Data Manipulation Attack Simulation

The following is a simulation of the data manipulation attack on an ICD.

Demo

https://dma-in-icd.netlify.app/.

Tachycardia detection - Rhythm ID algorithm

Refer to the paper: Synthesizing stealthy reprogramming attacks on cardiac devices.

Basic Parameters Working and Attack

Basic Heart Rate attack

The minimum heart rate supported is set to 30 (constant) value. If the parameter is set to less than 30 and the current heart rate recorded is less than 30, we report heart failure.

Upper Heart Rate attack

The upper heart rate is set to 220 (constant) value. If the parameter is set to more than 220 and the current heart rate recorded is more than 220, we report heart failure.

Note: The device is programmed to perform the upper heart rate check only between 8 and 20 hours. If the current time is not between 8 and 20 hours, the upper heart rate check is not performed.

Night Heart Rate attack

The night heart rate should lie between 50 and 120. If the parameter is set to less than 50 and the current heart rate recorded is less than 50, we report heart failure. If the parameter is set to more than 120 and the current heart rate recorded is more than 120, we report heart failure.

Note: The device is programmed to perform the night heart rate check only between 0 and 4 hours. If the current time is not between 0 and 4 hours, the night heart rate check is not performed.

Battery Depletion

Depletion due to pacing pulses (Atrial, Left Ventricular, Right Ventricular)

The battery depletion in the device is calculated based on the energy consumption during the delivery of shocks and pacing pulses. The code simulates the gradual reduction in battery level over time, taking into account the energy used for each pulse delivered.

Implementation: To simulate battery depletion, we use the following formula:

```
depletion = (amp - pacingThresholdSetup) * 0.05 + (width - 0.4) * 0.02
```

where:

- `amp` is the amplitude of the pacing pulse
- `pacingThresholdSetup` is the pacing threshold setup value
- `width` is the width of the pacing pulse
- `depletion` is the amount of battery depletion for the pacing pulse for each of atrial, left ventricular, and right ventricular pacing pulses

The above depletions are calculated for each pacing pulse and shock delivered. The total battery depletion is calculated by summing up the depletions for each pacing pulse and shock delivered. The total battery depletion is then subtracted from the initial battery level to get the current battery level.

Depletion due to shock therapies

Implementation: The battery depletion due to shock therapies is calculated as below:

```
newBatteryLevel = Mathf.max(oldBatteryLevel - depletion, 0)
```

where:

- `oldBatteryLevel` is the battery level before the shock therapy
- `depletion` is the amount of battery depletion due to the shock therapy which is calculated as:

```
depletion = 0.1 * shockEnergyValue
```

where:

- `shockEnergyValue` is the energy of the shock therapy

Shocking mechanism

Shocks are delivered whenever an arrhythmia is detected. When the mode is set to "Therapy On" (more about it in the next bullet point), tachycardia is detected and shock is delivered automatically. If the tachycardia detection is disabled, the programmer has to deliver the shock manually to cure tachycardia. Two possible shock deliveries are automatic and manual.

Automatic Shock Delivery

Automatic shock delivery doesn't rely on the programmer's availability to deliver the manual shock. This therapy depends on the first (`dose-v1`), second (`dose-v2`), and subsequent (`dose-vn`) shock energies. Every shock dose involves sending shocks a certain number of times, denoted with "Shocks per episode (`valuesh`)". Shock per episode is the maximum number of shock doses delivered per episode.

In the first step, shock doses of `dose-v1` are provided. If the appropriate therapy is not delivered, shock doses of `dose-v2` are given. If this still fails, shock doses of energy `dose-vn` are given. If this fails too, then we report heart failure. Each of the above doses is given `valuesh` times.

Implementation: Whenever the ICD detects tachycardia, it delivers the shock automatically. The shock delivery is done in the following way:

1. The code checks the shock values (`dose-v1`, `dose-v2`, `dose-vn`) for acceptable ranges and attempts shock treatment sequentially. If any shock treatment is unsuccessful, it reports heart failure.
2. For each episode in attempting shock treatment, we decide the outcome of the shock treatment probabilistically. For the shock dose `shockValue`, episode `episode`, and the current iteration of the shock dose `i`, we calculate the probability of success as:

```
prob = baseProb - shockPenalty - episodePenalty - 0.001 * heartRateDifference
```

where:

- `baseProb` is the base probability of success, which is set to 0.8
- `shockPenalty` is the penalty for the shock dose, which is set to 0.02 * `shockValue`
- `episodePenalty` is the penalty for the episode, which is set to 0.01 * `episode`
- We generate a random number between 0 and 1. If the random number is less than the probability of success, we report success, otherwise we repeat the process for the remaining iterations of the shock dose or the remaining shock doses.
- If all the shock doses are unsuccessful, we report heart failure.

Manual Shock Delivery

Manual shock delivery relies on the programmer's availability to deliver the shock.

Implementation: When the ICD detects tachycardia, it alerts the programmer about the detected tachycardia. The programmer can then deliver the shock manually. The shock delivery is done in the following way:

1. Whenever tachycardia is detected by the Rhythm ID algorithm, the ICD alerts the programmer about the detected tachycardia.
2. After the programmer is alerted, the timeout for the programmer to deliver the shock starts. If the programmer delivers the shock within the timeout, the shock is delivered successfully. If the programmer doesn't deliver the shock within the timeout, we report heart failure.
3. To determine if the shock delivery is successful, we calculate the probability of success as:

```
prob = baseProbability + shocksPenalty - deviationPenalty
```

where:

- `baseProbability` is the base probability of success, which is set to 0.8
- `shocksPenalty` is the penalty for the number of shocks already given, which is set to 0.002 * `shocksGiven`
- `deviationPenalty` is the penalty for the deviation of the shock value from the recommended shock value, which is set to 0.001 * `deviation`
- `shocksGiven` is the number of shocks already given for the current tachycardia episode, which should be less than or equal to the maximum number of shocks per episode
- `deviation` is the deviation of the shock value from the recommended shock value which is calculated as:

  ```
  deviation = manualShockEnergy - thresholdShockEnergy
  ```

  where:

  - `manualShockEnergy` is the energy of the shock delivered by the programmer
  - `thresholdShockEnergy` is the recommended energy of the shock which is again calculated as:

    ```
    thresholdShockEnergy =
      baseThreshold + heartRateMultiplier * currentHeartRate
    ```

    where:

    - `baseThreshold` is the base threshold energy for the shock, which is set to 30
    - `heartRateMultiplier` is the multiplier for the current heart rate, which is set to 0.5

    The threshold shock energy is clamped between 10 and 80.

- We generate a random number between 0 and 1. If the random number is less than the probability of success, we report success, otherwise we report failure. If the shock delivery is successful, the tachycardia episode is considered to be treated successfully. If the failures are more than the maximum number of shocks per episode, we report heart failure.
- If the programmer doesn't deliver the shock within the timeout, we report heart failure.

Rescue Shock

The rescue shock is delivered if the tachycardia detection is disabled for a certain timeout of a programmable value, that's nominally set to 6 hours because the ICD is in the MRI protection mode. Starting rescue shock terminates the MRI Protection Mode and the tachycardia detection is enabled.

Implementation: The rescue shock is delivered in the following way:

1. To start rescue shock, we give the command: `START RSHK`. The ICD then starts charging the ICD to deliver the shock. The ICD then delivers the shock to the patient.
2. The shock delivery is done in the same way as the manual shock delivery, with the same probability of success and failure.
3. If the rescue shock is successful, the tachycardia episode is considered to be treated successfully. If the rescue shock is unsuccessful, we report heart failure.
4. If rescue shock is started, the ICD exits the MRI Protection Mode and the tachycardia detection is enabled.

Mode

Shelf Mode

In shelf mode, the battery consumption is constant due to the battery's standby leakage. This is a one-time-only mode, and when this is set to use by the programmer, it can't be reverted to this mode.

Implementation: at the beginning of the simulation, the ICD is set to be in this mode by default. Once the "Start Simulation" button is pressed, there is no going back to the Shelf Mode. Constant battery depletion starts as soon as the site loads.

Therapy Off Mode

This mode is by default set when the "Start Simulation" button is pressed for the first time. Tachycardia detection: **OFF**.

Implementation: when this is set to **OFF**, the **Rhythm ID** algorithm still runs. If the algorithm decides that therapy is required, ICD doesn't deliver shock automatically, instead, manual shock therapy is administered if the medical programmer has been alerted about the detected tachyarrhythmia. More about the shock delivery methods are talked about in detail in the **Shock Delivery** section.

Therapy On Mode

This is the usual mode of the ICD, which detects tachyarrhythmias and responds by delivering appropriate shock treatment.

MRI Protection Mode

Tachycardia detection is disabled, beeper is disabled. Following are the methods to exit this mode:

1. manually change the mode
2. reverted after a certain timeout of a programmable value, that's nominally set to 6 hours
3. deliver a rescue shock

Once exited, the beeper needs to be separately enabled as it doesn't return to the previous settings.