Suitability of 3rd Round Signature Candidates for Vehicle-to-Vehicle Communication

3rd PQC Standardization Conference
June 7-9, 2021

Nina Bindel Sarah McCarthy Hanif Rahbari Geoff Twardokus

UNIVERSITY OF WATERLOO
IQC Institute for Quantum Computing

RIT Rochester Institute of Technology

Outline

- Introduction to **Secure** Vehicle-to-Vehicle (V2V) Communication
- Presentation of Existing Testbed **V2Verifier**
- **Integration of PQ** Algorithms to V2Verifier and **Experimental Results**
- Analysis of **Dense Environments** on Testbed
- Stating of **Future** Work


*

*All icons from flaticom.com using premium account.

Introduction to V2V Communication

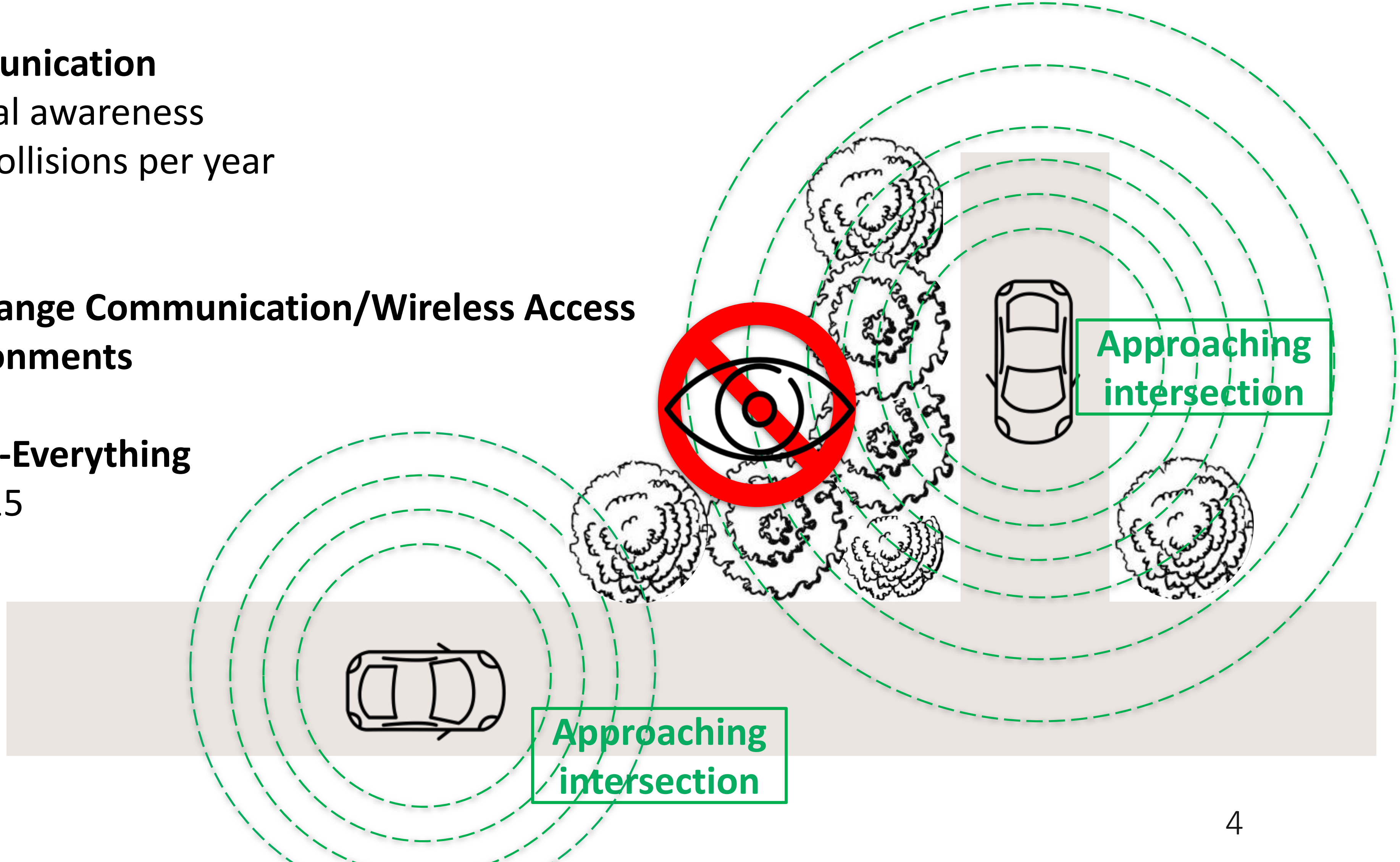
V2V Communication

Direct wireless communication

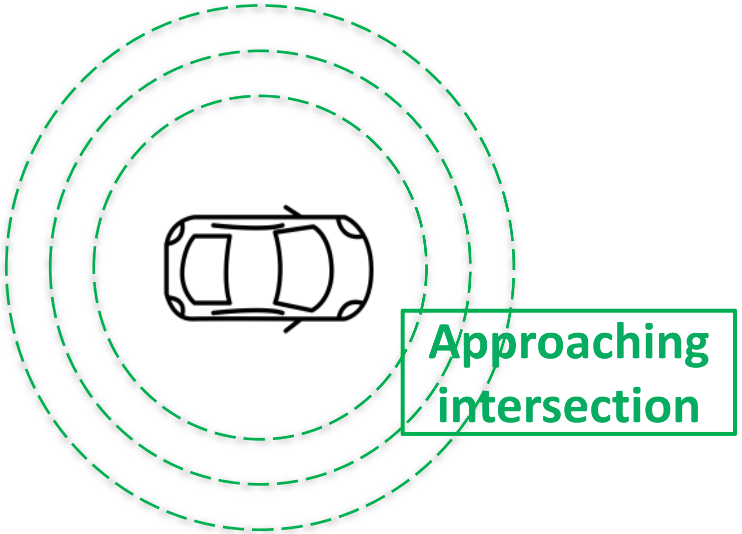
- Increases situational awareness
- Prevents 600,000 collisions per year

Described in

- **Dedicated Short Range Communication/Wireless Access in Vehicular Environments**
  IEEE 802.11p
- **Cellular Vehicle-to-Everything**
  3GPP Release 14/15

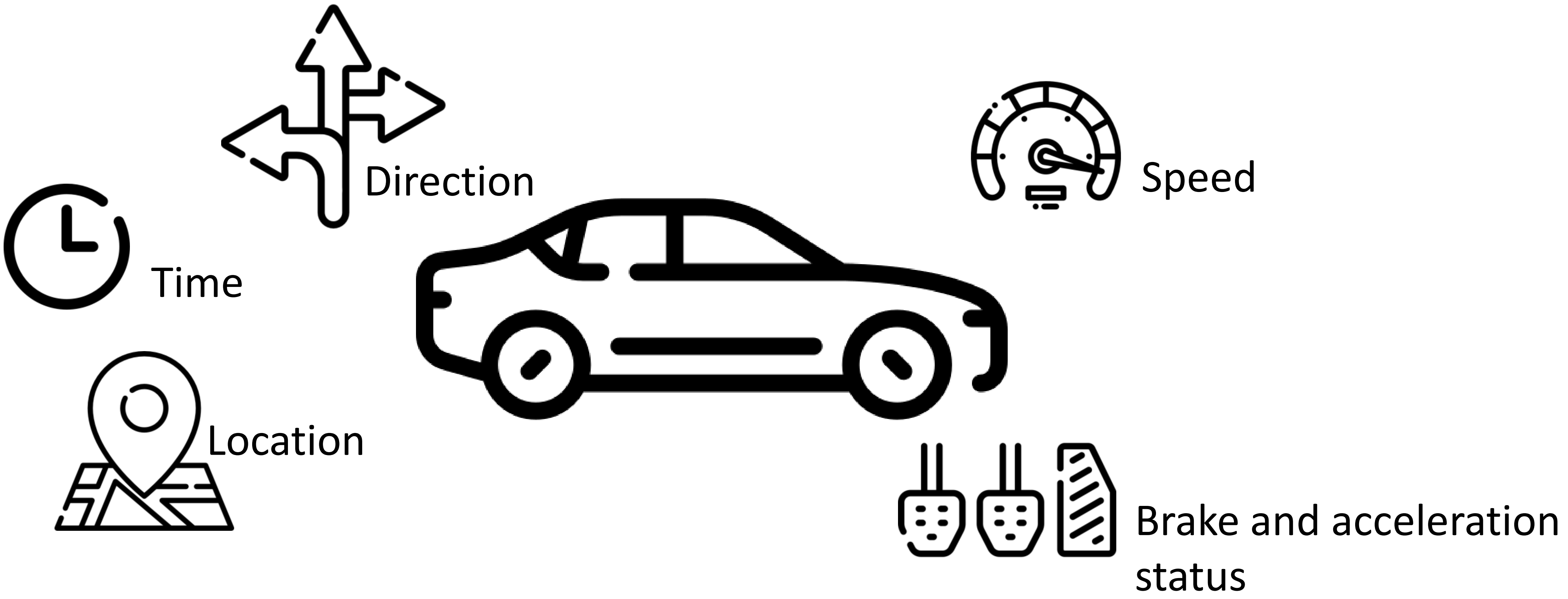


Basic Safety Messages (BSMs)



Every vehicle broadcasts 10 BSMs per second within transmission range

Information Collected in BSMs



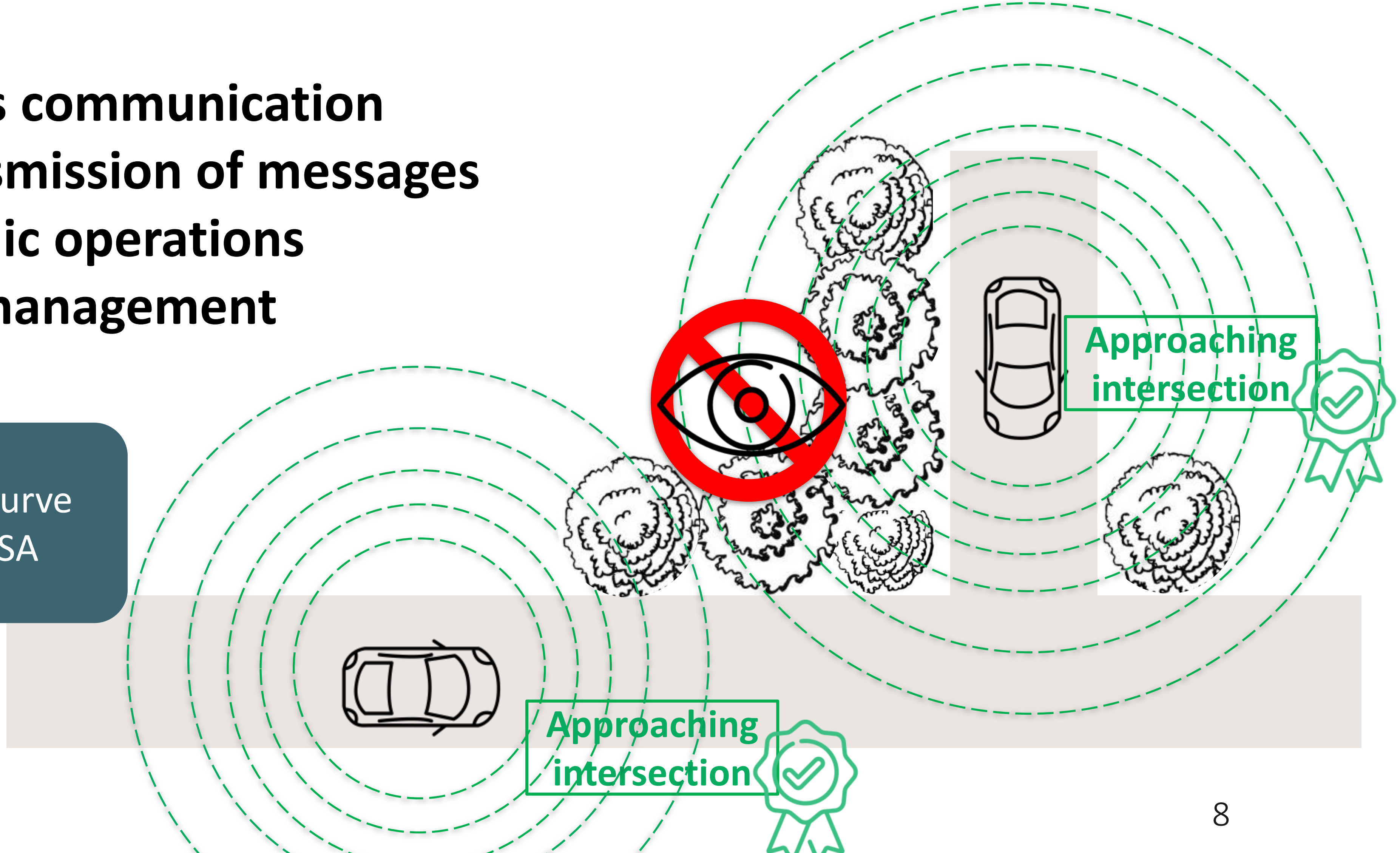
Introduction to **Secure** V2V Communication

IEEE 1609.2 Standard

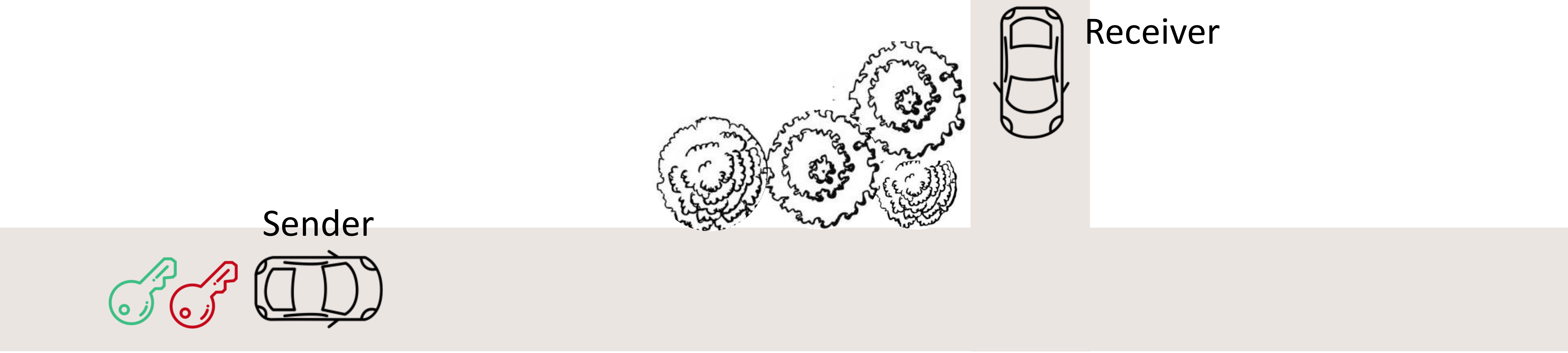
Secure wireless communication

- **secure transmission of messages**
- **cryptographic operations**
- **certificate management**

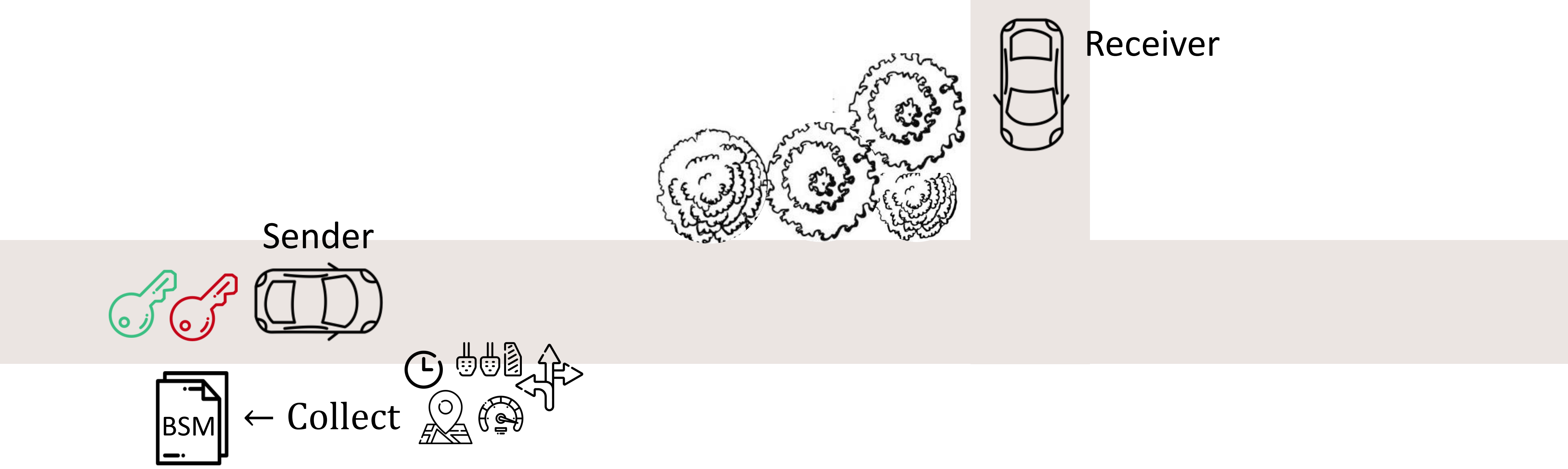
Based on elliptic curve crypto, e.g. ECDSA



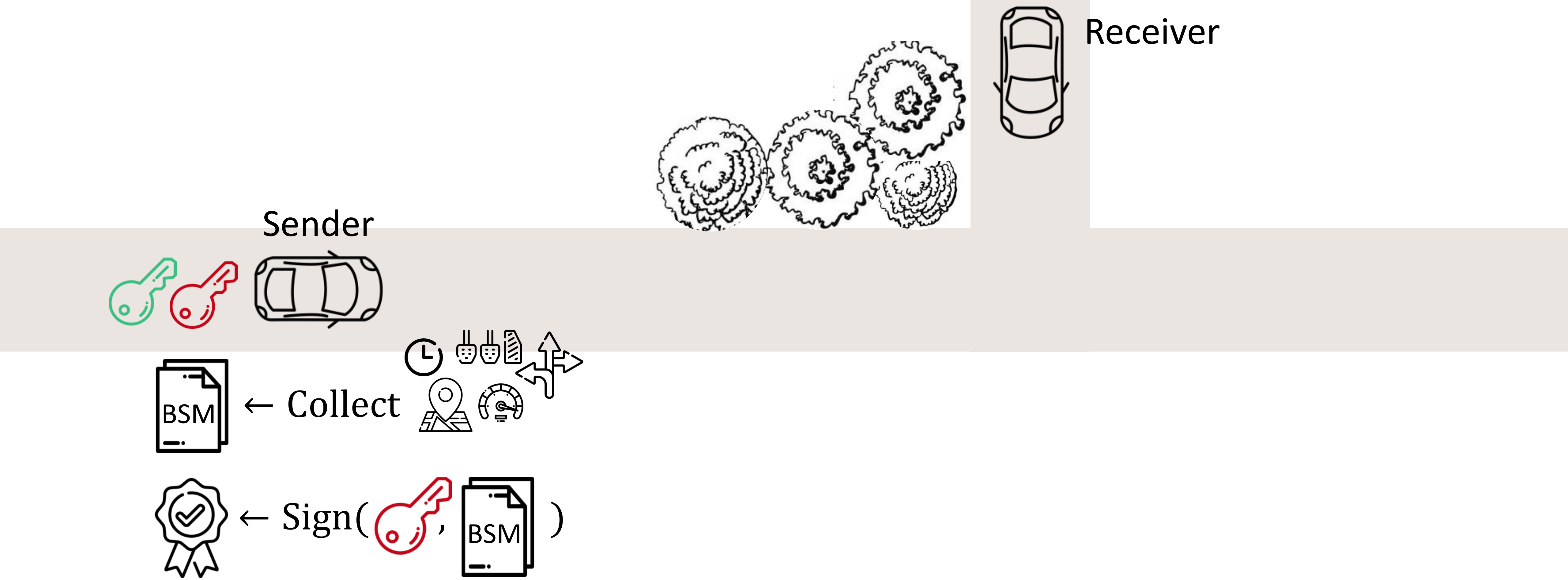
Secure BSM Exchange



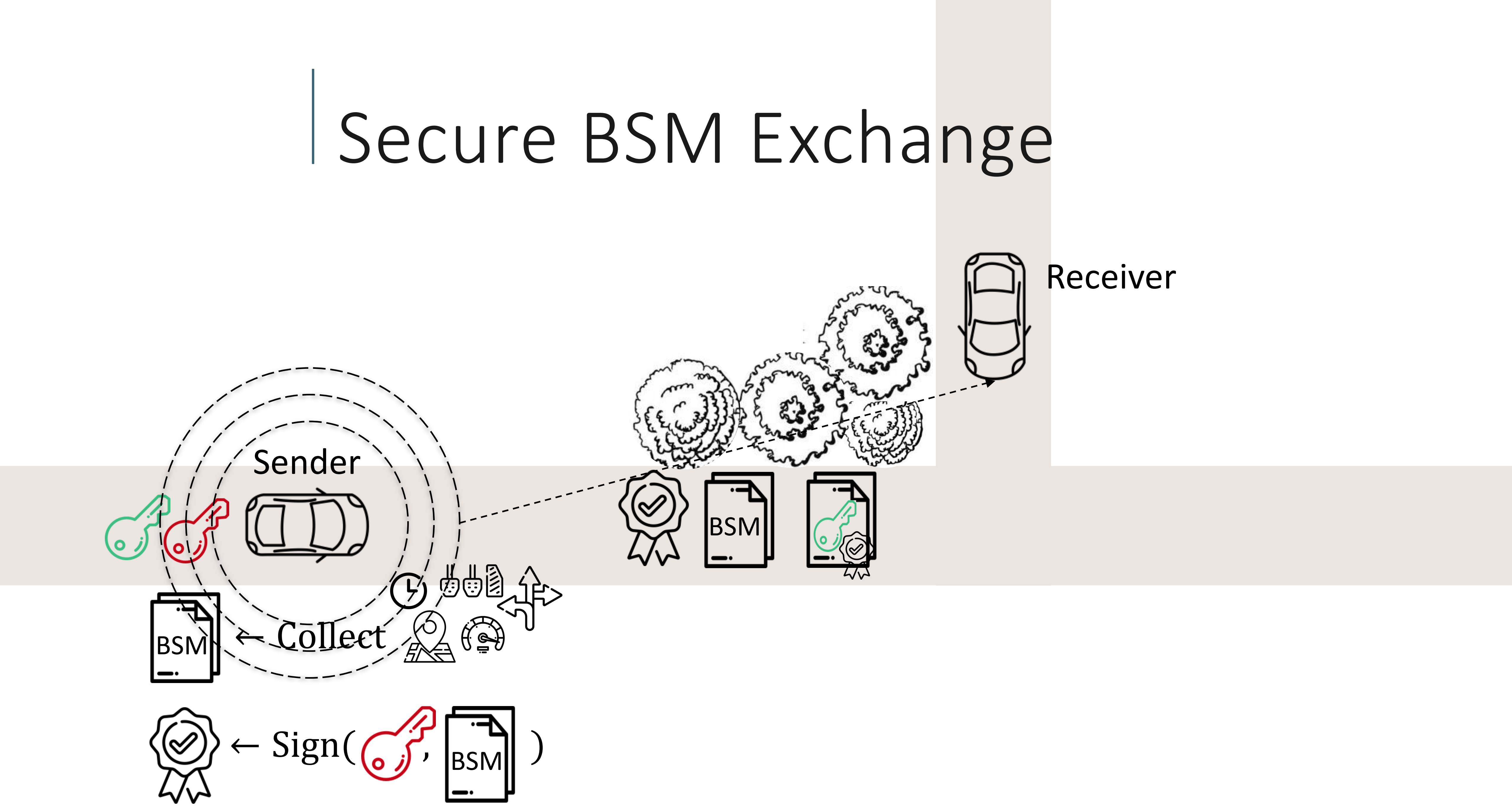
Secure BSM Exchange



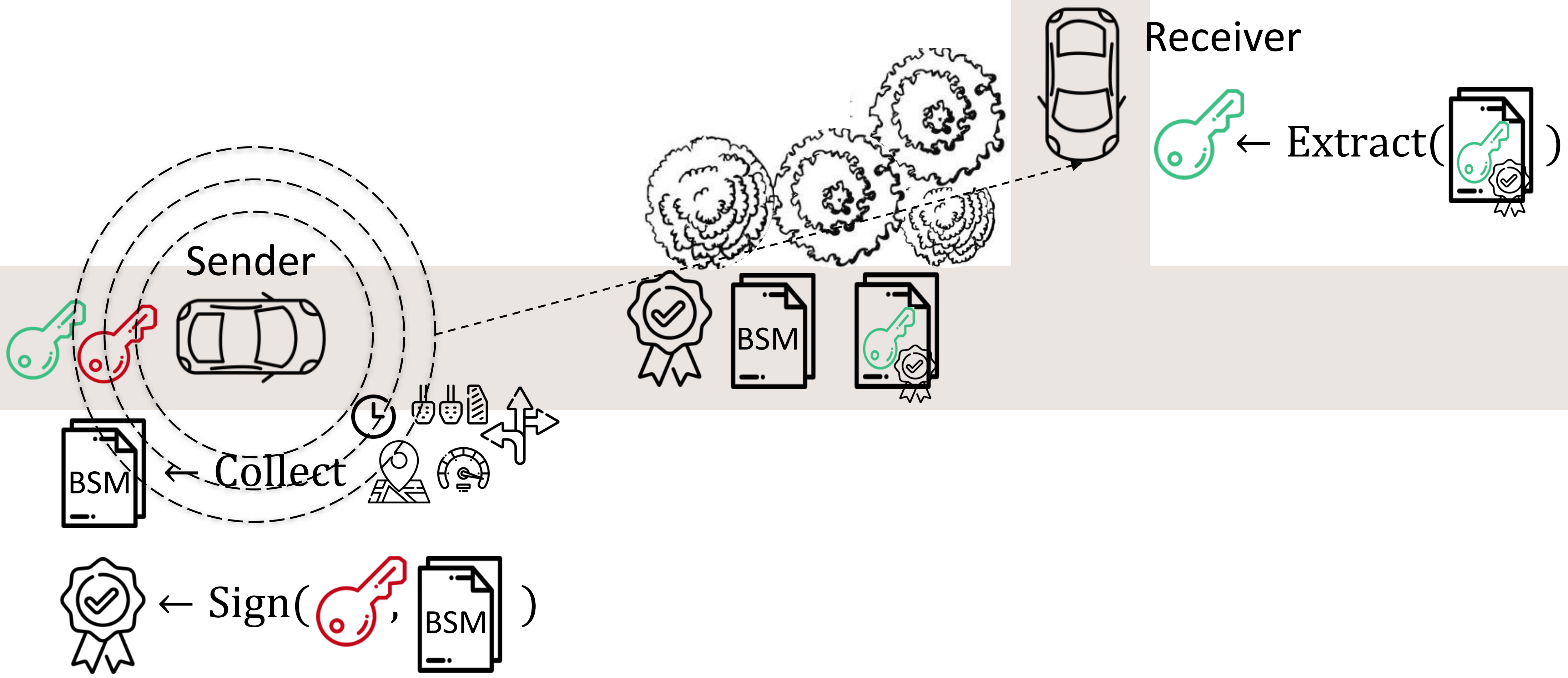
Secure BSM Exchange



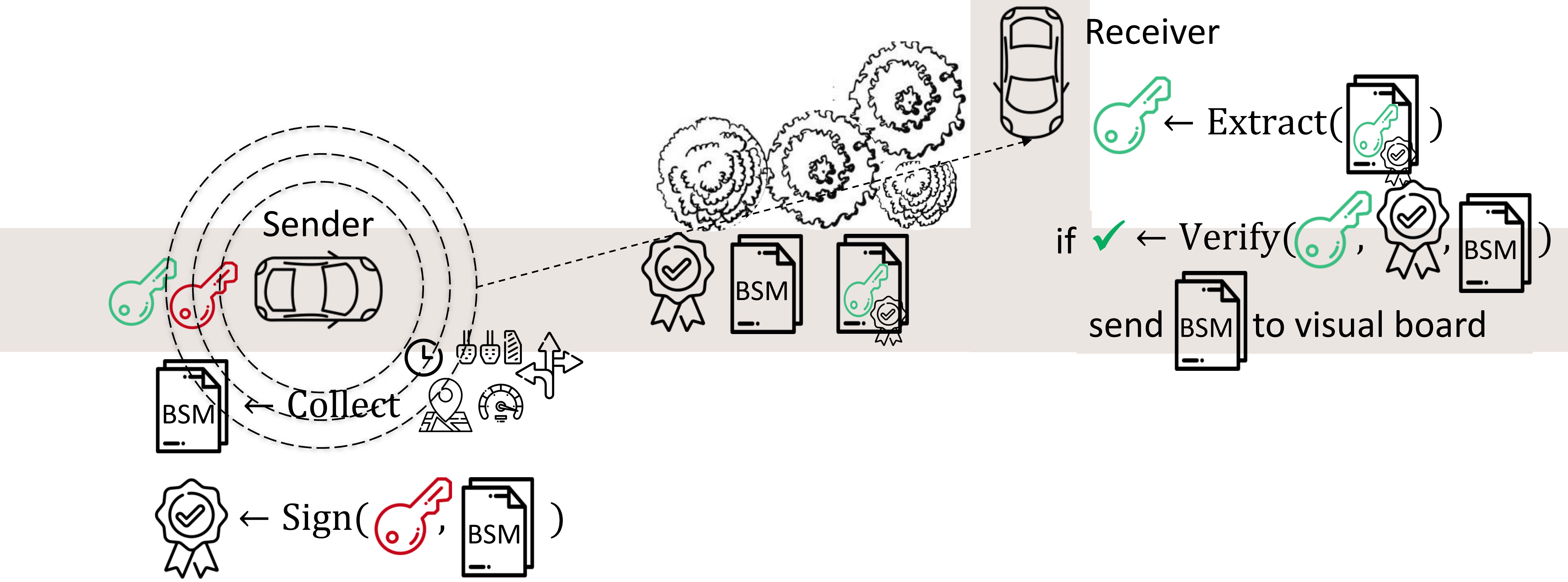
Secure BSM Exchange



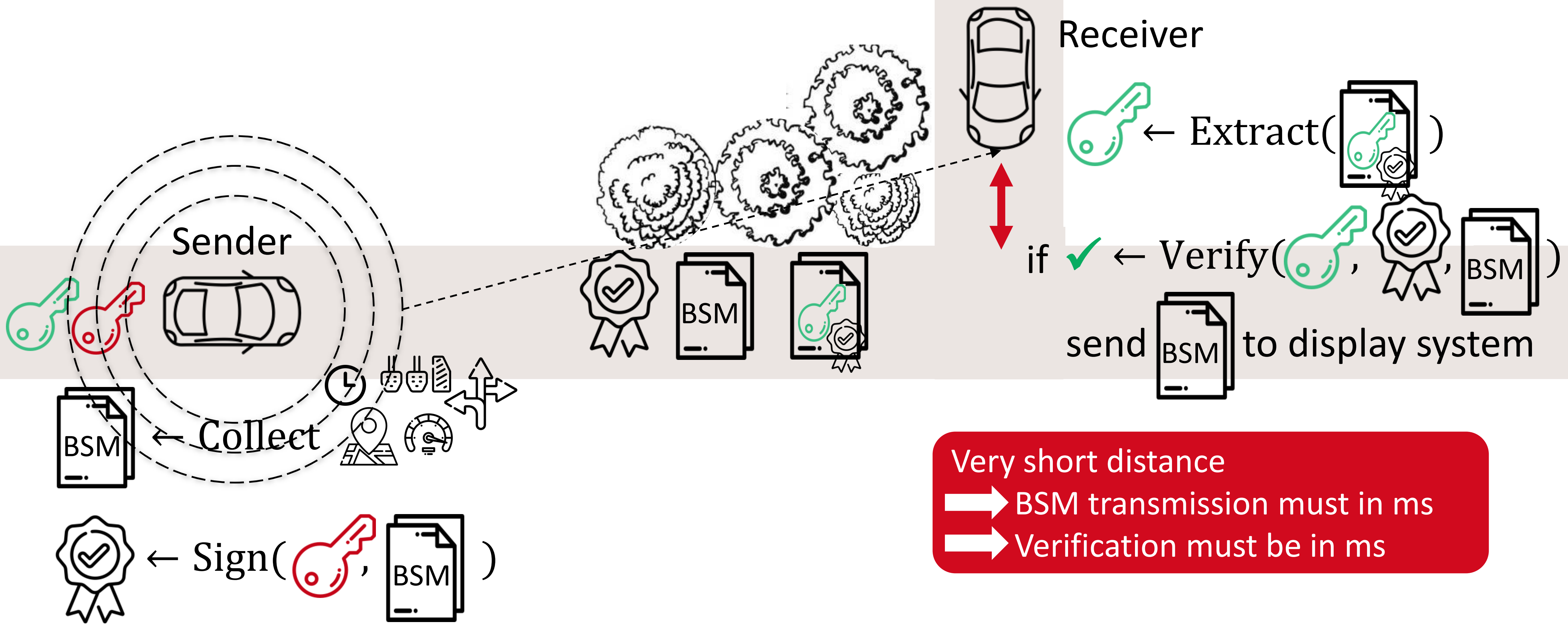
Secure BSM Exchange



Secure BSM Exchange



Secure BSM Exchange



Very short distance
- BSM transmission must in ms
- Verification must be in ms

Testbed V2Verifier

V2VERIFIER

= wireless hardware testbed for secure V2V communication [TR21]

- Based on IEEE 1609.2
- Open-source
- Written in Python

⇨ already used to find attacks and show effectiveness of mitigations [TPB+21]

[TR21] *Evaluating V2V Security on an SDR Testbed.* G. Twardokus, H. Rahbari. CNERT at IEEE INFOCOM 2021.

[TPB+21] *Targeted Discreditation Attack against Trust Management in Connected Vehicles.* G. Twardokus, J. Ponicki, S. Baker, P. Carenzo, H. Rahbari, S. Mishra. ICC 2021.

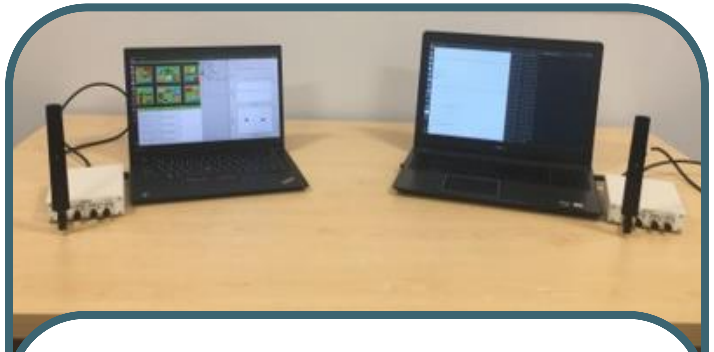
Laptop or Raspberry Pi to sign and verify BSMs

Emulates one car

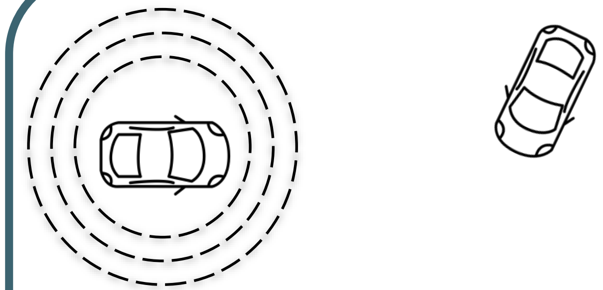
At least 2 meters apart during experiments

Software-defined radio (SDR) to send and receive signals

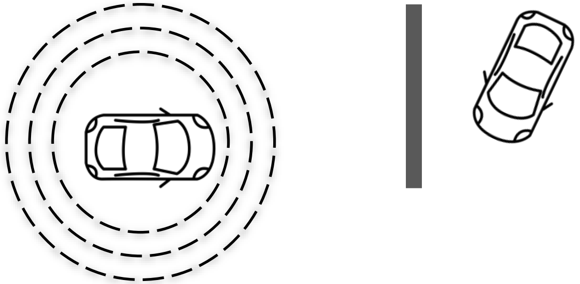
Considered Test Scenarios



Sender & Receiver static



Sender& Receiver moving, line-of-sight



Sender& Receiver moving, non-line-of-sight

Distance: at least 2 meters
Speed: 0 km/h

Distance: 2 - 300 meters
Speed: 0 - 50 km/h

Post-Quantum V2Verifier

Efficiency of Selected Schemes

Size (byte)

Cycle counts (k-cycles)

Algorithm	PK	Signature	Sign	Verify
ECDSA P-256	64	64	201	398
Dilithium-II	1 312	2 420	202	73
Falcon-512	897	666	831	141
Rainbow-I	157 800	66	4684	4913

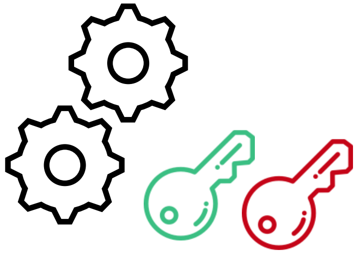
Danger of BSM loss?
Issue in jammed intersections?

Disadvantage due to slower sign?

Benefit due to faster verify?

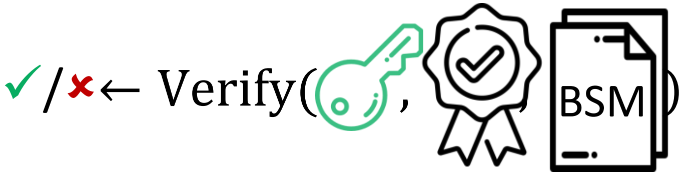
PQ EXTENSION OF V2VERIFIER

Integration of PQ signatures in V2Verifier is performed using liboqs implementations



Key generation called on demand

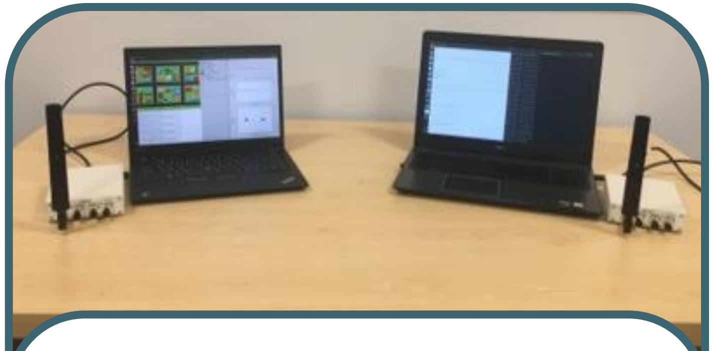
Signing is called from liboqs using Python bindings



Signature is extracted and passed to liboqs verify function

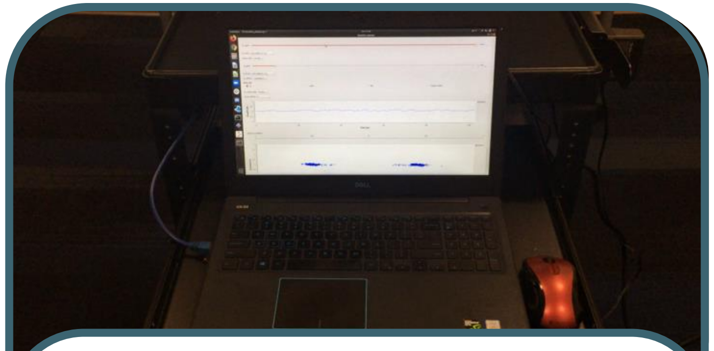
Experimental Results and Comparison

Considered Test Scenarios



Sender & Receiver static

Sender& Receiver moving, line-of-sight



Sender& Receiver moving, non-line-of-sight

Future work: test real environment with moving cars

Runtime and Sizes

Algorithm	Correct-ness	Sign (average)	Verification (average)
ECDSA P-256[1]	✓		
Dilithium-II	✗	0.063	0.054
Falcon-512	✓		
Rainbow-I	✓	1.526	1.664

⇨ Considering the fast verification, Dilithium and Falcon look like suitable replacements for ECDSA

[1]sign and verify approx., ms estimated from eBACs cycle counts

Runtime and Sizes

Algorithm	Correct-ness	Sign (average)	Verification (average)	BSM packet size[2] (bytes)	Packet loss (%)
ECDSA P-256[1]	✓				< 0.1
Dilithium-II	✗	0.063	0.054		N/A
Falcon-512	✓				< 0.1
Rainbow-I	✓	1.526	1.664		< 0.1

⇨ Considering the fast verification, Dilithium and Falcon look like suitable replacements for ECDSA

[1]sign and verify approx., ms estimated from eBACs cycle counts

Runtime and Sizes

Algorithm	Correct-ness	Sign (average)	Verification (average)	BSM packet size[2] (bytes)	Packet loss (%)
ECDSA P-256[1]	✓				< 0.1
Dilithium-II	✗	0.063	0.054		N/A
Falcon-512	✓				< 0.1
Rainbow-I	✓	1.526	1.664		< 0.1

⇨ Considering the fast verification, Dilithium and Falcon look like suitable replacements for ECDSA

2 304 byte= max. message size (IEEE 802.11p)

⇨ Signature size of Dilithium exceeds max. message size

[1]sign and verify approx., ms estimated from eBACs cycle counts

[2]included: BSM data, signature, **no** public key

Runtime and Sizes

Algorithm	Correct-ness	Sign (average)	Verification (average)	BSM packet size[2] (bytes)	Packet loss (%)	Packet size w/ explicit cert	Packet size w/ implicit cert
ECDSA P-256[1]	✓				< 0.1		
Dilithium-II	✗	0.063	0.054		N/A		--
Falcon-512	✓				< 0.1		--
Rainbow-I	✓	1.526	1.664		< 0.1		

⇨ Considering the fast verification, Dilithium and Falcon look like suitable replacements for ECDSA

2 304 byte= max. message size (IEEE 802.11p)

⇨ Signature size of Dilithium exceeds max. message size

⇨ Rainbow exceeds max. message size

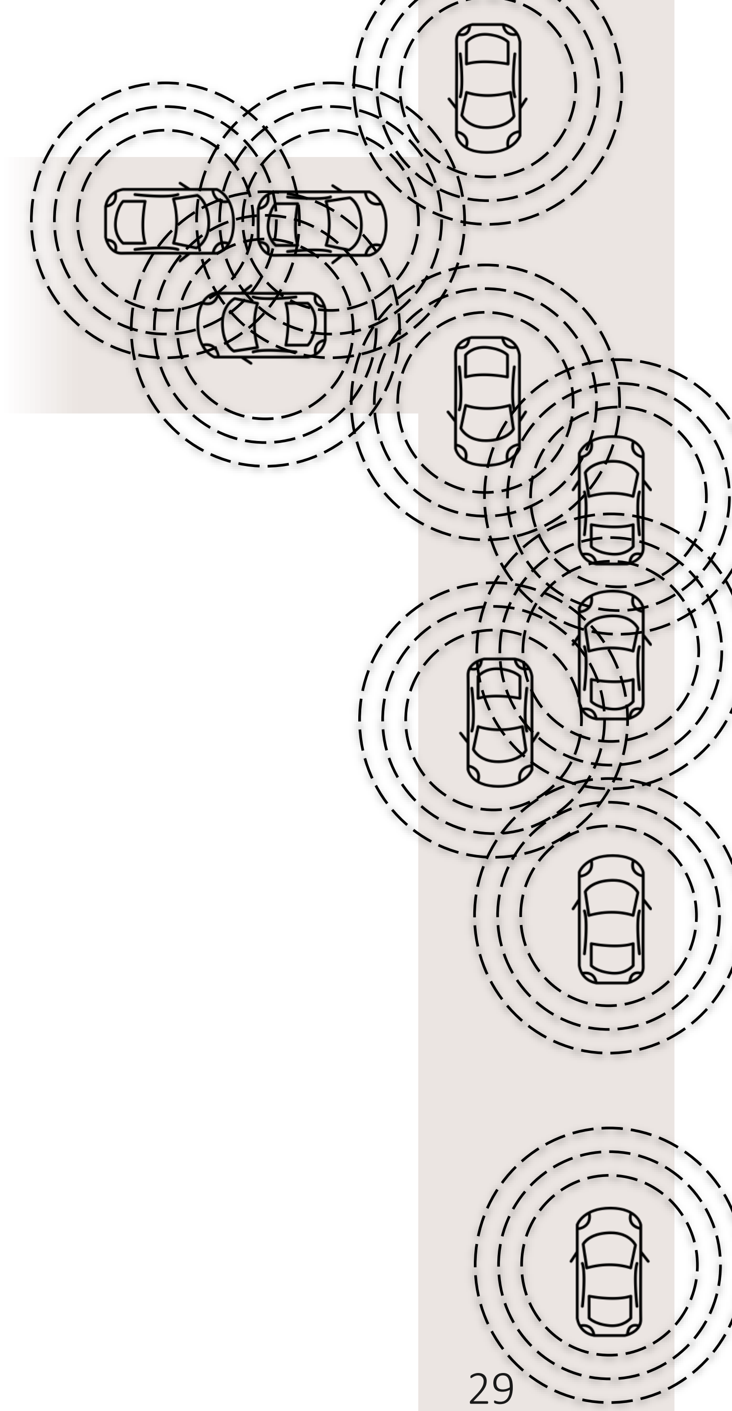
[1]sign and verify approx., ms estimated from eBACs cycle counts

[2]included: BSM data, signature, **no** public key

Analysis of Dense Environments

Dense Environments

Max number of ECDSA verifications: **2500 BSM/s**
(modern V2V equipment, e.g., Qualcomm 9150)



Dense Environments

Max number of ECDSA verifications: **2500 BSM/s**
(modern V2V equipment, e.g., Qualcomm 9150)

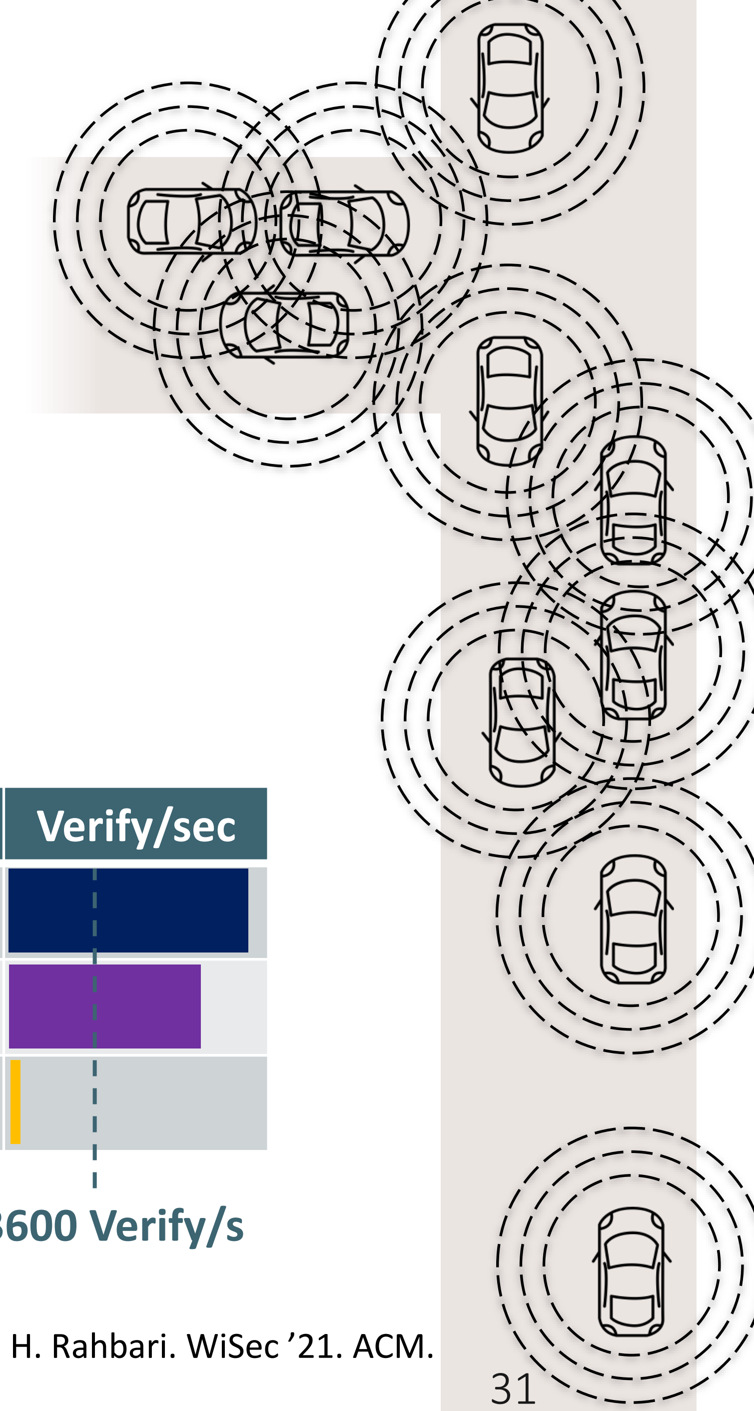
Example of dense environment: **3600 BSM/s**
peak hour on the I-490 highway, NY

- average vehicle speed: 50 mph
- vehicle spacing: 1.5 s
- Communication range: 1 km

Source under CC, Fig left
Open street map, Fig right

[1] More details in *Message Sieving to Mitigate Smart Gridlock Attacks in V2V.* S. Dongre, H. Rahbari. WiSec '21. ACM.

Dense Environments



Max number of ECDSA verifications: **2500 BSM/s**
(modern V2V equipment, e.g., Qualcomm 9150)

Example[1] of dense environment: **3600 BSM/s**
peak hour on the I-490 highway, NY

- average vehicle speed: 50 mph
- vehicle spacing: 1.5 *s*
- Communication range: 1 km

Algorithm	Correctness	Sign/sec	Verify/sec
Dilithium-II	✗		
Falcon-512	✓		
Rainbow-I	✓		

3600 Verify/s

Source under CC, Fig left
Open street map, Fig right

[1] More details in *Message Sieving to Mitigate Smart Gridlock Attacks in V2V.* S. Dongre, H. Rahbari. WiSec '21. ACM.

Future Work

Experiments on testbed

- Do benchmarks change when tested with real vehicles moving with higher speed?

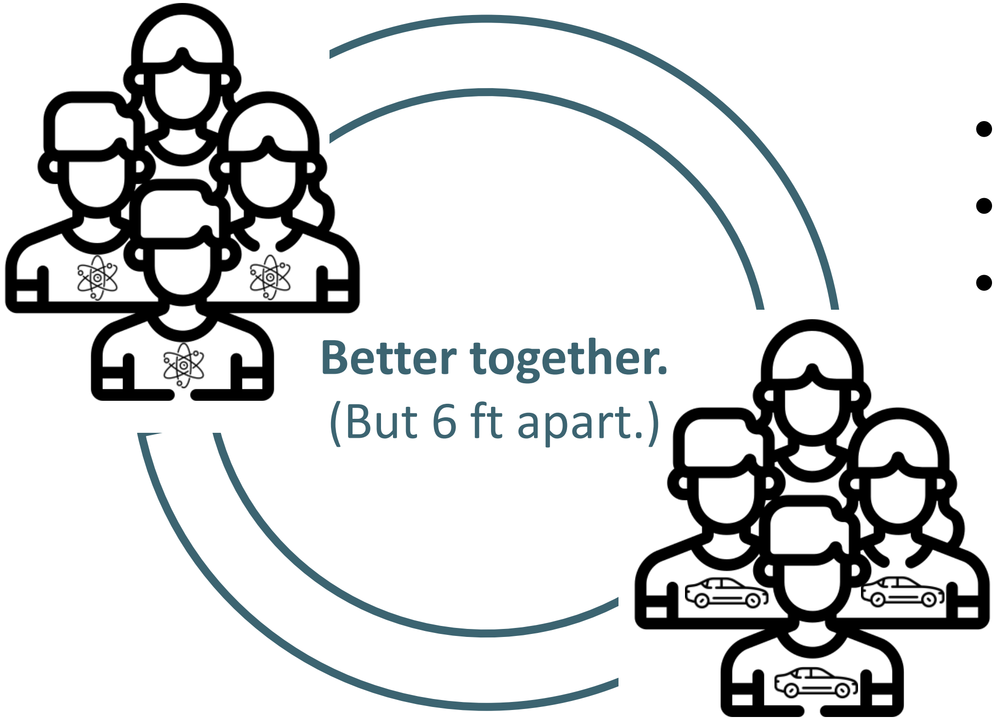
Analysis of scenarios

- How many messages can be sent at most, considering larger message sizes and faster runtimes?
- Is this number sufficient in scenarios, e.g., congested intersections?

Investigation of cert management

- Can we construct implicit certificates or alternatives from post-quantum assumptions?

Summary



- Customize post-quantum algorithms
- Adapt public-key infrastructure
- Agree on compromise between packet size and practicality/safety

Acknowledgment

Nina Bindel[1,2,3] Sarah McCarthy[3]

UNIVERSITY OF WATERLOO
IQC Institute for Quantum Computing

Hanif Rahbari[4] Geoff Twardokus[4]

RIT Rochester Institute of Technology

[1]supported by NSERC, RGPIN-2016-05146
[2]supported by NRC, program 927517
[3]supported by Public Works and Government Services Canada

[4]supported by NSA, grant H98230-19-1-0318

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Security Agency.

THANKS.