越境データのプライバシ問題に配慮した iKaaS アーキテクチャの提案

披田野清良 †　　清本晋作 †　　村上陽亮 ‡

†KDDI 研究所
356-8502 埼玉県ふじみ野市大原 2-1-15
{se-hidano, kiyomoto}@kddilabs.jp

‡KDDI 総研
102-8460 東京都千代田区飯田橋 3-10-10
yk-murakami@kddi.com

あらまし 複数のローカルクラウドが有機的に結合されたグローバルクラウド上において，IoT デバイスから収集されたデータをプライバシに配慮して利活用することを目的とした iKaaS プラットフォームのセキュリティアーキテクチャを提案する．本稿では，特に，越境データのプライバシ問題に着目しており，グローバルクラウドとローカルクラウドの接点にセキュリティゲートウェイを設置し，認証局により発行されたプライバシ証明書と，ローカルクラウドの管理者およびデータの所有者により階層的に定められたポリシ群を用いてアクセス制御を実施することにより，本問題の解決を図る．

iKaaS Platform Solves Privacy Issues on Cross-Border Data

Seira Hidano†　　Shinsaku Kiyomoto†　　Yosuke Murakami‡

†KDDI R&D Laboratories, Inc.
2-1-15 Ohara, Fujimino-shi, Saitama 356-8502, Japan
‡KDDI Research Institute, Inc.
3-10-10 Iidabashi, Chiyoda-ku, Tokyo 102-8640, Japan

Abstract The iKaaS (intelligent Knowledge-as-a-Service) platform integrates the data on multiple local clouds organically and provides the data to various types of applications as knowledge while taking security and privacy carefully into consideration. However, the access control on the iKaaS platform is not without complications because the application may access the personal data in different countries from the one where the application exists. We thus design a security gateway that is set at the entrance of each local cloud and can control access while interpreting the differences in regulations and guidelines between countries.

1 Introduction

現代のモバイル社会は，急激なパラダイムシフトの渦中にあり，IoT 技術への関心が非常に高まっている．特に，近年の IoT デバイスの成長には目を見張るものがあり，周囲の環境状況を観測するためのスマートセンサや，個人の健康状態を測定するためのウェアラブルデバイスなど，目的や形態を異にする多岐多様なデバイスが市場をにぎわせている．それらのデバイスの多くはクラウドサービスと連携しており，クラウド上には日々膨大な量のデータが集積されている．IoT デバイスから取得されたデータは，個人の嗜好の解析や，環境状況や行動の予測などに利用され，複数のクラウド上に保管されている異なる種類のデータを有機的に組み合わせることができれば，情報の有用性は高まるであろう [1, 2].

iKaaS（intelligent Knowledge-as-a-Service）プラットフォームはプライバシに配慮した情報集約プラットフォームの 1 つである [3]．本プラットフォームでは，グローバルクラウドと複数のローカルクラウドが階層的に配置され，グローバルクラウドはローカルクラウド上のデータを統合し，様々なアプリケーションに知識を提供する．iKaaS プラットフォームを用いることにより，アプリケーションは国境を越えたデータにもアクセスすることができ，国家間の比較など，これまでとは異なる規模の解析が可能となる．しかしながら，iKaaS プラットフォームはその実現に向けて解決すべき重要なプライバシ問題をはらんでいる．IoT デバイス等から取得されるデータの多くは，データの所有者のプライバシに配慮すべき個人データであり，アプリケーションが国境をまたいでそれらのデータにアクセスする場合，iKaaS プラットフォームはアプリケーション側とローカルクラウド側の双方の国の個人データの取り扱いに関する規則に準じなければならない．しかしながら，それらの規則はきわめて複雑であり，特に個人データの定義や他国へのデータ提供の可否等については国家間で大きく異なる場合がある．たとえば，現行の日本の個人情報保護法 [4] では越境データの取り扱いについて特段の定めがないのに対し，EU のデータ保護指令 [5] では日本へのデータ提供は許可されていない．また，分散クラウド環境に関する研究開発は近年のトレンドの 1 つであり，数は少ないがセキュリティやプライバシの問題について言及しているものもある [6, 7]．しかしながら，それらは一般的なセキュリティやプライバシの観点からプラットフォームに求められる要件と既存技術を整理するにとどまっており，越境データのプライバシ問題に取り組んでいるものは皆無である．

そこで，本稿では，個人データに関する規則の国家間の違いを解釈し，越境データのプライバシ問題を解決する iKaaS プラットフォームのセキュリティアーキテクチャを提案する．2 章では iKaaS プラットフォームの機能について概説するとともに，iKaaS プラットフォームに対する本稿の成果を整理する．3 章では，プライバシに配慮した iKaaS プラットフォームのセキュリティアーキテクチャを設計する．4 章では，3 章で提案したアーキテクチャに基づき，アプリケーションが iKaaS プラットフォームを利用してデータにアクセスする際のプロトコルについて述べる．

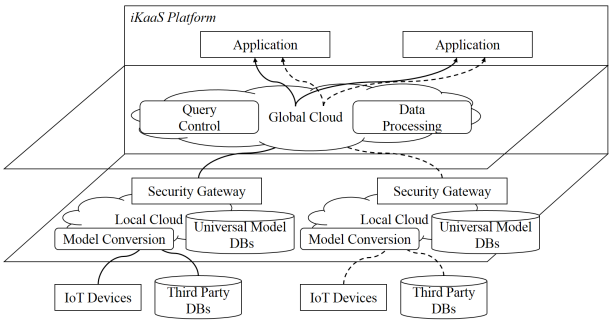

図 1: iKaaS プラットフォーム

2 iKaaS Platform

iKaaS（intelligent Knowledge-as-a-Service）は，複数のローカルクラウド上に蓄積されたデータを有機的に統合し，セキュリティおよびプライバシに配慮してそれらのデータを知識として様々なプリケーションに提供するためのコンセプトモデルである．図 1 に iKaaS プラットフォームの概要を示す．iKaaS プラットフォームは，グローバルクラウド，複数のローカルクラウド，IoT デバイスや既存の DB から成り，それらは階層的に配置される．ローカルクラウドは日本と UK など異なる国に設置される場合があり，ローカルクラウド上の DB には様々な種類のデータが蓄積される．ただし，データは新しく配置された IoT デバイスから取得されるだけでなく，他の目的で設計された既存の DB 等のデータも利用されるため，すべてのデータは iKaaS データモデルと呼ばれる普遍的なデータモデルに変換した上で保管される．グローバルクラウドは信頼できる機関であり，すべてのデータはグローバルクラウドを介してやり取りされる．グローバルクラウドは，主に，クエリ制御，データ処理の 2 つの機能を持つ．まず，グローバルクラウドはアプリケーションの目的

にあったデータが保管されているローカルクラウドを探索し，iKaaS データモデルに準じてクエリを作成してそれを該当のローカルクラウドに投じる（クエリ制御）．また，アプリケーションにデータを返す際は，ローカルクラウドから返ってきた生のデータをそのまま返す場合もあるが，アプリケーションからの要求に応じて，グローバルクラウド上で統計処理のような何らかの処理が加えられる場合がある（データ処理）．アプリケーションは，iKaaS プラットフォームを用いることにより，様々な国のローカルクラウドにアクセスすることができるが，IoT デバイス等から取得されるデータは機微な情報であることが多く，様々なプライバシ問題が浮上する．1 章で述べたように，個人データの第三国への提供に関しては，各国において様々な規則が存在する．また，個人データはデータの所有者の意思に基づき提供されるべきである．そこで，iKaaS プラットフォームは，グローバルクラウドとローカルクラウドの接点にアクセス制御機能とプライバシ制御機能を持つセキュリティゲートウェイ（セキュリティGW）を設置することにより，上述のプライバシ問題を解決する．

Our Contributions iKaaS プラットフォームに対する本稿の成果は以下の通りである．

- アプリケーション側とローカルクラウド側の双方の国の越境データに関する規則の違いを解釈して矛盾なくアプリケーションのアクセス制御を行うセキュリティGW を設計する．ただし，本機能は 3 章で定義するプライバシ証明書とセキュリティポリシを用いて実現される．また，セキュリティGW はプライバシポリシを用いて，プライバシ制御を行う．
- トークンを用いたアクセス制御機能を導入し，同一アプリケーションが連続してデータにアクセスする場合の効率性を向上する．トークンを用いることにより，セキュリティGW はプライバシ証明書やセキュリティポリシの都度の確認作業を省略することができる．これにより，IoT デバイス等から短周期で逐次的に取得されるデータに対するアプリケーションからの連続的なアクセス要求に対応する．
- プライバシ証明書を発行するための機関であるプライバシ CA の機能を利用して，セキュリティGW にアプリケーションの正当性を確認する機能を導入する．これにより，アプリケーションとセキュリティGW の間にグローバルクラウドが介在するために SSL のクライアント認証等の従来の認証プロトコルが使えない問題を解決する．

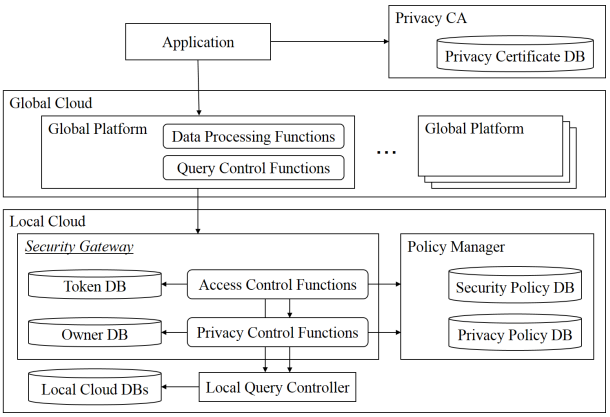

図 2: プライバシに配慮した iKaaS のセキュリティアーキテクチャ．

3 Security Gateway

図 2 にプライバシに配慮した iKaaS プラットフォームのセキュリティアーキテクチャを示す．セキュリティGW はローカルクラウド毎に，グローバルクラウドとの接点に設置され，アプリケーションからのクエリとローカルクラウド上のデータはすべてセキュリティGW を通してやり取りされる．また，セキュリティGW は，アプリケーション側とローカルクラウド側の双方の国の個人データに関する規則に配慮してアプリケーションにアクセス権を付与するアクセス制御と，データの所有者の意思に基づきデータの提供可否を決定するプライバシ制御の 2 つの機能を持つ．ただし，アクセス制御はプライバシ CA によって発行されるプライバシ証明書（3.1 節）およびセキュリティポリシ（3.2 節）を用いることにより実現される．また，プライバシ制

御の際には，プライバシポリシ（3.3 節）が用いられる．両機能の詳細については，3.4 節で述べる．

3.1 Privacy Certificate

プライバシ証明書は，セキュリティGWがアプリケーション側の国の規則を解釈する際に用いられる．したがって，アプリケーションがiKaaSプラットフォームを利用するためには，事前にプライバシ証明書を取得しておかなければならない．ただし，プライバシ証明書の発行局であるプライバシ CA は各国に設置され，アプリケーションは自国の CA から証明書を発行してもらう必要がある．プライバシ証明書は，個人データに関する国家レベルの規則と，アプリケーションに関する種々の情報に基づき作成され，以下の項目が記載される．

- *CA Country:* プライバシ CA の設置国.
- *Application IP:* アプリケーションの IP アドレス.
- *Application ID:* アプリケーションが提供するサービスを種別する識別子.
- *LC Countries:* アプリケーションがアクセスすることが許可されている国の名前．複数の値を指定可.
- *LC Data IDs:* アプリケーションがアクセスすることが許可されているデータを種別する識別子．本項目は，*LC Countries* の各値に入れ子で定義される．複数の値を指定可.
- *Expires:* プライバシ証明書の有効期限.
- *Application PK:* アプリケーションの公開鍵.（本項目の役割は，3.4 節参照）
- *Signature:* 署名は，プライバシ CA の秘密鍵を用いて生成される．公開鍵は事前にセキュリティGW に配布される.

表 1: アクセス権の有効期間

No	Data 1	⋯	Data N
1	UK 0/JP 2mo	⋯	UK 0/JP 0
2	UK 1h/JP 2h	⋯	UK 0/JP 0
⋮	⋮	⋱	⋮

表 2: プライバシの定義

No	Data 1	⋯	Data N
1	Non-privacy	⋯	Privacy
2	Privacy	⋯	Privacy
⋮	⋮	⋱	⋮

3.2 Security Policy

セキュリティポリシは，ローカルクラウドの管理者により，ローカルクラウド側の国の規則に基づき，ポリシマネージャを通して設定される．ポリシマネージャは，ローカルクラウドが設置されている国のプライバシ CA によって提供され，国家レベルの規則に準ずる基本ポリシはあらかじめ設定されているものとする．管理者は国家レベルよりも小さい単位，すなわち，県や市レベルの規則，もしくは，企業などの組織毎の規則，また，必要に応じて種々のガイドラインに基づき，セキュリティポリシを設定する．セキュリティポリシの例を表 1，2 に示す．セキュリティポリシは，アクセス権の有効期間とプライバシの定義に関する 2 つのテーブルを持つ．どちらのテーブルについても，管理者は，データの種類（データ ID）毎に値を設定し，各行はある 1 つの規則もしくはガイドラインに対応する（基本ポリシもいずれかの行に含まれているものとする）．アクセス権の有効期間は表 1 に示すように，国毎に設定される．ただし，値 “0” はその国に対してデータの提供を許可しないことを意味する．表 1 はアプリケーションの種類毎に設定される．表 2 の値 “Privacy” は，当該データ ID で表される種類のデータについては，データの所有者のプライバシに配慮する必要があることを表し，値 “Non-privacy” は配慮する必要がないことを表す．ただし，表 2 において値 “Non-privacy” が設定されたデータ ID

表 3: プライバシポリシ

Owner ID	Data 1	···	Data N
1	Yes	···	Yes
⋮	⋮	⋱	⋮
K	No	···	Yes

についても，セキュリティおよびデータの更新頻度等を考慮して表 1 においてアクセスの有効期間を設定しなければならない．

3.3 Privacy Policy

個人データについては 2 章でも述べたようにプライバシの観点からデータの所有者が自らの意思でデータ提供の可否を制御できるべきである．本モデルでは，iKaaS プラットフォームへのデータ提供に関するデータの所有者らの同意状況は，プライバシポリシ [8] に記載される．セキュリティGW は，プライバシポリシを参照してデータの所有者らのプライバシを制御する．表 3 にプライバシポリシの一例を示す．データの所有者らは，データ ID 毎にデータ提供の可否を設定する．値 “Yes” は当該所有者が iKaaS プラットフォームへのデータ提供に同意していることを表し，値 “No” は同意していないことを表す．ただし，表 3 も表 1 と同様にアプリケーション種類毎に設定される．

3.4 Access and Privacy Control

セキュリティGW によるアプリケーションのアクセス制御はトークンを用いて行われる．アプリケーションからローカルクラウド DB へのアクセス要求があった場合，セキュリティGW は当該アプリケーションにトークンを返す．そして，正しいトークンを持つアプリケーションのみがデータを取得することができる．アプリケーションは，トークンの有効期限が切れるまで同一トークンを用いて何度でもデータにアクセスすることができる．トークンを用いることにより，セキュリティGW によるプライバシ証明書やセキュリティポリシの確認処理に関わるコスト問題を回避し，短周期で逐次的に更新されるデータへの連続的なアクセスが可能となる．セキュリティGW は上記の処理を実現するために 2 つの関数 *Issue Token* と *Get Data* を提供する．

3.4.1 Issue Token

関数 *Issue Token* はアプリケーションがローカルクラウド DB にアクセスするためのトークンを取得する際に呼び出される．アプリケーションはアクセスしたいデータのデータ ID とプライバシ証明書をセキュリティGW に明示する必要がある．セキュリティGW はプライバシ証明書とセキュリティポリシを用いて以下の方法でアプリケーション側とローカルクラウド側の双方の国の規則を確認し，アプリケーションにトークンを発行する．

まず，プライバシ証明書に記載されている *LC Countries* と *LC Data IDs* の値を用いて，アプリケーション側の国の規則により指定されたデータ ID のデータについてアプリケーションがアクセスすることが許可されているかを確認する．次いで，セキュリティポリシに記載されているアクセスの有効期間を参照し，ローカルクラウド側の国の規則やガイドラインを確認するとともに，トークンの有効期限を決定する．この際，セキュリティGW は，プライバシ証明書に記載されている *Application ID* の値と指定されたデータ ID を用いて，該当するデータ ID の列を検索する．そして，その列の中で，プライバシ証明書に記載されている *CA Country* の国に関する有効期間に着目し，最も短い期間と現在の時刻を足し合わせ，トークンの有効期限を導出する．もし，アプリケーションにより複数のデータ ID が指定されている場合には，それぞれのデータ ID に対して，上述のプロセスを実行し，複数の有効期限を 1 つのトークンに関連付ける．ただし，すべてのデータ ID に対して，最も短い有効期間が値 “0” の場合，トークンは発行しないものとする．

次に，セキュリティGW は，セキュリティポリシの表 2 を参照し，指定されたデータ ID のプライバシの定義を確認する．トークンの有効

期限を決めた場合と同様に，表 2 の中の該当列を参照し，すべての行で値 “Non-privacy” が設定されていれば，そのデータ ID で表される種類のデータを Non-privacy データと呼び，プライバシに配慮する必要がないと判断する．いずれかの行に “Privacy” が設定されている場合には，当該データを Privacy データと呼び，アプリケーションに提供する際にデータの所有者のプライバシに配慮する必要があると判断する．このプロセスもまた指定されたすべてのデータ ID に対して行われる．本稿では，Non-privacy データもしくは Privacy データのことをプライバシタイプと呼ぶ．

最後に，セキュリティGW は，プライバシ証明書に記載されている *Application IP* と *Application ID* の値，トークン，アプリケーションにより指定されたデータ ID，トークンの有効期限，プライバシタイプを関連付けてトークン DB に保管する．複数のデータ ID が指定されている場合には，それぞれについて，データ ID，トークンの有効期限，プライバシタイプのセットを作成し，1 つのトークンにすべてのセットを関連付ける．そして，プライバシ証明書に記載されている *Application PK* を用いてトークンを暗号化し，それをアプリケーションに返す．iKaaS プラットフォーム上では，アプリケーションとセキュリティGW の間にはグローバルプラットフォームが介在するため，セキュリティGW が直接アプリケーションとコミュニケーションを取ることができず，アプリケーションの正当性の検証に SSL のクライアント認証のような既存のプロトコルを使用することができない．トークンの暗号化は本問題を解決するための一手段であり，プライバシ CA により正しくプライバシ証明書が発行されたアプリケーションのみが自身の秘密鍵を用いてトークンを復号することができる．

3.4.2 Get Data

関数 *Get Data* は，トークンを取得済みのアプリケーションがローカルクラウド DB にクエリを投じる際に呼び出される．本関数が呼び出されたとき，セキュリティGW は始めにトークンの真正性を確認する．ただし，この確認は MAC（Massage Authentication Code）を用いて行われる．すなわち，トークンは共通鍵として利用される．アプリケーションにデータを返す際は，データの所有者のプライバシに配慮するために，セキュリティGW はアプリケーションがアクセスしたいデータのプライバシタイプを確認する．データのプライバシタイプが Non-privacy データの場合，セキュリティGW は何もすることなく，すべてのデータをそのままアプリケーションに返す．一方，プライバシタイプが Privacy データの場合，セキュリティGW はプライバシポリシに基づき，データのフィルタリングを実施する．データの所有者の識別子（オーナ ID）と所有者の属性はオーナ DB の中で関係付けられており，セキュリティGW はクエリで指定された属性情報を利用して，該当するオーナ ID を抽出する．そして，抽出したオーナ ID を用いてプライバシポリシを検索し，クエリで指定されているアプリケーションの種類とデータ ID に対するデータの所有者らのデータ提供に関する同意状況を確認する．セキュリティGW は値 “Yes” が選択されている所有者のデータのみをアプリケーションに返す．

4 Protocol

本章では，iKaaS プラットフォーム上でセキュリティおよびプライバシに配慮してデータを提供するためのプロトコルを定義する．まず，セキュリティGW に投じられるクエリの構造について述べ，次いで，トークンの発行およびデータ要求に関するシーケンスを示す．ただし，セキュリティGW はそれぞれの関数を Web API として提供し，コミュニケーションはすべて HTTPS で行われるものとする．さらに，グローバルクラウドはアプリケーション毎にグローバルプラットフォームのインスタンスを生成し，正しいアプリケーションのみがその空間にアクセスできるものとする．

4.1 Query Formats

セキュリティGWはローカルクラウドDBに対する複雑なクエリ（*LCD-query*）を解釈する機能を持たない．このため，アプリケーションがデータを要求するときは，グローバルプラットフォーム上でセキュリティGWが解釈できるヘッダ（*SGW-headers*）が付加され，セキュリティGW用のクエリ（*SGW-query*）が生成される．*SGW-headers* の種類は以下の通りである．

- *Application IP:* アプリケーションのIPアドレス．
- *Application ID:* アプリケーションが提供するサービスの種別する識別子．
- *LC Data IDs:* アプリケーションがアクセスを要求するデータを種別する識別子．
- *Owner Attributes:* データの所有者の属性を表す情報．本ヘッダは，性別や年齢などの情報であり，*LCD-query* で属性が指定されている場合のみ設定される．3.4節で示したように，セキュリティGWはこの値を用いてオーナDBから該当するオーナIDを抽出する．
- *Time Stamp: SGW-query* が生成された時刻を表すヘッダ．本ヘッダによりクエリのリプライ攻撃を防止する．

4.2 Sequences

4.2.1 Token Issuance

1. アプリケーションはプライバシCAにプライバシ証明書の発行を要求する．
2. アプリケーションはグローバルプラットフォームの機能を利用して，目的に合ったDBが所在するローカルクラウドを検索し，トークンの発行を要求する．
3. グローバルプラットフォームはセキュリティGWの関数 *Issue Token* を呼び出す．その際，グローバルプラットフォームはアプリケーションがアクセスしたいデータのデータIDを明示するとともに，Step 1でアプリケーションが取得したプライバシ証明書をセキュリティGWに送信する．
4. セキュリティGWはプライバシ証明書に記載されている *Expires* と *Signature* の値を検証し，証明書の正当性を確認する．ただし，署名の検証は，プライバシ証明書に記載されている *CA Country* で示された国のプライバシCAの公開鍵を用いて行われる．
5. セキュリティGWはトークンを生成し，プライバシ証明書に記載されているアプリケーションの公開鍵を用いて暗号化した後に，グローバルプラットフォームを通して，それをアプリケーションに返す．
6. アプリケーションは自身の秘密鍵を用いてトークンを復号し，グローバルプラットフォーム上にそれを保管する．

4.2.2 Data Request

1. アプリケーションはグローバルプラットフォームの機能を利用して *SGW-query* を作成する．その際，グローバルプラットフォームは当該アプリケーションのトークンを用いて *SGW-query* のMACを生成する．
2. グローバルプラットフォームはセキュリティGWの関数 *Get Data* を呼び出し，*SGW-query* とMACをセキュリティGWに送信する．
3. セキュリティGWは *Application ID* および *Application IP* ヘッダの値を用いて，トークンDBから該当するトークンを抽出し，そのトークンの有効期限を確認する．
4. セキュリティGWは抽出したトークンと *SGW-query* からMACを生成し，クエリの真正性を確認する．*Time Stamp* ヘッダの値も合わせて確認する．
5. セキュリティGWは *LCD-query* をローカルクラウドのクエリ制御装置に送信する．

6. ローカルクラウド DB からデータが返ってきたら，セキュリティ GW はトークン DB を検索し，データのプライバシタイプを確認する．

7. データのプライバシタイプが Non-privacy データであれば，セキュリティ GW は何もすることなく，グローバルプラットフォームを通して，アプリケーションにデータを返す．データのプライバシタイプが Privacy データの場合は，Steps 8–10 の処理を実施する．

8. セキュリティ GW は，*Owner Attributes* ヘッダの値を用いて，オーナ DB から該当するオーナ ID を抽出する．

9. セキュリティ GW は，抽出したオーナ ID と，*Applica0tion ID* および *LC Data IDs* ヘッダの値を用いて，該当するデータの所有者らのデータ提供に関する同意状況を確認する．

10. セキュリティ GW はデータ提供に同意している所有者のデータのみを抽出し，グローバルプラットフォームを通してそれらのデータをアプリケーションに返す．

5 Conclusion

iKaaS（intelligent Knowledge-as-Service）プラットフォームは国境をまたぐ複数のローカルクラウド上のデータを有機的に統合し，目的が異なる様々なアプリケーションにそれを知識として提供する．本稿では，iKaaS プラットフォーム上にて，アプリケーションのアクセス制御を担務するセキュリティ GW に着目し，プライバシ CA が発行するプライバシ証明書と，ローカルクラウド上で管理されるセキュリティポリシを用いることにより，アプリケーション側とローカルクラウド側の双方の国の個人データに関する規則を解釈するプライバシに配慮したセキュリティアーキテクチャを提案した．

Acknowledgments

The work is supported by the EUJ-1-2014 Research and Innovation action: iKaaS; EU Grant number 643262, Strategic Information and Communications R&D Promotion Programme (SCOPE), Ministry of Internal Affairs and Communications, Japan.

参考文献

[1] EU FP7/ICT project 257115, “OPTIMIS: Optimized Infrastructure Services,” June 2010–May 2013.

[2] EU FP7/ICT project 287708, “iCore: Internet Connected Objects for Reconfigurable Ecosystems,” October 2011–September 2014.

[3] EU HORIZON 2020 project 643262, “iKaaS: intelligent Knowledge-as-a-Service,” 2014–2017.

[4] Japan, “Act on the Protection of Personal Information,” Act No. 57 of May 30, 2003.

[5] EU, “Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data,” 1995.

[6] EU FP7/ICT project 609094, “RERUM: REliable, Resilient and secUre IoT for sMart city applications,” 2013–2016.

[7] H. de Meer, H. C. Pohls, J. Posegga, and K. Samelin, “On the relation between redactable and sanitizable signature schemes,” in *Engineering Secure Software and Systems*. Springer, 2014, pp. 113–130.

[8] S. Kiyomoto, T. Nakamura, H. Takasaki, R. Watanabe, and Y. Miyake, “PPM: Privacy policy manager for personalized services,” in *Security Engineering and Intelligence Informatics*. Springer, 2013, pp. 377–392.