# Trustworthy Parking Communities: Helping Your Neighbor to Find a Space

Julian Timpner, Student Member, IEEE, Dominik Schurmann, € Student Member, IEEE, and Lars Wolf, Member, IEEE

*IEEE Transactions on Dependable and Secure Computing,*
vol. 13, no. 1, 2016, pp. 120-132.

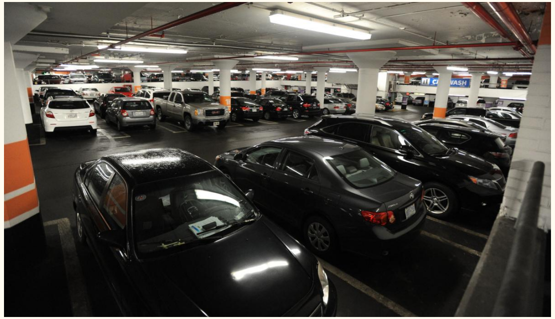
Presenter: Xuchao Zhang, Lijing Wang
2/2/2017

# Outline

- Introduction
- Related Work
- Parking Community Concept
- Attack Scenarios
- Implementation
- Discussion
- Simulation
- Conclusion

# Motivation

Drive back Home



Find a parking lot is extremely hard in city

Trustworthy Parking Communities: Helping to find a parking space

# Related Work

- Vehicular Network Fundamentals
  - ➢ ECC cryptographic fundamentals
  - ➢ ECIES (Elliptic Curve Integrated Encryption Scheme) – ECC variant using asynchronous communication
- Self-organizing Trust Models
  - ➢ Entity Oriented – modeling the trustworthiness of nodes only
  - ➢ Data Oriented – modeling the trustworthiness of data only

    *Drawback: Only ephemeral trust in data, no long-term trust relationships between nodes*

  - ➢ Hybrid Models – model trustworthiness of nodes, use the result to evaluate the data

***Contribution***: *First work of hybrid trust model with inherently trusted nodes and no additional infrastructure support*

# Related Work(cont.)

- Key Management
  - ➢ PKI (Public Key Infrastructure) – key generated by nodes; verified by additional CAs
  - ➢ Identity-based cryptography (IBC) – key pairs are generated and stored by a central trusted authority.

***Tradeoff***: *PKI achieves a limited form of anonymity, while IBC has advantage of binding keys to identifies without certificates.*

***Parking Community***: *Operate on a more abstract level and can choose most appropriate choice per use case.*

# Parking Communities

- Creating a Community

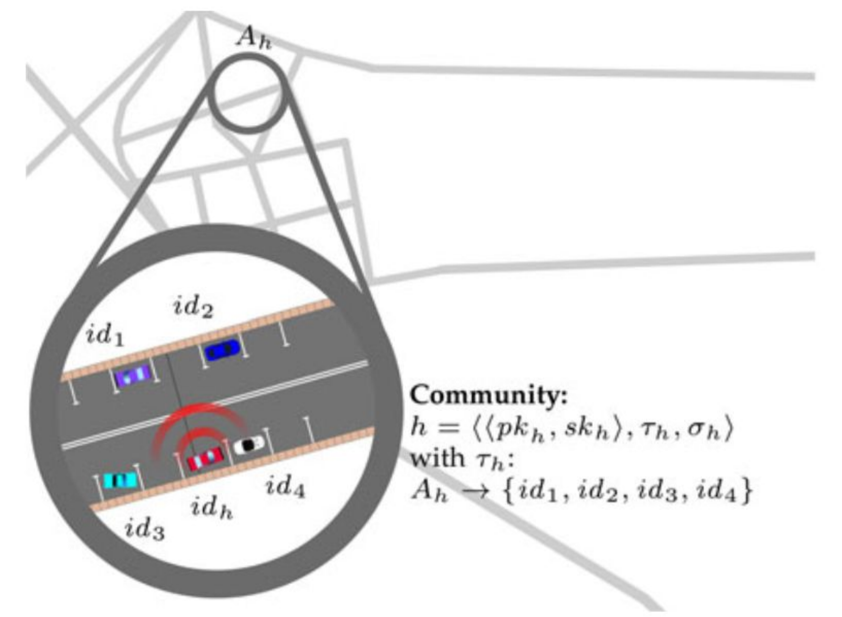
➢ A community is defined by the tuple

$$c = \langle\langle pk, sk \rangle, \tau, \sigma\rangle, \text{ with}$$

$$\tau : \mathcal{A} \to \mathcal{ID},$$

$$\sigma : ID_c \to \{r, s\}.$$

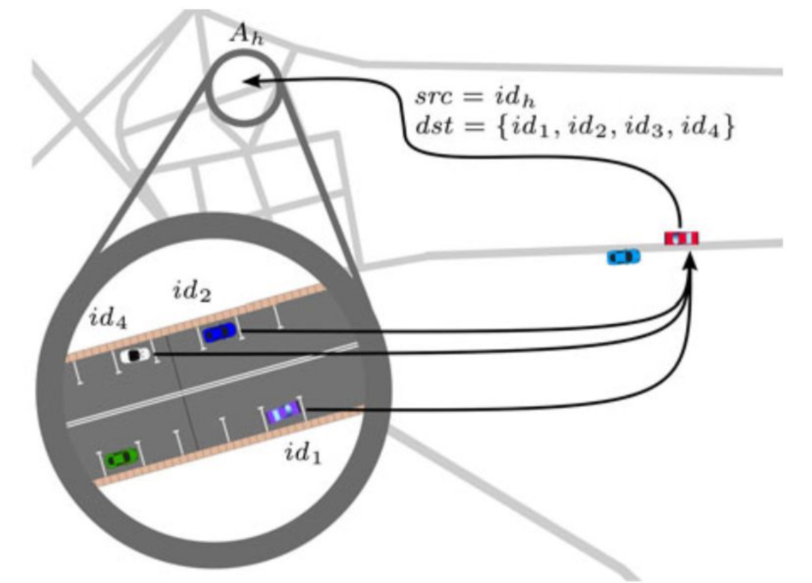
Encoding $pk_c$ directly as a vehicle's community ID, $id_c$.




(a) Collecting IDs via neighbor discovery with physical verification and establishing a trust anchor

# Parking Communities

- Querying
  - ➢ Scenario: When driving back home, previously collected IDs for $A_h$ will be queried.
  - ➢ Cryptographically signed with $h$'s private key $sk_h$.




(b) Encrypted and signed query/response for a free spot via geocast

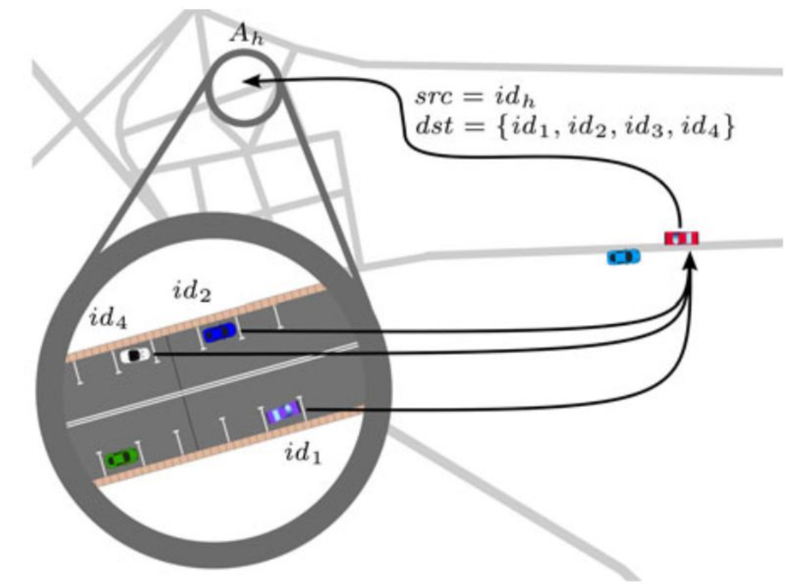
# Parking Communities

- Response
  - Each vehicle $v$ with $id_v \in ID_h$ that is located in $A_h$ (includes $id_1$, $id_2$, $id_4$) can decrypt the query
  - The response consists of an estimate $e$

$$e = \begin{cases} 1 & \text{if a space is available} \\ -1 & \text{if no space is available.} \end{cases}$$

**Estimate:** *use on-board sensor system*

  - Encrypt responses using the source ID $src$ of the message.




(b) Encrypted and signed query/response for a free spot via geocast

# Parking Communities

- Rating

  - For each community vehicle $v$, the originator keeps a count of correct and incorrect estimates: $r_v$ and $s_v$

  - Reputation rating

$$Rep_v(r_v, s_v) = E(\varphi(p|r_v, s_v)) = \frac{r_v + 1}{r_v + s_v + 2},$$

**Beta Reputation Function:**

$$\varphi(p|r, s) = \frac{\Gamma(r+s+2)}{\Gamma(r+1)\Gamma(s+1)} p^r (1-p)^s$$

  - Likelihood of a free parking spot

$$\omega = \frac{\Sigma_i^n (Rep_i(r_i, s_i) \cdot e_i)}{n}$$

**Threshold:** $\omega_{thresh} = 0$

# Parking Communities

- Prioritization

➢ Receiving vehicles can prioritize incoming queries based on the reputation rating of the originator.

➢ Vehicles receiving a query will typically favor community members over non-member requests to save resources, e.g., computing power.

***No reputation rating for non-members, so lowest priority.***

# Parking Communities

- Robustness
    - ➢ Problem: If vehicle density is sparse, there might not be sufficient vehicles in a destination area.
    - ➢ Non-members are able to respond to the query to increase the robustness of the protocol.
    - ➢ Signing but not encrypting queries also allows vehicles to query for parking spots in irregularly or newly visited locations.

***Sybil attacks become possible!!***

# Attack Scenarios

- Impersonation
  - ➢ As message is encrypted, an attacker need to generate a private key corresponding to an existing public key.
  - ➢ In case of an ECC based protocol, the success probability is $1/2^{256}$. So the attack is considered infeasible.
- Sybil Attack
  - ➢ Propose a trust on first use (TOFU) model to verify the existence of an actual vehicle for each identity used for answering parking spot queries.

*In a Sybil attack, the attacker subverts the reputation system of a peer-to-peer network by creating a large number of pseudonymous identities, using them to gain a disproportionately large influence.*

# Attack Scenarios

- Interception of Parking Spot Availability

  ➢ Without being part of the community, intercepted information is of no value for eavesdropping adversaries.

- Denial of Service

  ➢ An attacker could try to exhaust available resource of a parking vehicle by querying many many many many many times.

  ➢ Responders can decide to only answer queries originating from reputable members of their own parking community.

# Attack Scenarios

- Location Tracking

  ➢ Using a Key Derivation Function (KDF) to change pseudonyms regularly but in a deterministic and reproducible way for members of the parking community.

  ➢ A common secret is shared besides the ID during neighbor discovery.

  ➢ The secret as well as the last valid pseudonym ID are input parameters to the KDF for computing the new ID.

  ➢ The dedicated pseudonym can be change once per day to provide a mean for anonymity and location privacy.

# Implementation

- On top of existing networking stacks, implement a prototype by extending IBR-DTN, to provide integration of:
  - ECDSA and ECIES
  - key management for ECC keys
  - encoding public keys as IDs
  - trust rating model
- DTN: delay-tolerant networking
  - ID - endpoint identifier (EID)
  - Messages - bundles

# Implementation

- Crypto libraries
  - Crypto++
- Bundle Security Protocol
  - Signature scheme: ECDSA
  - Encryption scheme: ECIES
  - Only generate one key pair for signing and encrypting
  - Advs. only one public key needs to be encoded as an EID, resulting in short EIDs

# Implementation

- Key management
  - Each community's $eid_c \in EID_v$ is derived from its public key $pk$ according to:
  - $$eid_c := \text{'sec://'} \,\|\, base64url(pk).$$
  - base64url() corresponds to URL-safe Base64 encoding;
  - 'sec' is a new URI scheme indicating the SSP consists of the encoded public key instead of the typical node part and optional client/application specific parts
  - An ECC public key is 32b long. Base64 uses four characters to represent 3b, thus the length of n bytes encoded in Base64 is:
  - $$len_{ssp}(n) = \left\lceil \frac{n}{3} \right\rceil \cdot 4.$$
  - The SSP consumes 44b without the application/client specific part.

# Discussion

- A comparison of key and trust management schemes from the literature
    - Certificate-based schemes:
        - **PKI** - Public Key Infrastructure
        - **IBC** - Identity-Based Cryptography
        - **HIBC** - Hierarchical Identity-Based Cryptography
    - Incentive-based schemes: (protect against selfish behavior)
        - Barter-based
        - Credit-based
            - Virtual bank (**Bank**)
            - Self organizing (**SO**)
        - Reputation-based

TABLE 1
Comparison of Key and Trust Management Approaches

| Property | Parking Com. | Key Management: PKI[a] | Key Management: IBC[b] | Key Management: HIBC[c] | Credit/Reputation: Bank[d] | Credit/Reputation: SO[e] |
|---|---|---|---|---|---|---|
| No TTP Required | ✓ | ✘ | ✘ | ✘ | ✘ | ✓(setup) |
| Revocation/Expiry | ✓ | ✓ | ✓(expiry) | ✓(expiry) | – | – |
| Anonymity | –[g] | ✓/✘[f] | ✘ | ✓(limited) | ✘ | ✘ |
| Confidentiality | ✓/✘ | ✓ | ✓ | ✓ | – | – |
| Integrity and Authenticity | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Forward Secrecy | –[g] | ✓ | ✓(limited) | ✓(limited) | – | – |
| No Physical Encounters Required | ✘ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Required Network Connectivity | sparse | high | medium | medium | medium | sparse |
| Protocol Complexity | medium | low | low | low | medium | high |
| No Single Point of Failure | ✓ | ✘ | ✘ | ✓ | ✘ | ✓ |
| Protects against Impersonation | ✓ | ✓ | ✓ | ✓ | – | – |
| Protects against Sybil Attacks | ✓/✘ | ✓ | ✓ | ✓ | – | – |
| Protects against Selfish Behavior | ✓ | ✘ | ✘ | ✘ | ✓ | ✓ |

[a] *PKI schemes with traditional (X.509) or pseudonym certificates [1]*
[b] *IBC schemes: [20]*
[c] *HIBC schemes: [19], [21], [22], [24]*
[d] *Credit schemes, virtual bank: [49], [50], [51]*
[e] *Credit schemes, self organizing: [52]*
[f] *✓(limited): pseudonym certificates [1]; ✘: X.509 certificates*
[g] *Depending on underlying key management*
*✓/✘ Only true for specific scenarios/proposed protocols*
*–Not part of this scheme's objectives.*

# Discussion

- In summary, parking communities:
  - Does not require a security infrastructure to retrieve trust ratings
  - Offers protection against impersonation attacks despite its distributed design
  - Provides trust anchor concept to mitigate Sybil attacks
  - Allows prioritization on require/response messages to protect against selfish behaviour
  - Is a lightweight approach that integrates aspects from the wide range of existing architectures creating a novel approach for highly decentralized scenarios

# Simulation

- The ONE - Working Day Movement Model
    - Helsinki, Finland: area size is 7,000 x 8,500 $m^2$
    - Over 1,000 nodes (regular vehicles), 25% malicious nodes
    - Transmit range: 100m
    - Home zone radii: 300m
- Probability of a free spot in the home zone (the ground truth) is : 0.5
- Probability of malicious nodes lie: $\psi$=0.5
- Initial reputation rating: 0.5
- Computing a weighted consensus: ω
- Simulating 8 full days
- Repeating 10 times

Fig. 2. Map of Helsinki with artificial districts [26].

# Results

- After five days, 50% of communities have 2 to 4 members
- Values increase day by day
- Small communities - remote/isolated areas
- Large communities - densely populated areas (e.g. district A)
- Max community size: 24

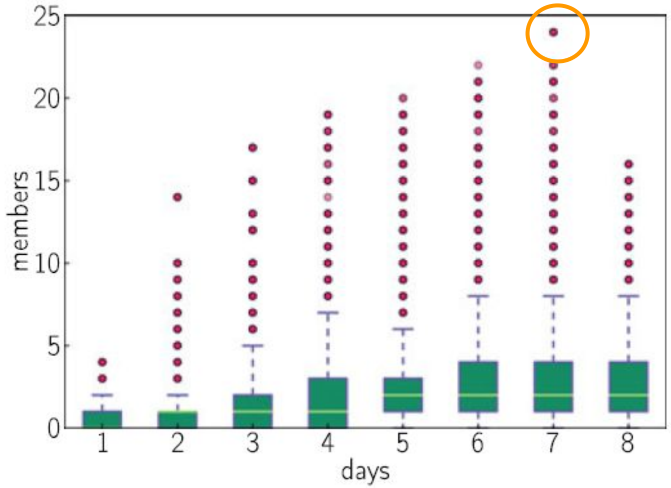



Fig. 3. Parking Community sizes.

# Results

- From day 3 on, vehicles receive average two responses
- 25% of vehicles received more responses, up to 15
- Max number of responses: 23

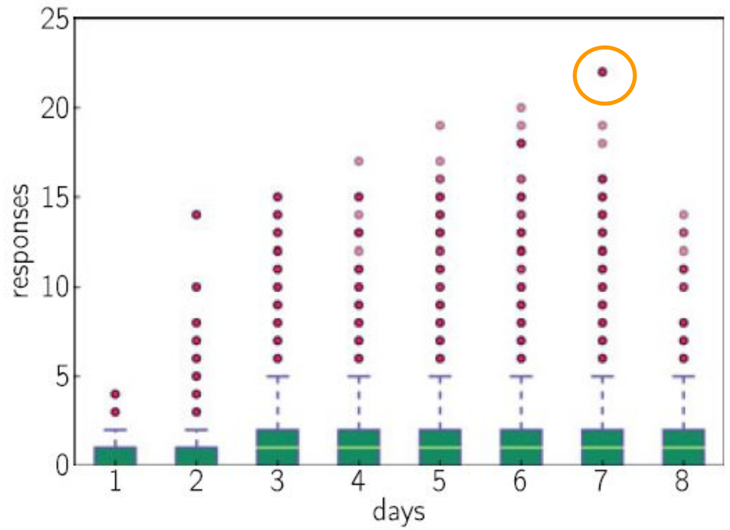
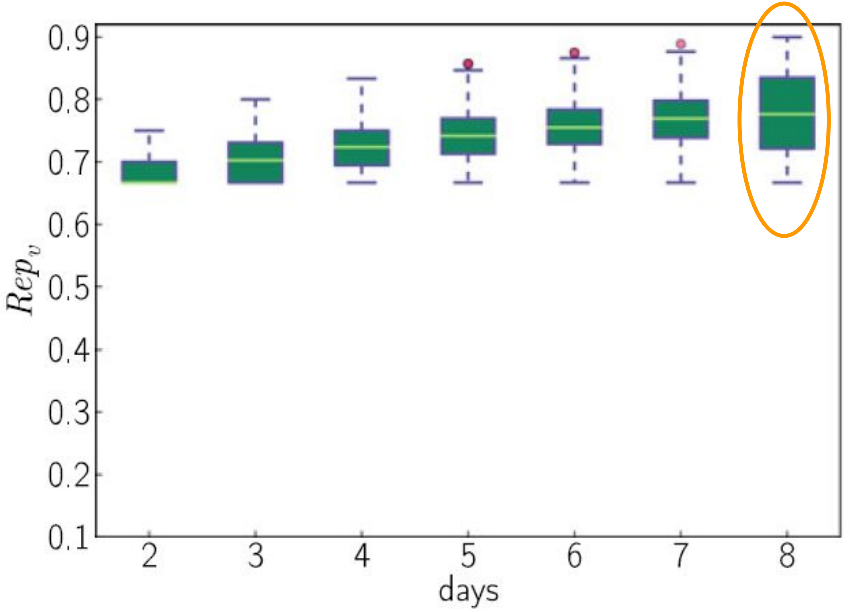



Fig. 4. Number of responses received per day.

# Results

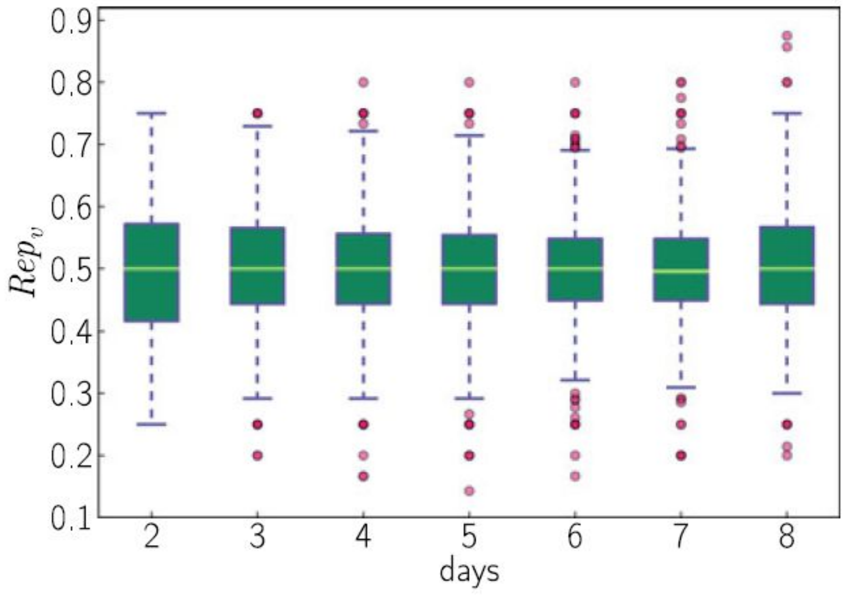
- Decentralized model
- Computing reputation *Rep(r,s)*
- Continually increases over the time
- Uprated quickly
- Large variance on the last day




(a) Reputation ratings for honest nodes per day

# Results

- Remains at 0.5 on average, with some outliers
- Malicious nodes arbitrarily lie or tell the truth (with $\psi$=0.5)

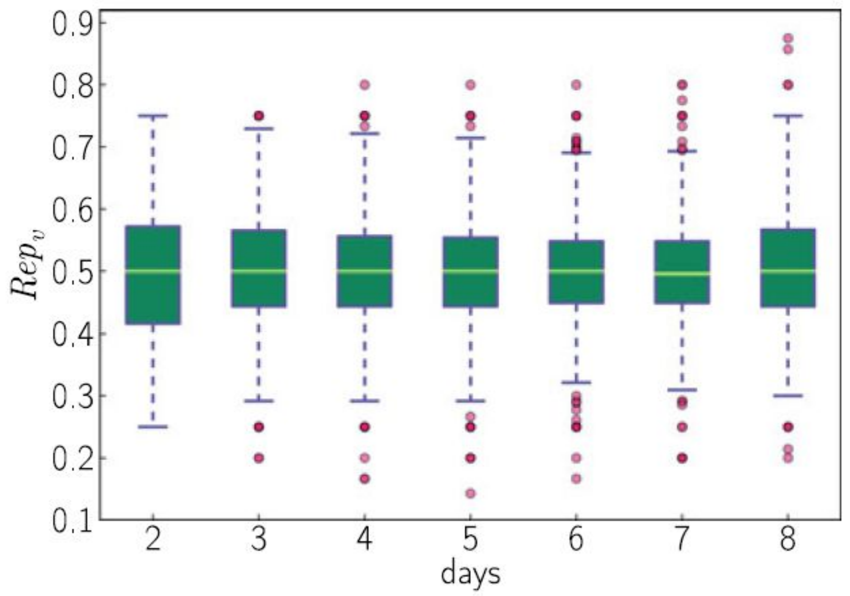


(b) Reputation ratings for malicious nodes per day

# Results

- $\psi$=0.5 vs. $\psi$=0.85

Malicious vehicles can clearly be identified and are downrated significantly from day 2 on.




(b) Reputation ratings for malicious nodes per day

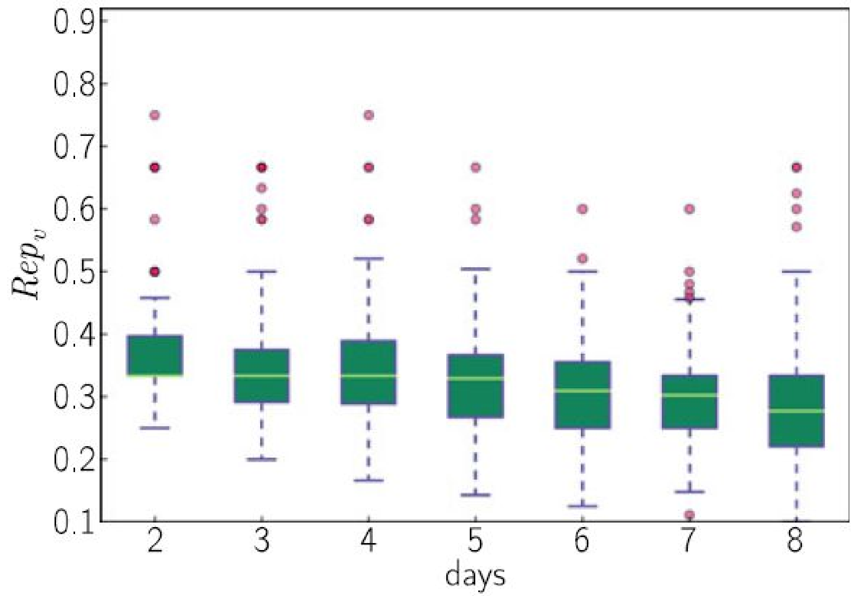



Fig. 6. Reputation for malicious nodes, $\psi = 0.85$.

# Results

- Correct decision:
  - A spot is free and $\omega \geq \omega_{thresh} = 0$
  - No spot is available and $\omega < \omega_{thresh} = 0$
- The rate of correct decisions increases over time
- Good values are achieved after only a few days, showing feasibility of the approach

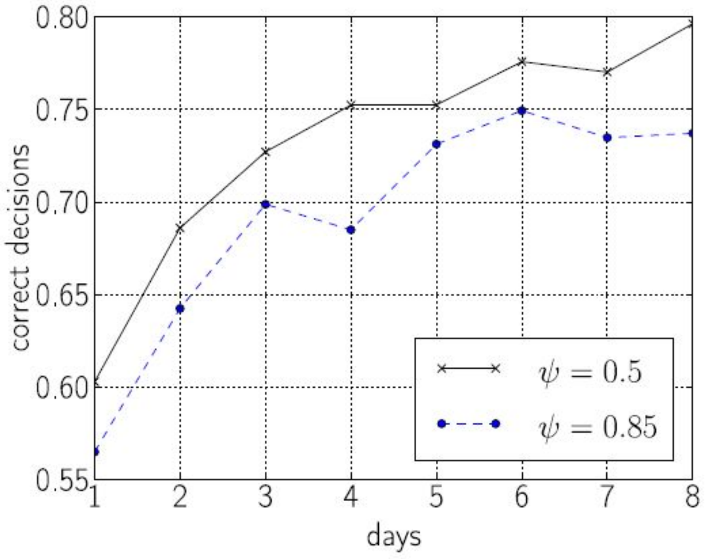



Fig. 7. Rate of correct decisions over time.

# Conclusion

- Parking community:
  - A **novel trust management**, without reliance on a central TTP for retrieving trust ratings
  - **Trust anchors** enable signed and encrypted request-response communication in disrupted environments
  - Based on high-performance state-of-the-art encryption and signature algorithms, in particular **ECC**, as well as a well-understood mathematical **trust rating model**
- Outstandings:
  - Provided protection against impersonation and Sybil attacks utilizing trust anchors and physical verification
  - Implemented in open-source IBR-DTN
  - Compared with existing key and trust management schemes
  - Simulated with the ONE

# Q & A

Thanks!