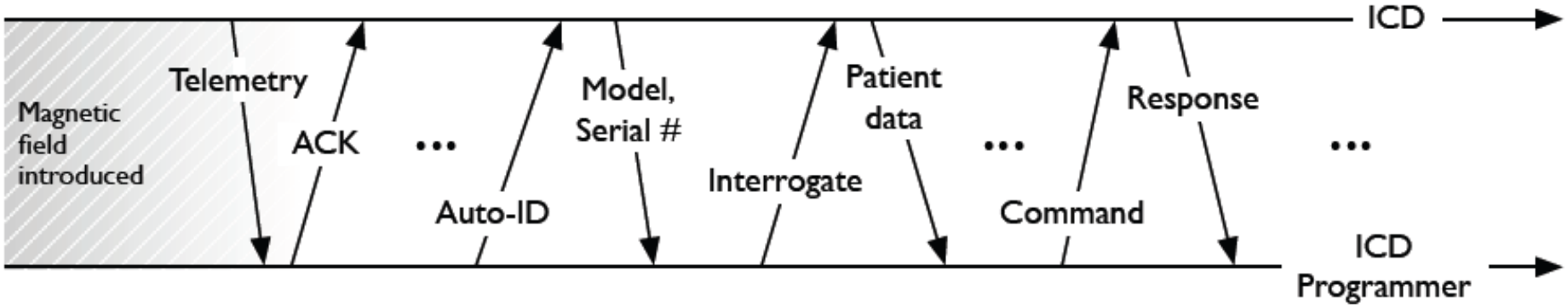
Equipment Used



Methodology to Reverse Engineeer Protocol

- Connected antenna to recording device (oscilloscope, USRP – universal s/w radio peripheral)
- Placed antenna within cms of ICD/programmer
- Demodulated the RF signals picked up – used spectral analysis to determine modulation technique (e.g., 2-FSK for programmer)
- Decoded demodulated output – by transmitting a known message like AAAA – and trying different well-known encoding schemes (e.g., Non-Return-to-Zero-Inverted – NRZI)

Transaction Timeline of the Protocol



This timeline can be followed in a “replay attack”

Replay Attacks using a Commodity Software Radio

- Triggering ICD identification
- Disclosing patient data (name, diagnosis, etc.)
- Disclosing cardiac data
- Changing patient name
- Setting the ICD's clock
- Changing therapies (e.g. disabling therapies)
- Inducing fibrillation (using a feature to test the device)
- Safeguards can be bypassed (because they are built into the commercial programmer)
- Power denial of service attack: make ICD keep transmitting

Proposed Defenses: Goals

1. Prevent/Deter attacks using both custom equipment and commercial programmer devices
2. Security should draw no power from the primary battery of the ICD
3. Security-sensitive events should be effortlessly detectable by the patient
4. New security mechanisms should not introduce new failure modes

→ Qn. What about a ‘traditional’ security approach that encrypts all data using a key that is embedded into the ICD at manufacture time?

Proposed Defenses: Goals

1. Prevent/Deter attacks using both custom equipment and commercial programmer devices
2. Security should draw no power from the primary battery of the ICD
3. Security-sensitive events should be effortlessly detectable by the patient
4. New security mechanisms should not introduce new failure modes

Solutions: “Zero-Power” Techniques for Notification, Authentication, and Key-Exchange

Wireless Identification and Sensing Platform (WISPer) for Notification

Goal: Cause the ICD to beep when programmer initiates RF communication

Approach: When WISPer gets requests from RFID reader (programmer), it 'chirps' to alert patient

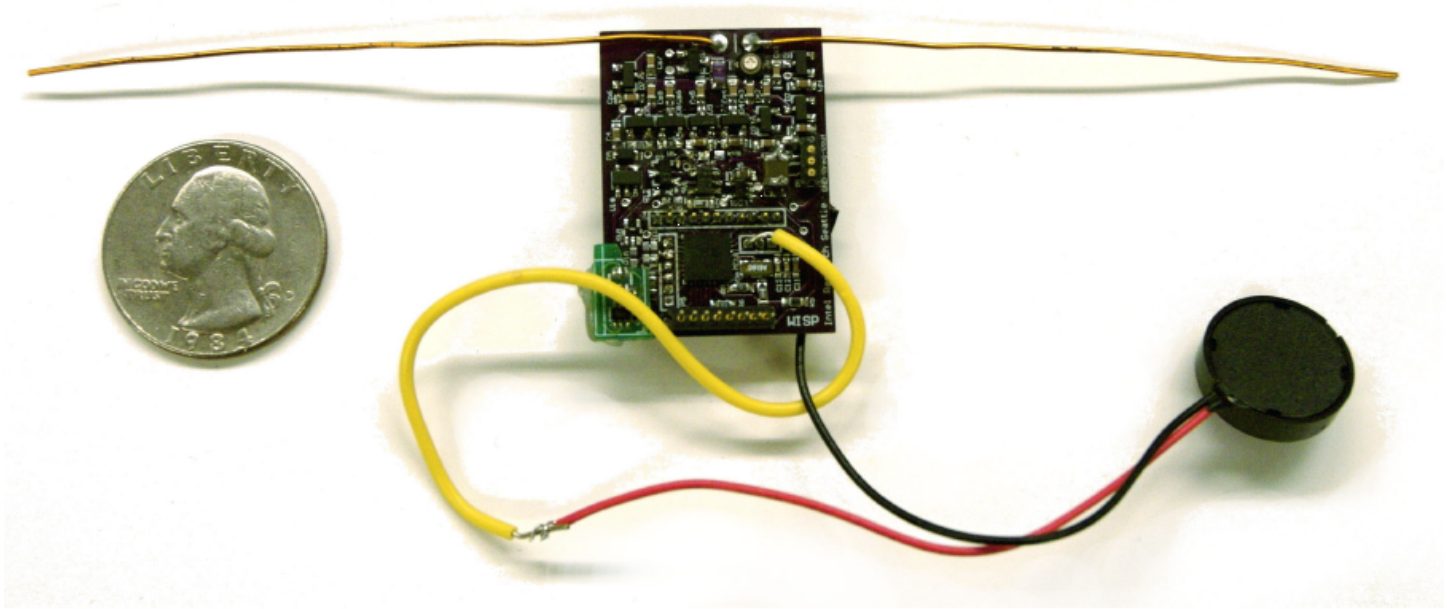
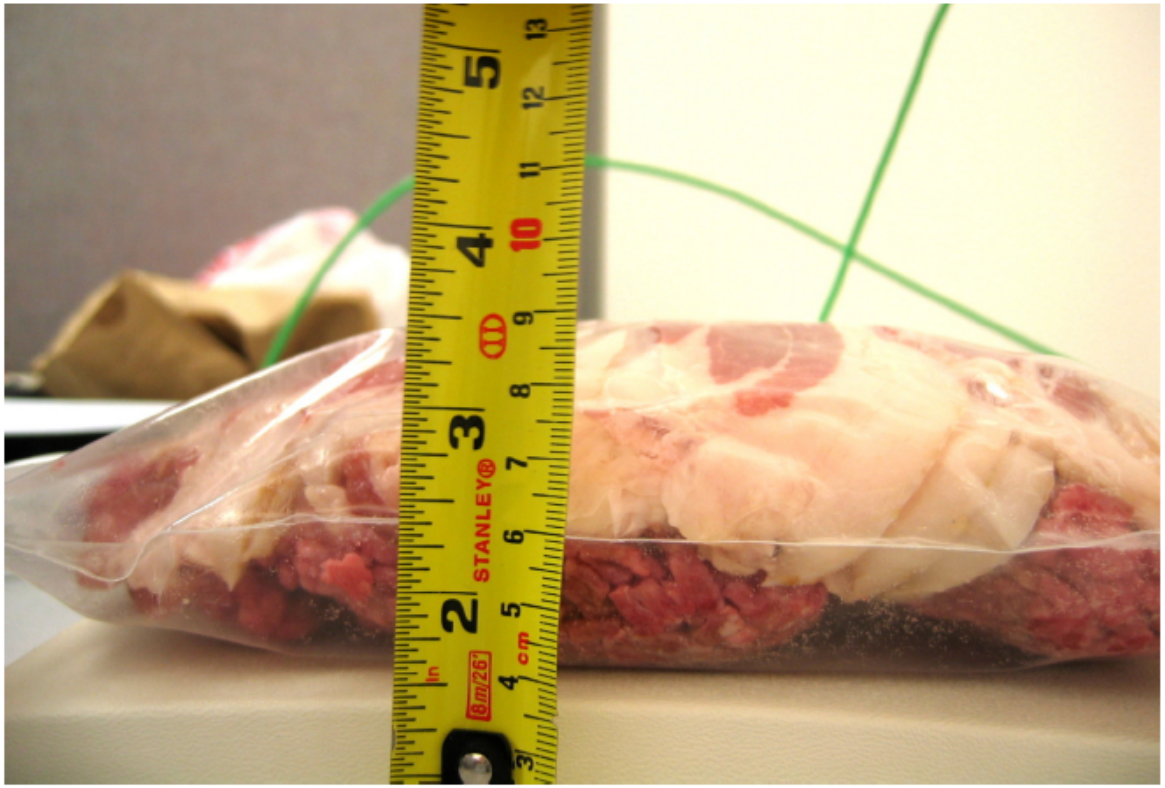


Fig. 7. The WISP with attached piezo-element.

Notification works even through “meat”

Max buzzing volume between 60 dB (normal conversation) and 70 dB (vacuum cleaner at 3m)

Should be audible when implanted



Zero-Power Authentication

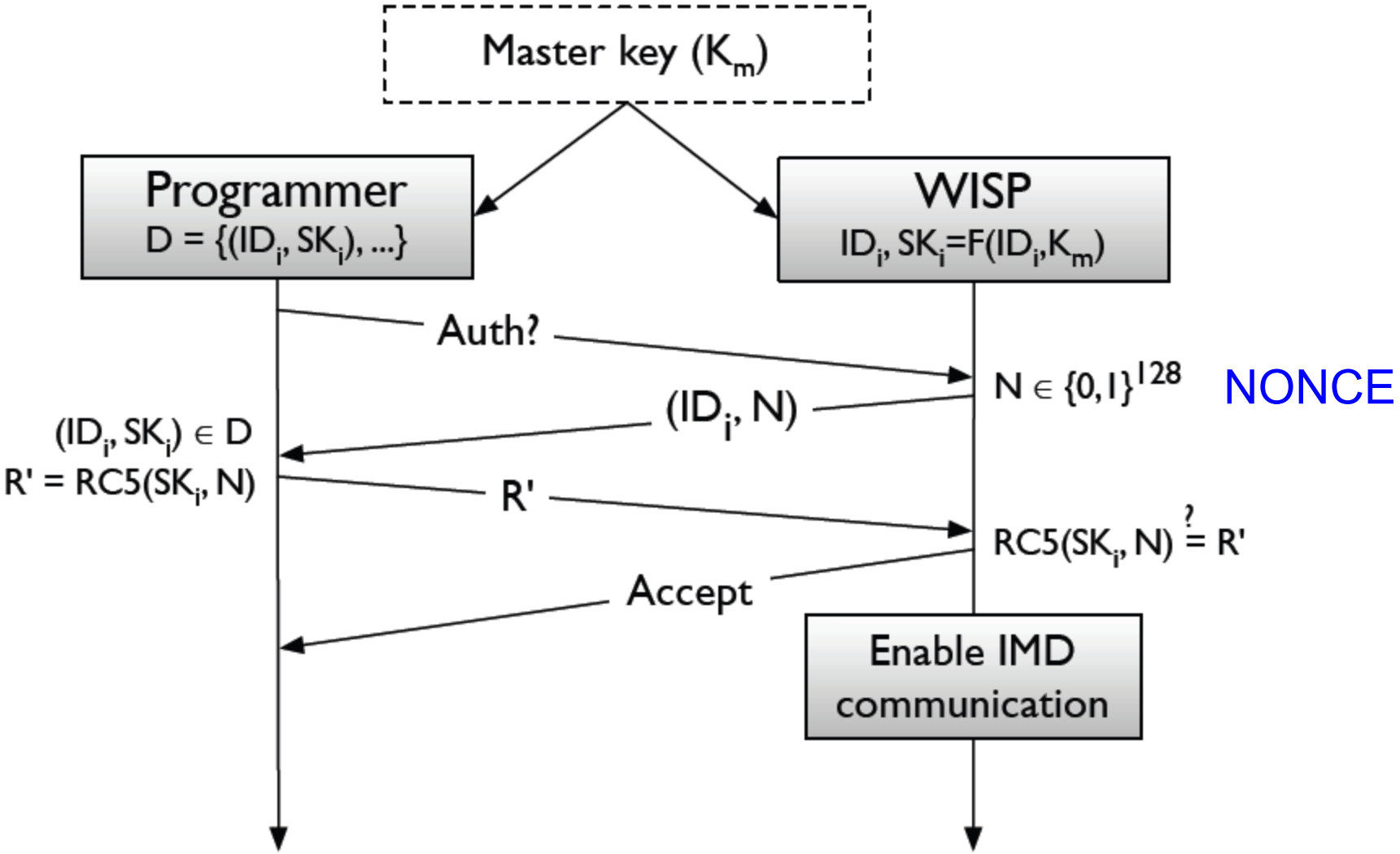
Authors devised a “challenge-response” protocol

Each IMD has an inbuilt identity/serial number ID

IMD and programmer share a secret key K_m

Uses RC5 encryption algorithm which is performed on WISP platform using “zero power” (only energy harvested from RFID reader)

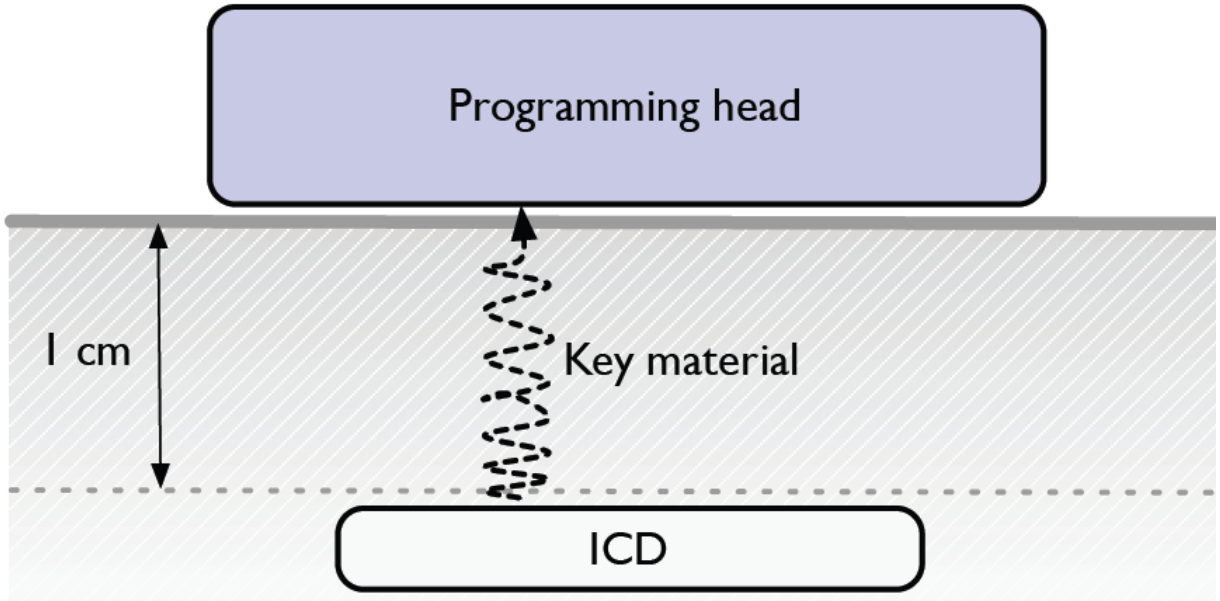
Proposed Authentication Protocol



Key Exchange: Where does K_m come from?

Idea: Key exchange is possible only by close proximity

1. Programmer is placed very close to the skin of the patient (indicates patient awareness and consent?) and transmits an RF signal to IMD
2. IMD then responds with a random value to be used as K_m, transmitted as a modulated sound wave that cannot be sensed at appreciable distance (> 1 cm ?)



Trusted Computing Base (TCB)

Should only include the ICD

Not the Programmer Device

→ Current designs assume the programmer device (and its user) is trusted

In general, the TCB must be as small as possible

Main Concepts to Discuss

- Dangers of sending messages in cleartext
  - How protocols can be reverse engineered
- Power matters
  - Use of “Zero-power” protocols
- Security fundamentals
  - Encryption, nonces, Trusted Computing Base (TCB)

A Few Other Contexts for Security in Embedded Systems

Electronic voting

Sensor networks