CECNET2021: Use Noise Protcol Framework to build secure communication channel for IoT scenarios

Pan Lanlan (潘蓝兰)

abbypan@gmail.com

Guangdong OPPO Mobile Telecommunications Corp. Ltd., China

2021.08

1. **IoT Secure Communication**
   - Main Process
   - Initial: Pairing
   - Initial: Provision/Exchange Long-Term credential
   - Communication: Mutual-Authentication based on Long-Term credential
   - Communication: Build secure communication channel
2. **IoT Secure Scenario**
   - Secure IoT Communication
   - Example: Matter
   - Example: CCC Digital Key
   - Noise
   - Our Proposal
3. **Conclusion**
   - Conclusion
   - Resources

Main Process

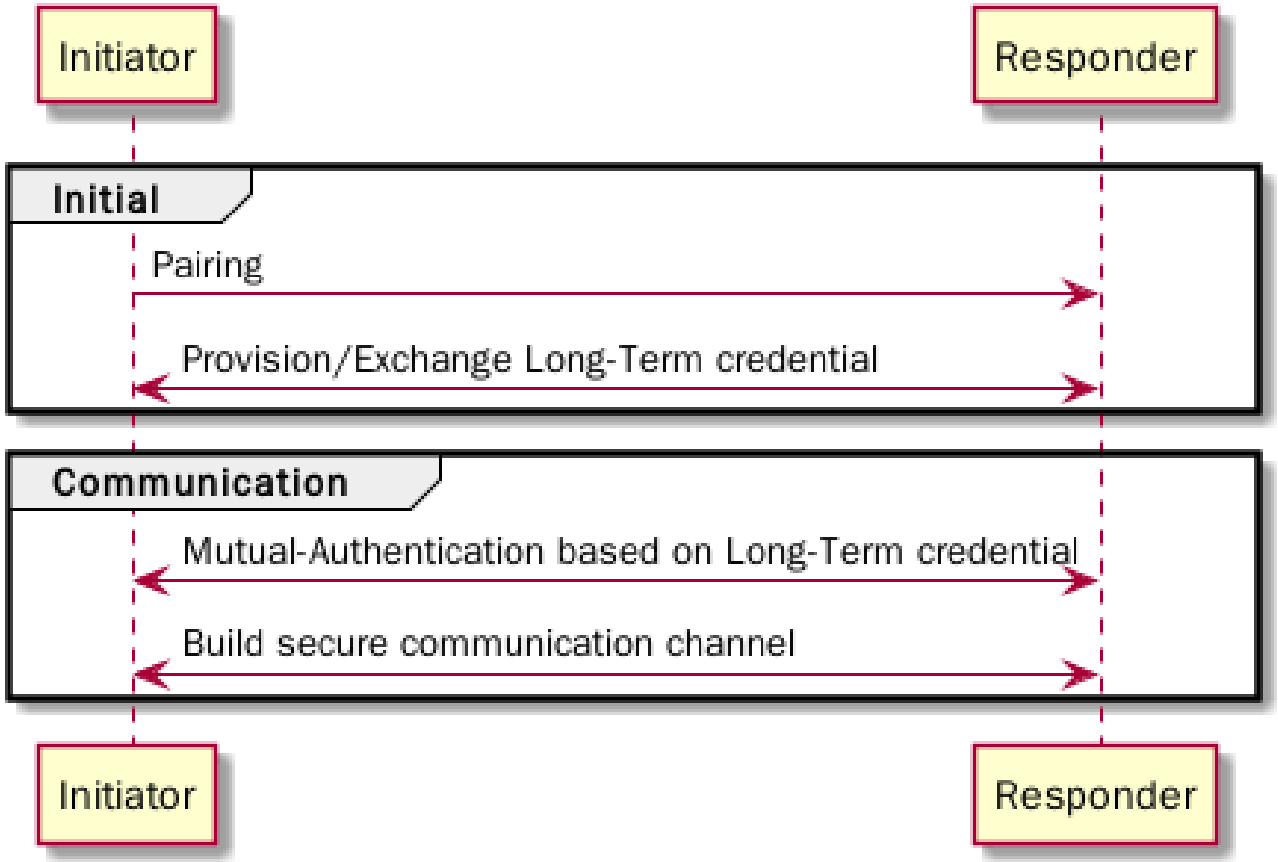

Fig: Main Process

Initial: Pairing

Pairing Category	Example
Balanced PAKE	Dragonfly, J-PAKE, ...
Augmented PAKE	SRP, Spake2+, ...
Other	Noise-NNpsk0, TLS-PSK, ...

Initial: Provision/Exchange Long-Term credential

- Share Secret
- Endpoint's Raw Public Key
- Endpoint's Certificate

Communication: Mutual-Authentication based on Long-Term credential

Authentication Category	Example
STS	Full STS, Basic STS, ...
SIGMA	SIGMA-I, SIGMA-R, ...
TLS	mutual TLS, TLS-PSK, ...
MAC	HMAC, C-MAC, ...
Noise	Noise-KK, Noise-IX, Noise-XX, ...
Other	...

Communication: Build secure communication channel

- Single Key: Symmetric Encryption(AEAD)
- Separated Keys: Symmetric Encryption + Symmetric Authentication

Secure IoT Communication

- Resource Constrained: Limited CPU/Memory/Battery
- Payload Size: Cut Down
- Compatible: Different Credentials
- Privacy: ID Protection

Example: Matter

Process	Selection
Pairing	Spake2+
Credential	Endpoint's Certificate
Authentication	SIGMA-I
Communication	Single Key

Fig: Matter

Example: CCC Digital Key

Process	Selection
Pairing	Spake2+
Credential	Endpoint's Certificate
Authentication	Vehicle's signature is plaintext, Endpoint's signature is encrypted
Communication	Separated Keys

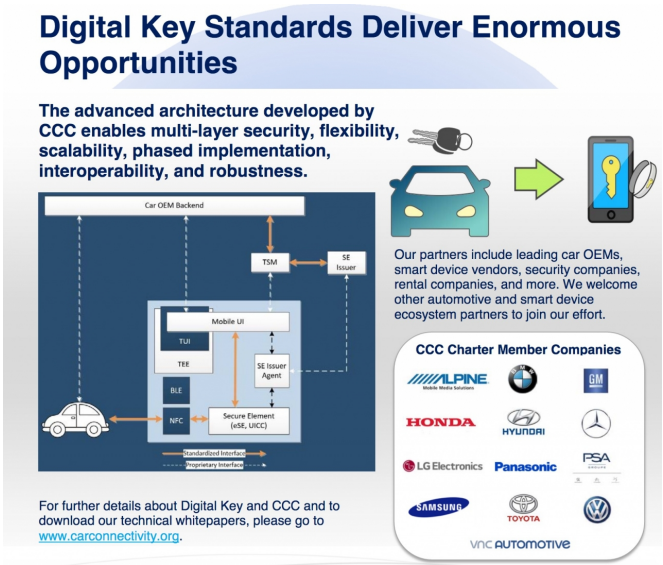

Fig: CCC

Noise

Pattern	Detail	Note
NNpsk0	-> psk, e <- e, ee, psk	share secret simpler than balanced PAKE, TLS-PSK
KK	-> s <- s ... -> e, es, ss <- e, ee, se	Public Key/Certificate similar to Basic STS
IX	-> e, s <- e, ee, se, s, es	Public Key/Certificate Responder's ID Protection similar to SIGMA-R
XX	-> e <- e, ee, s, es -> s, se	Public Key/Certificate Initiator's ID Protection similar to SIGMA-I

Our Proposal

Noise:

- use DH operations to cut down the handshake signature size, which is helpful for constrained communication channel.
- support different credentials: share secret, public key, and certificate.
- adjust ID protection quickly by changing handshake pattern, compliance with privacy requirement, no more code development.

Process	**Selection**		
Pairing	Spake2+, Noise-NNpsk0		
Credential	Share Secret	Raw Public Key	Certificate
Authentication	Noise-NNpsk0	Noise-KK/IX/XX	
Communication	Separated Keys		

Conclusion

It is important to enhance security and privacy for IoT communication.

Noise is suitable for lightweight IoT devices, compatible with different credentials, and flexible for ID protection compliance.

We can use Noise Protcol Framework to build secure communication channel for IoT scenarios, compatible with various IoT Devices.

Resources

- STS https://en.wikipedia.org/wiki/Station-to-Station_protocol
- Matter https://github.com/project-chip/connectedhomeip
- CCC Digital Key https://carconnectivity.org/
- Noise https://noiseprotocol.org/
- SIGMA https://webee.technion.ac.il/ hugo/sigma-pdf.pdf