スマートグリッドにおけるグループ署名を用いた利用者認証プロトコル

岸本 光[1,a)]　矢内 直人[1,b)]　岡村 真吾[2,c)]

概要：スマートグリッドでは電力利用に関する様々な情報が IP ネットワークを介して通信されることから，利用者の情報の正当性とプライバシー両方の保護が必要となる．本稿では将来的には第三者的に電力利用を提供するサービスが普及されると見込み，スマートグリッド上で電力利用時の匿名性を保証する利用者認証プロトコルを提案する．この核となる要素技術として，あるグループに所属するメンバのみが匿名で署名ができ，かつ，ある権限を与えられた利用者のみが同じ署名者による署名をリンク可能なグループ署名も新たに提案する．

キーワード：グループ署名，リンク制御可能なグループ署名，トークン依存型リンク制御可能グループ署名，匿名電力利用プロトコル，スマートグリッド

An Anonymous Authentication Protocol for Smart Grid

Hikaru Kishimoto[1,a)]　Naoto Yanai[1,b)]　Shingo Okamura[2,c)]

Abstract: Preserving privacy of users over Smart Grid is necessary since information about electric usage is communicated over public channel. In addition, falsifying by user or a third party becomes problem because the information is sent to electric utility via smart meter. In this paper, we propose an anonymous authentication protocol using group signatures, only members belonging to a group are able to generate signatures anonymously and only an specified entity can link signatures generated by a same signer.

Keywords: Group Signatures, Controllable-Linkability Group Signatures, Controllable-Linkability Group Signatures with Token Dependent Linking, Anonymous Protocol for Electric Use, Smart Grid

1. はじめに

近年，電気インフラの IT 化として，スマートグリッドが注目されている．スマートグリッド [23] とは，米国国立標準技術研究所（NIST: National Institute of Standards and Technology）によって定められた，送電網に接続されているすべての要素を情報インフラと統合することにより，電力会社と需要家側の双方に様々なメリットをもたらす電力網である．従来の電力網と比べて，IP 網を活用して顧客と電力会社間で相互通信を行う AMI（Advanced Meterming Infrastructure）[26] や，電力の需要に応じて効率よく電力供給を行うことができるデマンドレスポンスなどによるより高度なエネルギー管理が実現可能となる．その中でも各家庭に設置されるスマートメータは，従来の電力計測器とは異なり，各家庭における電気機器の利用状況や，一定期間における総電力利用量をリアルタイムに確認することができる他，従来の電力網にとは異なり，スマートメータから一定期間毎の電力利用量を瞬時に取得することができるため遠隔検針を行うことが可能となる．

スマートグリッドの普及に伴い，セキュリティ対策が重

[1] 大阪大学
Osaka University, 1-5 Yamadaoka, Suita, Osaka, 565–0871, Japan
[2] 奈良工業高等専門学校
National Institute of Technology, Nara College, 22 Yata-cho, Yamatokoriyama, Nara, 639–1080, Japan
[a)] hikaru.kishimoto@ist.osaka-u.ac.jp
[b)] yanai@ist.osaka-u.ac.jp
[c)] okamura@info.nara-k.ac.jp

要な課題になっている [9,27]．従来の電力網の制御システムに対して，スマートグリッドでは，システムが独自の仕様から汎用的な IT を活用する仕様に置き換わり，他のネットワークと接続された形態となる．そのため，スマートグリッドにおいても IT 網において問題となっている不正アクセスなどのリスクが高まることとなる．例えば，スマートグリッドの制御情報が改ざんされる危険性や，個人情報の流出などの危険性が考えられる．一方，スマートグリッドでは電気機器をスマートメータに直接接続することによって電力網の監視やデマンドレスポンスを実現している．しかし，電力の利用量や利用時間といった利用者の詳細な電力利用情報が電力事業者に送信されるため，それらの情報が解析されることは利用者のプライバシーの侵害につながることが指摘されている [6,19]．2009 年のオランダ議会においても利用者のプライバシ保護が不十分であるとしてスマートメータ取り付けの義務化を否決される [3] など，スマートグリッドにおけるプライバシー保護は重要な課題の一つとされている．

電力自由化後のスマートグリッドにおいては，第三者によるサービス事業が期待されている [14]．そのようなサービスの一種として，電気自動車の充電スポットのように公共の場所に設置されたスマートメータを経由して電力を利用するように，他者が管理するスマートメータを経由した電力利用が考えられる．その際，電力の供給側の電力利用量は実際に電力の供給側が使った電力利用量と外部からの利用者による電力利用量の和になるため，財政負担などを鑑みると電力を供給する側と利用者の電力利用を厳密に管理する営利的な状況も将来的には起こり得る．また，利用者がいつどこで電気を利用するかの情報が電力の供給側に漏れてしまうので，匿名で電力を利用するなど利用者のプライバシーを保護する必要がある．

本稿ではスマートグリッドにおいて利用者が自身の管理下以外のスマートメータを経由して電力を利用する場合を想定し，利用者のプライバシーを保護したまま電力を利用するプロトコルおよびそれを実現するためのグループ署名の提案する．

1.1 貢献

本稿では，スマートグリッドを想定した電力網において利用者自身の管理下以外のスマートメータを経由して電力を利用するプロトコルおよびそれを実現するためのグループ署名の提案する．

前節で述べたような利用者と供給側相互の厳密な管理をしたまま電力を利用するプロトコルを実現するためには，利用者の観点からは匿名性を保ったまま利用者認証を行うことと，供給側の観点からは同じ利用者の情報を紐付けることが必要となる．このため，本稿ではある権限を与えられた利用者のみが同じ署名者による署名をリンク可能なグループ署名 (CL-GS) を採用する．特に，提案プロトコルに用いるためのグループ署名として，スマートメータ毎に異なるトークンを利用することによって，同じトークンを持つ署名に対してリンク鍵と呼ばれる特殊な鍵を持つリンク管理者のみがリンク関係を判定できるトークン依存型リンク制御可能グループ署名を提案する．提案方式は，署名サイズが既存の CL-GS と比べて小さいという特徴もある．

提案プロトコルは，利用者が利用する複数のスマートメータの管理者が結託した場合でも同一の利用者かどうかを判定できないリンク不可能性，利用者が不正に電力事業者が正当と認める電力利用情報を生成できない否認不可能性，第三者が電力利用者が正当と認める電力利用情報を生成できない偽造不可能性を要件としており，それぞれを満たしていることを示す．また，提案プロトコルの効率評価として，計算量および処理時間の見積もりを行いスマートグリッドにおける端末における実装に対して有用であることも確認する．

本稿の構成は以下の通りである．2 章で関連研究について述べ，3 章で提案する電力利用プロトコルが満たすべき要件について述べる．4 章でトークン依存型リンク制御可能グループ署名を提案し，5 章で提案する電力利用プロトコルについて述べる．

2. 関連研究

2.1 スマートグリッドにおけるプライバシー保護

欧州ではスマートグリッドのための通信プロトコルとして，OSGP（Open Smart Grid Protocol）[7] の導入が検討されていたが，Jovanovic ら [16] による最新の研究にて，OSGP の暗号技術に脆弱性があることを示している．Chan ら [11] は，スマートグリッドにおける重要なセキュリティ技術として PKI や Trusted Computing の導入について議論している．Metke ら [21] はスマートグリッドに対して暗号および関連する管理基盤の導入が必要と結論付けており，実際に暗号技術を導入したプロトコルが数多く提案されている [1,4,8,15,18,25]．また，近年では利用者の匿名性のためにグループ署名を利用した手法も提案されている [3,10]．このうち，Diao ら [3] の手法は Qu ら [24] によって破られている．これらの手法は匿名性を制御可能という性質を持つため，通常は電力事業者に対して誰がその電力利用情報を生成したのかを明かさないが，一定条件の下で匿名性を低下させることが可能となっている．これにより，利用者のデータ同士を連携させるなどしてスマートグリッドをより有効に活用することが可能となる．

これらの研究における利用者のプライバシーはいずれも自身の管理下にあるスマートメータを経由して電力を利用した際のプライバシーの問題である．本稿では，自分の管理下以外のスマートメータを経由して電力を利用した際に，スマートメータの管理者に対してプライバシーを保護

する手法を提案する．著者らがこれまでに提案した電力料金の利用者課金プロトコル [17] では，ID 連携技術を用いて電力の匿名利用を実現している．本稿では匿名性を保ったまま利用者認証を行うことができる技術としてグループ署名を用いる点が異なる．

2.2 グループ署名

グループ署名は，あるグループに所属するメンバのみが匿名で署名をすることが可能なデジタル署名である [2]．グループを管理しているグループ管理者のみが，利用者をグループへのメンバーの追加・失効，グループ署名から署名者個人を特定することが可能である．グループ署名の中には，署名者の匿名性を損なうことなく同一の署名者によって生成されたか否かの判定を行うことが可能なリンク可能グループ署名 [22] がある．Hwang ら [12] は二つの署名のリンク関係をリンク鍵と呼ばれる秘密情報を持っているエンティティのみがチェックできる性質をもつリンク制御可能なグループ署名 (CL-GS) を提案しており，それをベースにした手法が提案されている [13,20]．また，Emura ら [5] は，トークンを用いることによって期間に応じて匿名性を制御可能なグループ署名を提案しており，別の期間で生成された署名に関しては通常のグループ署名と同レベルの Unlinkability を満たす一方で同じ期間内に生成された署名はリンク可能となる．

本稿では，利用者が自身の管理下以外の複数のスマートメータの下で電力を利用することを想定し，トークンを用いることによって同じスマートメータのもとで電力を利用した場合に限り，リンク管理者のみがリンク関係を判定できるグループ署名を提案する．

3. 要件定義

3.1 参加者

本稿では，以下の参加者を想定している．

消費者: 電力を利用する人であり $\mathcal{C}$ と表記する．

管理者: スマートメータを管理している人であり，$\mathcal{M}$ と表記する．本稿では，$\mathcal{C}$ の管理下にないスマートメータの管理者のことを指すことに注意されたい．

電力事業者: $\mathcal{C}$ や $\mathcal{M}$ と電力契約を結んでいる事業者であり，$\mathcal{U}$ と表記する．尚，特に $\mathcal{C}$ と $\mathcal{M}$ と契約している電力事業者についてはそれぞれ $\mathcal{U}_C$，$\mathcal{U}_M$ と表記する．

電力網: 各エンティティに対して電力供給するものであり，各エンティティは電力網を経由して接続されている．電力網の概要図を図 1 に示す．

3.2 安全性要件

本節では，電力利用プロトコルの安全性要件を定義する．

仮定: 電力網に接続されている $\mathcal{U}$ は信頼できる第三者機関によって承認されており，それぞれ識別子 ID_U が割り当てられているため，各電力事業者は信頼できるものとし，$\mathcal{U}_C$ と $\mathcal{U}_M$ は結託することは無いとする．

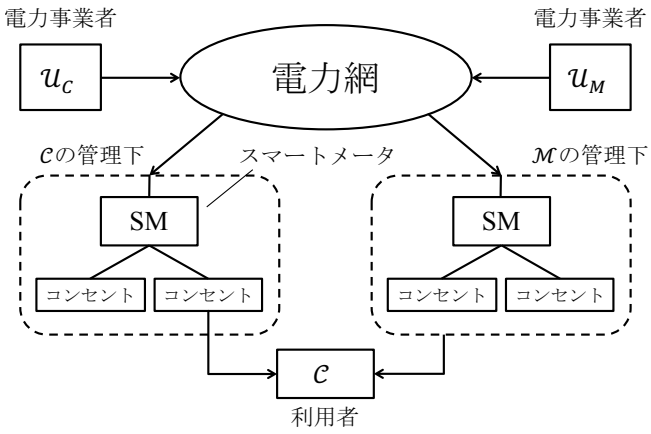

図 1 電力網の概要図

リンク不可能性: この要件の敵は，他の $\mathcal{M}$ とは独立して $\mathcal{C}$ とやり取りを行う $\tilde{\mathcal{M}}$ である．$\tilde{\mathcal{M}}$ が $\mathcal{M}$ と結託した場合でも，各種公開パラメータと $\mathcal{C}$ が $\mathcal{M}$ と $\tilde{\mathcal{M}}$ の下で電力を利用した際に生成した電力量情報が同一の $\mathcal{C}$ によって生成されたものかどうかを判別できないとき，リンク不可能性を満たすとする．

否認不可能性: この要件の敵は，$\mathcal{M}$ とやり取りを行う悪意のある $\mathcal{C}$ である．$\mathcal{C}$ が各種公開パラメータを用いて，過去に $\mathcal{M}$ の下で利用した電力利用情報とは別に $\mathcal{U}_C$ が正当な電力利用情報と認めるような電力利用情報を出力できるとき，$\mathcal{C}$ は否認可能であるという．$\mathcal{C}$ が電力利用情報を否認可能でないとき否認不可能性を満たすとする．$\mathcal{M}$ に対しても同様に各種公開パラメータを用いて，過去に $\mathcal{C}$ が $\mathcal{M}$ の下で利用した電力利用情報とは別に $\mathcal{U}_C$ が正当な電力利用情報と認めるような電力利用情報を出力できるとき，$\mathcal{M}$ は否認可能であるといい，$\mathcal{M}$ が電力利用情報を否認可能でないとき否認不可能性を満たすとする．

電力利用情報の偽造不可能性: この要件の敵は $\mathcal{C}$ を除く全てのエンティティである．攻撃者が各種公開パラメータを用いて $\mathcal{M}$ や $\mathcal{U}_C$ による検証を通過できるような偽の電力利用情報を任意の $\mathcal{C}$ に対して生成できるとき，電力利用情報を偽造可能であるとする．攻撃者が任意の電力利用情報において偽造が不可能であるとき，電力利用情報の偽造不可能性を満たすとする．

4. トークン依存型リンク制御可能グループ署名

本稿では，スマートメータ毎に異なるトークンを利用することによって，同じスマートメータの下における電力利用情報に対する署名に対してリンク管理者のみがリンク関係を判定できるグループ署名を提案する．この署名方式は Emura ら [5] による GS-TDL をベースとしている．GS-TDL のリンクアルゴリズムが誰でも実行できる点に対

して，提案方式では，リンク鍵を持っている特定のリンク管理者のみがリンク関係を判定できる．

4.1 定義

トークン依存匿名グループ署名は以下のアルゴリズム（Setup, GKeyGen, TKeyGen, Join, TokenGen, GSign, Revoke, GVerify, Link）から構成される．

Setup: セットアップアルゴリズムはセキュリティパラメータ λ を入力として，公開パラメータ $params$ を出力する．

GKeyGen: グループ鍵生成アルゴリズムは，$params$ を入力とし，グループ公開鍵 gpk，グループ秘密鍵 gsk，リンク鍵 lk，削除用ストレージ grs $:= \emptyset$，削除リスト $\mathsf{RL}_0 := \emptyset$ を出力する．

TKeyGen: トークン鍵生成アルゴリズムは，$params$ を入力とし，公開鍵 tpk，秘密鍵 tsk を出力する．

Join: 署名者追加アルゴリズムは gsk，grs，署名者の識別子 ID と $params$ を入力とし，署名鍵 $\mathsf{sigk}_{\mathsf{ID}}$ と更新された grs を出力する．

TokenGen: トークン生成アルゴリズムは tsk と T と $params$ を入力とし，トークン t_T を出力する．

GSign: 署名アルゴリズムは，gpk，tpk，t_T，$\mathsf{sigk}_{\mathsf{ID}}$，署名するメッセージ m と $params$ を入力とし，グループ署名 σ を出力する．

Revoke: 削除アルゴリズムは，gpk，grs，T に対する削除署名者の集合 $\{\mathsf{ID}_{T,1}, \mathsf{ID}_{T,2}, ..., \mathsf{ID}_{T,n}\}$ と $params$ を入力とし，T に対する削除リスト RL_T を出力する．

GVerify: 署名検証アルゴリズムは，gpk，tpk，RL_T，σ，m と $params$ を入力とし，$true$ または $false$ を出力する．尚，場合によりこのアルゴリズムはリンク鍵を持つエンティティとのインタラクティブアルゴリズムとなる．

Link: リンク判定アルゴリズムは gpk，tpk，RL_T と二つのグループ署名 $(\sigma_0, m_0), (\sigma_1, m_1))$ と lk と $params$ を入力とし，二つの署名の生成者が同じ場合は $true$ を，そうでない場合は $false$ を出力する．

Open: 開示アルゴリズムは，σ と grs と lk と $params$ を入力とし，署名 σ を生成したメンバの ID を出力する．

一般的なグループ署名には Open アルゴリズムの出力の正当性を検証する Judge アルゴリズムがあるが，Emura ら [5] は効率化のために Judge アルゴリズムを省略している．本稿では Emura ら同様，Open アルゴリズムのみを導入した定義を採用する．

4.2 安全性定義

新しい署名方式における安全性の定義について述べる．本定義では，正当性，匿名性，偽造不可能性，リンク健全性を満たすとき，安全であると定義する．

正当性: 署名者が削除されていない場合，正しく生成された署名は正当であり，かつ，署名者が同じ場合にはリンク判定アルゴリズム Link が $true$ を出力し，Open アルゴリズムを用いて署名者を特定できること．

匿名性: tsk を所持していた場合でも同じトークンを通じて生成された二つの署名が同じ署名鍵で生成されたか否かを判定できないこと．

偽造不可能性: 署名鍵を持たない攻撃者が GVerify が $true$ を出力するような署名を生成できないこと．

リンク健全性: 二つの署名が異なる署名鍵で生成された，もしくは異なるトークンに対して生成された場合に Link が $true$ を出力することがないこと．

追跡可能性: 不正が発覚した際には，Open によって，入力として与えられたグループ署名の真の生成者を一意に特定できること．

4.3 安全性仮定

$\mathcal{G}$ をセキュリティパラメータ λ を入力として，双線型群パラメータ $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$ を出力する確率的多項式時間アルゴリズムとする．ここで，p は λ-bit の素数であり，$\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ は位数 p の群，e は $\mathbb{G}_1 \times \mathbb{G}_2$ から $\mathbb{G}_T$ への写像，g_1, および g_2 はそれぞれ $\mathbb{G}_1, \mathbb{G}_2$ の生成元とする．また，$\mathbb{G}_1, \mathbb{G}_2$ は非対称なものを選択する．

本稿では，以下の安全性仮定が成り立つものとする．

SDDHI 仮定: 全ての PPT 攻撃者 $\mathcal{A}$,$(\Pr[(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_\gamma, e, g_1, g_2) \xleftarrow{\$} \mathcal{G}(1^\lambda); x \xleftarrow{\$} \mathbb{Z}_p; (\gamma, st) \leftarrow \mathcal{A}^{\mathcal{O}_x}(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_\gamma, e, g_1, g_2, g_1^x); \tau_0 = g_1^{\frac{1}{x+\gamma}}; \tau_1 \xleftarrow{\$} \mathbb{G}_1; b \xleftarrow{\$} \{0,1\}; b' \leftarrow \mathcal{A}^{\mathcal{O}_x}(y_b, st) : b = b'] - \frac{1}{2})$ が無視できるとき SDDHI 仮定が成り立つという．ここで，$\mathcal{O}_x$ は入力を $z \in \mathbb{Z}_p^* \backslash \{\gamma\}$ として，$g_1^{\frac{1}{x+z}}$ を出力するオラクルである．

q-SDH 仮定: 全ての PPT 攻撃者 $\mathcal{A}(\Pr[(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2) \xleftarrow{\$} \mathcal{G}(1^\lambda); \gamma \xleftarrow{\$} \mathbb{Z}_p; (x, g_1^{\frac{1}{x+\gamma}}) \leftarrow \mathcal{A}(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_1^\gamma, ..., g_1^{\gamma^q}, g_2, g_2^\gamma); x \in \mathbb{Z}_p^* \backslash \{-\gamma\})$ が無視できるとき q-SDH が成り立つという．

デジタル署名: デジタル署名のアルゴリズムとして (Gen, Sign, Verify) を定義する．Gen は，鍵生成アルゴリズムであり，セキュリティパラメータ λ を入力として，署名生成および検証に用いる鍵ペア (vk, sigk) を出力する．Sign は署名生成アルゴリズムであり，署名鍵 sigk と署名するメッセージ m を入力として，署名 Σ を出力する．Verify は，署名検証アルゴリズムであり，署名検証鍵 vk，署名 Σ，メッセージ m を入力として，$true$ もしくは $false$ を出力する．ここで，全ての $(\mathsf{vk}, \mathsf{sigk})$ と m について，$\Pr[\mathsf{Verify}(\mathsf{vk}, \mathsf{Sign}(\mathsf{sigk}, m), m) = 1] = 1$ が成り立つ必要がある．また，攻撃者が任意の m_i に対する署名 Σ_i を自由に入手できる状況において，任意の平文 m' に対して $\mathsf{Verify}(\mathsf{vk}, \Sigma, m) = true$ となるような署名 Σ' を出力できないとき，デジタル署名 $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ は適応的選択平文攻撃に対して存在的偽造不可能であるという．

4.4 提案方式

提案方式では，署名の生成時に位置情報（本稿の場合スマートメータの識別子）を表すトークンを用いており，同じトークンを用いて同じ署名者によって生成された署名の場合のみ，リンク鍵を持つリンク管理者によってリンク関係を判定することが可能である．そのために，GS-TDL に対して新たにリンク鍵を追加し，リンク鍵を所有するエンティティをリンク管理者として定義する．

Setup(1^λ): $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_3)$ を素数位数 p を持つ双線型群 ($\langle g \rangle = \mathbb{G}_1$ かつ $\langle g \rangle = \mathbb{G}_2$)，$e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ を双線型写像とし，(Gen, Sign, Verify) を偽造不可能性を満たすディジタル署名のアルゴリズムとする．$params = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2, (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify}))$ を出力する．

GKeyGen$(params)$: $\gamma, l \xleftarrow{\$} \mathbb{Z}_p$ および $h, u \xleftarrow{\$} \mathbb{G}_1, k \xleftarrow{\$} \mathbb{G}_2$ を選び，$W = g_2^\gamma, f \leftarrow u^l$ を計算し，$\mathsf{gpk} = (params, h, u, W, f, e(g_1, g_2), e(h, W), e(h, g_2), H), \mathsf{gsk} = \gamma$，リンク鍵 $\mathsf{lk} = k^l$，grs，RL_0 を出力する．ここで，$H : \{0,1\}^* \to \mathbb{Z}_p$ はハッシュ関数，$\mathsf{grs} := \emptyset$ および $\mathsf{RL}_0 := \emptyset$ と定義する．

TKeyGen$(params)$: Gen アルゴリズムによってトークン鍵 $(\mathsf{tpk}, \mathsf{tsk}) \leftarrow \mathsf{Gen}(1^\lambda)$ を生成する．

Join$(\mathsf{gsk}, \mathsf{grs}, \mathsf{ID}, params)$: $x, y \xleftarrow{\$} \mathbb{Z}_p$ を選び，$A = (g_1 h^{-y})^{\frac{1}{\gamma+x}}$ を計算し，$\mathsf{sigk}_{\mathsf{ID}} = (x, y, A)$ を出力し，$\mathsf{grs} := \mathsf{grs} \cup \{\mathsf{ID}, x\}$ と更新する．

TokenGen$(\mathsf{tsk}, T, params)$: $T \in \mathbb{Z}p$ とする．$W_T = g_2^T$ と $\Sigma \leftarrow \mathsf{Sign}(\mathsf{tsk}, W_T)$ を計算し，$t_T = (T, W_T, \Sigma)$ を出力する．

GSign$(\mathsf{gpk}, \mathsf{tpk}, t_T, \mathsf{sigk}_{\mathsf{ID}}, m, params)$: $\mathsf{Verify}(tpk, W_T, \Sigma)$ が $false$ である場合，$false$ を出力する．それ以外の場合，$\alpha, \beta \xleftarrow{\$} \mathbb{Z}_p$ を選び，$\delta = \beta x - y$ とし，$C = Ah^\beta$ と $\tau = g_1^{\frac{1}{x+T}} f^{\frac{\alpha}{x+T}}, \tau' = u^{\frac{\alpha}{x+T}}$ を計算する．また，$r_x, r_\delta, r_\beta \xleftarrow{\$} \mathbb{Z}_p$ を選び，

$$R_1 = \frac{e(h, g_2)^{r_\delta} e(h, W)^{r_\beta}}{e(C, g_2)^{r_x}}, R_2 = e(\tau, g_2)^{r_x}$$
$$c = H(\mathsf{gpk}, \mathsf{tpk}, C, \tau, \tau', R_1, R_2, m)$$
$$s_x = r_x + cx, s_\delta = r_\delta + c\delta, s_\beta = r_\beta + c\beta$$

を計算し，$\sigma = (C, \tau, \tau', c, s_x, s_\delta, s_\beta)$ を出力する．

ここで，$e(h, g_2)$ と $e(h, W)$ は事前計算可能であり，かつ gpk に含まれると仮定できる．また，$e(C, g_2)^{r_x} = e(A, g_2)^{r_x} e(h, g_2)^{\beta r_x}$ ことができるため，$e(A, g_2)$ を事前計算し，$e(A, g_2)$ に含んでおくと仮定すると，ペアリング計算を削減することが可能である．

Revoke$(\mathsf{gpk}, \mathsf{grs}, \{\mathsf{ID}_{T,1}, \mathsf{ID}_{T,2}, ..., \mathsf{ID}_{T,n}\}, params)$: Join で署名鍵を発行していない $\mathsf{ID} \in \{\mathsf{ID}_{T,1}, ..., \mathsf{ID}_{T,n}\}$ の場合，$false$ を出力する．それ以外の場合，$(\mathsf{ID}_{T,1}, x_{T,1}), ..., (\mathsf{ID}_{T,n}, x_{T,n})$ を grs から取得し，$\mathsf{RL}_T := \{(\mathsf{ID}_{T,1}, e(g_1^{\frac{1}{x_{T,1}+T}}, k)), ..., (\mathsf{ID}_{T,n_T}, e(g_1^{\frac{1}{x_{T,n_T}+T}}, k))\}$ を出力する．

GVerify$(\mathsf{gpk}, \mathsf{tpk}, \mathsf{RL}_T, \sigma, m, params)$: $\mathsf{Verify}(tpk, W_T, \Sigma)$ が $false$ のとき，$false$ を出力する．それ以外の場合，$\sigma = (C, \tau, \tau', c, s_x, s_\delta, s_\beta)$ に対して，(τ, τ') が RL_T に含まれているかどうかをグループ管理者に問い合わせる [*1]．(τ, τ') が RL_T に含まれている場合は $false$ を出力する．それ以外の場合，

$$R_1' = \frac{e(h, g_2)^{s_\delta} e(h, W)^{s_\beta}}{e(C, g_2)^{s_x}}$$
$$R_2' = e(\tau, g_2)^{s_x} \frac{e(g_1, g_2) e(f^\alpha, g_2)}{e(\tau, W_T)}^{-c}$$

を計算し，$c = H(\mathsf{gpk}, \mathsf{tpk}, C, \tau, \tau', R_1', R_2', m)$ が成り立つ場合 $true$ を，そうでない場合は $false$ を出力する．

Link$(\mathsf{gpk}, \mathsf{tpk}, \mathsf{RL}_T, (\sigma_0, m_0), (\sigma_1, m_1)), \mathsf{lk}, params)$: $\mathsf{GVerify}(\mathsf{gpk}, \mathsf{tpk}, \mathsf{RL}_T, \sigma_0, m_0, params) = false$ または $\mathsf{GVerify}(\mathsf{gpk}, \mathsf{tpk}, \mathsf{RL}_T, \sigma_1, m_1, params) = false$ の場合，$false$ を出力する．二つの署名 (σ_0, σ_1) とリンク鍵 lk を用いて $e(\tau_0/\tau_1, k) \stackrel{?}{=} e(\tau_0'/\tau_1', k^l)$ が成り立つ場合，$true$ を出力し，そうでない場合は $false$ を出力する．

Open$(\sigma, \mathsf{grs}, \mathsf{lk}, params)$: 署名 σ とリンク鍵 lk を用いて，grs に格納されている秘密鍵 x から $(\frac{e(\tau,k)}{e(\tau',k^l)})^{x+T}$ を計算し，$e(g_1, k)$ と一致するような x に対応する ID を出力する．

4.5 安全性解析

提案方式が安全性定義を満たしているか議論する．

正当性: GVerify，Link，Open における計算過程を (1) 式，(2) 式，(3) 式にそれぞれ示す．これら一連の式において計算が成り立つため正当性が成り立っているといえる．

$$\begin{aligned} R_2' &= e(\tau, g_2)^{s_x} \cdot (\frac{e(g_1, g_2) e(f^\alpha, g_2)}{e(\tau, W_T)})^{-c} \\ &= e(\tau, g_2)^{r_x + cx} \cdot e(g_1, g_2)^{-c} e(f^\alpha, g_2)^{-c} e(g_1 f^\alpha, g_2)^{\frac{cT}{x+T}} \\ &= e(\tau, g_2)^{r_x + cx} \cdot e(g_1, g_2)^{\frac{-cx-cT+cT}{x+T}} e(f^\alpha, g_2)^{\frac{-cx-cT+cT}{x+T}} \\ &= e(\tau, g_2)^{r_x + cx} \cdot e(g_1^{\frac{1}{x+T}}, g_2)^{-cx} e(f^{\frac{\alpha}{x+T}}, g_2)^{-cx} \\ &= e(\tau, g_2)^{r_x + cx} \cdot e(\tau, g_2)^{-cx} = e(\tau, g_2)^{r_x} = R_2 \end{aligned} \quad (1)$$

$$\begin{aligned} e(\frac{\tau_0}{\tau_1}, k) &\stackrel{?}{=} e(\frac{\tau_0'}{\tau_1'}, k^l) \\ e(\frac{g_1^{\frac{1}{x_0+T_0}} f^{\frac{\alpha_0}{x_0+T_0}}}{g_1^{\frac{1}{x_1+T_1}} f^{\frac{\alpha_1}{x_1+T_1}}}, k) &\stackrel{?}{=} e(\frac{u^{\frac{\alpha_0}{x_0+T_0}}}{u^{\frac{\alpha_1}{x_1+T_1}}}, k^l) \\ e(\frac{f^{\frac{\alpha_0}{x_0+T_0}}}{f^{\frac{\alpha_1}{x_1+T_1}}}, k) &\stackrel{?}{=} e(\frac{u^{\frac{\alpha_0}{x_0+T_0}}}{u^{\frac{\alpha_1}{x_1+T_1}}}, k)^l \\ e(\frac{u^{\frac{\alpha_0}{x_0+T_0}}}{u^{\frac{\alpha_1}{x_1+T_1}}}, k)^l &\stackrel{?}{=} e(\frac{u^{\frac{\alpha_0}{x_0+T_0}}}{u^{\frac{\alpha_1}{x_1+T_1}}}, k)^l \end{aligned} \quad (2)$$

[*1] RL_T に含まれているかの確認を不要とするなら，GVerify の検証者はグループ管理者との対話なしに計算可能である．

$$
\begin{aligned}
(\frac{e(\tau,k)}{e(\tau',k^l)})^{x+T} &= (\frac{e(g_1^{\frac{1}{x+T}} f^{\frac{\alpha}{x+T}},k)}{e(u^{\frac{\alpha}{x+T}},k)^l})^{x+T} \\
&= (\frac{e(g_1^{\frac{1}{x+T}},k)e(u^{\frac{\alpha}{x+T}},k)^l}{e(u^{\frac{\alpha}{x+T}},k)^l})^{x+T} \\
&= e(g_1^{\frac{1}{x+T}},k)^{x+T} = e(g_1,k) \qquad (3)
\end{aligned}
$$

匿名性: 任意の 2 人の署名者において，片方の署名者が真の DH タプル，もう片方の署名者が乱数に見える．これは SDDHI 仮定の分布と同じである．もし敵が匿名性を敗ることが可能な場合，任意の署名を識別できるため，その敵の能力を用いることで SDDHI 仮定を破ることができる．

偽造不可能性: GSign で出力する σ に含まれる τ および τ' について，$\tilde{g_1} = g_1 f^{\alpha}$ とすると，$\tau = g_1^{\frac{1}{x+T}} f^{\frac{\alpha}{x+T}} = \tilde{g_1}^{\frac{1}{x+T}}$ とみなすことができ，$\tilde{u} = u^{\alpha}$ とすると，$\tau' = u^{\frac{\alpha}{x+T}} = \tilde{u}^{\frac{1}{x+T}}$ とみなすことができる．これは任意の T に対する $g_1^{\frac{1}{x+T}}$ を計算する q-SDH 仮定とみなせる．これにより署名の偽造不可能性は q-SDH 仮定に帰着される．

リンク健全性: Link に入力として与えられた二つの署名において $x_0 + T_0 = x_1 + T_1$ であるか，$\alpha_0 = \frac{x_0+T_0}{x_1+T_1}$ または $\alpha_1 = \frac{x_1+T_1}{x_0+T_0}$ であるとき，(2) 式が成り立つ．後者二つは群位数 p に対して $\frac{1}{p}$ の確率でしか起こらないため，無視できる確率を除いて，同じ署名者以外の署名を受け入れることはない．

追跡可能性: Open アルゴリズムに対して正当な署名を入力として与えた場合，(3) 式における $e(g_1,k)$ は公開パラメータから一意に定まるため，署名の生成者が一意に定まる．

5. 電力利用のための認証プロトコルへの応用

提案グループ署名を応用して，スマートグリッドにおける電力利用のための匿名認証プロトコルを提案する．ここではある利用者が自身の管理下以外のスマートメータを経由して電力を利用する場合において，匿名で利用者を認証する．このプロトコルを利用することによって，公共の電源から電源を利用する際に，利用者のプライバシーを保護したままで利用者課金を実現することが可能となる．提案プロトコルにおいて，$\mathcal{U}_C$ をグループ管理者，$\mathcal{C}$ をそのグループのメンバとしてグループ署名を適用する．

5.1 概要

本プロトコルは，準備処理，署名処理，検証処理の 3 つの処理から成り立っている．準備処理は，電力事業者と利用者の間で事前に実行される処理であり，プロトコル内で利用する鍵などの生成を行う．署名処理は，$\mathcal{M}$ の下で $\mathcal{C}$ が電力を利用した際の終了時に実行される処理であり，$\mathcal{C}$ が $\mathcal{M}$ が電力を利用する度に実行される．この処理では $\mathcal{M}$ によって計測された $\mathcal{C}$ の電力利用量に対して $\mathcal{C}$ および $\mathcal{M}$ が署名を生成し，$\mathcal{U}_C$ に渡す．検証処理は，$\mathcal{C}$ が電力利用の決済を行う処理であり，月に一回などの定期的な間隔で実行される．この処理では，$\mathcal{U}_C$ が $\mathcal{C}$ や $\mathcal{M}$ により生成された署名の検証を行う．

5.2 仮定

提案プロトコルにおける仮定を以下に述べる．グループ管理者となる $\mathcal{U}_C$ はグループ公開鍵 gpk およびグループ秘密鍵 gsk とリンク鍵 lk を持っている．同様に $\mathcal{M}$ は $\mathcal{U}_M$ によって識別子を与えら，グループ秘密鍵 tsk とそれに対するグループ公開鍵 tpk を所有している．$\mathcal{M}$ は自身の管理しているスマートメータの下 $\mathcal{C}$ が電力利用を行った場合，電力量の計測を行う．各エンティティが所有している鍵は PKI によって正当性を保証されているものとし，誰でもその証明書付き公開鍵を入手できる．また，$\mathcal{U}_C$ は，決済情報としてスマートメータの識別子 T，$\mathcal{C}$ および $\mathcal{M}$ によって生成された署名，電力利用情報 m とそのハッシュ値 h を一つの組 $(T, \sigma_C, \sigma_M, m, h)$ として管理するデータベースを管理しており，$\mathcal{C}$ の毎回の電力利用毎に，データベースの更新を行う．尚，Setup や GKeyGen は事前に実行しているものとする．

5.3 構成

5.3.1 準備処理

準備処理の流れを以下に示す．

(1) $\mathcal{C}$ は $\mathcal{U}_C$ に対して登録リクエストを送信する．

(2) $\mathcal{U}_C$ は，利用者 $\mathcal{C}$ の識別子 ID を生成し，$\mathsf{Join}(\mathsf{gsk}, \mathsf{grs}, \mathsf{ID}, params)$ を実行することによって $\mathsf{sigk}_{\mathsf{ID}} = (x, y, A)$ を出力し，grs の更新を行う ($\mathsf{grs} := \mathsf{grs} \cup \{(\mathsf{ID}, x)\}$)．

(3) $\mathcal{U}_C$ は，$(\mathsf{sigk}_{\mathsf{ID}}, \mathsf{ID})$ を $\mathcal{C}$ に送信する．

5.3.2 署名処理

署名処理の流れを以下に示す．

(1) $\mathcal{M}$ は (q, N, t_T) を生成する．ここで q は電力利用情報における電力利用量であり，N は電力利用情報を一意なものにするために割り当てられる固有番号，t_T は $\mathcal{M}$ が管理しているスマートメータの識別子 T を入力として $\mathsf{TokenGen}(\mathsf{tsk}, T, params)$ を用いて生成されたトークンである．$\mathcal{M}$ は，$\mathsf{Sign}(\mathsf{tsk}, (q, N, t_T))$ を用いて (q, N, t_T) に対する署名 Σ を生成する．$\mathcal{M}$ はメッセージ $m = ((q, N, t_T), \Sigma)$ を $\mathcal{C}$ に送る．

(2) $\mathcal{C}$ は m における電力利用量 q の署名 $\mathsf{Verify}(\mathsf{tpk}, \Sigma, m)$ を用いて検証する．検証が通れば，$\mathcal{C}$ は署名 $\sigma_C = \mathsf{GSign}(\mathsf{gpk}, \mathsf{tpk}, t_T, \mathsf{sigk}_{\mathsf{ID}}, m, params)$ を生成する．

(3) $\mathcal{C}$ は，(σ_C, m) を $\mathcal{U}_C$ に送る．

(4) $\mathcal{U}_C$ は，データベースから $\mathsf{Link}(\mathsf{gpk}, \mathsf{tpk}, \mathsf{RL}_T, (\sigma_C, m), (\tilde{\sigma_C}, \tilde{m}), \mathsf{lk}, params) = true$ である最も新しい $\tilde{m}$ に対応する $\tilde{h}$ の値を h として $\mathcal{C}$ に送る．ここで，h は $\mathcal{C}$ が過去に $\mathcal{M}$ の元において電力を利用した際の電力利用情報のハッシュ値で

ある．

(5) $\mathcal{C}$ は，(σ_C, m, h) を $\mathcal{M}$ に送る．

(6) $\mathcal{M}$ は，$\mathsf{GVerify}(\mathsf{gpk}, \mathsf{tpk}, \mathsf{RL}_T, m, \sigma_C)$ によって署名 σ_C の検証を行う．検証結果が *false* である場合，$\mathcal{M}$ は (1) からやり直す．検証結果が *true* であれば，$\mathcal{M}$ は，$m' = m \parallel h$ に対する署名 $\sigma_M = Sign(m', \mathsf{tsk})$ を生成し，(σ_M, m') を $\mathcal{U}_C$ に送る．

(7) $\mathcal{U}_C$ は，m' のハッシュ値 $h' = H(m')$ を計算し，$(m, h) = (m', h')$ として $(T, \sigma_C, \sigma_M, m, h)$ の組をデータベースに追加する．

5.3.3 検証処理

検証処理の流れを以下に示す．

(1) $\mathcal{U}_C$ は Link アルゴリズムを用いて，データベースからある T における署名 σ_C を同じ署名者毎に抽出する．

(2) $\mathcal{U}_C$ は，抽出したグループにおいて $h'_1 = H(m_1)$ を計算し，$2 \leq i \leq n$ について $h'_i = H(m_i \parallel h'_{i-1})$ を計算する．また，このとき $\mathcal{U}_C$ は m_i における固有番号 N_i が他のメッセージ m_i の N_i と重複していないかどうかをチェックする．

(3) 計算した h'_n が抽出後のデータベースにおける h_n と一致している場合，$\mathcal{U}_C$ は $\mathsf{GVerify}(\mathsf{gpk}, \mathsf{tpk}, \mathsf{RL}_T, m_n, \sigma_{C,n})$ および $\mathsf{Verify}(\mathsf{tpk}, \sigma_{M,n})$ を用いて署名 $\sigma_{C,n}$ および $\sigma_{M,n}$ の検証を行う．

(4) 検証結果が双方とも *true* である場合，$\mathcal{U}_C$ は Open を用いて署名 $\sigma_{C,n}$ の生成者を特定し，q_i を電力利用量として通知する．$h'_i \neq h_i$ または検証結果が *false* である場合は，$\mathcal{U}_C$ は全ての署名を検証および開示することによって不正の特定を行う．

5.4 評価

5.4.1 安全性評価

提案プロトコルは安全性要件を次のように満たす．

リンク不可能性: グループ署名の匿名性から，任意のエンティティは，トークン生成鍵 tsk を持っている場合でもリンク鍵 lk を持っていない限り，二つの署名が同じ署名鍵で生成されたかどうかを判別することができない．提案プロトコルにおいて，リンク鍵を所有しているのはグループ管理者である $\mathcal{U}_C$ のみであり，仮定より $\mathcal{M}$ との結託は認めていない．また，$\mathcal{M}$ 毎にトークン T は一意に割り当てられており，グループ署名のリンク健全性から Link はトークン T が異なる限り *false* を出力することから，$\mathcal{M}$ は他の $\mathcal{M}$ と結託した場合であっても $\mathcal{C}$ の電力利用情報をリンクすることができない．以上より，提案プロトコルはリンク不可能性を満たしている．

否認不可能性: 各グループ署名には $\mathcal{C}$ の秘密鍵 $\mathsf{sigk}_{\mathsf{ID}}$ が含まれているため，$\mathcal{U}_C$ が Open を用いて署名を開示することで，署名の追跡可能性から真の署名の生成者を一意に特定できる．これにより，$\mathcal{C}$ の否認を防ぐことが可能である．また，$\mathcal{M}$ に対する否認不可能性は，5.3.2 節の (2) において $\mathcal{M}$ の署名に対して $\mathcal{C}$ が署名していることから，問題が発生した際は $\mathcal{C}$ は $\mathcal{M}$ の署名を出力することで否認を防ぐことができる．以上より，提案プロトコルは否認不可能性を満たしている．

偽造不可能性: $\mathcal{C}$ が電力利用情報に署名していること，また，グループ署名の偽造不可能性から帰着可能である．

表 1 計算量の比較

処理内容	Hajy ら [10]	提案方式
署名生成	$7\mathcal{P} + 16\mathcal{E} + 7\mathcal{B} + \mathcal{H}$	$\mathcal{P} + 4\mathcal{E} + 4\mathcal{B} + \mathcal{H}$
署名検証	$7\mathcal{P} + 16\mathcal{E} + 4\mathcal{B} + \mathcal{H}$	$6\mathcal{P} + 5\mathcal{B} + \mathcal{H}$
署名のリンク	$3\mathcal{P}$	$2\mathcal{P}$
署名者の開示	$10\mathcal{E} + \mathcal{H}$	$2\mathcal{P} + \mathcal{E}$

表 2 TEPLA における処理時間の見積もり（単位：msec）

処理内容	Hajy ら [10]	提案方式
署名生成	102.43	35.82
署名検証	82.42	71.75
署名のリンク	19.19	12.80
署名者の開示	7.14	13.48

5.4.2 効率評価

本節では，提案方式の効率評価として暗号方式の観点からグループ署名の計算量の評価を行う．比較対象として Hajy ら [10] による CL-GS をベースとした手法を用いる．ここではセキュリティレベルとして 128bit を想定しており，ペアリング計算，スカラー倍算，ペアリング関数のべき乗，ハッシュ関数の回数をそれぞれ $\mathcal{P}, \mathcal{E}, \mathcal{B}, \mathcal{H}$ とする．計算量の評価対象として，署名サイズおよび署名生成，署名検証，署名のリンク，署名者の開示を対象とする．署名サイズは，CL-GS が 512byte であるのに対して提案方式では 320byte であった．計算量の比較を表 1 に示す．尚，ここでは gpk 内で事前に計算されているペアリング計算の結果を用いることによって，署名生成において三回と署名検証において一回のペアリング計算を省略している．

これらの計算量の見積もりを用いて各手法における処理時間の見積もりを行った結果を表 2 に示す．尚，処理時間についてはペアリング関数ライブラリ TEPLA[*2] のベンチマークを参考にしており事前計算は処理時間に含んでいない．また，ハッシュ関数の処理時間については高速に処理できるためここでは無視している．表 2 より，署名生成，署名検証，署名のリンクについて Hajy らの手法よりも高速に処理できているということが分かる．また，署名サイズも小さいためスマートグリッド上で用いられる計算能力が制限されたスマートメータのような端末上における実装に向いていることがいえる．

[*2] http://www.cipher.risk.tsukuba.ac.jp/tepla/

6. まとめ

本稿ではスマートグリッドを想定した電力網において利用者が自身の管理下以外のスマートメータを経由して電力を利用する場合を想定し，利用者のプライバシーを保護したまま電力を利用するプロトコルおよびそれを実現するためのトークン依存型リンク制御可能グループ署名の提案を行った．今後の予定として，実際のスマートメータのプラットフォームや端末上での実装および評価を検討している．また，4 章の提案方式に関するより厳密な安全性証明も今後の課題である．加えて，Hajy ら [10] は，CL-GS における検証処理のバッチ検証を提案しており，本稿におけるグループ署名についてもバッチ検証を検討する．

7. 謝辞

本研究の一部は，JSPS 科研費 16K16065 の助成を受けている．また，大阪大学の藤原融教授には有益なアドバイスを頂きましたことを感謝致します．

参考文献

[1] Bobba, R., Khurana, H., AlTurki, M. and Ashraf, F.: PBES: A Policy Based Encryption System with Application to Data Sharing in the Power Grid, *Proc. of ASIACCS 2009*, ACM, pp. 262–275 (2009).

[2] Chaum, D. and Van Heyst, E.: Group Signatures, *Proc. of EUROCRYPT 1991*, LNCS, Vol. 547, Springer-Verlag, pp. 257–265 (1991).

[3] Diao, F., Zhang, F. and Cheng, X.: A Privacy-Preserving Smart Metering Scheme Using Linkable Anonymous Credential, *IEEE Transactions on Smart Grid*, Vol. 6, No. 1, pp. 461–467 (2015).

[4] Dimitriou, T. and Karame, G.: Privacy-Friendly Planning of Energy Distribution in Smart Grids, *Proc. of SEGS 2014*, ACM, pp. 1–6 (2014).

[5] Emura, K. and Hayashi, T.: A Light-Weight Group Signature Scheme with Time-Token Dependent Linking, *Proc. of LightSec 2015*, LNCS, Vol. 9542, Springer-Verlag, pp. 37–57 (2016).

[6] Erkin, Z. and Veugen, T.: Privacy Enhanced Personal Services for Smart Grids, *Proc. of SEGS 2014*, ACM, pp. 7–12 (2014).

[7] ETSI: Open Smart Grid Protocol (OSGP), Refference DGS/OSG-001, European Telecommunications Standards Institute, Sophia Antipolis Cedex (2012).

[8] Garcia, F. D. and Jacobs, B.: Privacy-friendly energy-metering via homomorphic encryption, *Proc. of STM 2010*, LNCS, Vol. 6710, Springer-Verlag, pp. 226–238 (2011).

[9] Hahn, A. and Govindarasu, M.: Cyber Attack Exposure Evaluation Framework for the Smart Grid, *IEEE Transactions on Smart Grid*, Vol. 2, No. 4, pp. 835–843 (2011).

[10] Hajy, S., Zargar, M. and Yaghmaee, M. H.: Privacy Preserving via Group Signature in Smart Grid, *Proc. of EIAC 2013* (2013).

[11] He, D., Chan, S., Zhang, Y., Guizani, M., Chen, C. and Bu, J.: An enhanced public key infrastructure to secure smart grid wireless communication networks, *IEEE Network*, Vol. 28, No. 1, pp. 10–16 (2014).

[12] Hwang, J. Y., Lee, S., Chung, B. H., Cho, H. S. and Nyang, D.: Short Group Signatures with Controllable Linkability, *Proc. of LightSec 2011*, IEEE, pp. 44–52 (2011).

[13] Hwang, J. Y., Lee, S., Chung, B.-H., Cho, H. S. and Nyang, D.: Group signatures with controllable linkability for dynamic membership, *Information Sciences*, Vol. 222, pp. 761 – 778 (2013).

[14] Ito, S., Shimada, T. and Kanada, M.: Cybersecurity Technologies for Smart Grids, Technical Report 11, TOSHIBA CORPORATION (2011).

[15] Jawurek, M., Johns, M. and Kerschbaum, F.: Plug-in Privacy for Smart Metering Billing, *Proc. of PETS 2011*, LNCS, Vol. 6794, Springer-Verlag, pp. 192–210 (2011).

[16] Jovanovic, P. and Neves, S.: *Practical Cryptanalysis of the Open Smart Grid Protocol*, LNCS, Vol. 9054, pp. 297–316, Springer-Verlag (2015).

[17] Kishimoto, H. and Okamura, S.: Secure consolidation of charging information over Smart Grid using ID federation, *Proc. of ISITA 2014*, IEEE, pp. 226–230 (2014).

[18] Kursawe, K., Danezis, G. and Kohlweiss, M.: Privacy-Friendly Aggregation for the Smart-Grid, *Proc. of PETS 2011*, LNCS, Vol. 6794, Springer-Verlag, pp. 175–191 (2011).

[19] Lam, H. Y., Fung, G. S. K. and Lee, W. K.: A novel method to construct taxonomy electrical appliances based on load signatures, *IEEE Transactions on Consumer Electronics*, Vol. 53, No. 2, pp. 653–660 (2007).

[20] Mamun, M. S. I. and Miyaji, A.: Secure VANET applications with a refined group signature, *Proc. of PST 2014*, IEEE, pp. 199–206 (2014).

[21] Metke, A. R. and Ekl, R. L.: Security Technology for Smart Grid Networks, *IEEE Transactions on Smart Grid*, Vol. 1, No. 1, pp. 99–107 (2010).

[22] Nakanishi, T., Fujiwara, T. and Watanabe, H.: A Linkable Group Signature and Its Application to Secret Voting, *Transactions of Information Processing Society of Japan*, Vol. 40, No. 7, pp. 3085–3096 (1999).

[23] National Institute of Standards and Technology: *NIST framework and roadmap for smart grid interoperability standards, release 1.0* (2010).

[24] Qu, H., Shang, P., Lin, X.-J. and Sun, L.: Cryptanalysis of A Privacy-Preserving Smart Metering Scheme Using Linkable Anonymous Credential, Cryptology ePrint Archive, Report 2015/1066 (2015).

[25] Rial, A. and Danezis, G.: Privacy-preserving Smart Metering, *Proc. of PETS 2011*, ACM, pp. 49–60 (2011).

[26] Sui, H., Wang, H., Lu, M. S. and Lee, W. J.: An AMI System for the Deregulated Electricity Markets, *Proc. of IAS 2008*, IEEE, pp. 1–5 (2008).

[27] Wang, W. and Lu, Z.: Cyber security in the Smart Grid: Survey and challenges, *Computer Networks*, Vol. 57, No. 5, pp. 1344 – 1371 (2013).