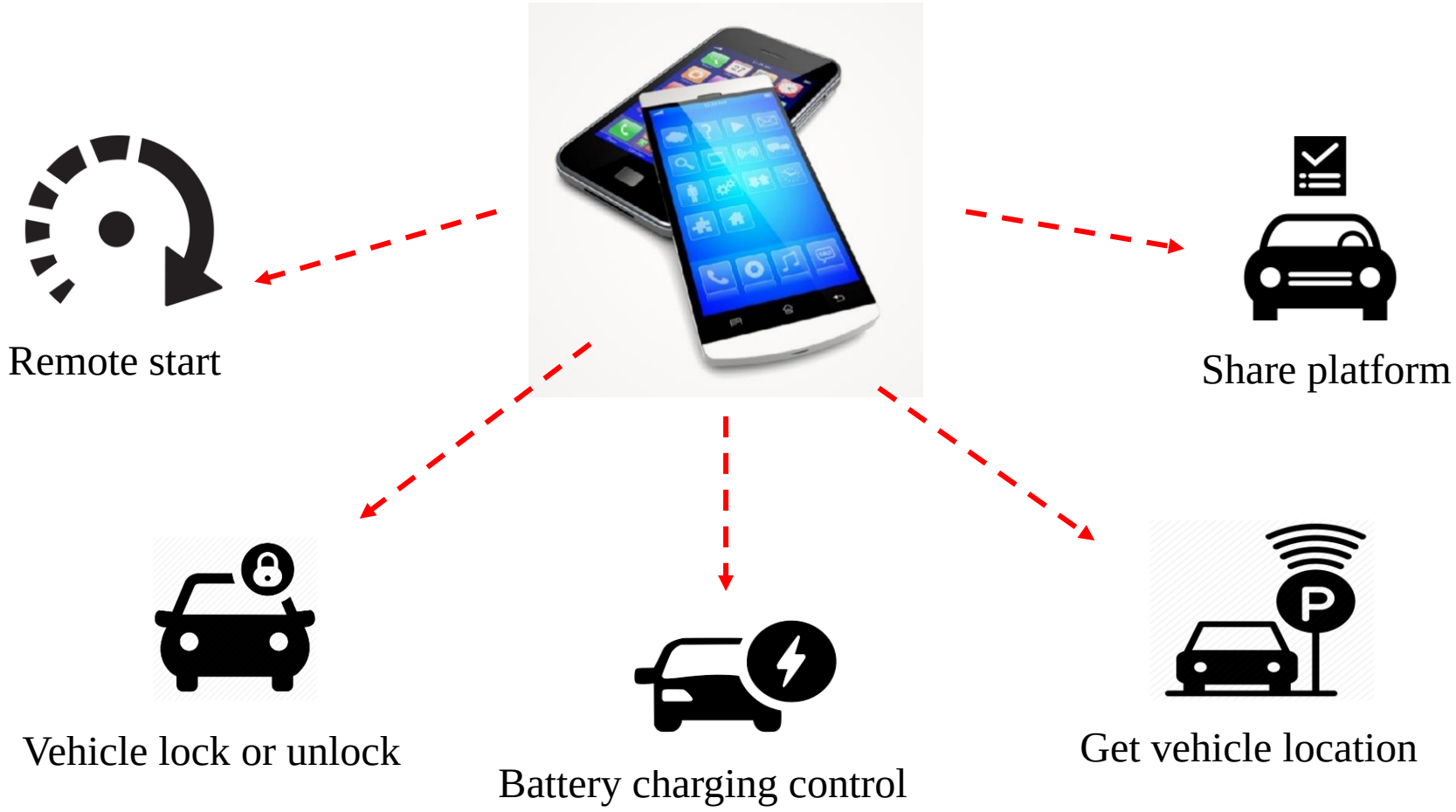
ICCPS 2019

Preventing Battery Attacks on Electric Vehicles based on Data-Driven Behavior Modeling

Liuwang Kang and Haiying Shen
Department of Computer Science, University of Virginia

Background

Users control an electric vehicle (EV) remotely through a smartphone



https://www.windowscentral.com/official-viper-smartstart-app-now-available-windows-phone
https://www.pcmag.com/news/353026/how-to-use-a-smartphone-app-to-find-your-car
https://www.123rf.com/photo_100862604_hand-with-phone-on-a-background-of-electric-car-charging-point-on-a-device-screen-indicator-of-power.html

Background

A malicious adversary may attack batteries in EVs through a smartphone:

- Modify battery current during the battery charging process
- Consume battery energy without user's notice

Battery over-heat or explosion

Driving plan change

Driving range anxiety

https://www.dailymail.co.uk/news/article-4679416/Electric-car-left-charging-overnight-destroyed-fire.html
https://insideevs.com/german-psychologist-says-range-anxiety-vanishes-after-3-months-of-driving-electric/
http://learn.pechnol.com/2018/02/24/electric-vehicles-range-anxiety/

Background

Battery Management System in EVs

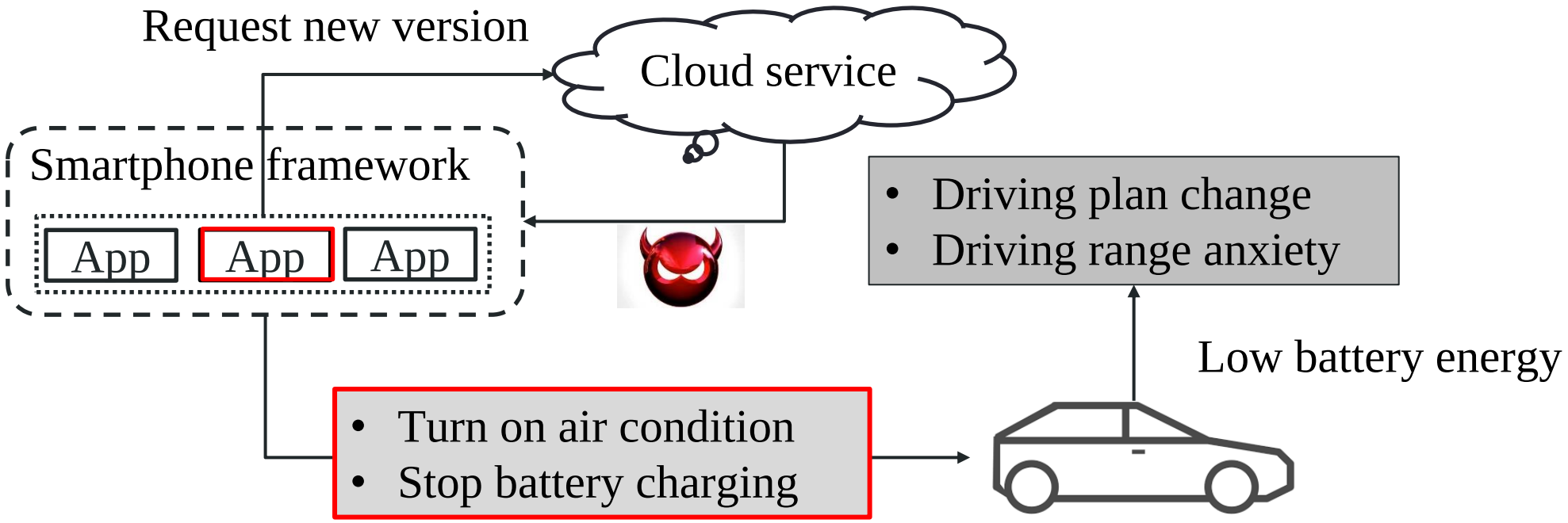
➢Monitors battery voltage and current during charging and discharging processes to avoid over-high battery temperature and battery explosion

- Malicious battery charging current changes ✓
- Unexpected battery energy consumption ✗

Focus on attacks which can result in unexpected battery energy consumption

Battery Attack

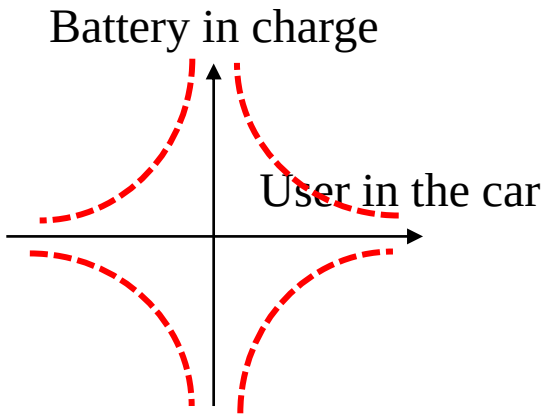
Attack batteries in EVs by remotely turning on AC or stopping battery charging process through a smartphone



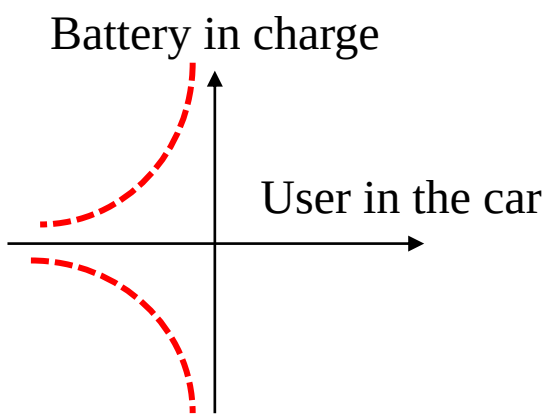
Battery Attack

Battery Management System in EVs

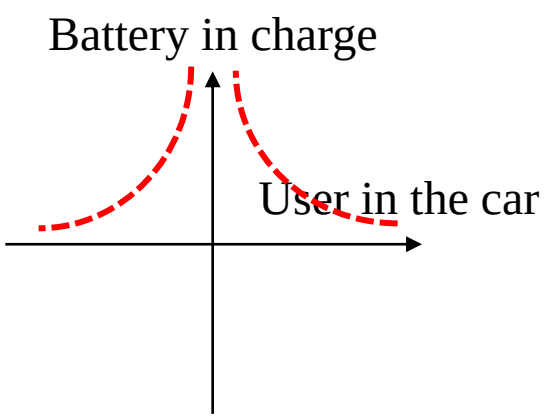
- **No-effort attack** sends malicious action requests randomly without considering whether a user is in the vehicle
- **Smart attack** sends malicious action requests only when the user is not in the vehicle



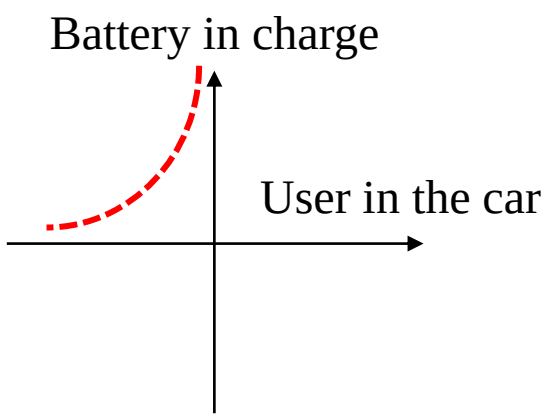
AC-turn-on
(No-effort attack)



AC-turn-on
(Smart attack)



battery-charge-stop
(No-effort attack)



battery-charge-stop
(Smart attack)

Related Work

- Several methods [SECURITY'11, NDSS'14] try to remove malicious Apps from centralized mobile marketplaces

➢Malware authors keep developing new methods to help malicious Apps penetrate into marketplaces [CS'18]

- Other methods [NDSS'11, TOCS'14, TI'18] develop frameworks in the smartphone operating system to provide security protection

➢Frameworks are not always trustworthy and malicious adversaries can play as privileged system daemons to attack smartphones [CRC'2018]

Challenges

Propose a battery authentication system (Bauth) to authorize AC-turn-on and battery-charge-stop requests based on user's habits

Challenge 1: How to build an user behavior model to accurately describe a user's habits?

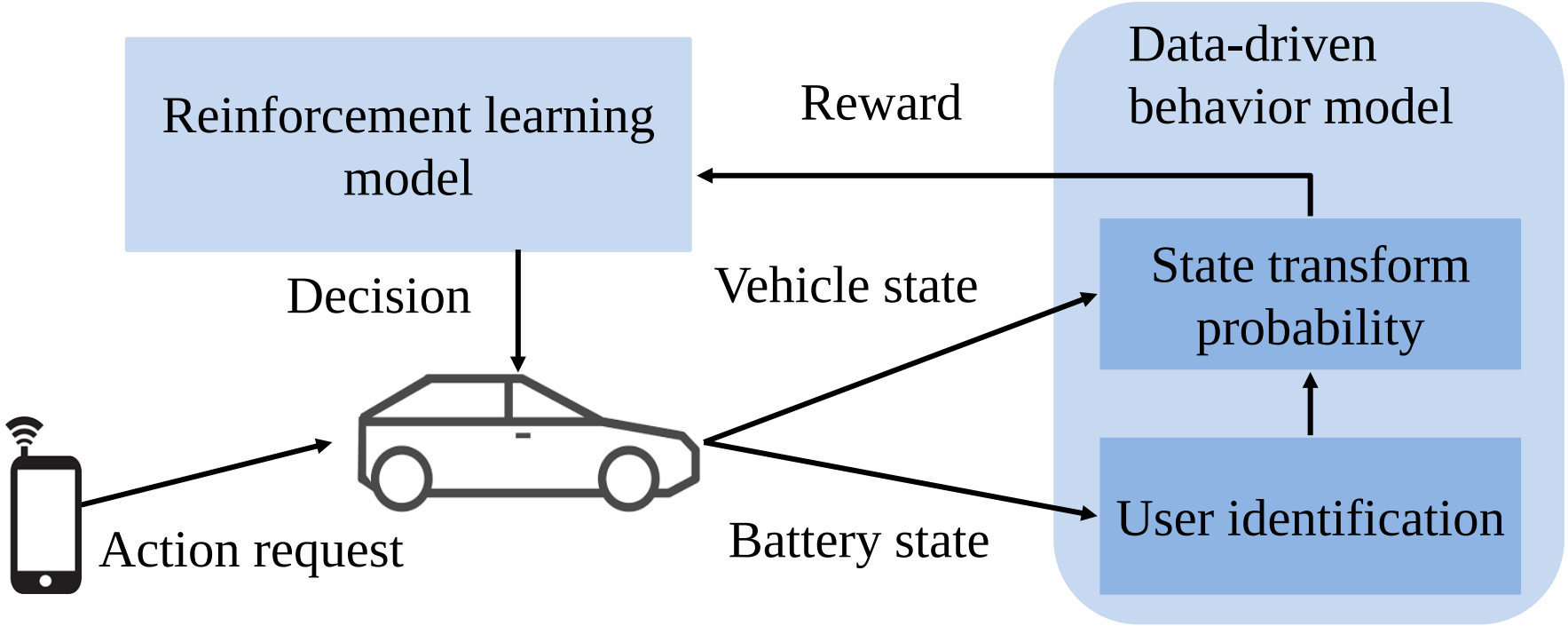
- A user's habit is affected by vehicle driving environments
- The situation where more than one users share one EV exists

Challenge 2: How to make a correct authentication decision based on the user behavior model?

- The user behavior model only provides statistical probabilities of user behaviors (accept or reject) under different vehicle states

Battery Authentication System (Bauth)

Bauth detects malicious AC-turn-on and battery-charge-stop requests from a smartphone



Vehicle state: vehicle indoor temperature and battery SOC
Battery state: current, battery power and battery SOC

Challenge 1

How to build an user behavior model to accurately describe a user’s habits?

Data-driven Behavior Model

Describe a user's habits in turning on AC or stopping charging battery based on the recorded vehicle usage data

- Situations where more than one users share one EV exists

➢ User identification

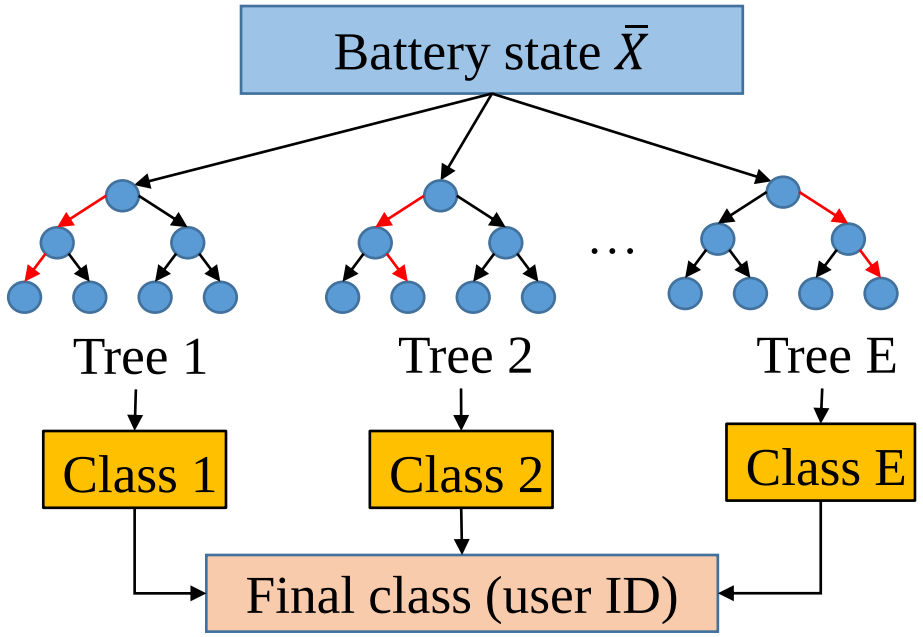
- A user's habit changes under different vehicle driving environments

➢ State transform probability calculation

Data-driven Behavior Model

User identification

Apply random forest technology to identify the user based on real-time battery state (battery current - c, battery power - p, and battery State-of-Charge - SOC)



Process battery state $x = [c, p, SOC]$:

- Normalize battery state: $X = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)}$
- Calculate average value $\bar{X}$ of X from $t - \Delta$ to t

Data-driven Behavior Model

State transform probability calculation

Probability of state transform from one vehicle state to another vehicle state when the user conducts an action

- Step 1: Calculate the number of the action d_t at vehicle state s_t in the historical vehicle usage data

$$B_{s_t,d_t} = \sum_{(s',d')\in M} \delta_{s_t,s'}\delta_{d_t,d'}$$

where $\delta_{s_t,s'}$ and $\delta_{d_t,d'}$ equal to 1 if s_t and d_t are the same as s' and d'

- Step 2: Normalize statistical matrix B to obtain statistical probability matrix $\bar{B}$

$$\bar{B}_{s_t,d_t} = \frac{B_{s_t,d_t}}{\sum_{d_t\in\{0,1\}} B_{s_t,d_t}}$$

Data-driven Behavior Model

State transform probability calculation

Probability of state transform from one vehicle state to another vehicle state when the user conducts an action

- Step 3: Obtain state transform matrix D by calculating the probability of state transform from s_t to s_{t+1} with action d_t

$$D_{s_t,d_t,s_{t+1}} = \sum_{s' \in S} \delta_{(s_t \to s_{t+1}),(s_t \to s')} \bar{B}_{s_t,d_t}$$

- Step 4: Normalize state transform matrix D to get state transform probability matrix $\bar{D}$

$$\bar{D}_{s_t,d_t,s_{t+1}} = \frac{D_{s_t,d_t,s_{t+1}}}{\sum_{s_{t+1} \in S} D_{s_t,d_t,s_{t+1}}}$$

Data-driven Behavior Model

Example: State transform probability calculation

Calculate the number of each action

Normalization

Vehicle usage data

	The vehicle state	d_t	Subsequent state
s_1	[60.02, 0.953]	0	[60.02, 0.953]
s_2	[60.02, 0.953]	0	[60.03, 0.953]
s_3	[60.03, 0.953]	1	[60.03, 0.954]
s_4	[60.03, 0.954]	1	[60.03, 0.955]
s_5	[60.03, 0.955]	0	[60.03, 0.955]
s_6	[60.03, 0.955]	1	[60.03, 0.956]

Statistical matrix B

S	$B_{s,0}$	$B_{s,1}$
[60.02, 0.953]	2	0
[60.03, 0.953]	0	1
[60.03, 0.954]	0	1
[60.03, 0.955]	1	1

Statistical probability matrix $\bar{B}$

S	$\bar{B}_{s,0}$	$\bar{B}_{s,1}$
[60.02, 0.953]	1	0
[60.03, 0.953]	0	1
[60.03, 0.954]	0	1
[60.03, 0.955]	0.5	0.5

Data-driven Behavior Model

Example: State transform probability calculation

Calculate probability

Normalization

Vehicle usage data

The vehicle state	d_t	Subsequent state
[60.02, 0.953]	0	[60.02, 0.953]
[60.02, 0.953]	0	[60.03, 0.953]
…	…	…
[60.03, 0.955]	1	[60.03, 0.956]

State transform $D_{s_t,0,s_{t+1}}$

s_t \ s_{t+1}	s^1	s^2	s^3	s^4
s^1	1	1	0	0
s^2	0	0	0	0
s^3	0	0	0	0
s^4	0	0	0	0.5

$\bar{D}_{s,0,s_{t+1}}$

s_t \ s_{t+1}	s^1	s^2	s^3	s^4
s^1	0.5	0.5	0	0
s^2	0	0	0	0
s^3	0	0	0	0
s^4	0	0	0	1

Statistical probability matrix $\bar{B}$

	S	$\bar{B}_{s,0}$	$\bar{B}_{s,1}$
s^1	[60.02, 0.953]	1	0
s^2	[60.03, 0.953]	0	1
s^3	[60.03, 0.954]	0	1
s^4	[60.03, 0.955]	0.5	0.5

$D_{s_t,1,s_{t+1}}$

s_t \ s_{t+1}	s^1	s^2	s^3	s^4
s^1	0	0	0	0
s^2	0	0	1	0
s^3	0	0	0	1
s^4	0	0	0	0

$\bar{D}_{s_t,1,s_{t+1}}$

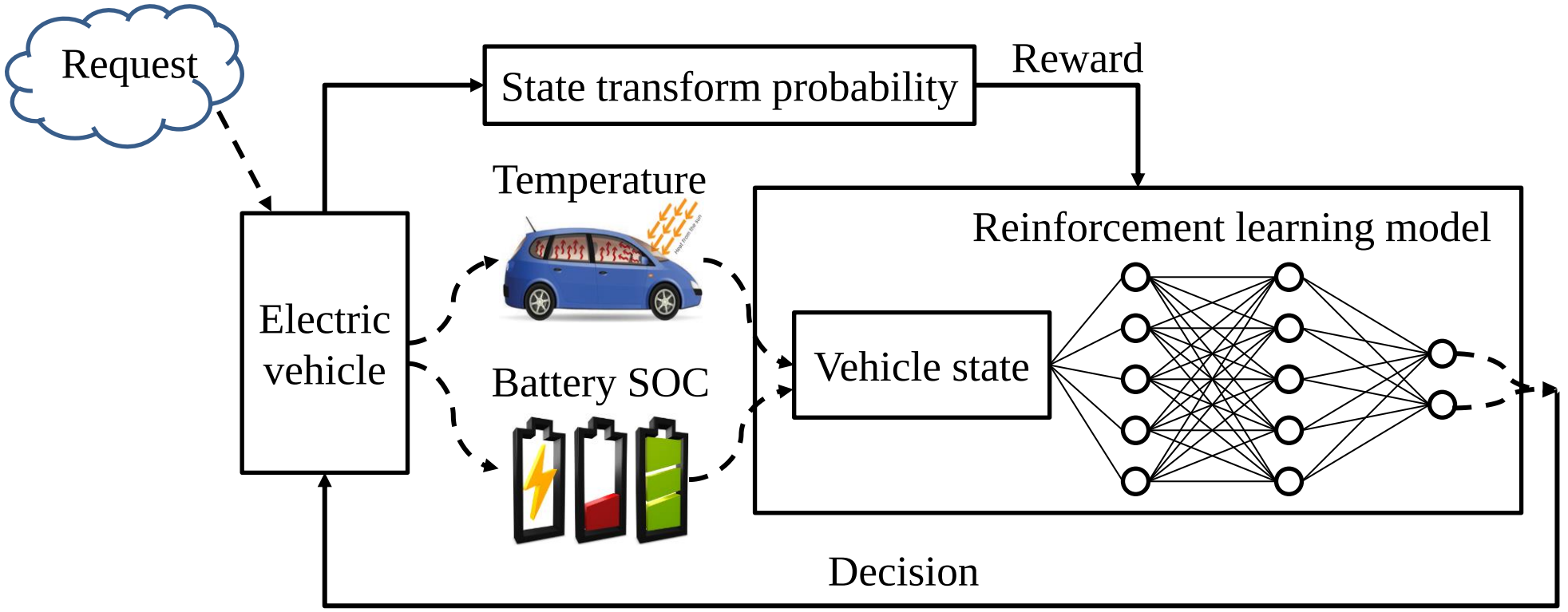
s_t \ s_{t+1}	s^1	s^2	s^3	s^4
s^1	0	0	0	0
s^2	0	0	1	0
s^3	0	0	0	1
s^4	0	0	0	0

Challenge 2

How to make a correct authentication decision based on the user behavior model?

Reinforcement Learning based Authentication

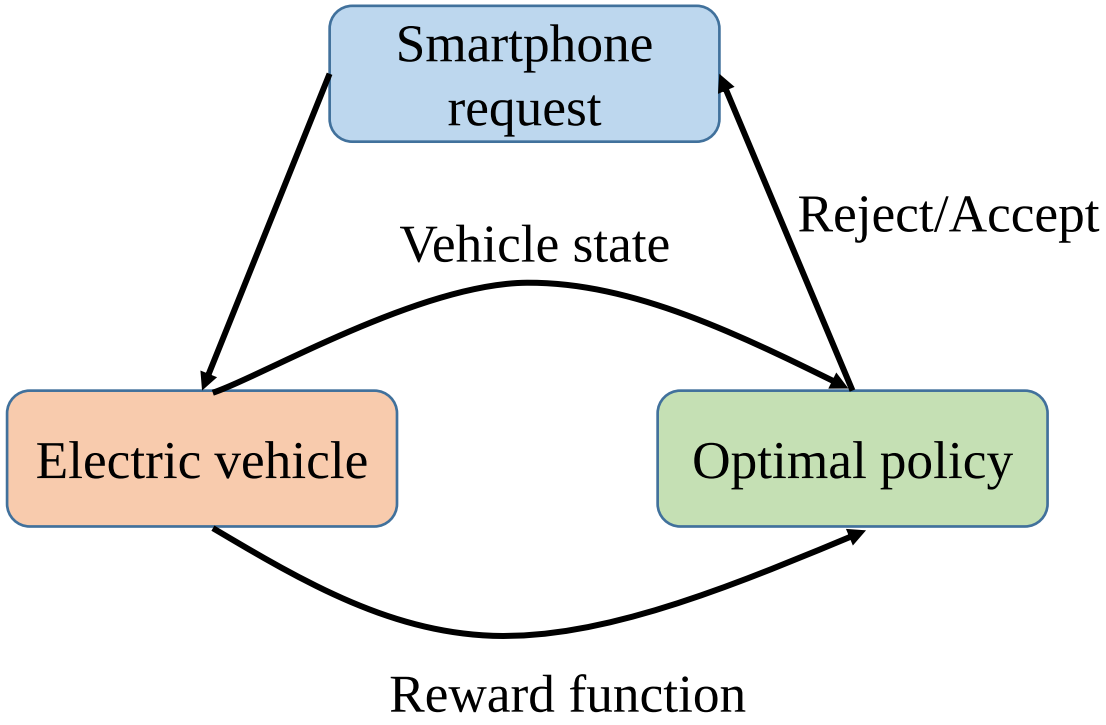
Bauth uses Reinforcement learning model to make authentication decisions on requests



Reinforcement Learning based Authentication

Reinforcement learning model mechanism

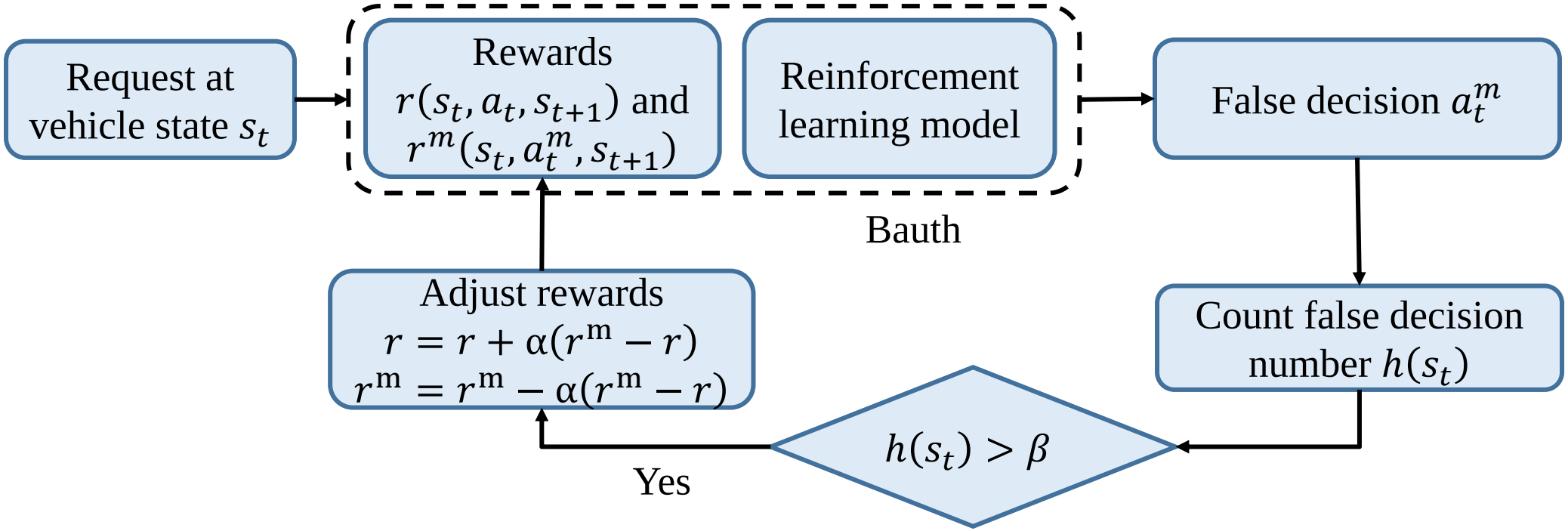
- Reward function
  - ➢ Defined as $r(s_t, a_t, s_{t+1})$ and equals to state transform probability in the user behavior model
- Optimal policy
  - ➢ Defined as one map $\pi : s_t \rightarrow a_t$ and guides to make authentication decisions by maximizing the expected cumulative rewards
  - ➢ Use neural network [ICML'2016] to form the optimal policy



Reinforcement Learning based Authentication

Reward self-adjustment method

- Reward function is built by statistically analyzing historical vehicle usage data
- Data samples are limited in practice, which results in less accurate reward function



Performance Evaluation

Experiment settings

- Drive 7 EVs for total 25 days (21-day data for training and 4-day data for testing)
- Implement Bauth in a laptop
- Install one vehicle App into a smartphone to send malicious requests with 0.1 $time/second$

Comparison method

- Statistical Method (SM) obtains statistical probabilities of the user's actions under different vehicle states and utilizes statistical probabilities to authorize action requests

Performance Evaluation

Evaluation aspects

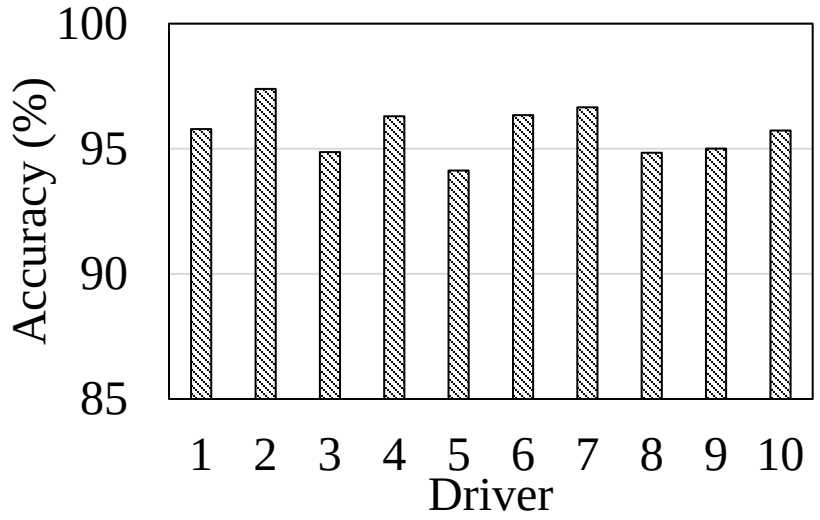
- How is Bauth’s performance in identifying different users?
- How is Bauth’s performance in attack detection accuracy?
- How is the effectiveness of user identification and reward adjustment to improve Bauth’s attack detection accuracy?
- Is Bauth effective in different vehicle usage cases?

Performance Evaluation

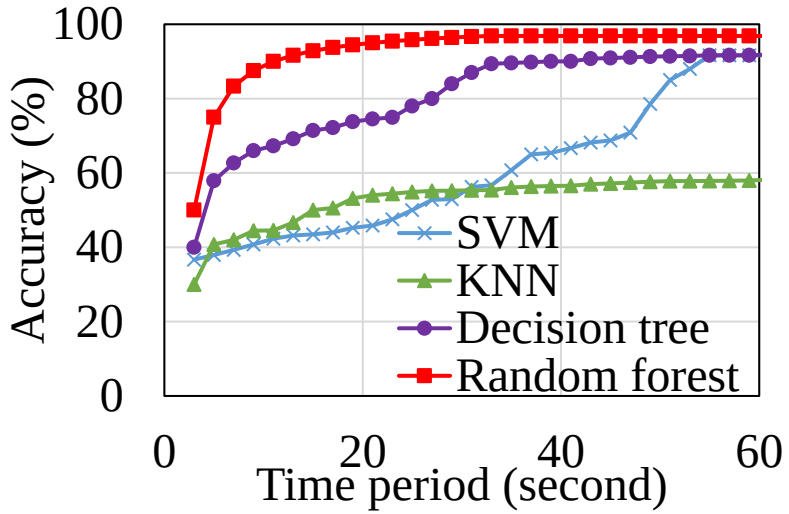
User identification evaluation

Total 10 participants drove one EV on a 7.4 mile long road

- Average identification accuracy reaches around 95%
- Random forest method has higher identification accuracy



User identification accuracies among participants

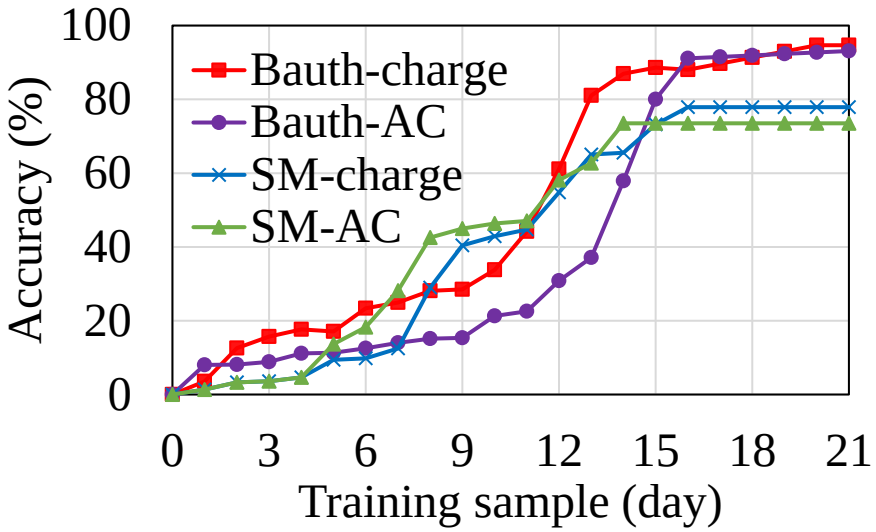


User identification accuracy comparison

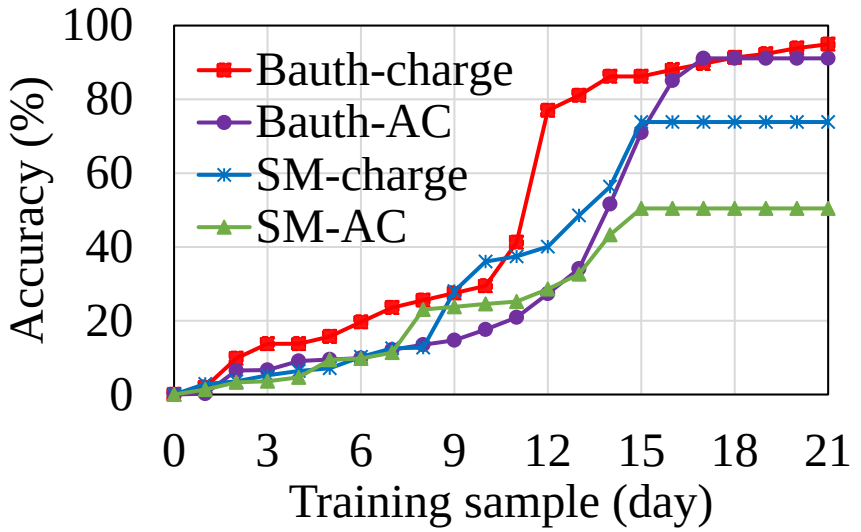
Performance Evaluation

Effects of training samples on attack detection accuracy

- Bauth has more than 20% higher attack detection accuracy on both No-effort attacks and Smart attacks than SM
- Bauth's detection accuracy keeps increasing while SM's detection accuracy keeps constant as more than 15-day samples are used for training



Attack detection accuracy for No-effort attacks

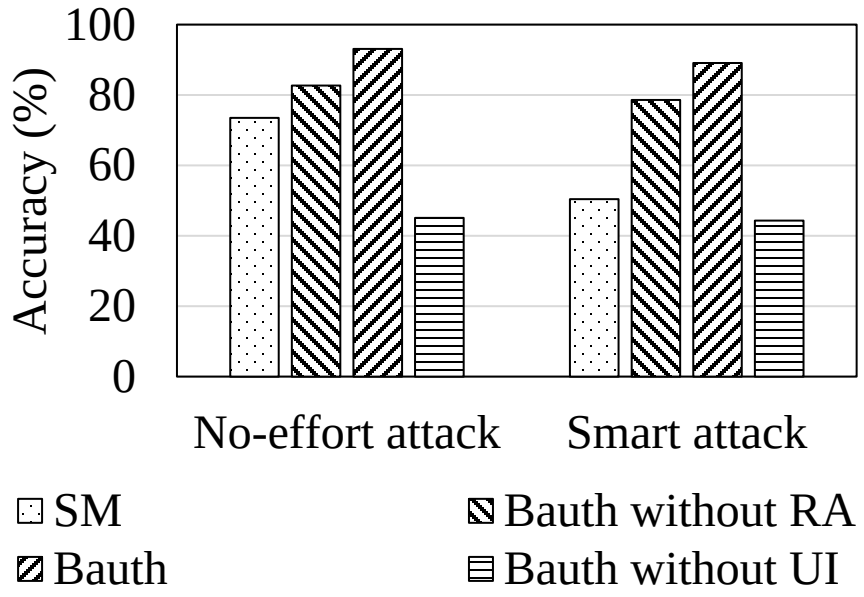


Attack detection accuracy for Smart attacks

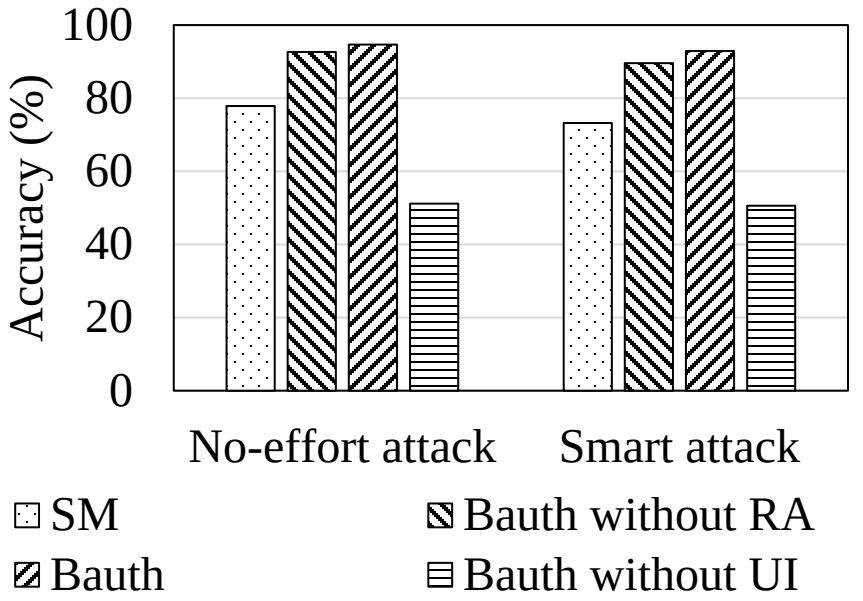
Performance Evaluation

Effects of user identification (UI) and reward adjustment (RA)

- Attack detection accuracies are reduced greatly for Bauth without UI because of multi-user sharing one vehicle situation
- Bauth without RA has lower attack detection accuracy because of less accurate rewards



Malicious AC-turn-on request detection

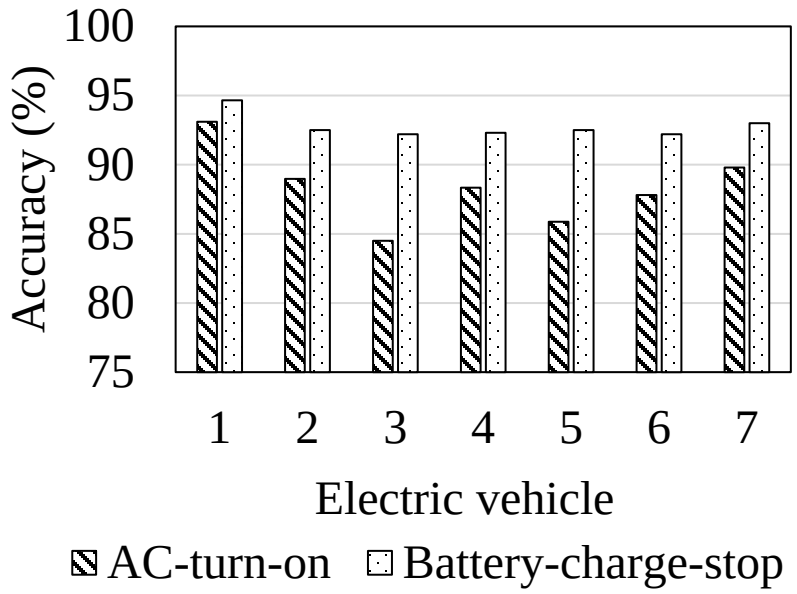


Malicious battery-charge-stop request detection

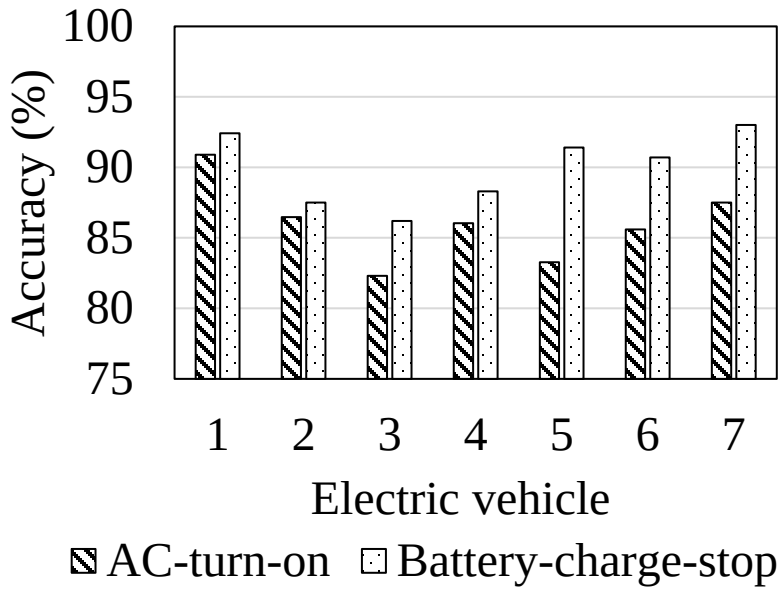
Performance Evaluation

Attack detection accuracies for different vehicle usage cases

- Average attack detection accuracies reach 91% and 88% for No-effort and Smart attacks
- Lower attack detection accuracy on AC-turn-on requests since the user considers both indoor temperature and SOC



Attack detection accuracy for No-effort attacks



Attack detection accuracy for Smart attacks

Summary

Propose Bauth to authorize requests from a smartphone to ensure EV battery security:

- Build a data-driven behavior model to describe user's habits
- Apply reinforcement learning model to authorize requests from the smartphone
- Conduct real EV driving experiments to verify Bauth

Future work:

- Consider more driving environments to improve attack detection accuracy
  - ➢ Humidity and wind speed
- Detect other malicious request types
  - ➢ Turn on headlights or rain wiper

Thank you!