**Introduction:**

In recent years, the popularity of people possessing their own automobiles is exponentially growing all over the world, in particular, some developing countries. Many automobile companies have proposed their novel types that drivers can have access to their vehicles by authorizing their biometric information via IoT. However, how to protect against opponents to ensure the system is secure is the major issues[1].

**Major Security Threats:**

There are two main sources of attack. One comes from car thefts, who start your car without authorization. Although the performance of biometric authorization (e.g. fingerprint, voice, face or even retina verification) is highly achievable in terms of accuracy, our automobile verification system cannot simply reject those unauthorized drivers, but track their behaviors. The attempted vehicle thefts may not steal our cars, but do some damage, thus we need to design a system that can capture bio-information of any attempted drivers, before they may commit crimes. Another attacker is cyber hacker, who intrudes into the IoT system between car owner and car and tampers key information, e.g. your profile, engine information, alarm system, verification system, etc.

**Possible solutions:**

In order to identify unauthorized drivers, the system should give a command to camera to capture an image of the attempted driver and store the image into the database. After verification phrase, if the captured image does not match with pre-stored verification image, the system will send the captured image and GPS location information through an email to the vehicle owner, which is helpful to catch thieves. Furthermore, the system also provides a sensor that can sense heavy vibration. When a criminal attempts to damage our car, the sensor will detect shaking and vibration. If it exceeds the predefined threshold, the sensor will report to the system a detection of high vibration, and the controller system will then command camera system to capture images nearby and alerts the vehicle owner by sending images and geographic location of vehicles through email. In some extreme cases, the system can even call the police.

To avoid cyber hackers, one passive protection is disconnect our car from the Internet, when it is idle, i.e., car will not be used in a time period, so attackers cannot hack our embedded car system. An active protection is design a protocol that increases the complexity of cryptanalysis to ensure security for each communication. In the next section, we will detail this protocol.

**Persistent Streaming Protocol:**

Persistent Streaming Protocol can divide into three stages: master key initialization; chained session key generation; session key verification.

Stage 1: Similar to Diffie-Hellman key exchange, the first stage is to exchange a permanent key between owner and vehicle. Specifically, two entities share two common secret numbers and before key exchange; owner generates a private number , and sends to automobile; automobile also generates a private number , sends to owner and computes common master key ; after owner receives , it also computes common master key by .

Stage 2: The chained session keys are generated for all communications between two entities. For each communication, each entity will generate their own secret number and , let and be public and sends and to each other. After that, session key for each communication is generated by

Where and is the hash value of and .

As we can see in the above equation, each stage of session key is generated based on the hash value of the previous one, except , which is based on the master key, so the chained framework guarantees security after verification in Stage 3.

Stage 3: The verification is performed by comparing hash digests using their common secret keys. For owner side, it computes -th digest by the following:

where is -th nonce number.

The automobile side also computes -th hash digest of the expected owner using its own secret key:

If the owner wants to authenticate automobile’s identity, he can ask automobile to send verification message:

Then the owner verify the -th hash digest by comparing the expected -th hash digest of automobile using their own secret key.

**Critical Evaluation**

Since PSP protocol generates a chained session keys, that the current session key is based on the hash value of the previous session key, and each session key is valid for only one communication. In other words, after the current communication has done, we never use this session key any more. In addition, each session key will be verified by both owner and car before their communication, and if one session key happens to be stolen by opponents, a security mechanism will be performed that flush all temporary session data to reduce the cost of information leak. In terms of master key, we can guarantee its security with very high confidence. One reason is the master key is distributed when we bought this car, which means there are no other entities that knows this key; Another supportive idea is the key space of master key is twice as the session key(256-bits vs 128-bits)[2].

On the other hand, if the master key is compromised, all session keys generated will be insecure and all communications will be invalid. Furthermore, since it generates a chain of session keys, the efficiency of this algorithm can be also a problem.

**Conclusion:**

Although the rapid development of IoT technology has brought great convenience for us, the importance of security cannot be ignored. As connected car owner, we will not worry about car thefts or cyber hackers, because of captured images and GPS information provided as well as a well-designed protocol that we mentioned above.