Authenticating Drivers Using Automotive Batteries

LIANG HE, University of Colorado Denver
YUANCHAO SHU, Microsoft Research
YOUNGMOON LEE, Hanyang University
DONGYAO CHEN, University of Michigan at Ann Arbor
KANG G. SHIN, University of Michigan at Ann Arbor

Automakers have been improving, or even trying to replace, key-based driver authentication solutions, owing to their vulnerability to cyber attacks and single-point-of-failures, as well as their inability of driver identification. In line with this effort, we design a novel driver authentication system using automotive batteries, called *Batteries-as-Authenticators* (BAuth), to mitigate the limitations of key-based solutions by providing a second-factor authentication. BAuth is an add-on module installed between vehicles and their batteries, which uses the batteries as *sensors* to validate drivers' identities and *actuators* to enable/disable the cranking of vehicle's engine. We have prototyped and evaluated BAuth on 6 regular/hybrid/electric vehicles. Our evaluation shows BAuth to authenticate the drivers with a 98.17 (2.84)% averaged true (false) positive rates and tolerate the dynamics caused by the aging/temperature/state-of-charge of batteries. Our user study corroborates BAuth's attractiveness to car owners.

CCS Concepts: • **Computer systems organization → Sensors and actuators**.

Additional Key Words and Phrases: Driver authentication, multi-factor authentication, batteries as authenticators

ACM Reference Format:
Liang He, Yuanchao Shu, Youngmoon Lee, Dongyao Chen, and Kang G. Shin. 2020. Authenticating Drivers Using Automotive Batteries. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4, Article 130 (December 2020), 27 pages. https://doi.org/10.1145/3432198

1 INTRODUCTION

Background. Key-based driver authentication systems [18, 21, 29] have been pervasively deployed on 1.32 billion vehicles in the world [69] to secure the interactions between vehicles and their drivers by allowing the engine to start only upon successful authentication. These key-based driver authentication systems can be categorized into two classes:

(1) *Metal keys:* a physical medium to match the key with the vehicle, but is vulnerable to hot-wiring [23].

Authors' addresses: Liang He, liang.he@ucdenver.edu, University of Colorado Denver, 1380 Lawrence Street, Denver, CO, 80204; Yuanchao Shu, yuanchao.shu@microsoft.com, Microsoft Research, Microsoft Building 99, Redmond, WA, 98052; Youngmoon Lee, youngmoonlee@hanyang.ac.kr, Hanyang University, 55 Hanyangdaehakro Sangnokgu, Ansansi, 15588; Dongyao Chen, dychen@umich.edu, University of Michigan at Ann Arbor, 2260 Hayward St., Ann Arbor, MI, 48109; Kang G. Shin, kgshin@umich.edu, University of Michigan at Ann Arbor, 2260 Hayward St., Ann Arbor, MI, 48109.

2474-9567/2020/12-ART130 $15.00
https://doi.org/10.1145/3432198

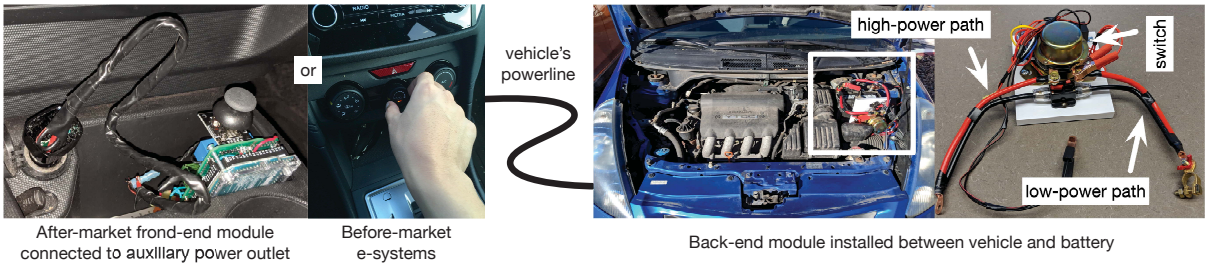


Fig. 1. Prototype of BAuth and its installation on 2008 Fit.

(2) *RF-integrated keys (or key-fobs):* integrating an RF chip with the key prevents hot-wiring by matching the digital code exchanged wirelessly between key and vehicle: (i) the vehicle's transponder electronic control unit (ECU) — usually located in the steering column — communicates wirelessly with the key for cyber authentication; (ii) the transponder ECU notifies the vehicle's power control module of cyber-authentication success via the in-vehicle network, such as CAN [9]; (iii) the power control module enables the cranking of engine.

Limitations. Despite their pervasive deployment, automakers, like GM/Ford/Volvo/BMW/Tesla, are still seeking improvement (or even replacement) of these key-based authentication solutions [7, 17, 24, 32, 39], due to the following three limitations.

- *Vulnerable to cyber attacks.* The cyber components of existing authentication systems, i.e., RF-based and in-vehicle communications, are vulnerable to cyber attacks [34, 43, 72]. RF communications suffer from a variety of jamming/relay attacks [22, 59, 74] while all communications via the in-vehicle network suffer from potential eavesdropping/fabrication via OBD-II port [49, 52, 63]. See [22, 27, 41] for real-life examples of attacks that hack the RF communications or in-vehicle network. Our user study with 165 car owners uncovers that people do not trust key-based solutions highly (see Appendix A for details).
- *Key is a single-point-of-failure.* Existing authentication systems rely solely on a key (or a key-fob) to validate the driver's identity, making the key a *single-point-of-failure*, i.e., anyone with the key or key-fob gains full control of the vehicle. Such a single-point-of-failure not only amplifies the vulnerability of vehicles, as corroborated by the increasing auto thefts using (cloned) keys [13, 25, 33], but also impedes the emerging service of peer-to-peer car rental (e.g., Turo [35] and Getaround [20]), which allows drivers to lease their vehicles to strangers but needs a reliable way to transfer the key.
- *Inability of driver identification.* Key-based authentication solutions are unable to identify different (but intended) drivers, which is required for driver-dependent services, such as personalized speed control to reduce crashes involved with teen drivers [12, 30, 31] and/or the coverage of insurance. By "identification", we mean the determination of a driver among a (usually *small*) set of candidate/intended drivers [51].

Authenticating Drivers Using Batteries. To mitigate these limitations, we design a novel driver authentication system using automotive batteries, called *Batteries-as-Authenticators* (BAuth). BAuth is an add-on module installed between vehicles and their batteries,[1] providing a second-factor authentication atop prevalent key-based solutions to allow/disallow a driver to drive a vehicle. Fig. 1 shows our BAuth prototype and its installation on 2008 Honda Fit.

BAuth exploits three physical facts: (i) the vehicle's engine would not start if the battery does not supply enough power, (ii) the power required to start the engine is much higher than that to power the vehicle's other

[1] BAuth could also be built in vehicles by automakers.

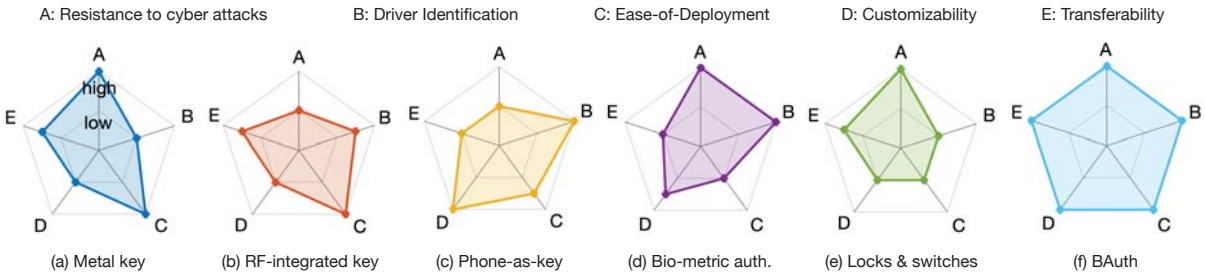


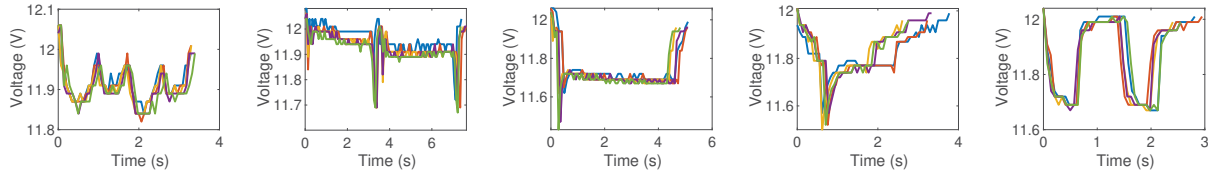
Fig. 2. BAuth vs. existing driving prevention solutions.

electric (e-) systems like lights/wipers or after-market accessories powered by the 12V auxiliary power outlet, and (iii) operating a vehicle's e-systems draws/causes unique power/voltage from its battery. BAuth uses these facts to authenticate a driver by having him/her perform a customized sequence of vehicle's e-system operations as an "authentication code" — thus allowing for user-preferred trade-offs between security and usability — and then authenticates the driver by matching the online-triggered battery voltages with his/her authentication code set up *a priori*. In the case of absence/failure of this authentication, BAuth disables the cranking of engine by reducing the vehicle battery's power supply. BAuth further uses (i) an alarming module to detect (and respond to) attacks, such as cranking the engine without authentication and uninstalling/tempering BAuth, and (ii) a reset module to allow the driver to change the authenticating operations if/when s/he forgets the authenticating operations or the e-systems (e.g., headlight bulbs) used by BAuth failed. The reset module also prevents the use of BAuth for denial-of-service (DoS) attacks.

BAuth mitigates holistically the limitations of existing key-based authentication solutions by:

- *Authenticating Drivers Physically.* BAuth first authenticates drivers using battery voltages as the "physical carrier" of their customized authenticating code/operations, which are delivered to the battery via the vehicle's power-line network (and thus reducing the attack surface for adversaries to eavesdrop/modify the authentication code), and then "physically" dis/enables the cranking of engine based on the authentication result. This way, BAuth does physical authentication that neither requires wireless communications nor depends on the in-vehicle network, which is especially critical due to the ever-increasing difficulty of securing the cyber space — all of the top 10 security risks of vehicles listed in [34] are cyber-related.
- *Using Batteries as Authenticators.* BAuth mitigates the single-point-of-failures of keys by using the battery as a second-factor authenticator, which augments keys' ownership-based authentication by authenticating drivers based on knowledge. Specifically, besides using battery as a vehicle's power supply/storage, BAuth exploits the battery as a *sensor* to validate a driver's identity and also as an *actuator* to dis/enable the cranking of engining. Using batteries as authenticators also facilitates pervasive deployment of BAuth on commodity vehicles, including aged/hybrid/electric vehicles.
- *Enabling Identification of Drivers.* BAuth allows different (but intended) drivers to customize their own authenticating operations and thus enables their identification, facilitating the ever-increasing vehicle sharing, e.g., one of every 10 (3) cars sold in 2030 (2050) is predicted to be a shared vehicle [14]. The fact that BAuth can be used with vehicles' after-market electronic accessories further enlarges its code space and thus increases the number of drivers BAuth can differentiate/identify.

Also, BAuth has several salient features in that it (i) controls the complexity of authenticating operations — from simple operations taking <1s to complex ones on multiple e-systems — based on the user's preference of security and usability; (ii) requires no additional devices to be carried by drivers; (iii) allows authentication code



(a) Swipe front windshield twice (b) Move driver-side window down & up (c) Turn headlight on and then off (d) Turn fan to max and then off (e) Unlock and then lock doors

Fig. 3. Operating a vehicle's e-systems triggers unique battery voltages, which BAuth uses to authenticate/identify drivers.

sharing among trusted parties (e.g., family members). Fig. 2 compares the key properties of BAuth with other driving prevention solutions. Note that BAuth is compatible to all these solutions (see Appendix B for details).

Evaluation Results. We have prototyped and installed BAuth on 6 vehicles of/from different types/makers: 2018 Subaru Crosstrek SUV, 2008 Honda Fit hatchback, 2019 Nissan Frontier pickup, 2019 Dodge Grand Caravan minivan, 2015 Chevrolet Volt hybrid sedan, and 2016 Nissan Leaf EV, and evaluated it with 20 authenticating operations of different complexities/strength. BAuth is shown to:

- authenticate/identify drivers with 98.17/2.84% true/false positive rates;
- dis/enable cranking of engine based on the authentication results without failure;
- tolerate voltage dynamics due to battery aging, temperature, and state-of-charge (SoC);
- be operable with after-market accessories powered by the vehicle's 12V auxiliary power outlet;
- be resistant against observation attacks with proper configuration.

We have also conducted a survey to collect users' opinions on BAuth, with 174 car owners recruited via Mechanical Turk. The survey results indicate BAuth's attractiveness to car owners, and thus its potential for wide deployment.

2 THREAT MODEL

We consider adversaries who want to breach vehicles' driver authentication systems, e.g., to steal/control vehicles via unauthorized driving, and classify them according to their knowledge of BAuth.

BAuth-Oblivious Attackers. BAuth-oblivious attackers attempt to (illegally) control vehicles via common schemes, such as use of stolen/cloned keys, radio jamming, and OBD hacking [2, 6, 37], without tailoring their attacks to BAuth.

BAuth-Aware Attackers. *Security-by-obscurity* is known to be not secure [64], as attackers are likely to accumulate knowledge of BAuth over time. BAuth-aware attackers, in addition to all the ability of BAuth-oblivious attackers, have sufficient knowledge of BAuth to deliver customized attacks, including but not limited to: (i) uninstalling BAuth from vehicle/battery, (ii) following the driver to steal his/her authenticating operations via observation attacks [53, 61, 75], (iii) mounting DoS attacks by physically destroying BAuth or breaking/altering the vehicle's e-systems used for authentication, and (iv) disabling BAuth by connecting a second battery in parallel with the original battery.

3 BASIC IDEA OF BAUTH

Automotive battery — commonly a rechargeable lead-acid battery with 12/24V nominal voltage depending on vehicle type — plays two distinct roles while discharging: ***Role-I*** to power the vehicle's e-systems, such as headlight, windshield wiper, etc., but not cranking the engine, and ***Role-II*** to power the starter motor to crank

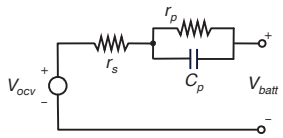


Fig. 4. The operation–voltage dependency can be explained by battery's Thevenin model.

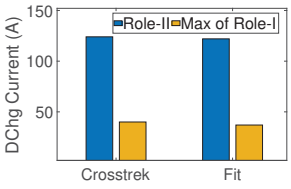


Fig. 5. Role-II is more power-demanding than Role-I.

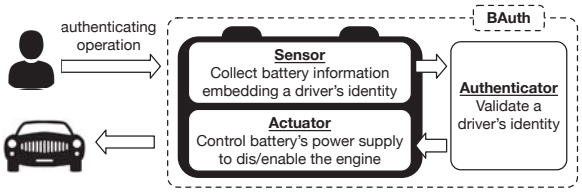


Fig. 6. Overview of BAuth.

the engine.[2] These two roles allow batteries to be used as a *sensor* to collect information on the driver's identity and also as an *actuator* to allow/disallow him/her to start the vehicle's engine.

3.1 As Sensors to Collect Information on Driver's Identity

Role-I allows use of batteries to collect a driver's authenticating code (in form of e-system operations) embedded in battery voltage. To corroborate this role, we plot the voltages of 2008 Honda Fit's battery in Fig. 3 while performing 5 different e-system operations. The voltages vary with different operations, but are similar for the same operation, revealing the possibility to fingerprint e-system operations (and hence driver's identity) using the thus-triggered voltages.

The dependency between e-system operations and the battery voltage can be captured analytically with battery's Thevenin circuit model in Fig. 4, which consists of an open-circuit voltage source V_{ocv}, a series resistance r_s, and a resistor-capacitor parallel network (i.e., r_p and C_p) [76]. The battery voltage V_{batt} for an operation requiring discharge current I_{DChg} will drop to

$$V_{batt} = V_{ocv} - r_s \cdot I_{DChg} - V_{C_p}, \tag{1}$$

where V_{C_p} is the voltage across the capacitor C_p, i.e.,

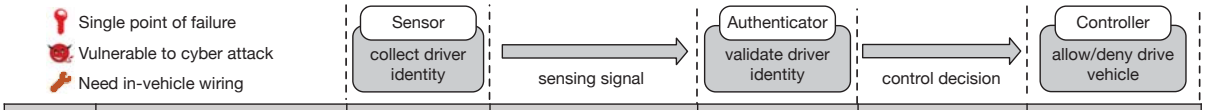
$$\dot{V}_{C_p} = -V_{C_p}/(r_p \cdot C_p) + I_{DChg}/C_p, \tag{2}$$

and then rise to $V_{batt} = V'_{ocv}$ when the discharging ends, where $V'_{ocv} \leq V_{ocv}$ due to the discharge. So, the operation's discharge current (and duration) determines the resultant battery voltages, capturing the operation–voltage dependency in Fig. 3. The above analysis also indicates the possibility of fingerprinting e-system operations using battery's discharge current, as discussed in Appendix C.

3.2 As Actuators to Allow/Disallow Starting of Engine

Role-II enables use of batteries to allow/disallow a person to drive a vehicle by controlling the battery's maximum power supply — the engine is not crankable without sufficient power, as we often experience in a cold weather [62]. Simply disconnecting the batteries from vehicles (e.g., with kill switches), however, is not the right solution because batteries are needed to run vehicles' monitoring functions even with the ignition off [48]. So, we selectively reduce the battery's output power, based on the fact that Role-II requires a much higher battery power output than Role-I. Fig. 5 plots our measurements on the battery's discharge current when playing the two roles on 2018 Subaru Crosstrek and 2008 Honda Fit, showing that Role-II requires a 3–4x higher discharge current than Role-I. More exemplary numbers on vehicles' power requirements can be found on pp. 461 of [45], showing, again, that Role-II demands much higher power than Role-I. These much different power demands encompass a power level that supports Role-I but not Role-II — we can (i) reduce the battery's output power to the level that

[2]These low-voltage batteries are also installed in hybrid/electric vehicles. For electric vehicles, Roll-II is to start all needed modules to make the vehicle drivable, which is an analogy of "cranking the engine" as with vehicles using internal combustion engines.



	Authentication Solutions Operated in Driver Compartment	Sensor (or Carrier of Identify)	Betw. Sensor & Authenticator	Authenticator	Betw. Authenticator & Controller	Controller (or Control Knob)
Existing Solutions	Metal Key	Keyhole	N.A.	N.A.	N.A.	Circuit Switch
	RF-Key (or Key-Fob)	Digital Code	RF Communication	Transponder ECU	In-Vehicle Network	Power Control ECU
	Phone + NFC	Digital Code	Near-Field Communication	Transponder ECU	In-Vehicle Network	Power Control ECU
	Biometric Solution	Bio-Fingerprint	Wireless Comm. or Additional Wiring	Transponder ECU Special OBD Dongle	In-Vehicle Network	Power Control ECU
	Kill-Switch	Digital Code Hidden Button	Wireless Comm. or Additional Wiring	Switch	N.A.	Switch
Proposed Solution	BAuth	Battery (Voltage as Sensing Signal)	Existing Powerline	BAuth's Controller	N.A.	Battery (Power Supply as Control Knob)
Other Design Choices (Potential Solutions Using Battery as Controller)	Biometric + Battery-as-Actuator	Bio-Fingerprint	Wireless Comm. or Additional Wiring	An authenticator similar as with BAuth	N.A.	Battery (power supply as control knob)
	Hidden Button + Battery-as-Actuator	Hidden Button				
	Keypad + Battery-as-Actuator	Keypad				
	Digital Code on Phone + Battery-as-Actuator	Digital Code	Wireless Communication			

Fig. 7. BAuth vs. other (potential) solutions to authenticate drivers, showing BAuth's advantage of: (i) solving the single-point-of-failure due to keys/key-fobs by using batteries as a second authentication factor, (ii) requiring no wireless communications and being decoupled from the in-vehicle network, thus becoming resilient to cyber attacks, and (iii) requiring no in-vehicle wiring and thus making it easy to deploy.

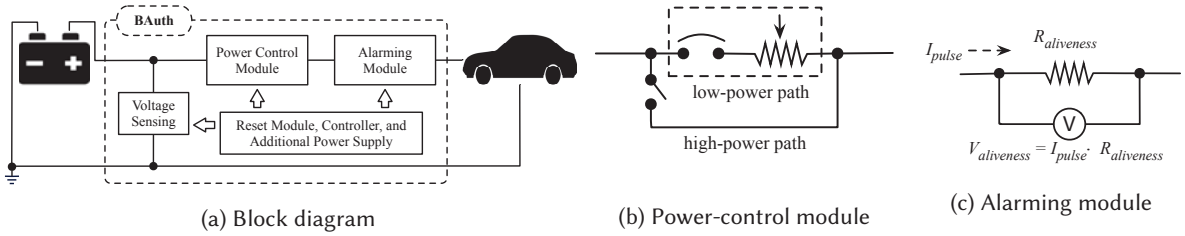
supports Role-I but not Role-II, (ii) use the thus-enabled Role-I operations to authenticate the driver, and (iii) raise the battery's output power (and hence enabling Role-II) upon successful authentication.

3.3 BAuth Overview

Based on the above observations, we design BAuth to authenticate drivers by using batteries as sensors and actuators (see Fig. 6), with first initialization and then online authentication.

- **Initialization.** During initialization, the driver operates the vehicle's e-systems with a customized sequence, reflecting his/her own preferred trade-off between usability and security, i.e., a more complex sequence of authenticating operations provide stronger protection but require more time/effort. BAuth records the resultant battery voltages to fingerprint the authenticating operations. Also, BAuth uses a reset module to allow drivers to change authenticating operations if/when necessary.
- **Online Authentication.** BAuth's online authentication assumes that intended drivers should know, and thus can perform, the customized sequence of authentication operations. Specifically, BAuth uses the above-constructed voltage fingerprint to authenticate drivers by checking if the correct sequence of operations have been performed, and, if successful, enables the cranking of engine. BAuth further uses an alarming module to detect and respond to attacks.

Note that the voltage fingerprints of e-system operations are *"delivered"* from the e-systems to the battery via the vehicle's power-line network — BAuth authenticates drivers without requiring any wireless communications nor using the in-vehicle network, as well as unneeding in-vehicle wiring. This differentiates BAuth from other



(a) Block diagram (b) Power-control module (c) Alarming module

Fig. 8. BAuth's physical architecture: (a) block diagram, (b) the power-control module regulates the battery's power mode by (dis)connecting the high-power path, (c) the alarming module detects, and responds to, attacks.

driver authentication schemes, like using a keypad or biometric fingerprints, with which the authenticating code is sent wirelessly (thus becoming susceptible to cyber attacks), or via wires (thus requiring additional wiring inside the vehicle or relying on vulnerable in-vehicle networks) to the authenticator, as compared in Fig. 7. Note the installation of BAuth on vehicle batteries is generally treated as *"external modification"* (i.e., the battery is exposed to the environment once opening the vehicle hood), which is much easier than in-vehicle modifications, such as adding/updating ECUs or adding wires traversing through the engine and driver compartments to connect front-end sensor (e.g., keypad and biometric collector) and the battery.

4 DESIGN OF BAUTH

We design BAuth with a *cyber-physical co-design* approach: the physical design enables the battery to be in either *low-power* or *high-power* mode, and the cyber design controls the transition of battery's power mode. Specifically, BAuth (i) uses a *low-power* battery to prevent the cranking of engine, and switches the battery to *high-power* mode (and thus enabling the cranking of engine) when the correct sequence of authenticating operations have been observed; (ii) keeps the battery in *high-power* mode until the engine stops and the vehicle is parked, at which time switches the battery to *low-power* mode again. BAuth works in either an *active* or *sleep* state — depending on whether or not the authentication will soon be needed — to improve its energy-efficiency.

4.1 Physical Design of BAuth

Fig. 8(a) shows the architecture of BAuth's physical components installed between a vehicle and its battery, consisting of: (i) a power-control module regulating the battery's power supply/mode, (ii) an alarming module detecting unauthorized cranking of engine and uninstalling/compromising BAuth, (iii) a voltage sensing module to monitor, in real time, the battery voltage, (iv) a reset module to change the authenticating operations, and (v) a controller and power supply to operate the above modules.

Power-Control Module. The power-control module regulates the power supply/mode of vehicle battery using the path-switching circuit in Fig. 8(b), consisting of 2 components.

- *Low-Power Path:* the low-power path between the battery and vehicle, implemented with a circuit breaker, flows only enough current to support the battery's Role-I (i.e., powering the vehicle's e-systems), but not Role-II (i.e., cranking the engine). This low-power path, or its circuit breaker more specifically, disconnects automatically when the current flowing through is larger than the maximum level I_{max}, where $I_{Role-I} < I_{max} < I_{Role-II}$.[3] We use I_{max} of 50A in our prototyping, based on the empirical results shown in Fig. 5. Note that unlike a fuse, a circuit breaker can be easily reset without replacing any physical component.

[3]We can make $I_{\max}$ programmable using a MOSFET-based current limiter.

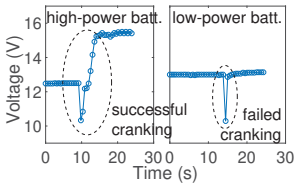


Fig. 9. Dis/enabling cranking of engine by controlling the battery's power mode.

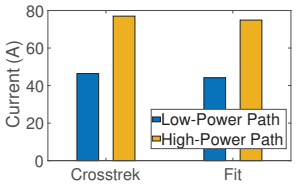


Fig. 10. Current on low/high-power paths when cranking with a high-power battery.

- *High-Power Path:* the high-power path allows sufficient power to support the battery's Role-II, which is implemented with a 6-AWG wire. A 12V power contactor, supporting a continuous/intermittent currents of 100A/1000A, respectively, is added to this high-power path to control whether or not to connect battery and vehicle via the high-power path. The contactor is operated by BAuth's controller with controlling signals amplified by first a TIP120-5A transistor and then a 12V/30A relay.

BAuth disconnects the high-power path — i.e., connecting the vehicle and battery solely via the low-power path — by default: excessive current breaks the circuit, thus regulating the battery to a low-power mode. On the other hand, closing the high-power path, and thus connecting the two paths in parallel, allows sufficient power to support the vehicle, thus restoring the battery's high power output. Let I_{total} denote the discharge current of the high-power battery, which will be distributed automatically over the two paths according to their resistance:

$$I_l = r_h/(r_l + r_h) \cdot I_{total} \quad \text{and} \quad I_h = r_l/(r_l + r_h) \cdot I_{total},$$

where $\{I_l,\ I_h\}$ are the current distributed over the low- and high-power paths, and $\{r_l,\ r_h\}$ are their resistance. To ensure $I_l < I_{max}$ (and hence the low-power path's connectivity when supporting high-power batteries), BAuth further uses a serial potentiometer to adjust r_l.

To corroborate the effectiveness/reliability of the power control module, we crank the engine of 2008 Fit with a low/high-power battery, and plot the resultant battery voltage in Fig. 9: the cranking is successful when the battery is in high-power mode, but failed with a low-power battery. We conducted similar experiments with 2018 Crosstrek and made similar observations. Note that the battery will be charged by the alternator upon successful cranking, thus raising its voltage to ≈15V. Fig. 10 plots the current flowed through the low/high-power paths when cranking the engine successfully, showing the low-power path flowed only a small (and adjustable with the potentiometer) portion of the total current, thus validating BAuth's keeping of the low-power path connected when supplying high power.

Alarming Module. The alarming module detects, and responds to using a siren, two illegitimate operations: (i) cranking the engine without passing the authentication (and hence using the low-power path/battery), and (ii) uninstalling BAuth physically from the vehicle, both of which are signs of attacks. These illegitimate operations will cause *open-circuit* between the vehicle and battery, either due to disconnected low-power path or disconnected terminals of BAuth interfacing the vehicle/battery. BAuth's alarming module detects such an open-circuit by monitoring the voltage drop over the aliveness detection resistor R_{alive} in Fig. 8(c), using the fact that the vehicle battery discharges a non-zero current (e.g., for monitoring functions) even when parked with ignition off [48]. BAuth will observe $V_{alive} = I \cdot R_{alive}$ if the circuit between the vehicle and battery is closed, and $V_{alive} = 0$ otherwise, in which case BAuth triggers an alarm.

With this alarming module, the only *theoretically feasible* way to undo BAuth's authentication without triggering an alarm is to **(i)** install a second battery to the vehicle while keeping the connections among the original battery, BAuth, and the vehicle intact, thus setting up a parallel connection to the two batteries, **and (ii)** tune the voltage

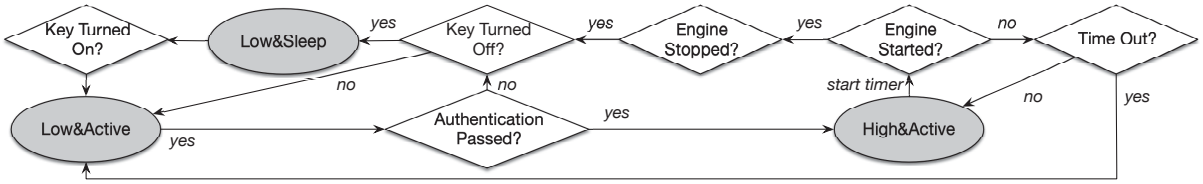


Fig. 11. Control flow of BAuth's cyber design: low/high-power states for physical components and sleep/active states for cyber components.

of the second battery based on the original battery's voltage (which fluctuates in practice), the resistance of the two paths, the current of cranking the engine, and the maximum current I_{max} of BAuth's low-power path, to ensure the original battery supplies only current less than I_{max} when cranking the engine, as we elaborate in Appendix D. These requirements make this hypothetical way of voiding BAuth practically infeasible.

System Reset. To ensure BAuth's real-life usability, we also implemented a system reset function, i.e., by entering the password configured *a priori*, to re-initialize BAuth if/when necessary (e.g., in case the driver forgets his/her customized operations or the used e-systems failed). We could also use this keypad to allow users to dis/enable BAuth based on their real-time needs, which is orthogonal to the design of BAuth and is thus omitted here.

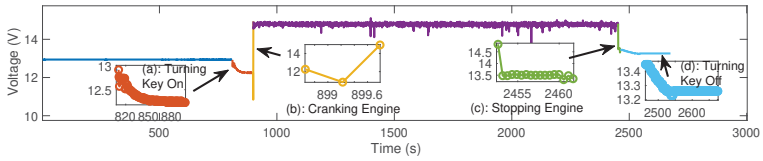
Other Modules. BAuth also includes: (i) a voltage sensor to monitor, in real time, battery voltage, (ii) a controller, and (iii) a 9V battery as the power supply. Note that using vehicle battery to power BAuth is not reliable, due to BAuth's constant operation (and thus power consumption) even when the car is parked, degrading the battery's ability to crank the engine. BAuth stores the voltage fingerprint of authenticating operations in the non-volatile EEPROM of its controller.

4.2 Cyber Design of BAuth

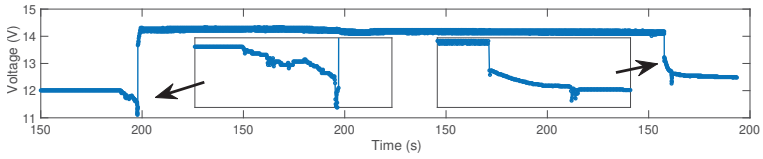
With its physical support, BAuth's cyber design reads battery voltage in real time to authenticate drivers, and controls the battery's power mode based on the authentication results.

Control Flow. Fig. 11 summarizes BAuth's control flow, including 5 key components: detecting the turning-on/off of ignition key, authenticating a driver by matching the real-time voltage readings with the customized voltage fingerprint, and detecting the cranking/stopping of engine. The turning-on/off of ignition key is defined as turning the key to the ignition position ON/LOCK. For vehicles with key-fobs, these two events correspond to the cases of turning the vehicle on/off.

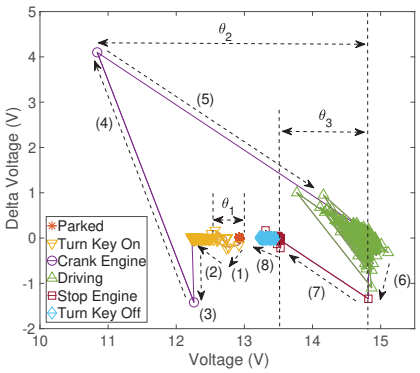
- *Detection of Turning-On/Off of Ignition Key.* BAuth operates in *sleep*/*active* states based on if a driver authentication is needed, which is, in turn, determined based on the events of turning on/off the ignition key: (i) turning on the ignition key indicates the authentication will soon be needed, at which time BAuth becomes *active* to monitor the battery voltage closely; (ii) turning off the ignition key indicates the vehicle has been parked and the authentication may not be needed soon, and thus BAuth returns to *sleep* to reduce its energy consumption.
- *Matching Battery Voltages.* Upon becoming *active*, BAuth attempts to authenticate the driver by matching real-time voltage readings with the *a priori* customized voltage fingerprint. BAuth establishes an authenticated session with the driver if the voltage matching is successful, and switches the battery to high-power mode.
- *Detection of Engine's Cranking/Stopping.* BAuth determines when to switch the battery back to low-power based on the events of cranking/stopping the engine. Specifically, the above-established authenticated session ends, and BAuth switches the battery back to the low-power mode, if (i) the engine is not cranked soon enough (e.g.,



(a) Battery voltage during a full usage cycle of Subaru Crosstrek



(b) Battery voltage during a full usage cycle of Nissan Leaf EV



(c) Transformed plot of Fig. 12(a)

Fig. 12. BAuth's operation is steered by 4 vehicle events: turning on/off the ignition key and cranking/stopping the engine.

within 5 minutes) after the session began, or (ii) the engine is cranked and then stopped after completing the driving.

Detection of Vehicle Events. As stated above, BAuth is steered by 4 vehicle events: turning-on/off the ignition key and cranking/stopping the engine. With battery voltage as the only input, BAuth detects these events based on their voltage patterns and sequential dependencies.

• *Voltage Patterns of Individual Events.* Fig. 12(a) plots a 45-min voltage trace of 2018 Subaru Crosstrek, covering a full usage cycle of the vehicle, i.e., parked, started, driven, and then parked again. A similar trace collected with Nissan Leaf EV is plotted in Fig. 12(b). Fig. 12(c) shows a transformed plot of the voltage trace in Fig. 12(a), where the x-axis is the voltage reading $v(t)$ and the y-axis is the change between two consecutive voltage readings, i.e., the markers in Fig. 12(c) are defined as

$$\mathbb{X} = \{x(t)\} = \{v(t)\},\ \mathbb{Y} = \{y(t)\} = \{v(t + t_\Delta) - v(t)\}, \tag{3}$$

and t_Δ is the sampling interval. We can clearly observe the vehicle's usage cycle — including all the four to-be-detected vehicle events — from the transitions of markers in Fig. 12(c). Also, Fig. 12 shows the battery voltage with the engine running (and hence charging the battery) is much higher than that with the engine off, demonstrating the ability to determine whether the engine is running or not based on battery voltage, especially in view of the standard voltage range with the engine running (i.e., [13.7, 15]V). Let a voltage reading $v \in \mathbb{O}$ denote if v is collected with the engine running, and $v \notin \mathbb{O}$ otherwise.

BAuth then detects the four events based on $\{\mathbb{X}, \mathbb{Y}\}$ using a moving time window of size w. For each window $[t - w,\ t]$ at time t, BAuth (i) identifies the max/min voltage readings in the window (i.e., v_{max} at t_{max} and v_{min} at t_{min}), (ii) calculates:

$$m_1 = \text{mean}\{v(t - w) : v(\min\{t_{min},\ t_{max}\})\}, \tag{4}$$

$$m_2 = \text{mean}\{v(\max\{t_{min},\ t_{max}\}) : v(t)\}, \tag{5}$$

$$m_3 = \text{mean}\{y(t - w) : y(t)\}, \tag{6}$$

and (iii) detects the events using $\{v_{max}, v_{min}, m_1, m_2, m_3\}$.

BAuth detects the tuning-on of the key based on two observations: (i) the voltage is stable before turning on the ignition (i.e., when the vehicle is parked); (ii) the voltage drops instantly and then deceases gradually when/after

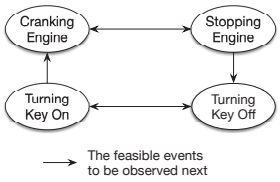


Fig. 13. BAuth detects four vehicle events using their sequential dependencies.

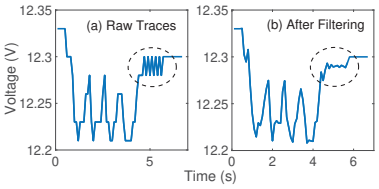


Fig. 14. Reducing the high-frequency fluctuations of voltages using a low-pass filter.

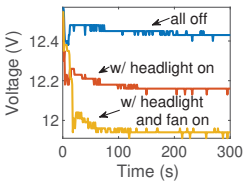


Fig. 15. Battery voltage varies with vehicle's background operations.

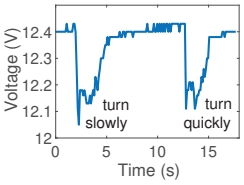


Fig. 16. Battery voltage varies with repetitions of operations.

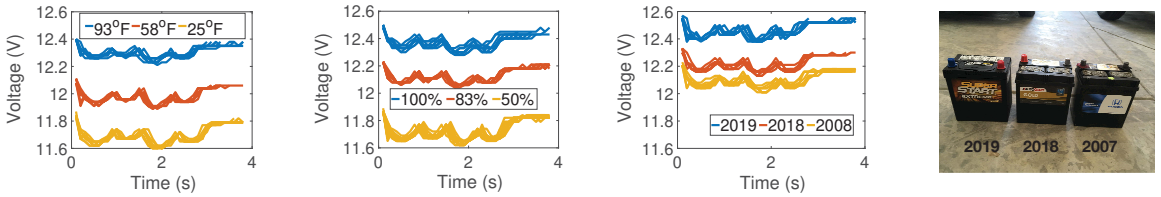
the ignition is turned on because of activated vehicle modules, such as ABS and airbag. These voltage patterns can be observed from Fig. 12(a) and transitions (1) and (2) of Fig. 12(c). So, BAuth concludes detection of turning on the ignition when $v_{max} \notin \mathbb{O}$ and $v_{max} - v_{min} > \theta_1$ are observed, where θ_1 captures the instant voltage drop when the ignition is turned on (see Fig. 12(c)). BAuth configures θ_1 during its initialization based on voltages observed from the vehicle/battery-of-interest, and updates θ_1 later during its online authentication. The battery voltage converges after turning off the ignition (see Fig. 12(d) and transition (8) of Fig. 12(c)), helping BAuth conclude the ignition is turned off if m_3=0 is observed.

Battery voltage drops significantly when the engine is cranked, and then returns to a higher level than before, as shown in Fig. 12(b) and transitions (3)–(5) of Fig. 12(c). The large voltage drop results from the starter motor's draw of a large discharging power/current from the battery to start the engine (i.e., battery's Roll-II). The alternator then generates power from the engine's rotation, which, in turn, charges the battery and thus restores the battery voltage. Also, battery voltage drops significantly when the engine is stopped, and then stays at low levels, due to the termination of charging current (see Fig. 12(c) and transitions (6) and (7) of Fig. 12(c)). Based on these observations, BAuth concludes (i) the engine is cranked if $v_{max} - v_{min} > \theta_2$, and (ii) a running engine is stopped if $v_{max} \in \mathbb{O}$ and $\theta_3 < m_1 - m_2 < \theta_2$. Fig. 12(c) illustrates θ_2 and θ_3, both of which can be configured at the initialization of BAuth and then updated online.

• *Sequential Dependencies among Events.* Besides the voltage patterns of individual events, BAuth further exploits the four events' sequential dependencies — e.g., it is impossible to turn off the ignition after cranking the engine without stopping the engine first — to improve their detections, as summarized in Fig. 13. Given the previous event, these dependencies define the feasible events to be observed next, thus improving/reducing the accuracy/complexity of BAuth's event detection.

Driver Authentication via Voltage Matching. Upon detection of turning-on of ignition key, BAuth becomes *active* to monitor the battery voltage at a high rate, and checks if the voltage fingerprint of customized authenticating operations has been observed. Let $\mathbb{F} = \{f_1, f_2, \ldots, f_n\}$ be the customized voltage fingerprint, and $\mathbb{U} = \{u(t)\}$ be the voltage readings after turning on the ignition key.

• *Sub-Trace Extraction.* BAuth attempts the voltage matching only when the driver has finished his/her authenticating operations, which can be observed from the recovered battery voltage (see Fig. 3), steering the extraction of $\mathbb{U}$'s sub-traces to match $\mathbb{F}$. The battery voltage, however, fluctuates significantly before its convergence, as shown in Fig. 14 with the voltage after *"swiping the front windshield wiper twice"* as an example. Such fluctuations are caused by both the background operations of vehicle's e-systems and the granularity of voltage sensing (e.g., 20mV in Fig. 14), magnifying the noises of voltage readings. We remove these fluctuations using a low-pass filter as they occur at a much higher frequency than the fluctuations during the e-system operations. BAuth smoothes the filtered voltage readings further with a moving average filter, and then records the time instants at which identical voltage readings have been observed consecutively (e.g., 3 times), denoted as $\mathbb{T} = \{t_1, t_2, \ldots\}$. At each



(a) Voltage varies w/ temp. (b) Voltage varies w/ SoC (c) Voltage varies w/ aging (d) Batteries of different ages

Fig. 17. Battery voltage varies with temperature, SoC, and aging.

$t_i \in \mathbb{T}$, BAuth extracts recursively a voltage sub-trace from the current time window:

$$\mathbb{U}_{j:i} = \{u(t_j), \ldots, u(t_i)\}\ (j = i-1, i-2, \ldots, k), \tag{7}$$

where $t_k + w \geq t_i$. BAuth then attempts to match $\mathbb{U}_{j:i}$ with $\mathbb{F}$ until (i) a match is found, or (ii) j reduces to k, thus concluding a matching failure at time t_i.

• *Voltage Matching.* Next, we describe how BAuth determines if a given $\mathbb{U}_{j:i}$ matches $\mathbb{F}$, which is non-trivial because even the voltages of the same operation vary with the contexts defined by the vehicle/people/battery.

(1) The background operations of the vehicle's e-systems lower the absolute levels of battery voltage. Fig. 15 plots the voltages after turning on the ignition key of 2018 Crosstrek, with different (but typical) background e-system operations, showing a clear dependency between voltage and the intensity of background operations.
(2) Both the relative levels of voltages and their durations may also vary for the same operation, due to the difficulty in repeating certain e-system operations exactly every time. Fig. 16 plots the voltages when the vehicle's fan is turned on to the maximum level and then off, with different speeds of rotating the speed dial: a slower operation increases both the magnitude and duration of the triggered voltages.
(3) The voltage dynamic is magnified further due to its dependency on temperature/SoC/aging, as shown in Fig. 17 with our empirical measurements when performing the operation of *"swipe front windshield wiper twice"* on 2008 Honda Fit 10 times.

BAuth makes its voltage matching transparent to contexts with three steps: fingerprint alignment, dual-dimensional similarity checking, and fingerprint updating.

BAuth first aligns $\mathbb{U}_{j:i}$ and $\mathbb{F}$ in both the voltage levels and durations. Specifically, BAuth aligns $\mathbb{U}_{j:i}$ and $\mathbb{F}$ with their first voltage readings (i.e., $u(t_j)$ and f_1) according to

$$\begin{cases} \mathbb{U}'_{j:i} = \{u(t_k) - u(t_j)\} & (k = j, j+1, \ldots, i), \\ \mathbb{F}' = \{f_k - f_1\} & (k = 1, 2, \ldots, n), \end{cases} \tag{8}$$

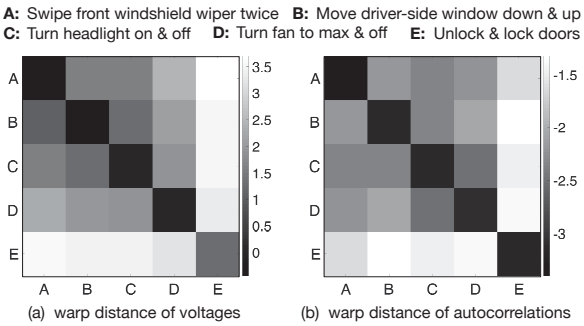
and then further aligns the durations of $\mathbb{U}'_{j:i}$ and $\mathbb{F}'$ by:

$$\begin{cases} \mathbb{U}''_{j:i} = \text{Interp}(\mathbb{U}'_{j:i},\ n), & (|\mathbb{U}'_{j:i}| \leq n), \\ \mathbb{U}''_{j:i} = \mathbb{U}'_{j:i}(1:n), & \text{otherwise}, \end{cases} \tag{9}$$

where Interp() is the interpolation function.

BAuth then examines the similarities between $\mathbb{U}''_{j:i}$ and $\mathbb{F}'$ in the time and frequency domains, respectively. BAuth uses *dynamic time warping* (DTW) [15] to quantify the similarity of $\mathbb{U}''_{j:i}$ and $\mathbb{F}'$ in the time domain as:

$$||\mathbb{U}_{j:i},\ \mathbb{F}||^{\text{time}}_{\text{BAuth}} = \frac{||\mathbb{U}''_{j:i},\ \mathbb{F}'||_{\text{dtw}}}{n}. \tag{10}$$



(a) warp distance of voltages (b) warp distance of autocorrelations

Fig. 18. The log-scale warp distances of operations in Fig. 3.

The effectiveness of BAuth's use of DTW is corroborated in Fig. 18(a), showing the minimum warp distance between the voltages of the 5 operations in Fig. 3 is always achieved between the same operation. BAuth further defines the similarity between $\mathbb{U}''_{j:i}$ and $\mathbb{F}'$ in the frequency domain using their autocorrelations, driven by the fact that users likely customize their authenticating code with repeated operations, e.g., *"swipe front wiper twice"*. The similarity between the autocorrelations of $\mathbb{U}''_{j:i}$ and $\mathbb{F}'$ is also quantified using DTW:

$$||\mathbb{U}_{j:i},\ \mathbb{F}||^{\text{freq}}_{\texttt{BAuth}} = \frac{||\text{AutoC}(\mathbb{U}''_{j:i}),\ \text{AutoC}(\mathbb{F}')||_{\text{dtw}}}{|\text{AutoC}(\mathbb{F}')|}, \tag{11}$$

where $|\text{AutoC}(\mathbb{F}')|$ is the number of lags when examining the autocorrelation. Fig. 18(b) compares the log-scale warp distance of the autocorrelations of 5 operations in Fig. 3, showing again the shortest warp distance is always achieved between the same operation. BAuth concludes a match between $\mathbb{U}_{j:i}$ and $\mathbb{F}$, and hence the driver successfully authenticated, if

$$||\mathbb{U}_{j:i},\ \mathbb{F}||^{\text{time}}_{\texttt{BAuth}} < \eta_1 \text{ and } ||\mathbb{U}_{j:i},\ \mathbb{F}||^{\text{freq}}_{\texttt{BAuth}} < \eta_2. \tag{12}$$

We use $\eta_1 = 1.5 \cdot \upsilon_\Delta$ and $\eta_2 = 0.35$ in our implementation of BAuth, where $\upsilon_\Delta = 20$mV is the voltage sensing granularity.

To mitigate the voltage dynamics further, BAuth updates $\mathbb{F}$ after each successful online voltage matching. One may record every successfully matched voltage trace and then match future voltages using conventional 1-Nearest-Neighbor DTW (1NN-DTW [66]) classification. However, this approach results in an ever-expanding dataset of $\mathbb{F}$s and thus may not work/scale for the resource-constrained platform. Instead, BAuth captures the voltage dynamics by finding the most representative voltage trace (denoted as $\mathbb{F}^*$) based on the previous observation ($\mathbb{U}_{j:i}$) and the current fingerprint ($\mathbb{F}$), defined as

$$\arg\min(\sqrt{(||\mathbb{F}^*,\ \mathbb{U}_{j:i})||^{\text{time}}_{\texttt{BAuth}})^2 + (||\mathbb{F}^*,\ \mathbb{F}||^{\text{time}}_{\texttt{BAuth}})^2} + \sqrt{(||\mathbb{F}^*,\ \mathbb{U}_{j:i})||^{\text{freq}}_{\texttt{BAuth}})^2 + (||\mathbb{F}^*,\ \mathbb{F}||^{\text{freq}}_{\texttt{BAuth}})^2}).$$

BAuth identifies the asymptotic equivalence of $\mathbb{F}^*$ using DTW Barycenter Averaging (DBA) [68], with the key idea to refine the average data trace via iterative expectation-maximization [67]:

$$\mathbb{F}^* \simeq \text{DBA}(\mathbb{F},\ \mathbb{U}_{j:i}). \tag{13}$$

DBA is known to achieve the smallest residual and requires only $\mathbb{O}(2n)$ to update a fingerprint. Therefore, BAuth's fingerprint matching/updating requires a time complexity of $\mathbb{O}(w(n^2 + 2n)) \approx O(wn^2)$, whereas the conventional 1NN-DTW requires $O(kwn^2)$ for k accumulated matchings.

Table 1. BAuth vs. attack schemes of BAuth-oblivious attackers (✓ means "being resistant to", and × otherwise).

Attack Schemes	Metal Key	RF Key (or Key-fob)	BAuth
Key Stolen in Burglary	×	×	✓
Key Left in the Vehicle	×	×	✓
Key Jamming/Relay	n/a	×	✓
OBD Port Hacking	×	×	✓
Twocking [36]	×	×	✓
Hot-Wiring [23]	×	✓	✓
Carjacking [11]	×	×	×/✓
Towing/Pushing	×	×	×

5 BAUTH'S DEFENSE AGAINST ATTACKS

Understanding the design of BAuth, we now discuss BAuth's resistance to attacks.

Against BAuth-Oblivious Attackers. By using batteries as a second factor to authenticate drivers physically, BAuth is resistant to most BAuth-oblivious attacks, as listed in Table 1. Note that it is advised to give up the vehicle and avoid resisting in case of carjacking, although BAuth remains intact so long as the authenticating operations are kept secret to robbers.

Against BAuth-Aware Attackers. Summarized below are potential schemes for BAuth-aware attackers to hack/disable BAuth and how BAuth defenses against them.

(1) **Attack:** Removing BAuth from the vehicle.
Defense: BAuth's alarming module detects such an attack and responds by turning on a siren. BAuth could also be integrated with vehicles if automakers provide it as a before-market product, in which case BAuth will not be exposed to attackers, voiding this removal attack.

(2) **Attack:** Stealing authenticating operations via observation.
Defense: BAuth could (i) be initialized with less-observable operations, such as turning on/off the fan and adjusting the seat, especially with BAuth's operability with after-market accessories (as we will experimentally validate in Sec. 6) which enlarges the space to define these less-observable operations, and (ii) use faked operations, such as *"touching the speed dial of the fan without turning it"*, to inject noise to the attacker's observations, which we will also experimentally corroborate in Sec. 6.

(3) **Attack:** Failing BAuth by hacking its individual modules, such as erasing the control algorithms from its controller or disconnecting its power supply.
Defense: BAuth can be implemented by (i) installing a protective case if provided as an add-on module to commodity vehicles, (ii) integrating with vehicles if provided by automakers as a before-market product (and thus not being exposed to the attacker), and (iii) using *hardware-based root-of-trust*, such as ARM TrustZone [5] and battery-backup IC [8], to thwart attacks dedicated to BAuth's individual components.

(4) **Attack:** Mounting DoS attacks by physically destroying BAuth or breaking the e-systems used for authentication.
Defense: Keeping the vehicle battery in low-power mode by default, BAuth's authentication preserves even if attackers have physically broken BAuth, in which case the battery would be either disconnected, or connected in low-power mode, from/to the vehicle, both of which prevent cranking of the engine. Also, BAuth's reset module facilitates change of authenticating operations (and hence the used e-systems) if any previously used e-systems fails.

(5) **Attack:** Evading BAuth by connecting a second battery in parallel with the original battery.
Defense: This attack is theoretically feasible to evade BAuth's authentication without triggering an alarm, but is practically difficult, as elaborated in Sec. 4.1.

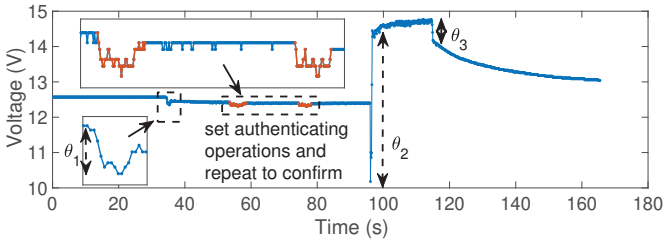


Fig. 19. Initializing BAuth on 2008 Fit with the authenticating operation of *swiping-the-front-windshield-wiper-twice*.

Case	Operation Description	Dur.
O-1	Swipe front windshield wiper once	2s
O-2	Swipe front windshield wiper twice	3s
O-3	Swipe front windshield wiper 5 times	7s
O-4	Blink left-turn signal 5 times	3s
O-5	Blink left-turn signal 10 times	6s
O-6	Blink emergency signal 5 times	3s
O-7	Blink emergency signal 10 times	6s
O-8	Move driver-side window down & up	7s
O-9	Move driver-side window down & up twice	13s
O-10	Turn on headlight for ≈1s and then off	2s
O-11	Turn on headlight for ≈3s and then off	5s
O-12	Unlock & lock doors	3s
O-13	Unlock & lock doors twice	5s
O-14	Turn fan to max for ≈1s and then off	2s
O-15	Turn fan to max for ≈3s and then off	4s
O-16	Move driver-side window down&up, then unlock&lock doors	9s
O-17	Turn fan to max & off, then blink emergency signal 5 times	5s
O-18	Turn headlight on & off, blink left-turn signal 5 times, then move driver-side window down & up	12s
O-19	Unlock & lock doors twice, then swipe front wiper 3 times	6s
O-20	Blink emergency signal 10 times, blink left-turn signal 10 times, then unlock & lock doors twice	16s

Fig. 20. Evaluating BAuth with 20 operations of different complexities/strengths. (See Appendix E for details)

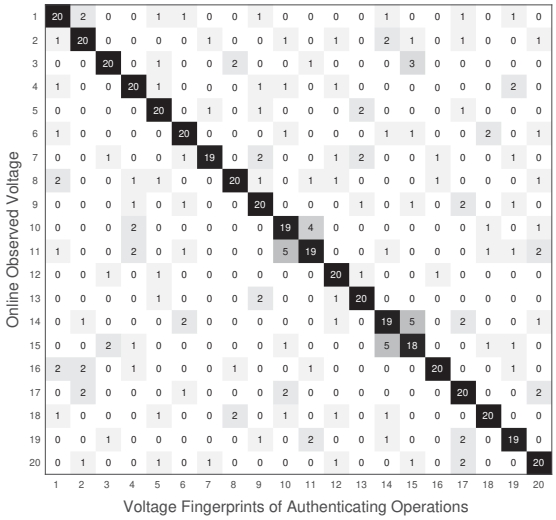


Fig. 21. BAuth achieves 98.25/2.07% true/false positive rates in recognizing the authenticating operations.

6 EVALUATION OF BAUTH

We have prototyped and evaluated BAuth with 6 vehicles of/from different types/makers — 2018 Subaru Crosstrek SUV, 2008 Honda Fit hatchback, 2019 Nissan Frontier pickup, 2019 Dodge Grand Caravan minivan, 2015 Chevrolet Volt hybrid sedan, and 2016 Nissan Leaf EV — using both real-life field-tests and trace-driven validation. We have also conducted a user study with 174 participants to survey car owners' opinion on BAuth.

6.1 Field-Tests on Vehicles

We first evaluate BAuth's end-to-end driver authentication in the field, with 20 authenticating operations of different complexities/strengths, as shown in Fig. 20 together with the average duration (rounded up to the next second) to perform these operations. These field-tests are performed with the ambient temperature varying from 10–60°F. The initial battery voltage of these field-tests — i.e., the battery voltage before inserting the key — varies in the range of 11.9–13.16V, spanning an SoC range of [≈15, 100]% according to [38].

Initialization. After its installation on a vehicle, BAuth keeps the battery in high-power mode by enabling the high-power path, and prompts the driver to perform a full cycle of the vehicle's usual operations, i.e., turning on the ignition key, performing his/her customized authenticating operations twice, cranking and then stopping the

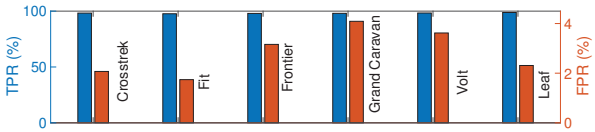


Fig. 22. BAuth achieves averaged true/false positive rates of 98.17/2.84% for the 6 vehicles.

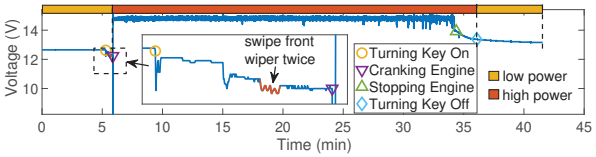


Fig. 23. Voltage trace taken while using BAuth for an ≈30-min drive in real life.

engine, and finally turning off the ignition key. BAuth records/uses the battery voltage during these initialization operations to: (i) construct the voltage fingerprint of users' authenticating operations, and (ii) configure the control parameters (i.e., $\{\theta_1, \theta_2, \theta_3\}$ Fig. 12(c)). Fig. 19 plots the recorded voltages during the initialization of BAuth with the authenticating operation of *swiping-the front-windshield-wiper-twice*.

Driver Authentication/Identification. After initializing BAuth with each authenticating operations in Fig. 20, we repeat 20 times (i) the customized authenticating operation to validate BAuth's ability of identifying the correct operations and thus authenticating the driver, and (ii) other operations in Fig. 20 to validate BAuth's ability of differentiating different operations and thus identifying different drivers. We used an active buzzer to "learn" the matching results, i.e., turn on the buzzer when BAuth concludes a match. Fig. 21 summarizes the in-field test results with Crosstrek, showing BAuth to identify the correct authenticating operations with an average true/false positive rates of 98.25/2.07%. Note in this experiment — repeating 20 times each of the operations in Fig. 20 for every authentication — the columns/rows of Fig. 21 do not necessarily add up to 20. Similar field-tests on Fit/Frontier/Grand-Caravan/Volt/Leaf show true/false positive rates of {97.65/1.74%, 97.96/3.16%, 98.04/4.09%, 98.33/3.62%, 98.8/2.31}, as plotted in Fig. 22, showing BAuth achieves averaged true/false positive rates of 98.17/2.84%. Note that the above results show no clear dependency between BAuth's authentication accuracy with the complexity of authentication operation, whose existence requires further exploration.

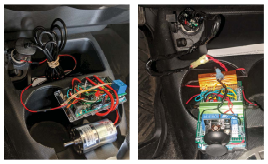
Fig. 21 also shows noticeable false detections between operations of {O-10, O-11} and {O-14, O-15}. Close examination of the related battery voltage reveals a relatively large variance in the durations when performing these operations, i.e., it is difficult for drivers to count for 1s (or 3s) consistently, thus magnifying the errors of voltage matching. This implies operations with clear timing, such as *"blinking the turn signal for 5 times"*, are preferred to improve BAuth's accuracy. The clear timing also makes these operations insensitive to specific operators. To corroborate this, we have 6 users perform O-4 on Fit, and use the thus-collected voltages to evaluate BAuth's voltage matching. All of the $6 \times (6-1) = 30$ pairs of cross-user voltages pass the matching, confirming BAuth's robustness to different operators when it is initialized with clearly-timed operations. BAuth's insensitivity to the operator also corroborates its sharing of authentication code between trusted parties (e.g., family members).

Event Detection. Next we evaluate BAuth's accuracy in detecting vehicle events by repeating 20 times the operations of turning on/off the ignition key and cranking/stopping the engine. Again, to observe the detection results, the buzzer is activated for $\{1, 2, 3, 4\}$s when the events of turning on key, turning off key, cranking engine, and stopping engine are detected, respectively. The results show BAuth to detect these events with 100/0% true/false positive rates.

Engine Dis/Enabling. To validate BAuth's reliability in dis/enabling the stater by controlling battery's power supply, we tried 20 times to crank the engine in low/high-power battery modes for both Crosstrek and Fit. All these attempts with low-power battery failed and those with high-power battery succeeded, thus exhibiting BAuth's 100% success in dis/enabling the cranking of engine.

Real-Life Usage. We have also evaluated BAuth while driving Fit in a real-life setting. Fig. 23 plots the battery voltage when the BAuth-enabled Fit is parked, started, driven, stopped, and parked again, with the authenticating

(a) Use BAuth with a uCosy massager



(b) Use BAuth with customized front-ends

Fig. 24. BAuth can be used with after-market vehicle accessaries powered by the 12V auxiliary power outlet.

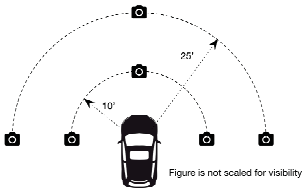


Fig. 25. Videoing the authenticating operations.

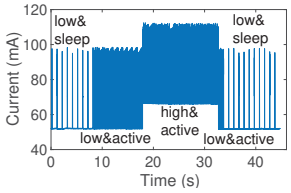


Fig. 26. Power consumption of our BAuth prototype.

operation O-2 in Fig. 20. The time at which BAuth detects turning on/off the ignition key and cranking/stopping the engine are labelled. BAuth accurately recognized the authenticating operation, enabled the high-power mode of battery, and then switched the battery back to low-power after turning off the ignition key.

BAuth with (Customized) After-Market Accessories. The authenticating operations in Fig. 20 are defined using vehicles' before-market e-systems. We could also use BAuth with after-market vehicle accessories, thus allowing more space to define authenticating operations. To corroborate this, we used an OSIM uCosy massager — connected/powered to/by a vehicle's 12V auxiliary power outlet, as shown in Fig. 24(a) — to define BAuth's authenticating operation as *"operating the massage for 3 revolutions"*. We then attempted this driver authentication for 60 times cumulatively on the 6 vehicles, all of which passed the matching. The feasibility of using BAuth with after-market accessories led us to design customized front-end modules to facilitate the defining of secure (e.g., not observable from outside of the vehicle) and convenient (e.g., quick/easy to perform) authenticating operations. As a proof of concept, we have designed/tested two front-ends of BAuth powered by the auxiliary power outlet — using a DC motor (load resistor) controlled by a push button (joystick) as the power drawer, respectively (see Fig. 24(b)).

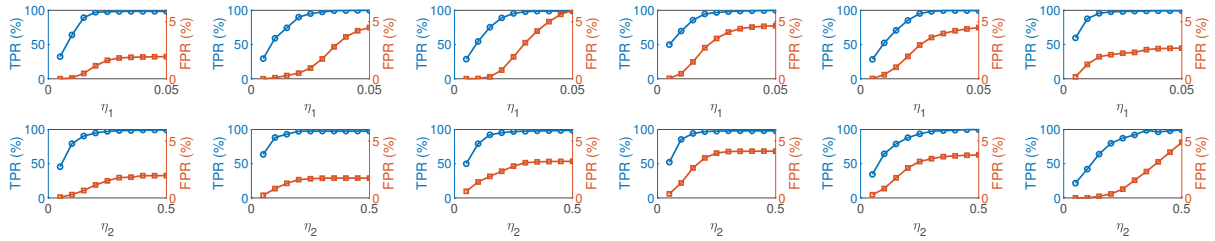
BAuth vs. Observation Attacks. BAuth could be configured with less-observable authenticating operations to defend against observation attacks (i.e., Attack-2 in Sec. 5), such as *"adjusting the seat level"*. The feasibility of configuring BAuth with after-market accessories, especially the customized front-end modules in Fig. 24(b), further facilitates the use of less-observable authenticating operations. We have also corroborated BAuth's resistance to observation attacks by using faked operations. We recorded the video of a driver *"locking/unlocking the door and then blinking emergency signal 5 times"* (as his custom sequence of operations) on a 2016 Ford Explorer, who ***"pretended"*** to turn the fan speed dial before performing his operation. We recorded 12 such videos at 10'/25' away from the vehicle, from the front/left-front/right-front angles, and with/without zoom-in, respectively (see Fig. 25). We asked 13 BAuth-aware participants to guess the authenticating operation by watching the videos. No participant correctly identifies the operation, confirming BAuth's resilience to observation attacks.

Power Consumption. Fig. 26 plots the power consumption of our BAuth prototype when operating in different states, measured using a Monsoon power monitor. Voltage sensing (and the corresponding computation) draws about 45mA current, and keeping the contactor connected requires another 20mA current. Note that the background current of ≈50mA is required by the controller, i.e., an Arduino Uno in this prototype, which can be reduced by using other low-power controllers, e.g., to 0.023mA with Arduino Pro Mini [4].

6.2 Trace-Driven Validation

We have further evaluated BAuth using empirical traces collected from the 6 vehicles.

Driver Authentication/Identification. We have collected 10–40 voltages to each of the operations in Fig. 20 with each of the 6 vehicles, and used these voltages to evaluate BAuth's voltage matching via cross-validation. We repeated this trace-driven validation by varying the threshold η_1 in Eq. (12) from $[0.005, 0.05]$V and η_2 from



(a) Crosstrek (b) Fit (c) Frontier (d) Grand Caravan (e) Volt (f) Leaf

Fig. 27. Trace-drive validation corroborates BAuth's high/low true/false positive rates in driver authentication.

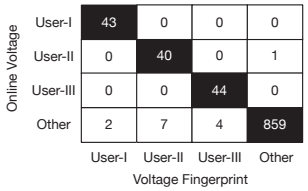
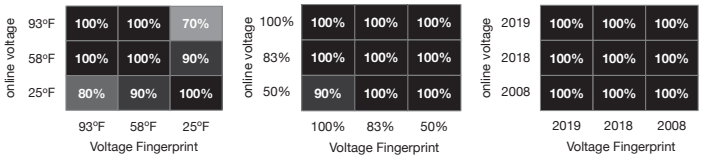


Fig. 28. BAuth identifies three users with 98.6% accuracy.



(a) Cross-temp. (b) Cross-SoC (c) Cross-age

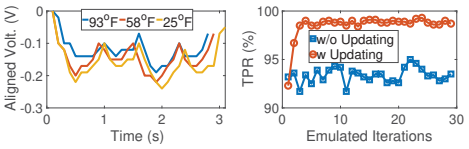
Fig. 29. BAuth achieves (w/o fingerprint updating) 91.95/98.85/100% true positive rates for cross-temp./SoC/age matchings.

$[0.05, 0.5]$, while keeping a default setting of $\eta_1 = 0.03$V and $\eta_2 = 0.35$. Fig. 27 plots the results, showing both the true/false positive rates increase with larger η_1 and η_2. The true positive rate increases to 99% when $\eta_1 \geq 0.035$V or $\eta_2 \geq 0.4$, while keeping the false positive rate <4%. Note that as a second-factor authentication, BAuth is desired to have a high true positive rate while being (relatively) tolerative to false positives.

To further validate BAuth's ability of driver authentication, we use the operations/voltages listed in Fig. 20 to emulate the case of three drivers sharing a vehicle and customizing their own (and different) authentication operations. We then evaluate BAuth's driver authentication by randomly selecting voltages from Fig. 20 as the online-observed voltage. As an example, Fig. 28 summarizes the authentication results when the three drivers customized their authentication operations as {O-2, O-6, O-10} over 1,000 randomly selected voltages, showing BAuth to identify drivers with an accuracy of $(43 + 40 + 44 + 859)/1,000 = 98.6\%$. We have repeated this emulation for 100 rounds with varying driver-customized authentication operations, showing that BAuth achieves a 98.06% accuracy on average.

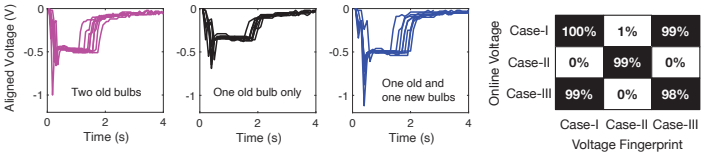
BAuth vs. Temperature/SoC/Aging Dynamics. We next examine BAuth's robustness to the voltage dynamics caused due to battery temperature/SoC/aging. We form and use $30 \times (30 - 1) = 870$ pairs of cross-temperature voltages, based on the battery voltages collected at different temperatures (see Fig. 17(a)), to examine if the voltage–temperature dependency degrades BAuth's robustness. 800 pairs of these voltages are shown by BAuth's voltage matching (i.e., Eq. (12)) to be of the same operation, achieving a 800/870 = 91.95% true positive rate (see Fig. 29(a)). Similar pairwise matchings are performed using the voltages collected with different SoC and aging (i.e., those plotted in Figs. 17(b) and (c)). BAuth achieves 98.85/100% true positive rates for the cross-SoC/age voltage matchings, as shown in Figs. 29(b) and (c).

Fig. 29(a) also shows a $1 - 91.95\% = 8.05\%$ false negative of cross-temperature matchings when matching the 25°F voltages with those collected at 93/58°F — the low temperature of 25°F increases battery resistance, which, in turn, magnifies the voltage changes when performing the authenticating operation (see Eq. (1)), thus leading to a larger deviation from the voltages collected with a 93/58°F battery (see Fig. 30(a)). BAuth's online fingerprint



(a) Low temp. magnifies voltage changes (b) Online updating improves matching

Fig. 30. BAuth's online fingerprint updating makes it robust to battery dynamics.



(a) Voltages collected in different scenarios (b) Matching results

Fig. 31. Exploring BAuth's stability on 2008 Fit with: (i) two old bulbs, (ii) one old and one new bulb, and (iii) one old bulb only.

Table 2. Erroneous operations of O-20 in Fig. 20.

Case	Operation Description
E-1	Blink emergency signal 5 times, blink left-turn signal 10 times, then unlock & lock doors twice
E-2	Blink emergency signal 10 times, blink left-turn signal 5 times, then unlock & lock doors twice
E-3	Blink emergency signal 10 times, blink left-turn signal 10 times, then unlock & lock doors once
E-4	Blink left-turn signal 10 times, then unlock & lock doors twice
E-5	Blink emergency signal 10 times, then unlock & lock doors twice
E-6	Blink emergency signal 10 times, blink left-turn signal 10 times)

updating mitigates this temperature-dependency of battery voltage. To corroborate this, we emulate the real-life usage of BAuth by (i) generating 1,000 random permutations of 20 voltages (i.e., 10 for 93°F and another 10 for 25°F), and (ii) emulating BAuth's real-life usage with each of these permutations by using the first voltage as the initialized fingerprint, and the next $20 - 1 = 19$ voltages as those subsequently observed online. Fig. 30(b) plots BAuth's true positive rate in matching the voltages during the emulated period, with and without updating the fingerprint, averaged over these 1,000 emulations. Without updating the fingerprint, the true positive rate of fingerprint matching fluctuates at ≈93%, which BAuth improves to (thanks to its fingerprint updates) ≈99% and with a much smaller fluctuation, justifying BAuth's robustness to battery dynamics.

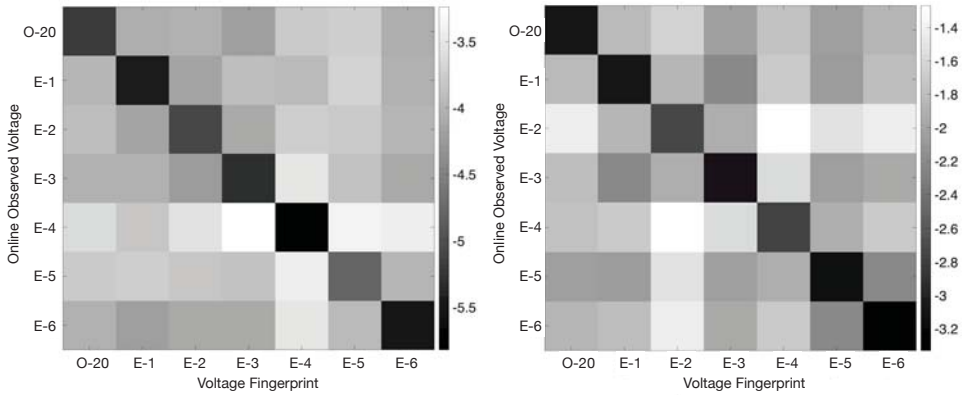
BAuth vs. Faulty/Replaced E-System Modules. To examine BAuth's stability against faulty/replaced e-system modules, we collected on 2008 Fit the voltages of *"turning on the headlight for 1s"* in three scenarios: (i) with two old (but working) bulbs, (ii) with only one old bulb to emulate the case when one bulb is broken, and (iii) with one old and one new bulb to emulate the case after replacing the broken bulb. The old bulbs are made by SYLVANIA, and the new bulb is from Panasonic. Fig. 31 plots/summarizes the thus-collected voltages and the results when matching the voltages collected in different scenarios, showing (i) (almost) identical voltages when both bulbs are working regardless whether they are old or new,[4] and (ii) very different voltages when only one bulb is working. Also, note that BAuth's reset function allows drivers to update the voltage fingerprints if/when needed.

BAuth vs. Erroneous Operations. Last but not least, we use O-20 in Fig. 20 to examine BAuth's performance when the user fails to correctly perform his/her customized operation. Specifically, we examine the distances of the voltages triggered by the erroneous operations listed in Table 2 (i.e., variations of O-20 in Fig. 20), as plotted in Fig. 32: incorrectly performing the authentication operation increases the distance to the correct operation (and hence reduces their similarity) in both the time and frequency domains, and thus should be avoided by the user.

6.3 User Survey on BAuth

We have conducted a second survey to collect users' opinion on BAuth. Specifically, after recruiting 174 car owners via Mechanical Turk and educating them how to use BAuth, we collected their opinions and found:

[4]This is because the bulbs for a given vehicle, regardless of their OEMs/modules, are of the same power rating, e.g., 60W12V for 2008 Fit.



(a) Time-domain distance (b) Frequency-domain distance

Fig. 32. Incorrectly performed operations (i.e., E-1 to E-6 in Table 2) increase the distance to the customized authentication operation (i.e., O-20 in Fig. 20).

- the participants score BAuth's *practical value* in driver authentication at 7.6/10 on average, and 62 (35.6%) of them scores ≥9;
- 78/42/22 (77.6%) of participants are willing to spend up to 5s/10s/≥10s to perform BAuth's authenticating operations. Cross-checking these results with Fig. 20 indicates all the authenticating operations we explored in Fig. 20 are acceptable;
- 105 (60.3%) participants are interested, with a ≥7 score, in installing BAuth on their vehicles, and 88 (50.6%) participates stated their interests in BAuth will increase with the value of their cars;
- 45 (83.3%) participants value BAuth to be >$50.

These results reveal BAuth's attractiveness to car owners and thus its potential for wide deployment.

7 SECURITY ANALYSIS

We analyze below the security of BAuth in code space, repeatability, uniqueness, and randomness, using 2018 Crosstrek as an example.

7.1 Code Space of Authentication Operations

BAuth uses the voltages triggered by e-system operations as the authentication code, leading to a code space of N^m, where N is the number of different types of e-system operations available on a vehicle and m is the number of elementary operations a driver used in his/her authenticating operations. Note that some e-system operations have different levels/options (e.g., "rolling" up/down different number of windows), each contributes to N separately. As an example, Table 3 lists the unit operations of available before-market e-systems on 2018 Crosstrek, together with their observability from the outside of the vehicle. It is important to note that we can (significantly) enlarge the code space by designing and providing customized front-ends of BAuth. Moreover, we have surveyed 16 car owners on their top-5 choices when using the unit operations in Table 3 to customize their authentication. Fig. 33 summarizes the number of votes each e-system received, showing *"(un)locking of doors"* and *"pressing braking pedal"* (and thus turning on the brake signal) are the most preferred operations/e-systems.

Table 3. Unit operations of the available before-market e-systems on 2018 Crosstrek.

Index	Unit Operations	External Observability
1	Swipe front windshield wiper once	high
2	Swipe rear windshield wiper once	high
3	Blink left/right turn signal once	high
4	Blink emergency signal once	high
5	Press brake for 1s	high
6	Turn on headlight for 1s	high
7–10	Roll 1–4 window(s) up	high
11–14	Roll 1–4 window(s) down	high
15–16	Turn 1–2 roof light(s) on for 1s	high
17	Turn fan to max	low
18	Lock doors	low
19	Unlock doors	low
20–21	Turn on 1–2 seat heater for 1s	low

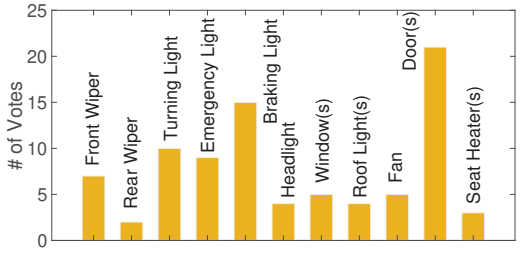
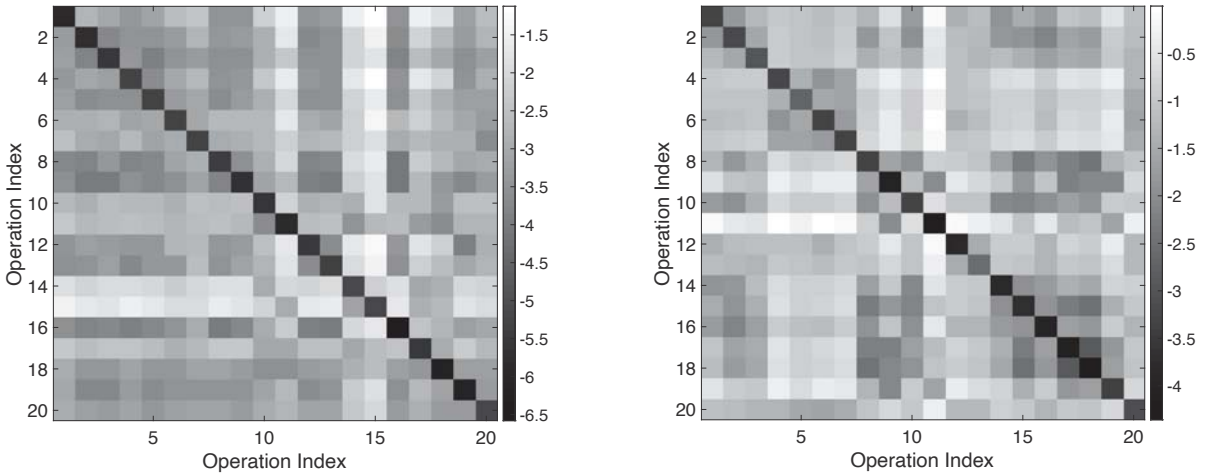


Fig. 33. Car owners' preference on using which e-systems to customize their authentication.



(a) Time-domain distance

(b) Frequency-domain distance

Fig. 34. The voltage fingerprints of authentication operations show high repeatability (i.e., small distance between the same operation) and uniqueness (i.e., large distance between different operations). The distances are in log scale.

7.2 Repeatability, Uniqueness, and Randomness of Voltage Signal

We next examine the repeatability and uniqueness of the voltage-fingerprint of the 20 authentication operations in Fig. 20. Fig. 34 plots the temporal and frequency space distance (in log scale). The small distances between voltages of the same operation corroborate the repeatability, and the large distances between voltages of different operations verify the uniqueness of the voltage fingerprint. We have also calculated the entropy of all the collected voltages in Fig. 20 to evaluate the randomness (and thus strength) of using the voltage as authentication signal. The voltage is shown to have an entropy of 3.10, which is close to its theoretical maximum (i.e., when the voltages are uniformly distributed) of 3.76. Note that the voltages in Fig. 20 are collected with a 20mV granularity.

8 DISCUSSION

Given below are a few possible directions to improve BAuth further.

- **Low-Power Path with Adjustable Rating.** BAuth controls the maximum current $I_{\max}$ of the low-power path using a circuit breaker, which is fixed for a specific implementation; we have to provide separate models of BAuth with different ratings of $I_{\max}$ for different vehicles. We can overcome this limitation by designing a low-power path with adjustable $I_{\max}$, e.g., using a current limiter built with MOSFETs and super-capacitors and operating with pulse width modulation. This power regulation will generate heat, so a heat sink will be needed to ensure reliability/safety.
- **Automatic Recharging Power Supply.** BAuth could be further equipped with an automatically recharging power supply which will be charged by the vehicle's alternator during driving, thus freeing car owners from maintenance burden.
- **Continuous Driver Authentication.** BAuth establishes authenticated sessions between the vehicle and the driver before the vehicle is driven. A continuous driver authentication that protects vehicles consistently during the authenticated session will improve vehicle security further. We conjecture that drivers can be fingerprinted using their driving behavior, which can be observed as vehicles' velocity/acceleration/gyroscope and estimated with battery voltage/current readings.

9 RELATED WORK

Multi-factor authentication has been widely explored to improve security [65], using such factors as token presence [44, 47, 60], voice biometrics [55, 56, 71], facial recognition [42, 58, 78], hand geometry [73, 77, 79], fingerprint scanner [50], and thermal image [54, 57]. A general authentication system structure is proposed in [1]. To the best of our knowledge, BAuth is the first to use batteries as an authentication factor.

10 CONCLUSION

We have presented a novel driver authentication system using automotive batteries, called BAuth, to provide vehicles a second-factor authentication atop existing key-based solutions. BAuth exploits batteries as *sensors* to authenticate a driver with his/her customized sequence of operations of the vehicle's e-systems, and then *actuators* to physically dis/enable cranking of engine by controlling battery output power. We have prototyped/installed/evaluated BAuth on 6 vehicles. Our user study with 174 car owners corroborates BAuth's attractiveness. BAuth's basic idea of battery-based physical authentication could also be applied to other electrical systems, where the core system function (e.g., cranking the engine) requires a higher power than other complementary functions (e.g., swiping the windshield wiper), thus allowing the use of low-power functions to authenticate users and then enable the high-power (and core) function. Examples of these systems include, but not limited to, drones, electric scooters, and a variety of handheld devices. This way, BAuth lays a basis for protecting systems physically, which is especially critical with the ever-increasing difficulty of securing the cyber space.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for constructive suggestions. The work reported in this paper was supported by NSF under Grant CNS-1446117 and CNS-1739577.

REFERENCES

[1] 2015. Measuring Strength of Authentication (Discussion Draft) Measuring Strength of Authentication.
[2] 2020. 4 Ways To Steal A Car. https://www.carthrottle.com/post/4-ways-to-steal-a-car-and-how-to-stop-it-happening-to-you/.
[3] 2020. Anti-Theft Vehicle Devices. https://www.whathappensnow.com/choosing-an-anti-theft-device-for-your-car/.
[4] 2020. Arduino Low Power. http://www.home-automation-community.com/arduino-low-power-how-to-run-atmega328p-for-a-year-on-coin-cell-battery/.

[5] 2020. ARM TrustZone. https://www.arm.com/products/silicon-ip-security.
[6] 2020. Australian Car Theft Statistics. https://www.budgetdirect.com.au/blog/the-state-of-car-theft-in-australia.html.
[7] 2020. Automakers Working Feverishly to Make Car Keys Disappear. https://www.thetruthaboutcars.com/2018/06/automakers-feverishly-working-getting-rid-keys/.
[8] 2020. Battery Backup IC. https://www.promelec.ru/pdf/pst523.pdf.
[9] 2020. CAN Bus. https://www.csselectronics.com/screen/page/simple-intro-to-can-bus/language/en.
[10] 2020. Car Security on the Cheap. https://www.instructables.com/id/Killswitch-Car-security-on-the-cheap/.
[11] 2020. Carjacking. https://en.wikipedia.org/wiki/Carjacking.
[12] 2020. Chevrolet's Teen Driver Technology. https://www.chevrolet.com/teen-driver-technology.
[13] 2020. Criminals cloning your key fob, easily taking your car. https://www.news5cleveland.com/news/criminals-cloning-your-key-fob-easily-taking-your-car.
[14] 2020. Disruptive trends that will transform the auto industry. https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/disruptive-trends-that-will-transform-the-auto-industry/de-de.
[15] 2020. Dynamic Time Warping. http://web.science.mq.edu.au/ cassidy/comp449/html/ch11s02.html.
[16] 2020. Electronics-Salon Panel Mount 150Amp AC/DC Current Sensor. https://www.amazon.com/Electronics-Salon-150Amp-Current-Sensor-Module/dp/B016M63GTM.
[17] 2020. Elon Musk says the Tesla 2020 Roadster "maybe won't need a key at all". https://www.businessinsider.com/elon-musk-tweets-tesla-roadster-might-not-need-key-2019-11.
[18] 2020. Ford PATS Antitheft System. http://ricksfreeautorepairadvice.com/ford-pats-antitheft-system/.
[19] 2020. Ford's Vehicle Security System. https://accessories.ford.com/alarm.html.
[20] 2020. Getaround. https://www.getaround.com/.
[21] 2020. GM Vehicle Theft Deterrent (VTD) Relearn Procedures. http://my.cardone.com/techdocs/pt%2077-0011.pdf.
[22] 2020. Hackers Can Steal a Tesla Model S in Seconds by Cloning Its Key Fob. https://www.wired.com/story/hackers-steal-tesla-model-s-seconds-key-fob/.
[23] 2020. How to Hotwire a Car. https://jalopnik.com/5871510/how-to-hotwire-a-car/.
[24] 2020. Lincoln Phone As A Key Technology Aims to Eliminate Traditional Key Fobs. https://www.caranddriver.com/news/a28751902/lincoln-phone-as-a-key/.
[25] 2020. More Cars Are Getting Stolen Because Owners Are Basically Asking for It. http://time.com/money/4553620/cars-stolen-keys-ignition-fob-unlocked/.
[26] 2020. NXP and Continental Demonstrate the World's First Concept Car Embedding NFC at Mobile World Congress. http://www.marketwired.com/press-release/nxp-continental-demonstrate-worlds-first-concept-car-embedding-nfc-mobile-world-congress-nasdaq-nxpi-1395814.htm/.
[27] 2020. Relay attack. https://www.youtube.com/watch?v=8pffcngJJq0.
[28] 2020. SecuriCode keyless entry keypad. https://owner.ford.com/how-tos/vehicle-features/locks-and-security/securicode-keyless-entry-keypad.html.
[29] 2020. Sentry Key Immobilizer System. https://cdn.xjjeeps.com/pdf/en-us/sentry-key-immobilizer-system.pdf.
[30] 2020. Teen Driver Car Accident Statistics. https://www.edgarsnyder.com/car-accident/who-was-injured/teen/teen-driving-statistics.html.
[31] 2020. Tesla's Valet Mode. https://electrek.co/2018/06/20/tesla-remotely-limit-speed-mobile-app-update/.
[32] 2020. The Future of Car Keys. https://www.nytimes.com/2015/06/26/automobiles/wheels/the-future-of-car-keys-smartphone-apps-maybe.html.
[33] 2020. Thefts of Autos With Keys are Rising. https://www.thebalancesmb.com/dont-leave-your-keys-in-your-car-or-truck-462432.
[34] 2020. Top 10 Security Challenges in the Automotive Industry for Connected Cars. https://www.trustonic.com/news/blog/top-10-security-challenges-for-connected-cars/.
[35] 2020. Turo. https://turo.com.
[36] 2020. Twocking. https://en.wikipedia.org/wiki/Taking_without_owner %27s_consent.
[37] 2020. Vehicle theft. https://en.wikipedia.org/wiki/Motor_vehicle_theft.
[38] 2020. Voltage & SoC. https://www.energymatters.com.au/components/ battery-voltage-discharge.
[39] 2020. Volvo's truly keyless entry: your smartphone. https://www.extremetech.com/extreme/224665-volvos-truly-keyless-entry-your-smartphone.
[40] 2020. WINGONEER Voltage Detector. https://www.amazon.com/WINGONEER-Voltage-Detector-Terminal-Arduino/dp/B06XHKZCD4.
[41] 2020. Your BMW can be stolen by any idiot with a $30 hacking kit. https://nakedsecurity.sophos.com/2012/09/18/bmw-stolen-hacking-kit/.
[42] T. Ahonen, A. Hadid, and M. Pietikainen. 2006. Face Description with Local Binary Patterns: Application to Face Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 12 (2006), 2037–2041.

[43] A. I. Alrabady and S. M. Mahmud. 2003. Some attacks against vehicles' passive entry security systems and their solutions. *IEEE Transactions on Vehicular Technology* 52, 2 (2003), 431–439.
[44] A. K. Awasthi and S. Lal. 2003. A remote user authentication scheme using smart cards with forward secrecy. *IEEE Transactions on Consumer Electronics* 49, 4 (2003), 1246–1248.
[45] Robert Bosch. 2014. *Bosch Automotive Electrics and Automotive Electronics*. Springer.
[46] C. Busold, A. Taha, C. Wachsmann, A. Dmitrienko, H. Seudié, M. Sobhani, and A. Sadeghi. 2013. Smart Keys for Cyber-cars: Secure Smartphone-based NFC-enabled Car Immobilizer. In *CODASPY'13*.
[47] Christoph Busold, Ahmed Taha, Christian Wachsmann, Alexandra Dmitrienko, Herve Seudie, Majid Sobhani, and Ahmad-Reza Sadeghi. 2013. Smart keys for cyber-cars: Secure smartphone-based NFC-enabled car immobilizer. In *CODASPY'13*.
[48] K. T. Cho. 2018. *From Attack to Defense: Toward Secure In-vehicle Networks*. Ph.D. Dissertation. University of Michigan.
[49] Kyong-Tak Cho and Kang G. Shin. 2016. Error Handling of In-vehicle Networks Makes Them Vulnerable. In *CCS'16*.
[50] A. De Luca and J. Lindqvist. 2015. Is secure and usable smartphone authentication asking too much? *Computer* 48, 5 (2015), 64–68.
[51] M. Enev, A. Takakuwa, K. Koscher, and T. Kohno. 2016. Automobile Driver Fingerprinting. *Proceedings on Privacy Enhancing Technologies* 2016, 1 (2016), 34 – 50. https://content.sciendo.com/view/journals/popets/2016/1/article-p34.xml
[52] A. Francillon, B. Danev, and S. Capkun. 2011. Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars. In *NDSS'11*.
[53] K. Fujita and Y. Hirakawa. 2008. A study of password authentication method against observing attacks. In *ISISI'08*.
[54] A. M. Guzman, M. Goryawala, J. Wang, A. Barreto, J. Andrian, N. Rishe, and M. Adjouadi. 2013. Thermal Imaging as a Biometrics Approach to Facial Signature Authentication. *IEEE Journal of Biomedical and Health Informatics* 17, 1 (2013), 214–222.
[55] Rosa GonzAlez HautamAki, Tomi Kinnunen, Ville HautamAki, and Anne-Maria Laukkanen. 2015. Automatic versus human speaker verification: The case of voice mimicry. *Speech Communication* 72 (2015), 13 – 31.
[56] Rosa González Hautamäki, Tomi Kinnunen, Ville Hautamäki, Timo Leino, and Anne-Maria Laukkanen. 2013. I-vectors meet imitators: on vulnerability of speaker verification systems against voice mimicry. In *INTERSPEECH*.
[57] Shuowen Hu, Jonghyun Choi, Alex L. Chan, and William Robson Schwartz. 2015. Thermal-to-visible face recognition using partial least squares. *J. Opt. Soc. Am. A* 32, 3 (2015), 431–442.
[58] I. A. Kakadiaris, G. Passalis, G. Toderici, M. N. Murtuza, Y. Lu, N. Karampatziakis, and T. Theoharis. 2007. Three-Dimensional Face Recognition in the Presence of Facial Expressions: An Annotated Deformable Model Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 4 (2007), 640–649.
[59] M. Kasper, T. Kasper, A. Moradi, and C. Paar. 2009. Breaking KeeLoq in a Flash: On Extracting Keys at Lightning Speed. In *AFRICACRYPT 2009*.
[60] Salman H. Khan, M. Ali Akbar, Farrukh Shahzad, Mudassar Farooq, and Zeashan Khan. 2015. Secure biometric template generation for multi-factor authentication. *Pattern Recognition* 48, 2 (2015), 458 – 472.
[61] Z. Ling, J. Luo, Y. Liu, M. Yang, K. Wu, and X. Fu. 2018. SecTap: Secure Back of Device Input System for Mobile Devices. In *INFOCOM'18*.
[62] H. Liu, Z. Wang, J. Cheng, and D. Maly. 2009. Improvement on the Cold Cranking Capacity of Commercial Vehicle by Using Supercapacitor and Lead-Acid Battery Hybrid. *IEEE Transactions on Vehicular Technology* 58, 3 (2009), 1097–1105.
[63] T. Matsumoto, M. Hata, M. Tanabe, K. Yoshioka, and K. Oishi. 2012. A Method of Preventing Unauthorized Data Transmission in Controller Area Network. In *VTC Spring'12*.
[64] Rebecca T. Mercuri and Peter G. Neumann. 2003. Security by Obscurity. *Commun. ACM* 46, 11 (2003), 160.
[65] Aleksandr Ometov, Sergey Bezzateev, Niko Makitalo, Sergey Andreev, Tommi Mikkonen, and Yevgeni Koucheryavy. 2018. Multi-Factor Authentication: A Survey. *Cryptography* 2, 1 (2018).
[66] Elzbieta Pekalska, Robert P.W. Duin, and Pavel Paclki. 2006. Prototype selection for dissimilarity-based classifiers. *Pattern Recognition* 39, 2 (2006), 189 – 208. http://www.sciencedirect.com/science/article/pii/S0031320305002633
[67] F. Petitjean, G. Forestier, G. I. Webb, A. E. Nicholson, Y. Chen, and E. Keogh. 2014. Dynamic Time Warping Averaging of Time Series Allows Faster and More Accurate Classification. In *ICDM'14*.
[68] F. Petitjean, A. Ketterlin, and P. Gancarski. 2011. A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition* 44 (2011), 678–693.
[69] Takeshi Sugashima, Dennis Kengo Oka, and Camille Vuillaume. 2016. Approaches for Secure and Efficient In-Vehicle Key Management. *SAE Int. J. Passeng. Cars – Electron. Electr. Syst.* 9 (2016), 100–106.
[70] R. Sykora. 2015. The Future of Autonomous Vehicle Technology as a Public Safety Tool. *Minn. JL Sci. & Tech.* 16 (2015), 811.
[71] Florentin Thullier, Bruno Bouchard, and Bob-Antoine Jerry Ménélas. 2017. A Text-Independent Speaker Authentication System for Mobile Devices. *Cryptography* 1 (2017), 16.
[72] J. Wetzels. 2014. Broken keys to the kingdom: Security and privacy aspects of RFID-based car keys. *CoRR* abs/1405.7424 (2014).
[73] Alexandra L. N. Wong and Pengcheng Shi. 2002. Peg-Free Hand Geometry Recognition Using Hierarchical Geomrtry and Shape Matching. In *MVA*.
[74] T. Yang, L. Kong, W. Xin, J. Hu, and Z. Chen. 2012. Resisting relay attacks on vehicular Passive Keyless Entry and start systems. In *ICFSKD'12*.

[75] Q. Yue, Z. Ling, X. Fu, B. Liu, K. Ren, and W. Zhao. 2014. Blind Recognition of Touched Keys on Mobile Devices. In *CCS'14*.
[76] Xiaoqiang Zhang, Weiping Zhang, , and Geyang Lei. 2016. A Review of Li-ion Battery Equivalent Circuit Models. *Transactions on Electrical and Electronic Materials* 17, 6 (2016), 311–316.
[77] Z. Zhang. 2015. Photoplethysmography-Based Heart Rate Monitoring in Physical Activities via Joint Sparse Spectrum Reconstruction. *IEEE Transactions on Biomedical Engineering* 62, 8 (2015), 1902–1910.
[78] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. 2003. Face Recognition: A Literature Survey. *ACM Comput. Surv.* 35, 4 (2003), 399–458.
[79] G. Zheng, C. Wang, and T. E. Boult. 2007. Application of Projective Invariants in Hand Geometry Biometrics. *IEEE Transactions on Information Forensics and Security* 2, 4 (2007), 758–768.

APPENDIX A: USER SURVEY ON KEY-BASED AUTHENTICATION SOLUTIONS

To learn public opinion on existing key-based authentication solutions, we conducted a user study with 165 car owners using Amazon Mechanical Turk. Over 70% of the respondents think the breaking of driver authentication is a serious problem with criticality scores of 7 or higher (on a scale of 1–10). However, only 3% of them use additional protection (e.g., steering wheel lock) besides the standard key-based solutions, even though (i) they only trust the key-based solutions with a 6.6/10 average confidence level, and (ii) 86% of them are aware of the possibility of stealing a car without the key. These results uncover the gap between people's desire to have an effective and easy-to-use driver authentication system and the lack of such a system in both the literature and the market.

APPENDIX B: EXISTING VEHICLE PROTECTION SOLUTIONS

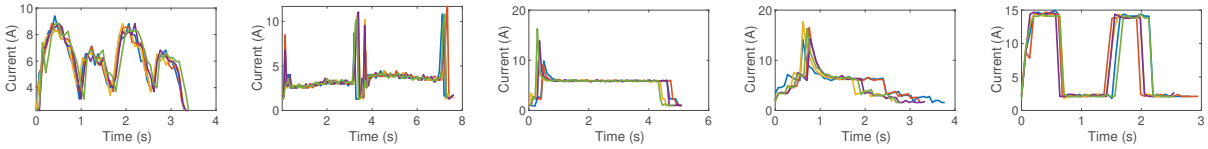
Summarized below are existing solutions to protect vehicles.

Entry Prevention. The first opportunity to protect vehicles is to prevent attackers from entering the vehicle, commonly achieved using door locks operated by keys/key-fobs, keypads, or even phones [28].

Driving Prevention. In case the entry prevention failed, the second protection opportunity is to prevent attackers from starting/driving the vehicle, using the following means.

- *Keys/Key-fobs.* Automakers have provided various key-based systems that disable the ignition/fuel/starter to prevent driving a vehicle [18, 21, 29]. However, these solutions suffer the limitations explained in Sec. 1.
- *Phone-based Immobilizer.* Smartphones could be turned to immobilizers using their NFC modules [26, 46], but suffer similar vulnerabilities to keys/key-fobs.
- *Physical Locks.* Various physical locks, such as tire/steering/pedal locks, are designed to reduce a vehicle's drivability [3]. These physical locks, however, suffer (relatively) low usability: drivers need to install & uninstall locks each time they leave the vehicle unattended, and are required to carry additional devices, such as keys of the locks.
- *Kill Switches.* Kill switches — cutting off the vehicle's flow of electric current to prevent the starting of its engine — are another type of driving-prevention solutions [70]. Kill switches are similar to `BAuth` in disabling the starter by controlling the electric power output, but are different from `BAuth` in two ways. First, `BAuth` *reduces* the power output from the battery, while kill switches *cut off* the vehicle's flow of electric current. Thus, kill switches also disable the monitoring function of parked vehicles or before-market authentication systems, making them unsuitable for daily use.[5] Second, kill switches rely on explicit control signals from either remote controls or switch buttons installed at hidden places inside a vehicle. Remote controls suffer from potential relay/jamming attacks and need to be carried by the driver all the time, while hidden switch buttons

[5]In fact, kill switches are often installed/used in vehicles to be stalled for a long period, to prevent the draining of their batteries.



(a) Swipe front windshield twice (b) Move driver-side window down & up (c) Turn headlight on and then off (d) Turn fan to max and then off (e) Unlock and then lock doors

Fig. 35. The discharge current of vehicle battery could also be used to fingerprint authenticating operations.

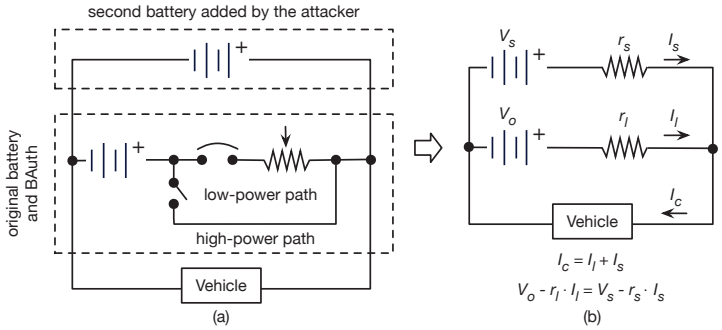


Fig. 36. A theoretically feasible (but practically challenging) way to bypass BAuth.

are *security-by-obscurity* in essence and thus not reliable [64], especially when the commonly-used hidden places in vehicles (e.g., under the driver seat) are not secret any more [10].

Attack Detection. Besides the above protection solutions, alarm systems are used by vehicles to detect, and respond to, attacks [19]. However, existing alarm systems rely on in-vehicle network to exchange monitored information and respond to detected attack attempts, and could thus be disabled by cyber attacks through the OBD-II port [41].

APPENDIX C: FINGERPRINTING AUTHENTICATING OPERATIONS USING CURRENT

Besides voltage, the discharge current of automotive battery when performing authenticating operations offers another opportunity to validate a driver's identify, as empirically observed in Fig. 35 with the traces collected on Fit. However, the sensing of battery's discharge current requires more engineering effort when compared to voltage sensing, especially for vehicles whose discharge currents vary widely from less than 1A to over 100A. In practice, this leads to the much higher cost of current sensors when compared to voltage sensors — e.g., the current sensor we used to collect the data in Fig. 35 costs $17.99 on Amazon [16], while the voltage sensor used in our BAuth prototype costs only $6.99 for a pack of 5 [40].

APPENDIX D: BYPASSING BAUTH USING A SECOND BATTERY

Fig. 36(a) illustrates a theoretically feasible approach to bypass BAuth's driver authentication by adding a second battery to the vehicle while keeping the original battery connected — using the second battery and BAuth's low-power path to provide the cranking current in parallel, as explained in Sec. 4.1. However, the attacker, after connecting the second battery according to Fig. 36(a), needs to further tune the second battery's voltage to make this approach work.

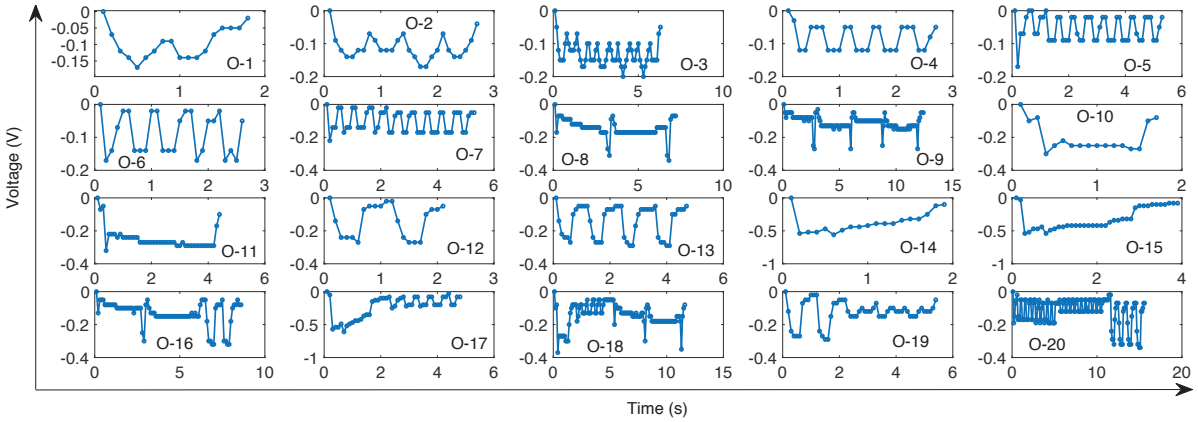


Fig. 37. Exemplary voltage fingerprints of the 20 authenticating operations listed in Fig. 20, aligned according to Eq. (8).

Focusing on only the discharge current, Fig. 36(a) can be simplified to the resistance circuit showing in Fig. 36(b), where $\{V_o,\ V_s\}$ are the voltage of the original/second battery, I_c is the cranking current, $\{I_l,\ I_s\}$ are the discharge current on BAuth's low-power path and that provided by the second battery when cranking, and $\{r_l,\ r_s\}$ are the equivalent resistances of BAuth's low-power path and that between the second battery and vehicle. When cranking the engine with this two-battery connection, we know:

$$I_l + I_s = I_c, \tag{14}$$

$$V_o - r_l \cdot l_l = V_s - r_s \cdot l_s. \tag{15}$$

To successfully bypass BAuth, the attacker needs to ensure $I_l < I_{max}$, where I_{max} is the maximum current the low-power path can flow. Based on Eqs. (14) and (15), we know the following condition has to be met to bypass BAuth:

$$I_l = \frac{V_o - V_s + r_s \cdot I_s}{r_l} = \frac{V_o - V_s + r_s \cdot (I_c - I_l)}{r_l} < I_{max}. \tag{16}$$

Clearly, the attacker must have accurate knowledge of $\{V_o,\ r_l,\ r_s,\ I_{max}\}$ to ensure Eq. (16) hold, which is nontrivial because: V_o varies over time, $\{r_l,\ r_s\}$ vary with the physical connections, and I_{max} varies from the specific implementation of BAuth. Note tuning V_s in a *trial-and-error* way does not work here because any cranking attempts without satisfying Eq. (16) trigger BAuth's alarm.

APPENDIX E: EXEMPLARY VOLTAGE FINGERPRINTS

Fig. 37 plots the exemplary voltage fingerprints of the authenticating operations listed in Fig. 20, collected with 2008 Honda Fit.