Battery Status-aware Authentication Scheme for V2G Networks in Smart Grid

Hong Liu, *Student Member, IEEE*, Huansheng Ning, *Senior Member, IEEE*, Yan Zhang, *Senior Member, IEEE*, and Mohsen Guizani, *Fellow, IEEE*

***Abstract*—Vehicle-to-grid (V2G) is emerging as an attractive paradigm in smart grid, and provides power and information services by periodically collecting power status of battery vehicles (BVs). During a BV's interaction with power grid, it may be in one of the following states: charging, fully-charged (FC), and discharging. In this paper, we identify that there are unique security challenges in a BV's different battery states. Accordingly, we propose a battery status-aware authentication scheme (BASA) to address the issue for V2G networks. In BASA, 1) an aggregated-identifier is proposed during the charging-to-FC state transition to ensure that BVs can be authenticated without disclosing their real identities; 2) selective disclosure based challenge-response authentication is presented during the FC-to-discharging phase to realize anonymous data transmission; 3) an aggregated-status is reported during the discharging-to-charging transition in order to hide a BV's power level from an aggregator. In addition, we perform comprehensive security analysis, which shows that BASA achieves both privacy preservation and security protection during battery state transitions. The analysis also indicates that battery status awareness is crucial for BVs' secure operations for V2G networks in smart grid.**

***Index Terms*—Authentication, battery status, privacy, security, smart grid, vehicle-to-grid (V2G).**

I. Introduction

THE SMART GRID is a critical power transmission infrastructure, and transforms the traditional power grid into the Internet of energy. Smart grid realizes bi-directional communications of electricity and information, which enables customers and utilities to jointly monitor and manage the power usage [1], [2]. Vehicle-to-grid (V2G) is an essential network component in smart grid, and has received lots of attention lately [3]. In V2G networks, interconnection is achieved by periodically collecting the power status data of each battery vehicle (BV) to provide information services for efficient power dispatching and management. In addition, the geographically dispersed BVs' charged power can be adopted as distributed energy resources to provide electricity services for power load balance. However, communications between BVs and smart grid may suffer from severe vulnerabilities, therefore security and privacy issues become noteworthy for V2G networks [4], [5].

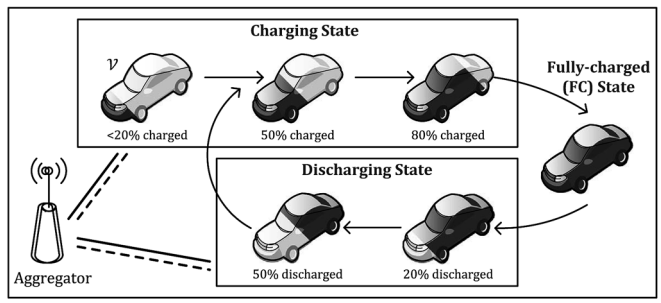


Fig. 1. The battery state transitions of a BV in V2G networks.

In this paper, we will identify a new security challenge owing to varying battery status in V2G networks, and propose a battery status-aware authentication scheme (BASA) to address this issue. Fig. 1 shows the battery state transitions of a BV (i.e., $\mathcal{V}$) in V2G networks. In the network, $\mathcal{V}$ interconnects with an aggregator, which acts as an intermediary communication entity between $\mathcal{V}$ and the power grid. Both power transmission and communication are established between the BV and the aggregator. This achieves bi-directional interaction of electricity and information. During the interaction between $\mathcal{V}$ and the power grid, it may be in one of the following states: charging, fully-charged (FC), and discharging. In the example, $\mathcal{V}$ starts in the charging state with the initial quantity of electricity (QoE) 20% charged battery. Gradually, its QoE increases from 20% to 80%, then to the FC state. During the FC state, $\mathcal{V}$ is not used. When the load of power grid is over balance, the fully-charged BV may perform discharging to feed the power back into the power grid. Then, $\mathcal{V}$ is in the discharging state and its QoE decreases. After its discharging operation, $\mathcal{V}$ may be connected to the power grid and perform charging again. This process will repeat during a BV's battery lifecycle.

Now, we have a close look at the new security challenge during the BV's battery state transitions. First, when $\mathcal{V}$ accesses the power grid and attempts to establish communication with the aggregator, $\mathcal{V}$ should be authenticated by the aggregator. In this case, the aggregator cannot correlate the BV's *location related privacy* with its real identity. Second, when $\mathcal{V}$ has been fully charged, it may be asked to perform discharging operation. $\mathcal{V}$ should have autonomy to decide whether or not to participate in the discharging operation. Hence, the aggregator cannot obtain the detailed response to deduce *interest related*

Manuscript received March 28, 2012; revised August 27, 2012; accepted October 02, 2012. Date of publication February 13, 2013; date of current version February 27, 2013. This work was jointly funded by National Natural Science Foundation of China (NSFC) and Civil Aviation Administration of China (CAAC) (61079019). This work was partially supported by the projects 217006/E20 funded by the Research Council of Norway, and the European Commission FP7 Project EVANS (2010-269323). Paper no. TSG-00135-2012.

H. Liu and H. Ning are with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China (e-mail: liuhongler@ee.buaa.edu.cn; ninghuansheng@buaa.edu.cn).

Y. Zhang is with Simula Research Laboratory, Norway; and Department of Informatics, University of Oslo, Oslo N-0373, Norway (e-mail: yanzhang@simula.no).

M. Guizani is with Qatar University, Qatar (e-mail: mguizani@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2012.2224387

privacy. Finally, when $\mathcal{V}$ completes the discharging operation and turns into the charging state, it can obtain its own detailed state-of-charge (SOC) for further bill purpose. Here, SOC refers to the charged percentage of the battery in a BV. However, the aggregator cannot obtain the BV's detailed power status with *SOC related privacy* concern. It is observed that there are different security and privacy requirements during battery state transitions in V2G networks. Accordingly, it is critical to design a systematic authentication scheme to achieve both security protection and privacy preservation for BVs in different battery states as well as during state transitions.

To address the identified security challenge, we will propose a battery status-aware authentication scheme for V2G in smart grid. The main contributions in this paper are as follows.

1) Identify a unique security challenge in V2G networks owing to varying battery status and introduce different privacy considerations (i.e., location, interest, and SOC) in different battery states;
2) Propose a battery status-aware authentication scheme to address the identified security challenge. Three inter-linked protocols are presented to guarantee the secure interaction between BVs and the power grid during the dynamic battery state transitions.
3) Perform comprehensive security analysis and show that the proposed scheme achieves both privacy preservation and security protection. The analysis also indicates that battery status awareness is crucial for BVs' secure operations in V2G networks.

In addition to the battery status awareness consideration, the proposed authentication scheme has the following propertie. *Privacy preservation*: an aggregator or illegal attackers cannot correlate BVs' identities with their sensitive information; *Hierarchical access control*: an aggregator and a central authority are assigned different authorities on BVs; *Data confidentiality and data integrity*: BVs and an aggregator ensure that the exchanged messages are never detected, tampered, or abused; *Dynamic participation*: BVs can freely join and leave the networks without influencing ongoing communications; *Mutual authentication*: BVs and an aggregator can establish mutual authentication so that any illegal entity cannot access system resources (e.g., power, and information).

The remainder of the paper is organized as follows. In Section II, we will discuss related works. Section III describes the system model. Section IV introduces the proposed authentication scheme, and the interrelation of the proposed protocols is discussed in Section V. Security analysis and performance analysis are presented in Sections VI and VII. Finally, Section VIII draws a conclusion.

II. Related Works

Studies have been performed to enhance generic security protection and privacy preservation in smart grid. Towards general security and privacy issues, universal cryptography algorithms have been recommended [6]. Meanwhile, several solutions have been proposed based on different mechanisms, including security framework [7]–[11], authentication protocols [12]–[16], cryptograph algorithm and secure management [17]–[21], and privacy preservation [22], [23]. Thereinto, Li *et al.* [12] proposed a one-time signature based multicast authentication scheme, which is able to reduce the storage cost and the signature size compared with existing schemes, and is appropriate for lightweight applications. Fouda *et al.* [14] proposed a lightweight message authentication scheme, in which mutual authentication and session keys are established by the hash-based authentication code and the Diffie-Hellman exchange protocol. Lu *et al.* [23] proposed a privacy-preserving aggregation scheme, which applies a super-increasing sequence to structure multi-dimensional data and encrypt the structured data by the homomorphic Paillier algorithm.

In the literature, there are only few studies on security and privacy issues in V2G networks. Yang *et al.* [24] identified privacy-preserving issues and proposed a precise reward architecture. Concretely, a reward scheme was proposed to realize the trade-off between the participants' freedom of using their BVs and full benefits provided by the power operators. A secure communication architecture was proposed to achieve privacy-preserving for BV monitoring and rewarding, in which an ID-based blind signature was introduced to realize anonymity. Guo *et al.* [25] proposed an authentication protocol to deal with multiple responses from a batch of vehicles. The proposed protocol introduces the concept of interval time for an aggregator verifying multiple vehicles, and applies the modified digital signature algorithm (DSA) algorithm to establish such batch verification scheme. Vaidya *et al.* [26] proposed a multi-domain network architecture, which incorporates a comprehensive hybrid PKI model which integrates hierarchical and peer-to-peer cross-certifications.

In the aforementioned studies, various security issues are observed and addressed. However, BVs' charging/discharging operations or battery states are not considered. As a consequence, distinctive security and privacy requirements in different battery states have not been studied yet in the literature.

III. System Model

Fig. 2 illustrates the battery status-aware V2G network architecture, which includes three main entities: BVs, a local aggregator (LAG), and a central authority (CA). A BV is owned by an individual consumer and has a specific group attribute. LAG is granted by a power operator to collect BVs' SOC for power scheduling. CA as a trusted party belongs to an independent institution. Towards the trust relationships in the system, CA is the only entity trusted by all other entities, and no other direct trust relationships exist between BVs and LAGs. In the network architecture, BVs access the power grid for power and information services via LAG, and LAG directly communicates with the power grid on behalf of the geographically dispersed BVs. CA participates in all the communications, and can derive the detailed power and information data to support bill services. It is noteworthy that communication between LAG and BVs is not limited to a specific communication manner. It can be based on either traditional computer networks or wireless communications. For instance, the interface between BV and LAG can use radio frequency identification (RFID).

The process of BVs' accessing the power grid can be categorized into three battery states and consequently three battery

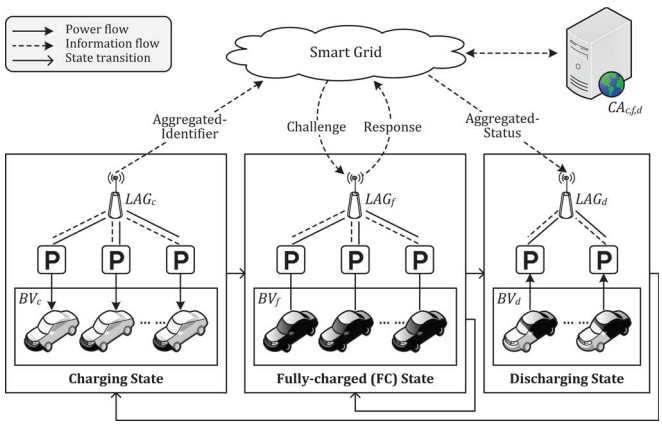


Fig. 2. The battery status-aware V2G network architecture.

state transitions. In a battery state, $\{BV, LAG, CA\}$ perform the following operations.

- In the charging state: For the sake of presentation, we use the notations $\{BV_c, LAG_c, CA_c\}$ for the variants of $\{BV, LAG, CA\}$ in the charging state. When BV_c enters into LAG_c's range, it attempts to establish communication with LAG_c for accessing the power grid.
- In the FC state: We use $\{BV_f, LAG_f, CA_f\}$ for the variants of $\{BV, LAG, CA\}$ in the FC state. BV_f has been fully charged, and becomes a potentially available energy source. It means that BV_f is a possible participant to feed its power back into the power grid. When BV_f receives the discharging request from CA_f, BV_f may agree or decline the request without being monitored.
- In the discharging state: We use $\{BV_d, LAG_d, CA_d\}$ for the variants of $\{BV, LAG, CA\}$ in the discharging state. If BV_f accepts the discharging request, it will perform discharging operation. Note that BV_f may terminate the discharging state in case when its power level decreases to a pre-defined threshold (e.g., 50%), or BV_f actively leaves the discharging operation.

During the battery state transition, there are different security and privacy requirements.

- During the charging-to-FC transition: $\{BV_c, LAG_c\}$ should establish mutual authentication to ascertain the identity validity. CA_c should perform authentication on $\{BV_c, LAG_c\}$ to avoid the conspiracy attack. Moreover, LAG_c can only obtain BV_c's basic group attribute without correlating BV_c with its real identity to protect the location related privacy.
- During the FC-to-discharging transition: BV_f should own full autonomy to decide whether or not to participate in the discharging operation. It means that BV_f may first agree to perform the discharging operation and turn into the discharging state, then withdraw the former response and decline the discharging request. BV_f may also first decline the discharging request and stay the current FC state, then change its mind and turn into the discharging state. Moreover, BV_f's response should be anonymously transmitted, and LAG_f cannot derive BV_f's detailed response to guess the interest related privacy.

TABLE I
THE SHARED SECRETS

	k_p	k_{si}	k_q	$s_{g_{BV}}^{LAG}$	$s_{LAG}^{g_{BV},CA}$	s_{CA}^{LAG}	$Cert_{BV_i}^{r}$	$Cert_{BV_i}^{h}$
BV_i	√	√	×	√	×	×	√	√
LAG	√	×	√	×	√	×	√	×
CA	×	√	√	×	×	√	√	√

√: The secret is available; ×: The secret is unavailable.

- During the discharging-to-charging transition: CA_d informs BV_d about its power status via LAG_d. LAG_d as an intermediary can only forward sensitive data to BV_d, while cannot obtain BV_d's individual power status for SOC related privacy consideration. Note that BV_{di} can change its discharging state to the charging state, and may freely perform immediate or later charging operation.

IV. THE PROPOSED AUTHENTICATION SCHEME: BASA

A. System Initialization

We consider the interactions of $\{BV_i, LAG, CA\}$ in V2G networks. $\{BV_i, LAG\}$ respectively have their own pseudonyms $\{PID_{BV_i}, PID_{LAG}\}$, and group identifiers $\{gid_{BV_i}, gid_{LAG}\}$. CA manages BV_i's pseudo power values $\{PST_{BV_{ci}}, PST_{BV_{fi}}, PST_{BV_{di}}\}$, which respectively represent the initially charging power value, fully charged power value, and discharged power value. Additionally, $\{BV_i, LAG\}$ have the full state identifiers $\{sid_{ci}, sid_{fi}, sid_{di}\}$, which are defined to update the pre-shared values into the corresponding charging/FC/discharging values, and CA also stores the charging state identifier sid_{ci}. $\{LAG, CA\}$ store state transition identifiers $\{sid_{fi|ci}, sid_{di|fi}\}$ which are used to transform the charging state values into FC state values, and the FC state values into discharging state values. The secret distribution is shown in Table I, and notations are listed in Table II.

1) The shared keys $\{k_p, k_{si}, k_q\}$: k_p is a secret key shared by BV_i and LAG, and is correlated with BV_i's group attribute gid_{BV_i}. k_p is applied for mutual authentication and selective secret disclosure. k_{si} is a secret key shared by BV_i and CA, and is correlated with BV_i's pseudonym PID_{BV_i}. k_q is also a secret key shared by LAG and CA, and is correlated with LAG's $\{gid_{LAG}, PID_{LAG}\}$. $\{k_{si}, k_q\}$ are used to realize mutual authentication and anonymous data transmission.
2) The shared values $\{s_{g_{BV}}^{LAG}, s_{LAG}^{g_{BV},CA}, s_{CA}^{LAG}\}$ are respectively owned $\{BV_i, LAG, CA\}$. Note that $s_{g_{BV}}^{LAG}$ is also owned by BVs attached with the same group attribute. $\{s_{g_{BV}}^{LAG}, s_{LAG}^{g_{BV}}\}$ and $\{s_{LAG}^{CA}, s_{CA}^{LAG}\}$ are pairwise secrets which satisfy the appointed mapping relationships: $s_{g_{BV}}^{LAG} = s_{LAG}^{g_{BV}} \oplus gid_{BV_i}$, and $s_{LAG}^{CA} = s_{CA}^{LAG} \oplus gid_{LAG}$.
3) The certificates $\{Cert_{BV_i}^{r}, Cert_{BV_i}^{h}\}$: The released certificate $Cert_{BV_i}^{r}$ is shared by all the legal entities. The hidden certificate $Cert_{BV_i}^{h}$ is owned by BV_i and CA, and is used by CA to determine BV_i's detailed identity for further billing purpose.

Two types of functions $F_{sid_{ci,fi,di}}$ and $F_{sid_{fi|ci,di|fi}}$ are respectively defined based on $\{sid_{ci}, sid_{fi}, sid_{di}\}$ and $\{sid_{fi|ci}, sid_{di|fi}\}$ for state-aware secret updating, in which

TABLE II
NOTATIONS

Notation	Description
BV_i	The i-th battery vehicle (BV) [24].
LAG, CA	The local aggregator, and central authority
BV_{xi}, LAG_x, CA_x	$x \in \{c, f, d\}$ represents the variants of $\{BV_i, LAG, CA\}$ respectively in the charging, FC, discharging state.
PID	The pseudonym.
ST, ΔST	The real power value, and its variation.
PST, ΔPST	The pseudo power, and its variation.
gid_{BV_i}, gid_{LAG}	The group identifier of BV_i, LAG.
sid_{ci}, sid_{fi}, sid_{di}	The state identifiers.
$sid_{fi\|ci}$, $sid_{di\|fi}$	The state transition identifiers.
ts, r	The timestamp, and random number.
k_p, k_{si}	The shared keys.
s_{gBV}^{LAG}, $s_{LAG}^{g_{BV},CA}$, s_{CA}^{LAG}	The shared values.
$Cert_{BV}^{r}$, $Cert_{BV}^{h}$	The released certificate, hidden certificate.
$Permit$, $Respon$	The access permit, and response of the challenged request.
$Pseudo/Status$, $Percen$	The real/pseudo power related value, and pseudo power percentage.
$H/H_k(.)$, $E(.)$, $R(.)$, $F_(.)$	The hash/keyed hash message authentication code (HMAC) function, encryption, pseudo-random function, and pre-shared defined function.
Ξ, $\rightarrow$, $\Rightarrow$	The multi-element cascade operator, extension operator, and mapping operator.

$sid_{fi} = sid_{ci} \oplus sid_{fi|ci}$ and $sid_{di} = sid_{ci} \oplus sid_{fi|ci} \oplus sid_{di|fi}$. Here, $\{F_{sid_{ci}}, F_{sid_{fi|ci,di|fi}}\} : \mathbb{R}^* \times \{0,1\}^* \rightarrow \mathbb{R}^*$, $F_{sid_{fi}} : \mathbb{R}^* \times \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{R}^*$, and $F_{sid_{di}} : \mathbb{R}^* \times \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{R}^*$. These functions $\{F_{sid_*}\}$ satisfy the following relationships, in which $sid_* \in \mathbb{R}^*$ represents all the state/state transition identifiers, and $\{x_1, x_2, x_3\} \in \{0,1\}^*$.

$$\begin{aligned} F_{sid_{ci}}(x_1) &= sid_{ci} \oplus x_1, \\ F_{sid_{fi}}(x_1 \oplus x_2) &= F_{sid_{fi|ci}}(F_{sid_{ci}}(x_1) \oplus x_2), \\ F_{sid_{di}}(x_1 \oplus x_2 \oplus x_3) &= F_{sid_{di|fi}}(F_{sid_{fi}}(x_1 \oplus x_2) \oplus x_3). \end{aligned}$$

Assume that there is a secret value M, and state/state transition identifiers based functions F_{sid_*} are applied to update M into the corresponding charging/FC/discharging state values $\{M_c, M_f, M_d\}$ that satisfy the following equations, in which $\{ts_1, ts_2, ts_3\}$ are the timestamps.

$$\begin{aligned} M_c &= F_{sid_{ci}}(M\|ts_1) = sid_{ci} \oplus (M\|ts_1), \\ M_f &= F_{sid_{fi}}((M\|ts_1) \oplus ts_2) \\ &= F_{sid_{fi|ci}}(M_c \oplus ts_2), \\ M_d &= F_{sid_{di}}((M\|ts_1) \oplus ts_2 \oplus ts_3) \\ &= F_{sid_{di|fi}}(F_{sid_{fi|ci}}(M_c \oplus ts_2) \oplus ts_3) \\ &= F_{sid_{di|fi}}(M_f \oplus ts_3). \end{aligned}$$

Note that standard algorithms can be introduced into F_{sid_*}, which are determined according to the computational capability in V2G networks. Furthermore, BV_i and CA establish three mapping relationships of the real power related values $\{ST_{BV_{ci,fi,di}}, \Delta ST_{BV_i}\}$ and the pseudo power related values $\{PST_{BV_{ci,fi,di}}, \Delta PST_{BV_i}\}$.

$$\begin{aligned} H(ST_{BV_{ci,fi,di}}) &\Rightarrow PST_{BV_{ci,fi,di}}, \\ H(\Xi_{x=c,f,d}(ST_{BV_{xi}})) &= H(ST_{BV_{ci}}\|ST_{BV_{fi}}\|ST_{BV_{di}}) \\ &\Rightarrow \Xi_{x=c,f,d}(PST_{BV_{xi}}) = PST_{BV_i}, \\ \Delta PST_{BV_i} &\Rightarrow \Delta ST_{BV_i}. \end{aligned}$$

The pseudo power values satisfy the following equations.

$$\begin{aligned} &R(PST_{BV}) \\ &= R(\Xi_{x=c,f,d}(PST_{BV_x/BV_{xi}} + PST_{BV_{xi}})) \\ &= R(PST_{BV_c/BV_{ci}}\|PST_{BV_f/BV_{fi}}\|PST_{BV_d/BV_{di}}) \\ &+ R(PST_{BV_{ci}}\|PST_{BV_{fi}}\|PST_{BV_{di}}). \end{aligned}$$

B. Charging-to-FC Phase: The Aggregated-Identifier Based Authentication Protocol (AIDP)

Fig. 3 shows the interaction among BV_{ci}, LAG_c, and CA_c in the charging-to-FC phase. Here BV_{ci} represents multiple charging vehicles $\{BV_{c1}, .., BV_{cI}\}(I \in \mathbb{N}^*)$, which simultaneously access the local charging aggregator LAG_c.

1) Query and LAG_c's Verification on BV_{ci}: LAG_c extracts a timestamp $ts_{LAG_c}^1$ and its group identifier gid_{LAG}. LAG_c generates a random number r_{LAG}, and extends r_{LAG} into $\{r_{LAG_1}, \ldots, r_{LAG_I}\}$ by Hamming distance based extension operation. Thereafter, LAG_c transmits $ts_{LAG_c}^1\|gid_{LAG}\|r_{LAG_i}$ to BV_{ci} as an query to initiate a new session.

Upon receiving the query, BV_{ci} extracts the timestamps $\{ts_{BV_{ci}}^{old}, ts_{BV_{ci}}\}$ and its group identifier gid_{BV_i}. Thereinto, $\{ts_{BV_{ci}}^{old}, ts_{BV_{ci}}\}$ are respectively the formerly stored timestamp and the currently extracted timestamp. BV_{ci} performs a quick check on LAG_c by comparing $ts_{LAG_c}^1$ with $\{ts_{BV_{ci}}^{old}, ts_{BV_{ci}}\}$. If $ts_{LAG_c}^1 - ts_{BV_{ci}}^{old} \leq 0$ or $ts_{BV_{ci}} - ts_{LAG_c}^1 > \Delta$ (Δ is an acceptable time interval), LAG_c will be regarded as an illegal aggregator and AIDP will terminate. Otherwise BV_{ci} will generate a random number r_{BV_i} and extract the stored static values $\{sid_{ci}, Cert_{BV_i}^r, s_{gBV}^{LAG}, k_p, k_{si}\}$. Thereinto, sid_{ci} is the charging state identifier to indicate that BV_{ci} is currently in the charging state and the corresponding charging state values should be applied in AIDP, $Cert_{BV_i}^r$ is BV_i's released certificate, and s_{gBV}^{LAG} is a secret owned by the BVs in the same group. BV_{ci} obtains the updated charging state values $\{Cert_{BV_{ci}}^r, s_{gBV_c}^{LAG_c}\}$ by applying $F_{sid_{ci}}$.

$$\begin{aligned} Cert_{BV_{ci}}^r &= F_{sid_{ci}}(Cert_{BV_i}^r\|ts_{BV_{ci}}) \\ s_{gBV_c}^{LAG_c} &= F_{sid_{ci}}(s_{gBV}^{LAG}\|ts_{LAG_c}^1). \end{aligned}$$

BV_{ci} computes authentication operators $\{S_{BV_{ci}}, \alpha_{BV_{ci}}\}$ based on the shared keys $\{k_p, k_{si}\}$ respectively.

$$\begin{aligned} S_{BV_{ci}} &= E_{k_p}(s_{gBV_c}^{LAG_c} \oplus gid_{BV_i}) \\ \alpha_{BV_{ci}} &= H_{k_{si}}(Cert_{BV_{ci}}^r \oplus r_{LAG_i}). \end{aligned}$$

BV_{ci} further transmits $ts_{BV_{ci}}\|gid_{BV_i}\|r_{BV_i}\|S_{BV_{ci}}\|\alpha_{BV_{ci}}$ to LAG_c. Afterwards, LAG_c extracts a timestamp $ts_{LAG_c}^2$

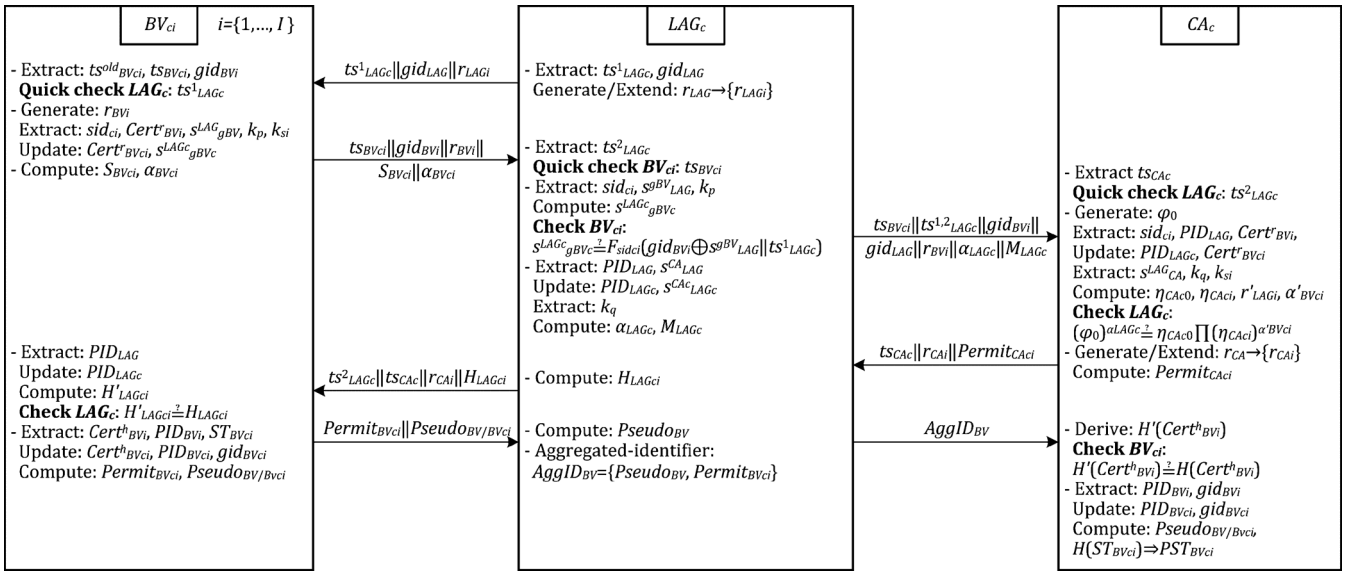


Fig. 3. Charging-to-FC phase: The aggregated-identifier based authentication protocol (AIDP).

to perform a quick check on BV_{ci}. If $ts_{BV_{ci}} - ts^1_{LAG_c} \leq 0$ or $ts^2_{LAG_c} - ts_{BV_{ci}} > \Delta$, LAG_c will regard BV_{ci} as an illegal vehicle and AIDP will terminate. Otherwise, LAG_c will determine BV_{ci}'s group attribute by gid_{BV_i}, and extract $\{sid_{ci}, s^{g_{BV}}_{LAG}, k_p\}$ to derive $s^{LAG_c}_{g_{BV_c}}$. Here, LAG_c can only obtain BV_{ci}'s general group attribute without determining its detailed identity.

$$s^{LAG_c}_{g_{BV_c}} = gid_{BV_i} \oplus E^{-1}_{k_p}(S_{BV_{ci}}).$$

LAG_c verifies BV_{ci} by checking whether the derived $s^{LAG_c}_{g_{BV_c}}$ and its local $s^{g_{BV}}_{LAG}$ satisfy the defined mapping relationship. If it holds, LAG_c will regard BV_{ci} as a legal vehicle. Otherwise LAG_c will eliminate BV_{ci} from AIDP.

$$s^{LAG_c}_{g_{BV_c}} \stackrel{?}{=} F_{sid_{ci}}((gid_{BV_i} \oplus s^{g_{BV}}_{LAG}) \| ts^1_{LAG_c}).$$

2) CA_c's Verification on LAG_c: LAG_c extracts and updates $\{PID_{LAG}, s^{CA}_{LAG}\}$ to obtain the charging state values $\{PID_{LAG_c}, s^{CA_c}_{LAG_c}\}$. LAG_c also extracts gid_{LAG}, which is LAG's group identifier and can be further applied by CA to determine LAG's detailed identity.

$$PID_{LAG_c} = F_{sid_{ci}}(PID_{LAG} \| (ts^1_{LAG_c} \oplus ts^2_{LAG_c}))$$
$$s^{CA_c}_{LAG_c} = F_{sid_{ci}}(s^{CA}_{LAG} \| ts_{BV_{ci}}).$$

Afterwards, LAG_c extracts the shared key k_q to compute operators $\{\alpha_{LAG_c}, M_{LAG_c}\}$, in which n is a large prime.

$$\alpha_{LAG_c} = s^{CA_c}_{LAG_c} + \sum_{i=1}^{I}(\alpha_{BV_{ci}} r_{BV_i}) \pmod n$$
$$M_{LAG_c} = E_{k_q}(PID_{LAG_c} \oplus r_{LAG_i})$$

LAG_c transmits $ts_{BV_{ci}} \| ts^{1,2}_{LAG_c} \| gid_{BV_i} \| gid_{LAG} \| r_{BV_i}$ and $\alpha_{LAG_c} \| M_{LAG_c}$ to CA_c. Upon receiving the messages, CA_c extracts a timestamp ts_{CA_c} to perform a quick check on LAG_c. If $ts_{CA_c} - ts^2_{LAG_c} > \Delta$, LAG_c will be regarded as an illegal aggregator and AIDP will terminate. Otherwise, CA_c will generate a random number φ_0 for further authentication. CA_c extracts and updates $\{PID_{LAG}, Cert^r_{BV_{ci}}\}$ into $\{PID_{LAG_c}, Cert^r_{BV_i}\}$ according to the same algorithm. CA_c further extracts $\{s^{LAG}_{CA}, k_q, k_{si}\}$ to compute $\{\eta_{CA_{c0}}, \eta_{CA_{ci}}, r'_{LAG_i}\}$ as authentication operators, and re-computes $\alpha'_{BV_{ci}}$ by the derived r'_{LAG_i}.

$$\eta_{CA_{c0}} = (\varphi_0)^{F_{sid_{ci}}((gid_{LAG} \oplus s^{LAG}_{CA}) \| ts_{BV_{ci}})} \pmod n$$
$$\eta_{CA_{ci}} = (\varphi_0)^{r_{BV_i}} \pmod n$$
$$r'_{LAG_i} = E^{-1}_{k_q}(M_{LAG_c}) \oplus PID_{LAG_c}$$
$$\alpha'_{BV_{ci}} = H_{k_{si}}(Cert^r_{BV_{ci}} \oplus r'_{LAG_i}).$$

CA_c verifies LAG_c by checking the following function. If it holds, CA_c will regard LAG_c as a legal aggregator. Otherwise, AIDP will terminate.

$$(\varphi_0)^{\alpha_{LAG_c}} \stackrel{?}{=} \eta_{CA_{c0}} \prod_{i=1}^{I} (\eta_{CA_{ci}})^{\alpha'_{BV_{ci}}}.$$

CA_c generates a random number r_{CA}, and extends r_{CA} into $\{r_{CA_i}\}$. CA_c computes $Permit_{CA_{ci}}$ to indicate that CA_c has authorized LAG_c and granted an access permit to LAG_c, and transmits $ts_{CA_c} \| r_{CA_i} \| Permit_{CA_{ci}}$ to LAG_c.

$$Permit_{CA_{ci}} = E_{k_{si}}(Cert^r_{BV_{ci}} \oplus (r_{BV_i} \| r'_{LAG_i}))$$

3) BV_{ci}'s Verification on LAG_c and Aggregated-Identifier Generation: LAG_c further computes $H_{LAG_{ci}}$, and transmits $ts^2_{LAG_c} \| ts_{CA_c} \| r_{CA_i} \| H_{LAG_{ci}}$ to BV_{ci}

$$H_{LAG_{ci}} = H_{k_p}(PID_{LAG_c} \oplus Permit_{CA_{ci}}).$$

Thereafter, BV_{ci} extracts PID_{LAG} to obtain the updated $PID_{LAG_c} = F_{sid_{ci}}(PID_{LAG} \| (ts^1_{LAG_c} \oplus ts^2_{LAG_c}))$, and re-computes $H'_{LAG_{ci}}$ by its local values.

$$H'_{LAG_{ci}} = H_{k_p}(E_{k_{si}}(Cert^r_{BV_{ci}} \oplus (r_{BV_i} \| r_{LAG_i})) \oplus PID_{LAG_c}).$$

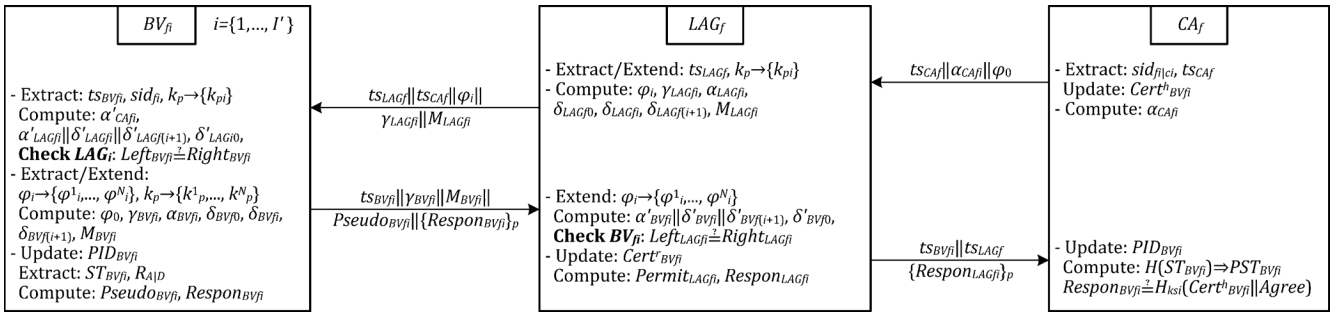


Fig. 4. FC-to-discharging phase: The selective disclosure based challenge-response authentication protocol (SDAP).

BV_{ci} verifies LAG_c by checking $H'_{LAG_{ci}} \stackrel{?}{=} H_{LAG_{ci}}$. If it holds, LAG_c will extract $\{Cert^h_{BV_i}, PID_{BV_i}, ST_{BV_{ci}}\}$, in which $ST_{BV_{ci}}$ is the current real power value. Thereafter, BV_{ci} obtains the updated charging state values $\{Cert^h_{BV_{ci}}, PID_{BV_{ci}}, gid_{BV_{ci}}\}$.

$$\begin{aligned} Cert^h_{BV_{ci}} &= F_{sid_{ci}}(H(Cert^h_{BV_i})\|ts_{BV_{ci}}) \\ PID_{BV_{ci}} &= F_{sid_{ci}}(PID_{BV_i}\|ts_{CA_c}) \\ gid_{BV_{ci}} &= F_{sid_{ci}}(gid_{BV_i}\|ts^2_{LAG_c}). \end{aligned}$$

BV_{ci} computes $\{Permit_{BV_{ci}}, Pseudo_{BV/BV_{ci}}\}$, in which $Permit_{BV_{ci}}$ indicates that BV_{ci} grants an access permit to LAG_c. $Pseudo_{BV/BV_{ci}}$ is a pseudo power value computed by wrapping the current power value $ST_{BV_{ci}}$ before charging, here $PID_{BV/BV_{ci}} = gid_{BV_{ci}} - PID_{BV_{ci}}$

$$\begin{aligned} Permit_{BV_{ci}} &= E_{k_{si}}(Cert^h_{BV_{ci}} \oplus (r_{LAG_i}\|r_{CA_i})) \\ Pseudo_{BV/BV_{ci}} &= E_{k_{si}}(PID_{BV/BV_{ci}} \oplus H(ST_{BV_{ci}})). \end{aligned}$$

Thereafter, BV_{ci} transmits $Permit_{BV_{ci}}\|Pseudo_{BV/BV_{ci}}$ to LAG_c, and LAG_c computes $Pseudo_{BV}$ to aggregate multiple BVs' $Pseudo_{BV/BV_{ci}}$ and $Permit_{BV_{ci}}$.

$$Pseudo_{BV} = \Xi^I_{i=1}(Pseudo_{BV/BV_{ci}} \oplus Permit_{BV_{ci}}).$$

LAG_c establishes an aggregated-identifier $AggID_{BV}$ for $\{BV_1, \ldots, BV_I\}$, and transmits $AggID_{BV}$ to CA_c

$$AggID_{BV} = \{Pseudo_{BV}, Permit_{BV_{ci}}\}.$$

Upon receiving the message, CA_c derives the hashed hidden certificate $H'(Cert^h_{BV_{ci}})$. CA_c checks BV_{ci} by comparing the derived $H'(Cert^h_{BV_i})$ with its locally re-computed $H(Cert^h_{BV_i})$. If $H'(Cert^h_{BV_i}) \neq H(Cert^h_{BV_i})$, CA will regard BV_{ci} as an illegal vehicle and eliminate it from AIDP.

$$\begin{aligned} &H'(Cert^h_{BV_i})\|ts'_{CA_c} \\ &= F^{-1}_{sid_{ci}}(E^{-1}_{k_{si}}(Permit_{BV_{ci}}) \oplus (r'_{LAG_i}\|r_{CA_i})). \end{aligned}$$

CA_c extracts $\{PID_{BV_i}, gid_{BV_i}\}$ to obtain the updated charging state values $\{PID_{BV_{ci}}, gid_{BV_{ci}}\}$, and derives $Pseudo_{BV/BV_{ci}}$ and $H(ST_{BV_{ci}})$. Here, $\Xi^{-1}_i(.)$ is the inverse operation that is used to derive the i-th element of the cascaded value $\Xi^I_{i=1}(Pseudo_{BV/BV_{ci}} \oplus Permit_{BV_{ci}})$.

$$Pseudo_{BV/BV_{ci}} = Permit_{BV_{ci}} \oplus \Xi^{-1}_i(Pseudo_{BV})$$

$$\begin{aligned} H(ST_{BV_{ci}}) = &E^{-1}_{k_{si}}(Pseudo_{BV/BV_{ci}}) \\ &\oplus (gid_{BV_{ci}} - PID_{BV_{ci}}) \end{aligned}$$

CA_c applies $H(ST_{BV_{ci}})$ to retrieve $PST_{BV_{ci}}$ according to the distributed hash table, which shows the mapping from the hashed real power values to the pseudo power values. Here, $PST_{BV_{ci}}$ is the initial pseudo power value of BV_{ci}. Till now, BV_{ci} and LAG_c have established mutual authentication, and CA_c has recognized the charging vehicles' identities and their corresponding pseudo power values. Based on the authentication, the aggregated-identifier is transmitted to CA_c via LAG_c without revealing any individual privacy.

C. FC-to-Discharging Phase: The Selective Disclosure Based Challenge-Response Authentication Protocol (SDAP)

Fig. 4 shows the interaction among BV_{fi}, LAG_f, and CA_f in the FC-to-discharging phase. Thereinto, BV_{fi} represents multiple fully-charged vehicles $\{BV_{f1,\ldots,}BV_{fI'}\}$ $(I' \leq I)$, which are the possible participants to feed their power back into the power grid. The vehicles have full autonomy to decide whether or not to participate in the discharging operation.

1) CA_f's Discharging Challenge: CA_f extracts the state transition identifier $sid_{fi|ci}$ that is an operator to transfer the charging state values into the FC state values. CA_f extracts a timestamp ts_{CA_f} to compute the FC state hidden certificate $Cert^h_{BV_{fi}}$ by $F_{sid_{fi|ci}}$. Then, CA_f computes $\alpha_{CA_{fi}}$, and transmits $ts_{CA_f}\|\alpha_{CA_{fi}}\|\varphi_0$ as a nonspecific discharging request to all the vehicles, in which φ_0 is a random number generated in AIDP.

$$\begin{aligned} Cert^h_{BV_{fi}} &= F_{sid_{fi|ci}}(Cert^h_{BV_{ci}} \oplus ts_{CA_f}) \\ \alpha_{CA_{fi}} &= H_{k_{si}}(Cert^h_{BV_{fi}} \oplus r_{CA_f}) \end{aligned}$$

2) BV_{fi}'s Verification on LAG_f: LAG_f generates ts_{LAG_f}, and extends the shared key k_p into $\{k_{pi}\} = \{k_{p1}, \ldots, k_{pI'}\} \in \{0,1\}^*$. Here, k_{pi} as a reconstructed secret key assigned to BV_i and LAG, is computed by pre-appointed algorithm. LAG_f wraps φ_0 into $\varphi_i = E_{k_{pi}}(\varphi_0 \oplus r_{CA_f})$, and computes the authentication operators $\{\gamma_{LAG_{fi}}, \alpha_{LAG_{fi}}\}$ and $\{\delta_{LAG_{f0}}, \delta_{LAG_{fi}}, \delta_{LAG_{f(i+1)}}, M_{LAG_{fi}}\}$. Thereafter, LAG_f transmits $ts_{LAG_f}\|ts_{CA_f}\|\varphi_i\|\gamma_{LAG_{fi}}\|M_{LAG_{fi}}$ to BV_{fi}.

$$\begin{aligned} \gamma_{LAG_{fi}} &= (\varphi_0)^{ts_{BV_{ci}}}(\varphi_i)^{r_{LAG_i}} \pmod n \\ \alpha_{LAG_{fi}} &= (\varphi_0)^{\gamma_{LAG_{fi}}}(\varphi_i)^{k_{pi}} \pmod n \\ \delta_{LAG_{f0}} &= H(\alpha_{LAG_{fi}}\|\gamma_{LAG_{fi}}\|r_{BV_i}) \end{aligned}$$

$$\delta_{LAG_{fi}} = \delta_{LAG_{f0}} k_{pi} + r_{LAG_i} \pmod n$$
$$\delta_{LAG_{f(i+1)}} = \delta_{LAG_{f0}} \gamma_{LAG_{fi}} + ts_{BV_{ci}} \pmod n$$
$$M_{LAG_{fi}} = \alpha_{CA_{fi}} \oplus E_{k_{pi}}(\alpha_{LAG_{fi}} \| \delta_{LAG_{fi}} \| \delta_{LAG_{f(i+1)}})$$

BV_{fi} extracts $\{ts_{BV_{fi}}, sid_{fi}\}$, in which sid_{fi} is the FC state identifier to indicate that BV_{fi} is currently in the FC state. Thereafter, BV_{fi} re-computes $\{\alpha'_{CA_{fi}}, \delta'_{LAG_{f0}}\}$, and derives $\alpha'_{LAG_{fi}} \| \delta'_{LAG_{fi}} \| \delta'_{LAG_{f(i+1)}}$.

$$\alpha'_{CA_{fi}} = H_{k_{si}}(F_{sid_{fi}}((H(Cert^h_{BV_i}) \| ts_{BV_{ci}}) \oplus ts_{CA_f})) \oplus r_{CA_i})$$
$$\alpha'_{LAG_{fi}} \| \delta'_{LAG_{fi}} \| \delta'_{LAG_{f(i+1)}} = E^{-1}_{k_{pi}}(M_{LAG_{fi}} \oplus \alpha'_{CA_{fi}})$$
$$\delta'_{LAG_{f0}} = H(\alpha'_{LAG_{fi}} \| \gamma_{LAG_{fi}} \| r_{BV_i}).$$

BV_{fi} computes $\{Left_{BV_{fi}}, Right_{BV_{fi}}\}$ to verify LAG_f. If $Left_{BV_{fi}} = Right_{BV_{fi}}$ holds, BV_{fi} will regard LAG_f as a legal aggregator. Otherwise, SDAP will terminate.

$$Left_{BV_{fi}} = \gamma_{LAG_{fi}} (\alpha'_{LAG_{fi}})^{\delta'_{LAG_{f0}}}$$
$$Right_{BV_{fi}} = (\varphi_0)^{\delta'_{LAG_{f(i+1)}}} (\varphi_i)^{\delta'_{LAG_{fi}}}$$

3) LAG_f's Verification on BV_{fi}: BV_{fi} extends $\{\varphi_i, k_p\}$ into $\{\varphi^n_i\}$ and $\{k^m_p\}$, in which $\{\varphi^n_i\} = \{\varphi^1_i, \ldots, \varphi^N_i\} \in \{0,1\}^*$, and $\{k^m_p\} = \{k^1_p, \ldots, k^N_p\} \in \mathbb{Z}^*$. Note that $\{\varphi^n_i\}$ attaches selective attributes for LAG_f. BV_{fi} derives φ_0 by computing $E^{-1}_{k_{pi}}(\varphi_i) \oplus r_{CA_f}$, computes the authentication operators $\{\gamma_{BV_{fi}}, \alpha_{BV_{fi}}, \delta_{BV_{f0}}, \delta_{BV_{fi}}, \delta_{BV_{f(i+1)}}\}$, and wraps $\alpha_{BV_{fi}} \| \delta_{BV_{fi}} \| \delta_{BV_{f(i+1)}}$ into $M_{BV_{fi}} = M_{LAG_{fi}} \oplus E_{k_{pi}}(\alpha_{BV_{fi}} \| \delta_{BV_{fi}} \| \delta_{BV_{f(i+1)}})$ for further transmission.

$$\gamma_{BV_{fi}} = (\alpha'_{LAG_{fi}})^{ts_{LAG_f}} \prod_{n=1}^{N} (\varphi^n_i)^{r_{BV_i}} \pmod n$$
$$\alpha_{BV_{fi}} = (\alpha'_{LAG_{fi}})^{\gamma_{BV_{fi}}} (\varphi_0)^{k_p} \times \prod_{n=1}^{N} (\varphi^n_i)^{\sum_{m=1}^{N} (k^m_p)} \pmod n$$
$$\delta_{BV_{f0}} = H(\alpha_{BV_{fi}} \| \gamma_{BV_{fi}} \| r_{LAG_i})$$
$$\delta_{BV_{fi}} = \delta_{BV_{f0}} \sum_{m=1}^{N} (k^m_p) + r_{BV_i} \pmod n$$
$$\delta_{BV_{f(i+1)}} = \delta_{BV_{f0}} \gamma_{BV_{fi}} + ts_{LAG_f} \pmod n$$

BV_{fi} obtains the updated FC state value $PID_{BV_{fi}}$ by computing $F_{sid_{fi}}((PID_{BV_i} \| ts_{CA_c}) \oplus ts_{BV_{fi}})$, and extracts $\{ST_{BV_{fi}}, R_{A|D}\}$ to compute $Pseudo_{BV_{fi}}$ and $Respon_{BV_{fi}}$. Here, $ST_{BV_{fi}}$ is the FC state real power value $ST_{BV_{fi}}$. The response $R_{A|D}$ (i.e., $\{Agree, Decline\}$) is alternatively chosen as a command which is linked with the actuator to perform or not perform the discharging operation.

$$Pseudo_{BV_{fi}} = E_{k_{si}}(PID_{BV_{fi}} \oplus H(ST_{BV_{fi}}))$$
$$Respon_{BV_{fi}} = H_{k_{si}}(Cert^h_{BV_{fi}} \| R_{A|D}).$$

BV_{fi} transmits $ts_{BV_{fi}} \| \gamma_{BV_{fi}} \| M_{BV_{fi}} \| Pseudo_{BV_{fi}}$ and $\{Respon_{BV_{fi}}\}_p$ to LAG_f, in which the former is used for authentication, and the latter is used to declare BV_{fi}'s response. $\{Respon_{BV_{fi}}\}_p$ indicates that $Respon_{BV_{fi}}$ is periodically transmitted to LAG_f, which realizes that BV_{fi} can freely join and leave the discharging state without revealing its private identity. The period can be determined according to actual demands (e.g., 15 minutes). Afterwards, LAG_f extends φ_i into $\{\varphi^n_i\} = \{\varphi^1_i, \ldots, \varphi^N_i\}$. LAG_f derives $\alpha'_{BV_{fi}} \| \delta'_{BV_{fi}} \| \delta'_{BV_{f(i+1)}}$, and re-computes $\delta'_{BV_{f0}}$ by the derived $\alpha'_{BV_{fi}}$.

$$\alpha'_{BV_{fi}} \| \delta'_{BV_{fi}} \| \delta'_{BV_{f(i+1)}} = E^{-1}_{k_{pi}}(M_{BV_{fi}} \oplus M_{LAG_{fi}})$$
$$\delta'_{BV_{f0}} = H(\alpha'_{BV_{fi}} \| \gamma_{BV_{fi}} \| r_{LAG_i}).$$

Thereafter, LAG_f computes $\{Left_{LAG_{fi}}, Right_{LAG_{fi}}\}$ to verify BV_{fi} by checking $Left_{LAG_{fi}} \stackrel{?}{=} Right_{LAG_{fi}}$. If it holds, LAG_f will regard BV_{fi} as a legal vehicle. Otherwise, SDAP will terminate.

$$Left_{LAG_{fi}} = \gamma_{BV_{fi}} (\alpha'_{BV_{fi}})^{\delta'_{BV_{f0}}}$$
$$Right_{LAG_{fi}} = (\alpha_{LAG_{fi}})^{\delta'_{BV_{f(i+1)}}} (\varphi_0)^{k_p \delta'_{BV_{f0}}} \prod_{n=1}^{N} (\varphi^n_i)^{\delta'_{BV_{fi}}}$$

4) LAG_f Forwards BV_{fi}'s Response: LAG_f obtains the updated $Cert^r_{BV_{fi}} = F_{sid_{fi}}(Cert^r_{BV_i} \| ts_{BV_{ci}}) \oplus ts_{CA_f})$, computes $Permit_{LAG_{fi}}$, and establishes $Respon_{LAG_{fi}}$ with the elements $\{Respon_{BV_{fi}}, Pseudo_{BV_{fi}}, Permit_{LAG_{fi}}\}$.

$$Permit_{LAG_{fi}} = E_{k_{pi}}(Permit_{BV_{ci}} \oplus Cert^r_{BV_{fi}}).$$

LAG_f transmits $ts_{BV_{fi}} \| ts_{LAG_f}$ and $\{Respon_{LAG_{fi}}\}_p$ to CA_f, and CA_f computes the updated $PID_{BV_{fi}}$ by computing $F_{sid_{fi|ci}}(PID_{BV_{ci}} \oplus ts_{BV_{fi}})$, and derives the hashed FC state real power value $H(ST_{BV_{fi}})$ to retrieve the corresponding pseudo power value $PST_{BV_{fi}}$.

$$H(ST_{BV_{fi}}) = E^{-1}_{k_{si}}(Pseudo_{BV_{fi}}) \oplus PID_{BV_{fi}} \Rightarrow PST_{BV_{fi}}$$

CA_f checks $Respon_{BV_{fi}} \stackrel{?}{=} H_{k_{si}}(Cert^h_{BV_{fi}} \| Agree)$ to determine BV_{fi}'s response. If it holds, CA_f will believe that BV_{fi} accepts the request, and turn BV_{fi} into the discharging state. Otherwise, CA_f will keep BV_{fi} in the current FC state.

D. Discharging-to-Charging Phase: The Aggregated-Status Based Authentication Protocol (ASTP)

Fig. 5 shows the interaction among BV_{di}, LAG_d, and CA_d in the discharging-to-charging transition. Thereinto, BV_{di} represents multiple discharging vehicles $\{BV_{d1,\ldots,}BV_{dI''}\}$ ($I'' \leq I'$), which have accepted CA's discharging request and performed discharging for a period. If the BV wants to quit the discharging operation or the vehicle's battery reduces to a certain power level, BV_{di} will turn into the charging state.

1) LAG_d's Verification on BV_{di}: BV_{di} extracts $\{sid_{di}, ts_{BV_{di}}, PID_{LAG}, ST_{BV_{di}}\}$, in which sid_{di} is the discharging state identifier to indicate that BV_{di} is in the discharging state, and $ST_{BV_{di}}$ is the remaining power value after performing the discharging operation. BV_{di} respectively computes the updated discharging state values $\{PID_{BV_{di}}, PID_{LAG_d}\}$.

$$PID_{BV_{di}} = F_{sid_{di}}((PID_{BV_i} \| ts_{CA_c}) \oplus ts_{BV_{fi}} \oplus ts_{LAG_f})$$
$$PID_{LAG_d} = F_{sid_{di}}((PID_{LAG} \| (ts^1_{LAG_c} \oplus ts^2_{LAG_c})) \oplus ts_{CA_f} \oplus ts_{BV_{di}}).$$

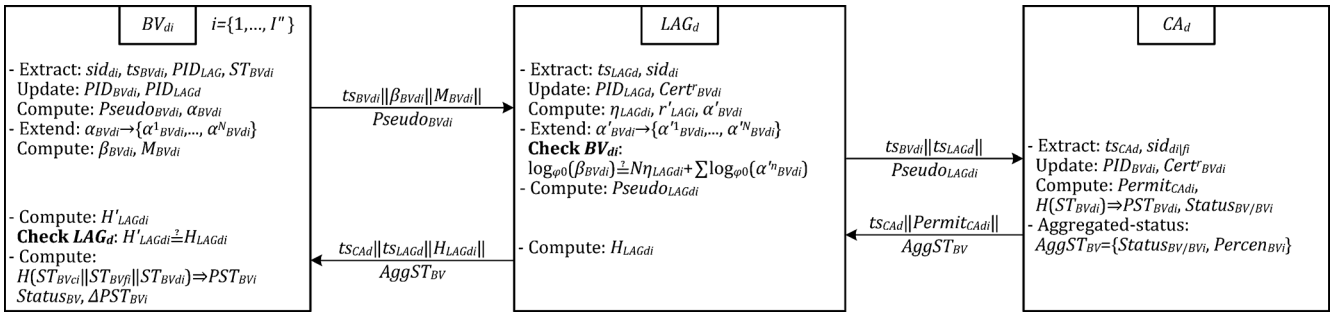


Fig. 5. Discharging-to-charging phase: The aggregated-status based authentication protocol (ASTP).

BV_{di} further computes $Pseudo_{BV_{di}}$ and $\alpha_{BV_{di}}$.

$$Pseudo_{BV_{di}} = E_{k_{si}}(PID_{BV_{di}} \oplus H(ST_{BV_{di}}))$$
$$\alpha_{BV_{di}} = H_{k_{pi}}(Pseudo_{BV_{di}} \oplus r_{LAG_i})$$

BV_{di} extends $\alpha_{BV_{di}}$ into $\{\alpha^n_{BV_{di}}\} = \{\alpha^1_{BV_{di}}, \ldots, \alpha^N_{BV_{di}}\}$, and computes $\beta_{BV_{di}}$ and $M_{BV_{di}}$. Thereafter, BV_{di} transmits $ts_{BV_{di}}\|\beta_{BV_{di}}\|M_{BV_{di}}\|Pseudo_{BV_{di}}$ to LAG_d.

$$\beta_{BV_{di}} = \prod_{n=1}^{N}(\alpha^n_{BV_{di}} r_{BV_i}) \pmod n$$
$$M_{BV_{di}} = E_{k_{pi}}(PID_{LAG_d} \oplus r_{LAG_i}).$$

Thereafter, LAG_d extracts $\{ts_{LAG_d}, sid_{di}\}$, and obtains the updated discharging state values $\{PID_{LAG_d}, Cert^r_{BV_{di}}\}$.

$$PID_{LAG_d} = F_{sid_{di}fi}(F_{sid_{fici}}(PID_{LAG_c} \oplus ts_{CA_f}) \oplus ts_{BV_{di}})$$
$$Cert^r_{BV_{di}} = F_{sid_{di}fi}(Cert^r_{BV_{fi}} \oplus ts_{LAG_d})$$

LAG_d first computes $\eta_{LAG_{di}}$, and derives r'_{LAG_i} to re-compute $\alpha'_{BV_{di}}$.

$$\eta_{LAG_{di}} = \log_{\varphi_0}(r_{BV_i}) \pmod n$$
$$r'_{LAG_i} = E^{-1}_{k_{pi}}(M_{BV_{di}}) \oplus PID_{LAG_d}$$
$$\alpha'_{BV_{di}} = H_{k_{pi}}(Pseudo_{BV_{di}} \oplus r'_{LAG_i})$$

LAG_d extends $\alpha'_{BV_{di}}$ into $\{\alpha'^n_{BV_{di}}\} = \{\alpha'^1_{BV_{di}}, \ldots, \alpha'^N_{BV_{di}}\}$, and verifies BV_{di} by checking the following function.

$$\log_{\varphi_0}(\beta_{BV_{di}}) \stackrel{?}{=} N\eta_{LAG_{di}} + \sum_{n=1}^{N}\log_{\varphi_0}(\alpha'^n_{BV_{di}}).$$

If it holds, LAG_d will regard BV_{di} as an legal vehicle. Otherwise, LAG_d will eliminate BV_{di} from ASTP. Afterwards, LAG_d computes $Pseudo_{LAG_{di}}$, and transmits $ts_{BV_{di}}\|ts_{LAG_d}\|Pseudo_{LAG_{di}}$ to CA_d.

$$Pseudo_{LAG_{di}} = Pseudo_{BV_{di}} \oplus H(Cert^r_{BV_{di}}\|r_{LAG_i})$$

2) CA_d's Aggregated-status Generation and BV_{di}'s Verification on LAG_d: CA_d extracts $\{ts_{CA_d}, sid_{di|fi}\}$, in which $sid_{di|fi}$ refers to the state transition identifier that transfers the FC state values into the discharging state values. CA_d obtains the updated values $\{PID_{BV_{di}}, Cert^r_{BV_{fi}}\}$.

$$PID_{BV_{di}} = F_{sid_{di}fi}(PID_{BV_{fi}} \oplus ts_{LAG_f})$$
$$Cert^r_{BV_{di}} = F_{sid_{di}fi}(F_{sid_{fici}}(Cert^r_{BV_{ci}} \oplus ts_{CA_f}) \oplus ts_{LAG_d})$$

Thereafter, CA_d computes $Permit_{CA_{di}}$, and derives BV_i's hashed remaining power value $H(ST_{BV_{di}})$.

$$Permit_{CA_{di}} = H_{k_{si}}(E^{-1}_{k_{pi}}(Permit_{LAG_{fi}}) \oplus Cert^r_{BV_{fi}} \oplus PID_{BV_{di}})$$
$$H(ST_{BV_{di}}) = E^{-1}_{k_{si}}(Pseudo_{LAG_{di}} \oplus H(Cert^r_{BV_{di}}\|r_{LAG_i})) \oplus PID_{BV_{di}} \Rightarrow PST_{BV_{di}}$$

CA_d computes $Status_{BV/BV_i}$ by the PRNG function $R(.)$, and extracts $Percen_{BV_i}$. Here, $Status_{BV/BV_i}$ is computed by multiple BVs' aggregated pseudo power value $Status_{BV}$ except BV_i's pseudo power value $Status_{BV_i}$, and $Percen_{BV_i}$ represents the pseudo percentage of ΔPST_{BV_i} in PST_{BV_i}.

$$Status_{BV/BV_i} = R(PST_{BV_c/BV_{ci}}\|PST_{BV_f/BV_{fi}}\|PST_{BV_d/BV_{di}})$$

CA_d establishes an aggregated-status $AggST_{BV}$, and transmits $ts_{CA_d}\|Permit_{CA_{di}}\|AggST_{BV}$ to LAG_d.

$$AggST_{BV} = \{Status_{BV/BV_i}, Percen_{BV_i}\}.$$

Upon receiving the message, LAG_d computes $H_{LAG_{di}}$, and transmits $ts_{CA_d}\|ts_{LAG_d}\|H_{LAG_{di}}\|AggST_{BV}$ to BV_{di}.

$$H_{LAG_{di}} = H_{k_{pi}}(Permit_{CA_{di}} \oplus PID_{LAG_d})$$

BV_{di} re-computes $H'_{LAG_{di}}$ by its local values to verify LAG_d by comparing $H'_{LAG_{di}}$ with the received $H_{LAG_{di}}$. If it holds, BV_{di} will regard LAG_d as a legal aggregator. Otherwise, ASTP will terminate.

$$H'_{LAG_{di}} = H_{k_{pi}}(H_{k_{si}}(Permit_{BV_{ci}} \oplus PID_{BV_{di}}) \oplus PID_{LAG_d})$$

BV_{di} obtains the pseudo power value PST_{BV_i} according to $H(ST_{BV_{ci}}\|ST_{BV_{fi}}\|ST_{BV_{di}})$, and computes $Status_{BV}$ and ΔPST_{BV_i} according to the defined relationship of $\{PST_{BV}, PST_{BV/BV_i}, PST_{BV_i}\}$.

$$Status_{BV} = Status_{BV/BV_i} + R(PST_{BV_i})$$
$$\Delta PST_{BV_i} = R^{-1}(Status_{BV})Percen_{BV_i} \Rightarrow \Delta ST_{BV_i}.$$

Till now, BV_{di} has derived the real power variation ΔST_{BV_i} according to the mapping from the pseudo power variations to the real power variations.

V. Scheme Inter-Relation Analysis in BASA

The proposed three protocols (i.e., AIDP, SDAP, ASTP) are essential components in BASA scheme, and they are inter-linked with each other. Considering the essential cryptographic primitives, we show the inter-relation among the protocols.

A. State Identifier and State Transition Identifier

The state identifiers $\{sid_{ci}, sid_{fi}, sid_{di}\}$ are introduced to determine entities' state attributes, and to adaptively update the pre-shared values. In AIDP, the charging state values are obtained by $F_{sid_{ci}}$, including the certificates $\{Cert^r_{BV_{ci}}, Cert^h_{BV_{ci}}\}$, pseudonyms $\{PID_{BV_{ci}}, PID_{LAG_c}\}$, and shared values $\{s^{LAG_c}_{g_{BV_c}}, s^{g_{BV_c},CA_c}_{LAG_c}, s^{LAG_c}_{CA_c}\}$. The charging state values are used by $\{BV_{ci}, LAG_c\}$ to perform mutual authentication, and by CA_c to verify $\{BV_{ci}, LAG_c\}$. In SDAP, the FC state values are updated by $F_{sid_{fi}}$, including the hidden certificate $Cert^h_{BV_{fi}}$ and the pseudonym $PID_{BV_{fi}}$. During the challenge-response process, $\alpha_{CA_{fi}}$ is computed by $Cert^h_{BV_{fi}}$ to wrap $\alpha_{LAG_{fi}} \| \delta_{LAG_{fi}} \| \delta_{LAG_{f(i+1)}}$, and $PID_{BV_{fi}}$ is applied to compute $Pseudo_{BV_{fi}}$, which includes the hashed real power value $H(ST_{BV_{fi}})$. In ASTP, the discharging state values are updated by $F_{sid_{di}}$, including the certificates $\{Cert^r_{BV_{di}}, Cert^h_{BV_{di}}\}$, and pseudonyms $\{PID_{BV_{di}}, PID_{LAG_d}\}$. Thereinto, $Cert^r_{BV_{di}}$ is applied to wrap $Pseudo_{BV_{di}}$ into $Pseudo_{LAG_{di}}$ for transmission.

The state transition identifiers $\{sid_{fi|ci}, sid_{di|fi}\}$ are available to CA, and are respectively invoked in the FC-to-discharging, discharging-to-charging, and the charging-to-FC phases to transfer the charging/FC/discharging state values into the FC/discharging/charging state values. For instance, $\{Cert^h_{BV_{fi}}, PID_{BV_{fi}}\}$ are obtained by applying $F_{sid_{fi|ci}}$ on $\{Cert^h_{BV_{ci}}, PID_{BV_{ci}}\}$ in SDAP. In ASTP, $PID_{BV_{di}}$ is obtained by applying $F_{sid_{di|fi}}$ on $PID_{BV_{fi}}$, and $\{Cert^r_{BV_{di}}, Cert^r_{BV_{di}}\}$ are obtained by successively applying $\{F_{sid_{fi|ci}}, F_{sid_{di|fi}}\}$ on $Cert^r_{BV_{ci}}$. The state transition identifiers realize that $\{LAG, CA\}$ can apply transition functions to obtain the corresponding state values to ensure BV_i's validity.

B. In-State Permit and Cross-State Permit

The in-state permit and the cross-state permit are introduced for authentication.

1) The In-State Permit: In AIDP, $Permit_{CA_{ci}}$ is first computed by randomizing $Cert^r_{BV_{ci}}$, which indicates that LAG_c has been authenticated by CA_c, and owns the qualified permits to access BV_{ci}. Thereafter, $Permit_{BV_{ci}}$ is computed by randomizing $Cert^h_{BV_{ci}}$, to indicate that BV_{ci} has authenticated LAG_c and distributes its permit to LAG_c for later message delivery. Finally, CA_c applies $\{Permit_{BV_{ci}}, Permit_{CA_{ci}}\}$ to perform authentication on BV_{ci}.

2) The Cross-State Permit: In SDAP, $Permit_{LAG_{fi}}$ is computed based on the cross-state permit $Permit_{BV_{ci}}$, which acts as a hint foreshadowing for later authentication. In ASTP, $Permit_{LAG_{fi}}$ is further applied to obtain $Permit_{CA_{di}}$ for assigning the cross-state authority to LAG_d. Therefore, $\{Permit_{BV_{ci}}, Permit_{CA_{ci}}\}$ are reused operators to provide enhanced safeguard.

C. The Shared Secret, Timestamp, and Random Number

The main secrets, including shared keys, pairwise secrets, and certificates, are assigned to the appointed entities.

- $\{k_p, k_{si}, k_q\}$: Functions $\{E_{k_p,k_{si},k_q}, H_{k_p,k_{si}}\}$ are defined based on the shared keys. Besides, k_p is applied to achieve selective secret disclosure in SDAP, and is extended into $\{k_{pi}\}$ (i.e., $\{k_{p1}, \ldots, k_{pI}\}$), and $\{k^m_p\}$ (i.e., $\{k^1_p, \ldots, k^N_p\}$), which can attach the selective available attributes. Accordingly, k_{pi} act as a reconstructed secret key assigned to BV_i and LAG for authentication.
- $\{s^{LAG}_{g_{BV}}, s^{g_{BV},CA}_{LAG}, s^{LAG}_{CA}\}$: In AIDP, the pairwise secrets $\{s^{LAG}_{g_{BV}}, s^{g_{BV}}_{LAG}\}$ are used by LAG_c to perform preliminary verification on BV_{ci}, and $\{s^{CA}_{LAG}, s^{LAG}_{CA}\}$ are used by CA_c to verify LAG_c.
- $\{Cert^r_{BV_i}, Cert^h_{BV_i}\}$: The released certificate $Cert^r_{BV_i}$ is mainly used for mutual authentication of $\{BV_i, LAG\}$, and the hidden certificate $Cert^h_{BV_i}$ is mainly applied by CA to ascertain BV_i's identity. Thereinto, $Cert^r_{BV_{ci}}$ is extracted to compute $\{\alpha_{BV_{ci}}, Permit_{CA_{ci}}\}$, thereafter it is updated into $Cert^r_{BV_{fi}}$ to obtain $Permit_{LAG_{fi}}$, and is updated into $Cert^r_{BV_{di}}$ to obtain $Pseudo_{LAG_{di}}$. Meanwhile, $Cert^h_{BV_{ci}}$ is computed by updating $H(Cert^h_{BV_i})$. It is used to compute $Permit_{BV_{ci}}$, and is updated into $Cert^h_{BV_{fi}}$ to obtain $\{\alpha_{CA_{fi}}, Respon_{BV_{fi}}\}$.

Meanwhile, timestamps and random numbers are introduced to achieve session freshness. The quick check is performed by checking whether the received timestamp has appeared in the former session, and whether it is within an acceptable time interval. The state-aware values are updated by timestamps to realize randomization. Besides, the random numbers $\{r_{BV_i}, r_{LAG}, r_{CA}\}$ are used throughout all protocols. The main functions are to resist the replay attack, defend the impersonation attack by randomizing the transmitted messages, and perform authentication as random operators.

D. Power Value Mapping and Integration

Three types of power values are collected during the battery state transition.

- $\{ST_{BV_{ci}}, PST_{BV_{ci}}\}$: The real/pseudo power values before BV_i performs charging operation.
- $\{ST_{BV_{fi}}, PST_{BV_{fi}}\}$: The real/pseudo power values when BV_i has fully charged.
- $\{ST_{BV_{di}}, PST_{BV_{di}}\}$: The real/pseudo power values after BV_i performs discharging operation.

The real power values are transmitted in the hashed values of $\{H(ST_{BV_{ci}}), H(ST_{BV_{fi}}), H(ST_{BV_{di}})\}$. Thereafter, CA retrieves the corresponding $\{PST_{BV_{ci}}, PST_{BV_{fi}}, PST_{BV_{di}}\}$ by the mapping relationships. Upon BV_i leaving the discharging state, CA computes $Status_{BV/BV_i}$, and then extracts $Percen_{BV_i}$ that represents the corresponding pseudo percentage of ΔPST_{BV_i} to PST_{BV}. BV_i computes $Status_{BV}$ based on its locally hashed value $H(ST_{BV_{ci}} \| ST_{BV_{fi}} \| ST_{BV_{di}})$, and obtains the pseudo power variation ΔPST_{BV_i}. Then, BV_i

retrieves the real power variation ΔST_{BV_i} according to the mapping between ΔPST_{BV_i}.

VI. Security Analysis

A. Privacy Preservation

Privacy preservation guarantees that LAG cannot correlate BV_i's identity with its sensitive information (e.g., location, interest, and SOC), and the private data cannot be exposed.

1) Location Related Privacy: LAG can obtain BV_i's general group identifier gid_{BV_i} without ascertaining its detailed pseudonym PID_{BV_i}. It realizes that LAG can only determine which group BV_i belongs to, but cannot correlate the location with BV_i's identity. In the charging-to-FC phase, $AggID_{BV}$ is established by aggregating multiple vehicles' identity information, in which $Pseudo_{BV}$ is obtained by randomizing and cascading $\{Pseudo_{BV/BV_{ci}}, Permit_{BV_{ci}}\}$ $(i = \{1, .., I\})$, to hide an individual BV's identity.

2) Interest Related Privacy: In the FC-to-discharging phase, LAG cannot ascertain whether BV_i accepts or declines the discharging request by the periodically transmitted $Respon_{BV_{fi}}$. Thereinto, $Respon_{BV_{fi}}$ can be obtained by a one-way function, therefore LAG_f can neither obtain the detailed response (i.e., $Agree$ or $Decline$), nor correlate $R_{A|D}$ with BV_i's detailed identity.

3) SOC Related Privacy: CA establishes an aggregated-status $AggST_{BV}$ by elements $\{Status_{BV/BV_i}, Percen_{BV_i}\}$, and transmits $AggST_{BV}$ to BV_i via LAG. The real power values $\{ST_{BV_{ci,fi,di}}, ST_{BV_{fi}}, ST_{BV_{di}}\}$ are transmitted in the forms of $\{H(ST_{BV_{ci}}), H(ST_{BV_{fi}}), H(ST_{BV_{di}})\}$. Only BV_i itself can deduce $Status_{BV}$ by adding its locally randomized power value $R(PST_{BV_i})$.

B. Hierarchical Access Control

Hierarchical access control indicates that $\{LAG, CA\}$ have different authorities on BV_i, and such capability is achieved by identifiers and certificates.

Two types identifiers $\{gid_{BV_i}, PID_{BV_i}\}$ are assigned to BV_i, in which gid_{BV_i} is a group identifier with nonspecific attribute compared with other BVs in the same group, and PID_{BV_i} is a particular pseudo identifier (i.e., pseudonym). Thereinto, gid_{BV_i} is shared by $\{BV_i, LAG, CA\}$ with different authorities. It realizes that LAG can recognize BV_i's general group attribute without obtaining the detailed identity, and CA has full authority on gid_{BV_i} by which CA can further ascertain BV_i's specific pseudonym PID_{BV_i}.

The released/hidden certificates (i.e., $Cert^r_{BV_i}/Cert^h_{BV_i}$) are introduced to achieve hierarchical access control. Similarly, $Cert^r_{BV_i}$ is owned by $\{BV_i, LAG, CA\}$, $Cert^h_{BV_i}$ is shared by $\{BV_i, CA\}$, in which different access authorities are granted to $\{LAG, CA\}$. LAG with limited authority can only obtain $Cert^r_{BV_i}$ for preliminary authentication, CA can further obtain the corresponding hidden certificate $Cert^h_{BV_i}$ to ascertain BV_i's identity. It turns out that CA owns full authority on BV_i's certificates, by which it can derive the identity for further billing purpose.

C. Data Confidentiality and Data Integrity

Data confidentiality is achieved by anonymous aggregated-proofs (i.e., aggregated-identifier, and aggregated-status). Functions $\{E_{k_*}, F_{sid_*}\}$ are defined to ensure that only legal entities can derive the consistent values. In the charging-to-FC phase, an aggregated-identifier $AggID_{BV}$ is established by combining $\{Pseudo_{BV}, Permit_{BV_{ci}}\}$, in which pseudonyms $\{Pseudo_{BV/BV_{c1}}, \ldots, Pseudo_{BV/BV_{cI}}\}$ are cascaded into $Pseudo_{BV}$. $AggID_{BV}$ realizes that BV_i's individual identity is never revealed by aggregating multiple pseudonyms. In the FC-to-discharging phase, BV_i's insensitive attributes can be attached on $\{\varphi_i^n\}$ $(n = \{1, \ldots, N\})$ by the selective disclosure mechanism. In the discharging-to-charging phase, an aggregated-status $AggST_{BV}$ is established with $\{Status_{BV/BV_i}, Percen_{BV_i}\}$, in which $Status_{BV/BV_i}$ is obtained by randomizing $\{PST_{BV_x/BV_{xi}}\}$ for $x = \{c, f, d\}$. Only BV_i itself can reconstruct $Status_{BV}$ by the locally re-computed $H(ST_{BV_{ci}} \| ST_{BV_{fi}} \| ST_{BV_{di}})$.

Data integrity is achieved by one-way functions $\{H, H_{k_*}\}$. Thereinto, the hash functions are applied to wrap $Cert^h_{BV_{ci}}$, and to protect $\{\gamma_{BV_{fi}}, \gamma_{LAG_{fi}}\}$ for selective disclosure. Furthermore, a distributed hash table is introduced to map the hashed real power values $H(ST_{BV_{ci,fi,di}})$ to the pseudo power values $PST_{BV_{ci,fi,di}}$, and map the pseudo power variation ΔPST_{BV_i} to the real power variation ΔST_{BV_i}. The HMAC functions are applied to ensure that the response $R_{A|D}$ cannot be derived, and to guarantee that the permits $\{Permit_{CA_{ci}}, Permit_{CA_{di}}\}$ cannot be deduced.

D. Dynamic Participation

Dynamic participation refers to the freedom that BVs can join or leave the V2G networks without influencing the ongoing communications.

- For the charging state vehicles BV_{ci} $(i = \{1, \ldots, I\})$, they have established communications with LAG_c, and the newly joined vehicles BV_{cj} $(j = \{1, \ldots, J\})$ access the power grid via LAG_c. Upon receiving new queries, BV_{cj} and LAG_c follow the corresponding procedures without interfering with the ongoing sessions of BV_{ci} and LAG_c.
- For the FC state vehicles BV_{fi} $(i = \{1, \ldots, I'\})$, they are assigned full autonomy to decide whether or not to participate in the discharging operation. BV_{fi} may straightforwardly participate in discharging, and it will turn into discharging state. BV_{fi} may first decline the discharging request, then it changes its mind and wants to agree with the request. The periodically transmitted $Respon_{BV_{fi}}$ realizes that BV_{fi} can freely change its decision, while the response is never exposed to LAG_f.
- For the discharging state vehicles BV_{di} $(i = \{1, \ldots, I''\})$, in a case that BV_{di}'s power value reduces to some extent, CA will perform mandatory termination and turn BV_{dj} into the charging state. In another case that BV_{di} actively quits the discharging operation, CA will immediately terminate BV_{di}'s discharging operation.

E. Mutual Authentication

Mutual authentications are established between BV_i and LAG. In the charging-to-FC phase, LAG_c checks

TABLE III
PERFORMANCE ANALYSIS

		AIDP	SDAP	ASTP
Computation Load	BV_i	$\{6B, 6F, 2R, 4H, 4E\}$	$\{13B, 2F, R, (4+n)H, 4E\}$	$\{12B, 2F, 3R, 4H, 2E\}$
	LAG	$\{(7+I)B, 3F, 3R, H, 2E\}$	$\{9B, F, R, 2H, 4E\}$	$\{11B, 2F, R, 3H, E\}$
	CA	$\{13B, 6F, 3R, 2H, 4E\}$	$\{4B, 2F, R, (1+n)H, E\}$	$\{7B, 3F, 2R, 2E\}$
Communication Overload	$BV_i \leftrightarrow LAG$	4	$1+n$	2
	$LAG \leftrightarrow CA$	3	$1+n$	2

n: The period number of response; I: The number of the charging BVs.
B: Bitwise/Arithmetic function; F: Defined function; R: Timestamp/PRNG function; H: Hash/HMAC function; E: Encryption.

BV_{ci} by verifying whether the updated charging state values $\{s_{g_{BV_c}}^{LAG_c}, s_{LAG_c}^{g_{BV_c}}\}$ satisfying the relationship that $s_{g_{BV}}^{LAG} = s_{LAG}^{g_{BV}} \oplus gid_{BV_i}$. BV_{ci} checks LAG_c by verifying whether $H_{LAG_{ci}}$ has valid PID_{LAG_c} and $Permit_{CA_{ci}}$. In the FC-to-discharging phase, BV_{fi} and LAG_f perform mutual authentication based on $Cert_{BV_{fi}}^h$, $\{\varphi_0, \varphi_i\}$, and $\{k_{pi}, k_p^m\}$. In the discharging-to-charging phase, LAG_d checks BV_{di} based on PID_{LAG_d}. BV_{di} verifies LAG_d based on the consistency of $\{Permit_{BV_{ci}}, Permit_{LAG_{fi}}, Permit_{CA_{di}}\}$. Besides, two-round unilateral authentications are performed by CA to authenticate BV_i and LAG. During the charging-to-FC phase, CA_c verifies LAG_c by checking the authentication operators $\{s_{LAG}^{CA}, s_{CA}^{LAG}\}$, and CA_c verifies BV_{ci} by checking the consistency of $H(Cert_{BV_i}^h)$.

VII. PERFORMANCE ANALYSIS

We evaluate the performance of BASA in terms of computation load and communication overhead. Table III shows the performance analysis.

In BASA, the computation load mainly includes the bitwise/arithmetic functions (e.g., XOR, and Modulo), defined functions (e.g., F_{sid_*}), timestamp/pseudorandom number generation (PRNG) function, and hash/HMAC function (e.g., H/H_{k_*}), and symmetric encryption (e.g., E_{k_*}). The maximum amount of computations refer to the bitwise/arithmetic functions, and the number of logic gates required to implement the functions can be minimized according to hardware conditions (e.g., in field-programmable gate array (FPGA)). The Boolean algebra based functions are more lightweight compared with cryptographic algorithms (symmetric/asymmetric encryption). Concretely, in AIDP, BV_{ci} performs $\{6B, 6F, 2R, 4H, 4E\}$ operations, which generate a random number $r_{BV_{ci}}$, extract a timestamp $ts_{BV_{ci}}$, and perform 6 times updating operations by $F_{sid_{ci}}$, 4 times one-way function H/H_{k_*}, and 4 times symmetric encryptions. LAG_c performs $\{(7+I)B, 3F, 3R, H, 2E\}$ operations, in which I times XORs are computed to obtain $Pseudo_{BV}$ by combining multiple BVs' $\{Permit_{BV_{c1}}, \ldots, Permit_{BV_{cI}}\}$. CA_c performs $\{13B, 6F, 3R, 2H, 4E\}$ operations, and the highest calculations belong to the basic algorithms. The major calculations include 8 times XORs, and 6 times arithmetic functions. In SDAP, BV_{fi} performs $\{13B, 2F, R, (4+n)H, 4E\}$ operations, in which the periodically transmitted $\{Respon_{BV_{fi}}\}p$ is obtained by performing n times $H_{k_{si}}(Cert_{BV_{fi}}^h \| R_{A|D})$. LAG_f performs $\{9B, F, R, 2H, 4E\}$ operations, in which 4 times encryptions are the main computations. CA_f performs $\{4B, 2F, R, (1+n)H, E\}$ operations, and n times $H_{k_{si}}$ are performed to determine $R_{A|D}$. In ASTP, BV_{di} performs $\{12B, 2F, 3R, 4H, 2E\}$ operations, where the main calculations includes 12times basic algorithms, 2 times $F_{sid_{di}}$ based updating operation, 4 times $H_{k_{si}}/H_{k_{pi}}$ functions, and 2 times $E_{k_{si}}/E_{k_{pi}}$ functions. LAG_d and CA_d respectively perform $\{11B, 2F, R, 3H, E\}$ and $\{7B, 3F, 2R, 2E\}$ operations, in which the computation load is moderate, and can be supported by the current hardware conditions.

The communication overhead depends on the total number of exchanged messages during the protocol execution. We assume that timestamps/random numbers are 16-bit length, the large prime, identifiers, shared secrets/keys, and certificates are 64-bit length, and the hashed values are 128-bit length. In AIDP, the mutual and unilateral authentications complete via 7 steps, and the communication overhead of $\{BV_{ci}, LAG_c\}$ and $\{LAG_c, CA\}$ are respectively estimated as 104 bytes and $(70+18I)$ bytes. In SDAP, the protocol completes via $(2+2n)$ steps, in which the periodical response is transmitted to achieve a consumer's participation freedom. In a single period, LAG_f exchanges 104 bytes with BV_{fi}, and 72 bytes with CA_f. In ASTP, the protocol completes via 4 steps, and data delivery is 80 bytes between BV_{di} and LAG_d, and is 64 bytes between LAG_d and CA_d. In summary, the communication overhead is mainly caused by the aggregated-proofs, which is appropriate in pervasive environments.

VIII. CONCLUSION

In this paper, we have identified different security and privacy requirements during the battery state transitions in V2G networks, and proposed a battery status-aware authentication scheme (BASA) that involves three protocols to ensure secure communications. The proposed BASA employs an aggregated-identifier to hide each BV's identity from disclosing location related information. The scheme also introduces challenge-response to achieve dynamic response without revealing a consumer's interest related privacy. Additionally, an aggregated-status is established to enhance anonymous power data transmission for state-of-charge related data protection. Security analysis shows that BASA can achieve security protection and privacy preservation. The identified problem and our proposed solution also indicate that battery status awareness is crucial for securing BVs' operations in V2G networks.

REFERENCES

[1] Z. M. Fadlullah, M. Fouda, N. Kato, A. Takeuchi, N. Iwasaki, and Y. Nozaki, "Towards intelligent machine-to-machine communications in smart grid," *IEEE Commun. Mag.*, vol. 49, no. 4, pp. 60–65, 2011.

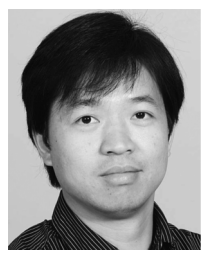
[2] Y. Zhang, R. Yu, M. Nekovee, Y. Liu, S. Xie, and S. Gjessing, "Cognitive machine-to-machine communications: Visions and potentials for the smartgrid," *IEEE Netw. Mag.*, vol. 26, no. 3, pp. 6–13, 2012.
[3] C. Guille and G. Gross, "A conceptual framework for the vehicle-to-grid (V2G) implementation," *Energy Policy*, vol. 37, no. 11, pp. 4379–4390, 2009.
[4] H. Khurana, M. Hadley, N. Lu, and D. A. Frincke, "Smart-grid security issues," *IEEE Security Privacy*, vol. 8, no. 1, pp. 81–85, 2010.
[5] J. Liu, Y. Xiao, S. Li, W. Liang, and C. Chen, "Cyber security and privacy issues in smart grids," *IEEE Commun. Surveys Tuts.* vol. 14, no. 4, pp. 981–997, 2012.
[6] A. R. Metke and R. L. Ekl, "Security technology for smart grid networks," *IEEE Trans. Smart Grid*, vol. 1, no. 1, pp. 99–107, 2010.
[7] Y. Zhang, R. Yu, W. Yao, S. Xie, Y. Xiao, and M. Guizani, "Home M2M networks: Architectures, standards, and QoS improvement," *IEEE Commun. Mag.*, vol. 49, no. 4, pp. 44–52, 2011.
[8] X. Wang and P. Yi, "Security framework for wireless communications in smart distribution grid," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 809–818, 2011.
[9] S. Sridhar, A. Hahn, and M. Govindarasu, "Cyber-physical system security for the electric power grid," *Proc. IEEE*, vol. 100, no. 1, pp. 210–224, 2012.
[10] U. Mo, T. H. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, and B. Sinopoli, "Cyber-physical security of a smart grid infrastructure," *Proc. IEEE*, vol. 100, no. 1, pp. 195–209, 2012.
[11] S. Kim, E. Lee, D. Je, and S. Seo, "A physical and logical security framework for multilevel AFCI systems in smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 3, pp. 496–506, 2011.
[12] Q. Li and G. Cao, "Multicast authentication in the smart grid with one-time signature," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 686–696, 2012.
[13] H. Son, T. Y. Kang, H. Kim, and J. H. Roh, "A secure framework for protecting customer collaboration in intelligent power grids," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 759–769, 2012.
[14] M. M. Fouda, Z. M. Fadlullah, N. Kato, R. Lu, and X. Shen, "A lightweight message authentication scheme for smart grid communications," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 675–685, 2012.
[15] S. Kim, E. Y. Kwon, M. Kim, J. H. Cheon, S. Ju, Y. Lim, and M. Choi, "A secure smart-metering protocol over power-line communication," *IEEE Trans. Power Del.*, vol. 26, no. 4, pp. 2370–2379, 2011.
[16] Y. Kim, V. Kolesnikov, H. Kim, and M. Thottan, "SSTP: A scalable and secure transport protocol for smart grid data collection," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm2011)*, pp. 161–166.
[17] M. Qiu, W. Gao, M. Chen, J. W. Niu, and L. Zhang, "Energy efficient security algorithm for power grid wide area monitoring system," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 715–723, 2012.
[18] D. Wu and C. Zhou, "Fault-tolerant and scalable key management for smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 375–381, 2011.
[19] M. Qiu, H. Su, M. Chen, Z. Ming, and L. T. Yang, "Balance of security strength and energy for a PMU monitoring system in smart grid," *IEEE Commun. Mag.*, vol. 50, no. 5, pp. 142–149, 2012.
[20] D. He, C. Chen, S. Chan, Y. Zhang, J. Bu, and M. Guizani, "Secure service provision in smart grid communications," *IEEE Commun. Mag.*, vol. 50, no. 8, pp. 53–61, 2012.
[21] Y. Zhang, L. Wang, W. Sun, R. C. Green, II, and M. Alam, "Distributed intrusion detection system in a multi-layer network architecture of smart grids," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 796–808, 2012.
[22] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *Proc. 1st IEEE Int. Conf. Smart Grid Commun. (SmartGridComm 2010)*, pp. 234–243.
[23] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen, "EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 9, pp. 1621–1631, 2012.
[24] Z. Yang, S. Yu, W. Lou, and C. Liu, "P^2: Privacy-preserving communication and precise reward architecture for V2G networks in smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 697–706, 2012.
[25] H. Guo, Y. Wu, F. Bao, H. Chen, and M. Ma, "UBAPV2U: A unique batch authentication protocol for vehicle-to-grid communications," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 707–714, 2012.
[26] B. Vaidya, D. Makrakis, and H. T. Mouftah, "Security mechanism for multi-domain vehicle-to-grid infrastructure," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM 2011)*.

Hong Liu (S'10) is currently working toward a Ph.D. degree at the School of Electronic and Information Engineering, Beihang University, China.

She focuses on the security and privacy issues in radio frequency identification, vehicle-to-grid, and wireless machine-to-machine networks. Her research interests include authentication protocol design, and security formal modeling and analysis.

Huansheng Ning (M'10–SM'13) received a B.S. degree from Anhui University, China, in 1996 and Ph.D. degree in Beihang University, China, in 2001.

He is an Associate Professor in School of Electronic and Information Engineering, Beihang University, China. His current research focuses on internet of things, aviation security, electromagnetic sensing, and computing. He has published more than 30 papers in journals and international conferences/workshops.



Yan Zhang (M'05-SM'10) received a Ph.D. degree from Nanyang Technological University, Singapore.

From August 2006, he has been working with Simula Research Laboratory, Norway. He is currently senior research scientist at Simula Research Laboratory, Norway. He is an adjunct Associate Professor at the University of Oslo, Norway. He is a regional editor, associate editor, on the editorial board, or guest editor of a number of international journals. He is currently serving the Book Series Editor for the book series "Wireless Networks and Mobile Communications" (Auerbach Publications, CRC Press, Taylor & Francis Group). He serves as organizing committee chairs for many international conferences. His research interests include resource, mobility, spectrum, energy, data, and security management in wireless communications and networking.

Mohsen Guizani (S'85–M'89–SM'99–F'09) received his B.S. (with distinction) and M.S. degrees in Electrical Engineering; M.S. and Ph.D. degrees in Computer Engineering in 1984, 1986, 1987, and 1990, respectively, from Syracuse University, Syracuse, New York.

He is currently a Professor and the Associate Vice President for Graduate Studies at Qatar University, Doha, Qatar. He was the Chair of the Computer Science Department at Western Michigan University from 2002 to 2006 and Chair of the Computer Science Department at University of West Florida from 1999 to 2002. He also served in academic positions at the University of Missouri-Kansas City, University of Colorado-Boulder, Syracuse University and Kuwait University. His research interests include computer networks, wireless communications and mobile computing, and optical networking.

Dr. Guizani currently serves on the editorial boards of six technical journals and the Founder and EIC of *Wireless Communications and Mobile Computing* (Wiley; http://www.interscience.wiley.com/jpages/1530–8669/). He is the author of eight books and more than 300 publications in refereed journals and conferences. He guest edited a number of special issues in IEEE Journals and Magazines. He also served as member, Chair, and General Chair of a number of conferences. He was the Chair of IEEE Communications Society Wireless Technical Committee (WTC) and Chair of TAOS Technical Committee. He was an IEEE Computer Society Distinguished Lecturer from 2003 to 2005. Dr. Guizani is a Senior Member of ACM.