Intrinsic ID's First ever IOT Security Procedure:

The Company “Intrinsic-ID” has introduced the first ever trustable working IOT Security in market. They use the following procedures to authenticate the devices:

**Step 1**: Generate a PUF Symmetric Enrollment Key (K) - which is the same as PUF key in Figure 1 using Intrinsic-ID’s key generator [SSFP]. The SRAM PUF response along with internally generated random data will be used to derive Helper Data. Helper Data is non-sensitive public data that is used by the Fuzzy Extractor to correct noisy PUF bits and extract the same cryptographic key every time. The Helper Data will be stored close to the end IoT device for retrieval during the reconstruction phase.

**Step 2** This symmetric enrollment key will be run through a PRNG to derive an asymmetric public/private key pair (PK, SK). The public key (PK) is stored on the ledger.

**Step 3** Create a data structure (PK, Ek(B), D-ID, H(PK), A) whereby:

• PK is the SRAM PUF based public key

• Ek(B) is the encryption of the extracted biometric template file (B) with the symmetric enrollment key (K),

• D-ID is a device ID1 (to associate the biometric data with)

• H(PK) is the hash (H) of PK, it is used as a unique fingerprint-ID for ledger lookup

• A stands for any other deterministic attributes.

This data structure (PK, Ek(B), D-ID, H(PK), A) is signed with the private key of the PUF (SK). We denote the signature with $(PK, Ek(B), D-ID, H(PK), A). Note that we encrypt the biometric template file before storing on the ledger to avoid transmitting the file in the clear which makes the communication prone to man-in-the middle attacks.

**Step 4** Register this signed data structure [(PK, Ek(B), D-ID, H(PK), A ),$ (PK, Ek(B), D-ID, H(PK), A)] in the IOT ledger. The device has now been added to the ledger (after checking this device has not been registered before). This ledger entry will now be associated with the user who owns the device. Note that a user can own multiple devices.

**Step 5** When a user wants to authenticate her/him self to the service, the following steps are executed. 1. The device captures a fresh biometrics B’ of the user.

2. The symmetric enrolled key is reconstructed using the current noisy PUF response and the Helper Data retrieved from the device

3. The corresponding asymmetric key pair (SK, PK) is derived

4. Using the key fingerprint ID (H(PK)), the corresponding encrypted biometric template file (Ek(B)) is retrieved from the ledger

5. The data (Ek(B)) is decrypted and compared locally to the fresh measurement B’ captured on the IoT end device. If the fresh measurement B’ matches with B, the device has authenticated the user and it will continue its operation.

**Step 6** After device initiation, it is assumed that biometric authentication is not needed for continued trust. When the device generates data (D), it is signed with the PUF private key (D, $D). The PUF signature allows the data to be authenticated as having come from the device. A hash of the data is then signed with KSI to ensure integrity: (H(D), $H(D))

**Step 7** The KSI signature is then registered in the ledger.

**References:**

A BLOCKCHAIN SOLUTION USING SRAM PHYSICAL UNCLONABLE FUNCTIONS in co-operation with intrinsic-ID.