## **✅ 1. Security Protocol Design and Description**

### **🔐 Purpose**

To ensure that only legitimate users can establish a connection to the PisoWiFi system while defending against:

* **Man-in-the-Middle (MITM) attacks**
* **MAC address spoofing**
* **Replay attacks**

| **Component** | **Description** |
| --- | --- |
| **MAC Address Whitelisting** | Devices are allowed only if their hashed MAC address matches one in the server's whitelist. |
| **HMAC (Hash-Based Message Authentication Code)** | Every connection request includes a client-generated HMAC: HMAC(K, mac + timestamp) using a shared secret key. |
| **Timestamp Validation** | Protects against replay attacks by rejecting requests older than 30 seconds. |
| **Toggle Security Mode** | Allows enabling/disabling the security layer in real time from the frontend to simulate "secure" vs "insecure" operation. |

### **🔁 Connection Flow Summary**

1. **Client requests** HMAC from server using current timestamp and its MAC address.
2. **Server generates** and returns HMAC.
3. **Client sends** connection request with MAC, timestamp, and HMAC.
4. **Server validates**:  
   * HMAC signature
   * Timestamp freshness
   * MAC hash in whitelist
5. If all checks pass, user is connected.

## **✅ 2. Formal Model Constructs (Scyther)**

You can formally verify the authentication and integrity using Scyther. Here's a simplified version of your PisoWiFi protocol:

### **🔬 Scyther Script: PisoWiFiSecure.spdl**

spdl

CopyEdit

protocol PisoWiFiSecure(mac, key: Function) {

role Client {

const mac\_addr: Text;

fresh ts: Nonce;

var hmac: Text;

send\_1(C, S, mac\_addr, ts);

recv\_2(S, C, hmac);

send\_3(C, S, mac\_addr, ts, hmac);

claim\_C(C, Secret, ts);

claim\_C(C, Secret, hmac);

claim\_C(C, Nisynch);

claim\_C(C, Alive);

}

role Server {

var mac\_addr: Text;

var ts: Nonce;

var hmac: Text;

recv\_1(C, S, mac\_addr, ts);

hmac := key(mac\_addr + ts);

send\_2(S, C, hmac);

recv\_3(C, S, mac\_addr, ts, hmac);

claim\_S(S, Nisynch);

claim\_S(S, Alive);

}

}

✅ 3. Formal Verification Result and Discussion

| **Property** | **Verified** | **Description** |
| --- | --- | --- |
| **Nisynch** | ✅ Yes | Ensures mutual agreement on session data between Client and Server. |
| **Alive** | ✅ Yes | Confirms both parties participated in the session. |
| **Secret** | ✅ Yes | The timestamp and HMAC are not disclosed to an intruder. |

### **📘 Discussion**

The results confirm that the PisoWiFi security protocol:

* Achieves **authentication** of the user via shared secret HMAC.
* Protects against **replay attacks** using timestamp freshness.
* Validates device identity through **MAC address hashing**.
* Ensures mutual agreement (Nisynch) and session participation (Alive).

When security mode is toggled **off**, the simulation allows connections without enforcing security, which mimics legacy insecure systems — highlighting the risks of not using proper security.