Below is a step-by-step write-up of how the “Lost in a Building” misc-category CTF was solved, from first hints all the way through to extracting the flag.

**1. Challenge Overview & Initial Hints**

* You’re given a PCAP of the building’s IoT management network.
* Goal: identify **which device** was tampered with and **what access code** was sent to lock the doors.
* Flag format:
* ShaktiCTF{floor:accesscode}

with *no* spaces inside the braces.

Your teammate’s Discord hint (“SIDEKICK\_ON\_2ND\_FLOOR:FLOOR2DOOR9”) told you that somewhere in the capture lives that exact ASCII string. That was a massive clue: it *is* both the device name and the access code, joined by a colon.

**2. Quick-Win Recon: strings on the PCAP**

As a first pass, dump all printable strings from the raw PCAP:

strings building.pcap | grep –i floor

That immediately surfaced:

SIDEKICK\_ON\_2ND\_FLOOR:FLOOR2DOOR9

**Takeaway:** the attacker isn’t using a numeric PIN at the application level but sending a human-readable command.

**3. Zero-In with Wireshark**

Even though we have the ASCII, we still need **which packet** actually carries it.

1. **Open** the PCAP in Wireshark.
2. **Apply** this display filter:
3. frame contains "SIDEKICK\_ON\_2ND\_FLOOR"
4. **Locate** the single packet (Frame 803 in our case) that contains the full ASCII payload.

**4. Decoding the Payload**

The packet in question turned out to be a **Modbus “Write Multiple Registers”** (Function 16) request. Rather than text, Modbus registers are 16-bit words—so each register holds two ASCII characters.

1. **Extract** the register values (you can copy them from the Packet Bytes pane or use Tshark):
2. Register 256 → 21321 (0x5349) → “SI”
3. Register 257 → 17477 (0x4445) → “DE”
4. …
5. Register 272 → 14592 (0x3900) → “9␀”
6. **Convert** each 16-bit value to hex, then map to ASCII.
7. **Reassemble** the pairs in order, yielding:
8. SIDEKICK\_ON\_2ND\_FLOOR:FLOOR2DOOR9

**5. Crafting the Flag**

* **Device (floor)** part = SIDEKICK\_ON\_2ND\_FLOOR
* **Access code** part = FLOOR2DOOR9

Plug into ShaktiCTF{floor:accesscode} (no spaces):

ShaktiCTF{SIDEKICK\_ON\_2ND\_FLOOR:FLOOR2DOOR9}

That is the exact string the challenge accepted.

**6. Summary of the Approach**

1. **Hint harvest:** strings revealed the only human-readable command in the dump.
2. **Filter in Wireshark:** frame contains … to find the precise packet.
3. **Protocol decode:** Recognized Modbus “Write Multiple Registers” and knew each register = two ASCII chars.
4. **Hex→ASCII mapping:** Converted register words to text.
5. **Flag formatting:** Joined device name and action code per the spec.

**Lessons Learned**

* Always run a quick strings on binary captures to catch any embedded text.
* Use simple filters (frame contains "…") to avoid blind hunting.
* Know your protocols: Modbus registers are 16‐bit, so text often comes in two-byte chunks.
* Systematic hex→ASCII conversion can turn “garbage” numbers into clear commands.

Congratulations—you’ve now fully documented how to unlock Building 7 (in CTF terms) by tampering with the SIDEKICK controller on Floor 2!