Summary of the ECU Dataset and ahmed2021ecu Paper

Dataset Context and Need:

The healthcare sector's integration with IoT devices, termed IoHT, has brought significant advancements but also introduced substantial security vulnerabilities. These devices, such as insulin pumps and pacemakers, are critical as they directly impact patient safety. However, many of these devices lack basic security features like encryption and authentication, making them susceptible to cyberattacks.

The ECU-IoHT dataset was developed as part of a study to analyze the security vulnerabilities and potential cyberattacks on IoHT devices. It is the first of its kind in the healthcare domain, as most other available datasets do not cover IoHT-specific security challenges.

Dataset Development:

The ECU-IoHT dataset was created in a controlled environment following a standard white hat penetration testing methodology. The testbed used for the dataset development included components like:

- **Libelium MySignals Healthcare Kit**: This kit was central to the IoHT testbed, providing a platform for developing eHealth applications and medical devices. It includes multiple sensors that monitor various biometrics, and the data is sent to the cloud for analysis.
- **Computing Environment**: The environment consisted of a Windows 10 system, a Kali Linux instance, a mobile Wi-Fi hotspot, and Bluetooth and wireless network adapters. The MySignals device was connected wirelessly to the Internet, allowing for the simulation of real-world IoHT device interactions.

Sensors Used:

Three specific sensors were utilized in the dataset:

1. **Temperature Sensor**: Selected for its ease of use and quick data transmission to the cloud. It sends data every 10 seconds.
2. **Blood Pressure Sensor**: Though it takes longer to send updates, it was chosen for its ability to generate large quantities of data, making it ideal for attack detection.
3. **Heart Rate Sensor**: More challenging to set up but provides frequent data updates.

Detailed Explanation of Dataset Fields

1. No. (Number)

- This is simply the index or serial number of the entry in the dataset, allowing you to reference specific records easily.

2. Time

- This field records the time at which the packet was captured, typically in seconds since the start of the capture. It allows for the analysis of network activity over time, helping to identify patterns such as bursts of traffic or delays that could be indicative of an attack.

3. Source

- This represents the originating address or device from which the network packet was sent. It could be an IP address or a MAC address depending on the layer of the network stack being examined.
- **Why it matters**: The source field helps in identifying which device initiated the communication. In an attack scenario, this could help trace back to the attacking device or compromised system.

4. Destination

- This field shows the address of the target device or endpoint that the packet is trying to communicate with.
- **Why it matters**: Similar to the source field, the destination helps in understanding which device is being targeted. The roles of source and destination can switch if the devices are engaged in a back-and-forth exchange, such as during an attack or response.

5. Protocol

- The protocol field indicates the type of communication protocol used in the packet, such as TCP, ARP, DNS, or ICMP.
- **Why it matters**: Different protocols serve different purposes. For example, ARP (Address Resolution Protocol) is used for mapping IP addresses to MAC addresses, while TCP (Transmission Control Protocol) is used for reliable data transfer. Certain attacks target specific protocols, so identifying the protocol helps in recognizing the type of attack.

6. Length

- This field indicates the size of the packet in bytes.
- **Why it matters**: The length of the packet can be indicative of normal or abnormal behavior. For instance, certain attack types like ARP Spoofing or Smurf Attacks tend to have fixed sizes. Recognizing these sizes can help in flagging suspicious packets.

7. Info

- The Info field provides a detailed summary of the packet's content, which can vary significantly depending on the protocol. For example:
    - **Normal Traffic Example**: `36954 > 110 [SYN] Seq=0 Win=1024 Len=0 MSS=1460`
        - This represents a SYN packet, which is the first step in establishing a TCP connection. The source port `36954` is trying to connect to destination port `110` (typically used for POP3 email services). The packet is initiating a connection with sequence number `0`, a window size of `1024` bytes, and a maximum segment size (MSS) of `1460` bytes.
    - **Nmap Port Scan Example**: `1720 > 36954 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0`
        - This represents a response to a port scan where the target is closing the connection with a reset (RST) flag. This typically happens when Nmap, a network scanning tool, tries to probe a closed port.
- **Why it matters**: The Info field provides insights into the behavior of the traffic. Understanding these behaviors is crucial for distinguishing between normal and malicious activities.

8. Type

- This field indicates whether the packet is part of a normal communication (`Normal`) or an attack (`Attack`).
- **Why it matters**: The Type field is essential for labeling data during the training of machine learning models. It helps the model learn the characteristics of normal versus malicious traffic.

9. Type of Attack

- If the packet is flagged as an attack, this field specifies the type of attack, such as ARP Spoofing, Nmap Port Scan, or Smurf Attack.
- **Why it matters**: Knowing the specific type of attack allows for more granular analysis and understanding of how different attacks manifest in network traffic. This can help in creating specialized detection mechanisms for each type of attack.

Summary of Model Architecture of wustl-ehms-2020 dataset using hady2020intrusion paper

The system comprises six key components, each contributing to a comprehensive solution for monitoring and securing patient data:

1. **Multi-Sensor Board (PM4100 Six Pe):**
    - **Electrocardiogram (ECG):** Measures the heart's electrical activity using three-electrode pads.
    - **Blood Oxygen Saturation (SpO2):** Monitors blood oxygen levels and heart rate.
    - **Temperature Sensor:** Records body temperature.
    - **Blood Pressure Sensor:** Measures systolic and diastolic arterial pressure.
2. **Gateway:**
    - A Windows-based laptop connected to the multi-sensor board via USB.
    - Displays biometric data through a graphical user interface (GUI) and transmits real-time data to the server using a C++ program.
    - Connected to the network switch via Ethernet.
3. **Server:**
    - An Ubuntu-based laptop that receives and stores data from the gateway.
    - Utilizes a C++ program for data collection and analysis.
4. **Network:**
    - An Ethernet switch links the server, Intrusion Detection System (IDS), and attacker.
    - A router dynamically assigns IP addresses, with the gateway connected via Wi-Fi.
5. **Intrusion Detection System (IDS):**
    - Monitors all packets directed to the server and forwards them to the IDS computer.
    - Employs Argus software for network flow monitoring and biometric data collection.
    - Provides real-time decisions on incoming traffic packets.
6. **Attacker:**
    - A Kali Linux-based computer used to simulate attacks on the system.
    - Utilizes a Python script with the Scapy library for packet sniffing, spoofing, and alteration.

Data Collection and Features

- **Data Collection**:
  - 16,000 data samples were collected and labeled as 0 (non-attack) or 1 (attack).
  - Source MAC address was used for labeling; samples with attacker MAC addresses were labeled as 1.
  - Unrelated samples to the gateway, attacker, and server MAC addresses were removed.
- **GUI Features**:
  - **HR**: Heart Rate in Beats Per Minute (BPM).
  - **RR**: Respiration Rate in BPM.
  - **ST**: Electrically neutral area between ventricular depolarization and repolarization in millivolts (mv).
  - **SYS**: Systolic blood pressure.
  - **DIA**: Diastolic blood pressure.
  - **SPO2**: Blood oxygen.
  - **PR**: Pulse Rate in BPM.
  - **TEMP**: Temperature in degrees Celsius
  -

This architecture ensures robust monitoring and security of patient data, effectively integrating both network and biometric information to detect and respond to potential threats.