

MARS : 相互認証を基盤とする 未来型 RSS 配信システムの開発

梶原 広輝[†], 廣安 知之^{††}, 三木 光範^{††}

[†] 同志社大学大学院 ^{††} 同志社大学工学部

Hiroki KAJIWARA[†] Tomoyuki HIROYASU^{††} Mitsunori MIKI^{††}

[†] Graduate School of Engineering, Doshisha University ^{††} Department of Knowledge Engineering,
Doshisha University

本稿では、各種センサから観測される現実世界の様々な情報データを、グリッドの有する相互認証と RSS (Really Simple Syndication) による標準データフォーマットを利用した情報配信システム、MARS (Mutual Authentication RSS, 相互認証を基盤とする未来型 RSS 配信システム) の開発について述べた。

近年のユビキタス・コンピューティングに代表されるように、センサ・ネットワークへの需要は年々高まっている。センサ・ネットワークの今後の発展を考えると、センサは今後ますます、小型・省電力・多様化することが考えられ、個人レベルでの利用形態が想定される。個人で利用するセンサ情報は、機密性が高く、強固なセキュリティと認証を考慮する必要がある。また、多くのユーザがセンサを利用することから、各センサ・リソースへの認可と利便性の高い利用形態を考慮する必要がある。そこで、近年、科学技術計算分野を中心に発展してきた、グリッド (グリッド・コンピューティング) の認証基盤を利用することを考えた。グリッドの認証基盤を利用することで、強固な暗号化、第 3 者認証局による相互認証、リソースやユーザ単位の認可、一度の認証で組織間をまたいだリソースの利用 (SSO : Single Sign-On) などが実現する。そして、Web サービスを基盤とする新たなグリッドの利用形態として、WSRF (WS-Resource Framework) と WSN (WS-Notification) に基づくメッセージ通知を利用したプッシュ型 RSS 配信で、効果的なセンサ・リソースの配信を実現する。また、グリッド上の Web サービス間のデータフォーマットとして、既存の RSS フォーマットを利用することで、Web サービス間で統一的就易にセンサ・リソースの利用が期待できる。グリッドの相互認証を基盤とし、RSS フォーマットによるデータ表現により、今までに無かった有益なサービスを創生する、新たなセンサ・プラットフォームを構築することを目的とする。

1 はじめに

コンピュータやセンサの小型・省電力化、高速インターネット網や ZigBee, WiMAX といった無線技術の普及、携帯端末や RFID など新たなデバイスの技術革新などに伴い、我々が生活する環境には様々な情報機器群が存在するようになった。このような背景を基に、近年ではユビキタス・コンピューティングやウェアラブル・コンピューティング、モバイル・コンピューティングといった、情報機器群 (センサ群) をネットワークを介して連携させ、有益なサービスを提供する構想 (センサ・ネットワーク) が生まれている。現在までに、多くの研究機関や組織、プロジェクト等で検証実験や実運用がなされてきた。

従来のセンサ・ネットワーク利用形態の多くは、データの機密性保持のため、センサ情報を提供する側と、享受する側は閉ざされた同一組織内に限られてきた。センサ情報が異なる組織間にまたが

る際は、センサ情報が広く公に公開されることが前提か、もしくはユーザ ID とパスワードを利用したベーシック認証や、独自プロトコルが用いられる。しかし、従来のベーシック認証では、セキュリティ上の信頼性が乏しく、サイトごとに管理を行うため、利便性に乏しく、独自プロトコルを利用する際は、組織間をまたいだ利用が困難となる。センサの小型・省電力・多様化に伴うセンサ・ネットワークの今後の発展を考えると、より多くのセンサ群が広域ネットワーク上に配置され、複数組織間での利用形態が進むことが考えられる。また、今までは限られた企業や組織間でのみ利用されてきたセンサ群が、今後は個人レベルに至るまで利用されるようになり、様々なセンサ・アプリケーションが開発・利用され、より一層のセキュアかつ透過的な利用形態が望まれると考えられる。

このように、組織間をまたいでセキュアに情報配信を行い、かつ利便性の高いプラットフォームを

考えた際に、広域ネットワーク上に接続されたあらゆる資源（CPU、メモリ、ストレージ、センサなど）を、仮想的に統合利用するグリッド¹⁾の利用が挙げられる。グリッドの利用形態は、リソースを提供する側と利用する側が、異なる組織間に存在しており、提供・利用する側の相互認証と強固な通信の暗号化、利便性の高いSSO（Single Sign-On）が基盤となる。そして、従来のWeb利用形態のようにサイトごとではなく、リソースやユーザごとに利用権限の認可が行われ、様々なアプリケーション・サービスがセキュアかつ透過的に利用される。

また、近年グリッドの新たな利用形態として、サービス・グリッド（ビジネス・グリッド）の利用が促進されてきた。サービス・グリッドは、Webサービス技術を基盤とし、アプリケーションやデータ、その他グリッドを実現する機能群を全てWebサービスとして表現することで、それらの統合的な利用を可能にするアプローチである。サービス・グリッドは、Webサービスを用いたサービス指向アーキテクチャ（Service Oriented Architecture：SOA）を採用しており、データ記述言語としてXML（eXtensible Markup Language）、インタフェース記述形式としてWSDL（Web Services Description Language）、サービス間のメッセージ記述形式としてSOAP（Simple Object Access Protocol）を用いている。そのため、センサ・ネットワークでのサービス・グリッドの利用を考えると、センサから観測されるデータを、ネットワーク間のアプリケーションで統一的に利用するための標準データフォーマットを定義する必要がある。一方で近年、Webの世界で情報配信の新たなデータフォーマットとして、XMLを利用したRSS（Really Simple Syndication, RDF Site Summary）が用いられるようになった。RSSはブログやニュースサイトを中心とし利用されはじめ、現在ではHTMLやテキスト情報のみならず、音楽、動画配信といったマルチメディア・コンテンツの配信にも利用されている。

そこで、本研究では、センサ・ネットワークの新たなプラットフォームとして、グリッドの有する相互認証とRSSによる標準データフォーマットを利用した情報配信システム、MARS（Mutual Authentication RSS, 相互認証を基盤とする未来型RSS配信システム）を開発する。グリッドの相互

認証基盤を利用することで、複数組織間にまたがるデータ配信や個人個人のプライベートにまで至るデータをセキュアかつ透過的に扱うことが可能となる。また、センサ・アプリケーションの開発・利用の際に定義するデータフォーマットに、既存のRSSを利用することで、データを統一的かつ容易にアプリケーション開発が行えることが期待できる。

以降より、提案システムである相互認証を基盤とするRSS配信について述べる。

2 相互認証を基盤とするRSS配信システム

2.1 MARSの利用形態

MARSの利用形態をFig. 1に示す。MARSでは現実世界に存在する情報を、各種センサで観測を行い情報配信を行う。センサ情報を提供する側と享受する側はグリッド環境上に存在し、セキュアに情報配信が行われる。そのため、Fig. 1にもあるように、ユーザによって利用できるリソースが異なる。これは、今後のセンサ・ネットワークの利用形態を想定し、リソースやユーザごとに権限の認可を必要とするためである

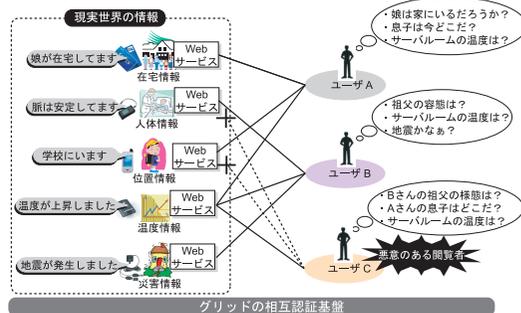


Fig. 1 MARS の利用形態

次節より、Fig. 1に示すMARSの利用形態を実現するための、グリッドの相互認証とRSSデータフォーマットについて述べる。

2.2 グリッド

2.2.1 グリッドの概要

近年のハードウェアにおける技術革新により、多くの機関で大規模な計算資源を保有できるようになった。しかし、大規模計算の要求は尽きることなく、複数の計算機やデータセンタ、希少な観測

機器を高速なネットワークで接続することで、莫大な計算資源や情報資源を得ようというニーズが生じている。このニーズを受けて開発された技術がグリッドである。

グリッドとは通常、電力網を意味する言葉であり、現在では電気を使う際、その電気がどこでどのように発電され、どんな経路を経て家庭まで供給されているかを気にする人はいない。同じようにコンピューティング資源として計算資源や情報資源も、自由に簡単に、いつでもどこでも必要なだけ使えるようにしたい、というコンセプトのもとグリッド・コンピューティングが生まれた。

グリッド・コンピューティングをより技術的な観点から見た場合、地理的・組織的に広範囲に分散した様々な種類のCPU、メモリ、ストレージ、情報、その他の資源を仮想的に連携させ、ユーザの需要と提供者のポリシーに応じて動的に共有し、それらを協調的に動作させ、問題を処理させるための技術基盤である。

グリッドの利用形態としては、構造解析やバイオ・インフォマティクス、数値シミュレーションなど、大規模リソースを利用した科学技術計算分野での利用形態が主であった^{2, 3)}。グリッドを実現するための、ツールやミドルウェア等も盛んに研究・開発されている^{4, 5)}。グリッドのアプリケーションやサービスは、科学技術計算分野を中心に利用されてきたが、全てのアプリケーションとサービスは、グリッドの有する認証基盤上で利用される。本研究では、グリッド上で標準ツールとして利用されている、Globus Toolkit⁶⁾ (バージョン4.0)を用いる。Globus Toolkitは、グリッドの認証基盤を提供し、様々な有益なライブラリ群を提供し、デファクト・スタンダードとして利用されている。

2.2.2 認証局を用いた相互認証基盤

グリッド上でのアプリケーション実行の一般的な流れを Fig. 2 に示す。一般的には、ジョブを実行させたいエンドユーザ、広域ネットワーク上に分散するリソースを保持するサーバ (複数ノードからなる PC クラスタシステム構成)、第3者認証局 (CA: Certificate Authority)、の3構成となる。グリッド上での相互認証は、公開鍵証明書に基づく X.509 電子署名を用いる。エンドユーザはあらかじめ用意した秘密鍵と公開鍵の対からなる

ユーザ証明書のうち、公開鍵を第3者認証局に署名してもらおう。同様に、サーバ側のホスト証明書の公開鍵を第3者認証局に署名してもらおう。これにより、ユーザとサーバが同一の第3者認証局を信頼したこととなる。その後、ユーザは第3者認証局に署名してもらったユーザ証明書から、サーバとの暗号化に用いる一時的 (通常、12時間の有効期間が設定される) なプロキシ証明書を生成する。ユーザはこのプロキシ証明書を用いて、サーバとの公開鍵暗号方式によるセキュアな通信を行う。サーバは「自身の信頼する第3者認証局が信頼した正規のユーザからのリクエストである」と判断し、ユーザは「自身の信頼する第3者認証局が信頼した正規のサーバである」と判断し、両者間で相互認証が行われる。

また通常の Web 上でのベーシック認証では、サイトごとにパスワードが必要となり、利便性に乏しい。しかしながら、グリッド上では、プロキシ証明書を用いて、一度認証を行うと組織間をまたいだ SSO が前提となっている。これは、Fig. 2 にも示すように、サイト.A やサイト.B も同一の第3者認証局から署名を受けているためである。そして、サイト.B からサイト.C へのジョブの委譲がなされているように、グリッド上ではユーザの権限 (プロキシ証明書) を委譲する機構 (Credential Delegation) が備わっている。

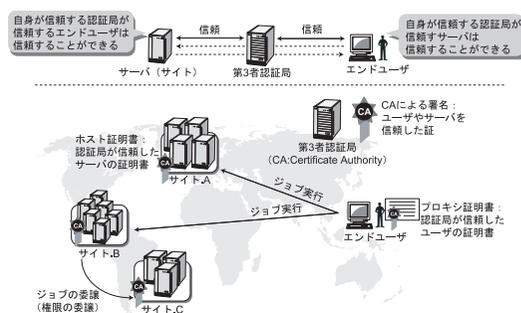


Fig. 2 グリッド上でのアプリケーション実行の流れ

第3者認証局は通常、以下に示すように複数存在するが、現在では特に simpleCA を用いた認証局が広く構築・運営されている。

- Globus CA :

Globus Alliance⁷⁾ が運営をする。メールで申請し、発行までに約 2 日かかる。

- simpleCA :
OpenSSL ベースの CA 構築パッケージ。Globus Toolkit と共に提供される。
- OpenSSL :
広く普及しているオープンソースの CA 構築パッケージ。
- Commercial CA :
ベンダーによる商業用の CA サービス。

2.2.3 Open Grid Services Architecture

ビジネス分野におけるグリッドの利用促進を目的として、OGSA (Open Grid Services Architecture)⁸⁾ が提案された。OGSA は、Web サービスを用いたサービス指向アーキテクチャを採用しており、データ記述言語として XML、インタフェース記述形式として WSDL、サービス間のメッセージ記述形式として SOAP に代表される Web サービス技術群をグリッド技術の基盤としている。そして、Web サービスの連携によってグリッドで利用可能なシステムを構築することを目標としている。グリッド上で開発・提供される数多くのアプリケーションは、今後全てが Web サービスのインタフェースを有するようになることが期待でき、それらをコンポーネントとして組み合わせるといった開発スタイルが、今後重要になると考えられる。

OGSA により、これまでは科学技術計算を中心に発展してきたグリッド技術が、今後ビジネス分野において高く注目されることが期待できる。本研究においても、この OGSA に準ずる Web サービス同士がデータ配信を行うことを目指す。

2.2.4 WS-Resource Framework と WS-Notification

OGSA は、WS-Resource Framework (WSRF)⁹⁾ と WS-Notification (WSN)¹⁰⁾ の 2 つの仕様に代表される Web サービス技術を基盤としている。以下にそれぞれの Web サービス技術について述べる。

- **WS-Resource Framework :**
WSRF は、Web サービスに状態を有するリソースを導入するための仕様である。リソースはリソースプロパティのセットにより構成

される。WSRF により、サービス実行中の動的な状態変化を表現することが可能になる。

- **WS-Notification :**
WSRF により状態を有するリソースを導入した Web サービスにおいて、状態変化を扱う操作は非常に重要である。WSN は状態変化により駆動する登録型の非同期メッセージ通知の枠組みである。

2.3 メッセージ通知によるプッシュ型 RSS 配信

状態を保持する WSRF と、状態変化を通知する WSN により、グリッド上の Web サービスで効率的に RSS 配信を行うことを考える。メッセージ通知によるプッシュ型 RSS 配信を Fig. 3 に示す。Web サービスクライアントはあらかじめ、リソースプロパティを保持する Web サービスに対して、通知予約 (subscribe) を行う。その後、センサ情報が更新され新たな RSS が生成されると、その旨を Web サービスクライアントに通知 (notify) する。

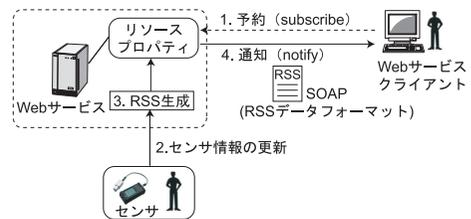


Fig. 3 メッセージ通知によるプッシュ型 RSS 配信

Web 上で用いられる通常のプル型 RSS 配信とは異なり、WSRF と WSN によるメッセージ通知では、リソース提供側がリソース要求側に情報更新の通知を行うプッシュ型の配信形態となる。プッシュ型配信はプル型配信に比べ、リアルタイム性が高く、特に本研究で対象とするようなセンサ・ネットワークでの利用価値は高いものと考えられる。

2.4 代理発行サーバ MyProxy

MARS が想定する一般のエンドユーザは、グリッドや Globus Toolkit といった専門知識を持たないユーザであり、MARS はエンドユーザ側からの処理を全て Web 上 (Web ブラウザ) で行う。そのため、前述のプロキシ証明書や権限の委譲といったグリッド特有の機構は、サポートされていない。プロキシ証明書を用いた認証を行うためには、あらかじめ作成したプロキシ証明書を、Web を介してサー

バ側に委譲する必要がある。Globus Toolkit を用いたプロキシ証明書の委譲として、MyProxy¹¹⁾ が実装されている。MyProxy を用いたグリッド上でのアプリケーション実行の流れを Fig. 4 に示す。

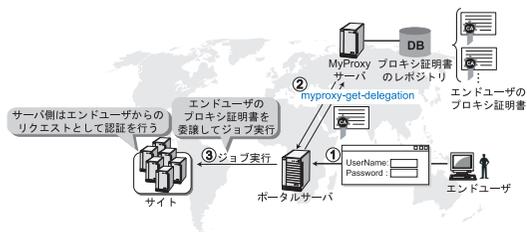


Fig. 4 MyProxy を用いたグリッド上でのアプリケーション実行の流れ

Fig. 4 において、MyProxy サーバは任意の場所にあり、エンドユーザのプロキシ証明書を保持している。Globus Toolkit が完備されたポータルサーバは、エンドユーザからのリクエスト（ユーザ名とパスワード）を基に、myproxy-get-delegation コマンドにより、MyProxy サーバからエンドユーザのプロキシ証明書をセキュアに取得する。その際エンドユーザが利用するパスワードは、MyProxy のレポジトリにプロキシ証明書を登録する際に作成する一時的なキーとなっている。そして、取得したエンドユーザのプロキシ証明書を委譲することで、グリッドアプリケーションを提供するサーバとの認証を行う。

MyProxy を用いることで、エンドユーザは専門的な知識や環境（Globus Toolkit 環境）を用意する必要がなく、グリッドの有する相互認証と権限の委譲、グリッドアプリケーションを Web 上で実行することができる。

2.5 RSS によるデータフォーマットの定義

2.5.1 スキーマの定義

Fig. 3 で示したように、WSRF と WSN のメッセージ通知によるプッシュ型 RSS 配信を行うためには、リソース提供側と要求側での Web サービス間で、互いに統一的に扱えるデータフォーマットを定義する必要がある。通常の Web サービスでは、データフォーマットの定義を、Web サービス開発者がサービスごとに独自のスキーマを定義して行う。しかしながら、独自にスキーマを定義する場

合、汎用性と拡張性に欠けたり、サービスごとにスキーマを定義する多大な労力がかかるなど問題が生じる。

そこで MARS では、データのスキーマ定義に RSS (Really Simple Syndication) フォーマットをベースにした XML Schema を利用することを考える。

2.5.2 Really Simple Syndication

RSS は、Web サイトの見出しや要約などのメタデータを、構造化して記述する XML 形式の標準フォーマットであり、主に Web サイトの更新情報を効率的に伝えるためのデータフォーマットとして利用されてきた。XML 形式の標準フォーマットであることから、ソフトウェア的に統一的かつ簡単に扱うことが可能であり、閲覧用の RSS リードアプリケーションや Web サイト間の連携に用いられてきた。現在までに RSS は、主に以下に示す 2 つの仕様が存在する。

- RSS 0.90, 1.0 :
1999 年に Netscape Communication が自社のポータルサイトにおいて、「チャンネル」の詳細を記すために策定した RDF (Resource Description Framework) がベースとなっている RSS (RDF Site Summary) .
- RSS 2.0 (0.91~0.94, 2.0) :
1997 年に UserLand Software により、RDF の構文とは異なる RSS 0.91 (Rich Site Summary) が開発された。その後、RSS 0.91 を拡張し後方互換性を持つ RSS 2.0 (Really Simple Syndication) が開発された。

MARS では、RSS データフォーマットとして、RSS 2.0 を用いる。

2.5.3 RSS によるデータフォーマット

XML 形式の標準フォーマットである RSS は、Fig. 5 に示すような構造を持つ。channel 要素には、title (チャンネル名)、link (チャンネルの URL)、description (チャンネルの説明) の要素が必須となる。その他のオプション要素として、language (チャンネル記述言語)、copyright (著作権の表記)、pubDate (発行日)、lastBuildDate (更新日) などの要素がある。channel 要素の子要素として、通常は複数 item 要素がある。item 要素には、title、description、pubDate、permaLink などがある。

```

<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0" xmlns:blogChannel=
"http://backend.userland.com/blogChannelModule">
<channel>
<title>Temperature and Humidity</title>
<link>http://mikilab.doshisha.ac.jp</link>
<description>
This information shows the temperature and
humidity of the server room.
</description>
<language>ja</language>
<copyright>2005-2006 (C) Hiroki Kajiwara</copyright>
<pubDate>Thu, 30 Nov 2006 10:10:02 +0900
</pubDate>
<lastBuildDate>Thu, 30 Nov 2006 10:10:02 +0900
</lastBuildDate>
<item>
<title>Temperature and Humidity</title>
<description>23.28,34.8834
</description>
<guid isPermaLink="true">http://mikilab.doshisha.ac.jp/</guid>
<pubDate>Thu, 30 Nov 2006 10:10:02 +0900</pubDate>
</item>
...
</channel>

```

Fig. 5 RSS データフォーマットの例

MARS では、グリッド上の Web サービスのデータスキーマとして、この RSS データフォーマットをベースに XML Schema で定義する。

以上より、グリッドの相互認証を基盤とし、メッセージ通知機構により RSS を配信することで、セキュアかつ透過的、効率的にセンサ・データを配信することができる。また、グリッド上の Web サービスのデータフォーマットに RSS を利用することで、汎用的にセンサ・アプリケーションを開発することができ、開発者の労力を軽減することが期待できる。

3 MARS の概要

MARS のシステム概要を Fig. 6 に示す。MARS は主に、エンドユーザのポータルとなるポータルサーバ、プロキシ証明書のレポジトリとなる MyProxy サーバ、RSS を配信する複数の RSS サーバから構成され、各サーバには Globus Toolkit によるグリッド環境が構築されている。そのため、複数の RSS サーバがネットワーク上に地理的・組織的に分散していたとしても、同一のグリッド環境として存在することとなる。

次節より、MARS の処理の流れと各ユーザの役割、機能および設計について述べる。

3.1 処理の流れとユーザの役割

MARS の処理の流れを以下に示す。

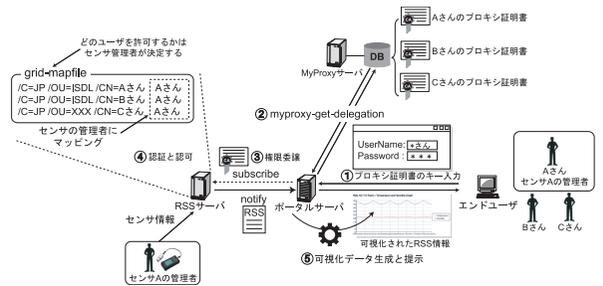


Fig. 6 MARS のシステム概要

1. プロキシ証明書のキー入力：
エンドユーザは、Web ブラウザを通じてポータルサーバに接続し、あらかじめ MyProxy サーバに登録しているプロキシ証明書のキーを入力する。
2. MyProxy からのプロキシ証明書の取得：
ポータルサーバは、要求のあったエンドユーザのプロキシ証明書を、MyProxy サーバから取得する。
3. プロキシ証明書の委譲と通知予約：
ポータルサーバは、エンドユーザのプロキシ証明書を RSS サーバに対して委譲する。その際、RSS サーバはエンドユーザのプロキシ証明書から、エンドユーザの認証を行い、要求のあった RSS データに対して、エンドユーザが認可されているかを判断する。そして、要求するリソースに対して通知予約 (subscribe) を行う。
4. RSS データの生成と通知：
RSS サーバは、定期的に更新されるセンサ情報から RSS データを生成する。新たに RSS が生成されると、通知予約をしていたポータルサーバの Web サービスに対して、通知 (notify) を行い RSS データの転送を行う。
5. RSS データの提示：
必要に応じて、Web サービスクライアントは RSS データの加工を行いエンドユーザに提示する。

Fig. 6 において、以下に示す各ユーザが MARS に関わることとなる。

- センサ管理者
- エンドユーザ
- ポータルサーバ管理者
- MyProxy サーバ管理者
- 第3者認証局管理者
- RSS サーバ管理者

通常、ある1つのセンサの管理者はエンドユーザにもなりえる。そのセンサ情報を扱えるエンドユーザを、センサ管理者が認可することとなる。RSSサーバは各地に分散することが想定され、1つのRSSサーバに複数のセンサ情報が登録されることも想定される。ポータルサーバ、MyProxyサーバ、第3者認証局はMARSの提供する枠組み上で、各管理者が運営することとなる。

3.2 機能および設計

3.2.1 プロキシ証明書の委譲

ポータルサーバから取得したエンドユーザのプロキシ証明書を、エンドユーザが指定するセンサ・リソースを保持するRSSサーバへ委譲するWebサービスを構築する。委譲を行うWebサービスを、コンポーネント化し、Webサービス開発者に提供する。

3.2.2 Webサービスによるメッセージ通知

ポータルサーバとRSSサーバ間で、メッセージ通知によるRSS配信を行うためのWebサービスを構築する。エンドユーザの指定するセンサ・リソースに対するメッセージ通知予約と、センサ・リソースがポータルサーバ側のWebサービスに状態変化を通知する、2つの機能である。

RSSサーバ側のWebサービスは、主にセンサ管理者が開発する必要がある。ポータルサーバ側のWebサービスは、センサ管理者もしくはエンドユーザが開発する。Webサービス開発の負荷を軽減させるための、統一データフォーマットとしてRSSフォーマットをベースとしたXML Schemaを提供する。

3.2.3 センサ情報からRSSデータへの変換

センサから観測したデータを、RSSデータに変換する必要ある。センサ情報からRSSデータを生成するプログラムは、各センサの管理者が作成する必要がある。

3.2.4 エンドユーザの認証とリソースへの認可

プロキシ証明書には、そのエンドユーザの所属や署名されたCAの情報が記述されている。RSSサーバは、ポータルサーバから委譲されたプロキシ証明書が、自信が信頼するCAが信頼(署名)したプロキシ証明書であるかを認証する。その後、エンドユーザが指定したセンサ・リソースに対して、センサ管理者(RSSサーバ上のローカルユーザ)が認可しているかを、Fig. 7に示すようにgrid-mapfileを用いて権限認可のマッピングを確認する。

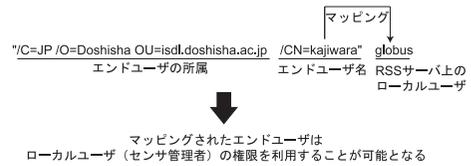


Fig. 7 grid-mapfileを用いた権限マッピング

4 まとめと今後の課題

本稿では、今後ますますの発展が想定されるセンサ・ネットワークの新たなプラットフォームとして、グリッドの有する相互認証とRSSによる標準データフォーマットを利用した情報配信システム、MARS (Mutual Authentication RSS, 相互認証を基盤とする未来型RSS配信システム)の開発について述べた。

MARSはグリッドの相互認証を基盤とすることで、相互認証や強固な通信の暗号化、組織間をまたいだSSO、エンドユーザごとにリソースへの権限を認可する仕組みを実現し、センサ情報をセキュアかつ透過的に配信することができる。そして、Webサービスを基盤とするサービス・グリッドの仕様のうち、WSRFとWSNに基づくメッセージ通知によるプッシュ型RSS配信により、効果的なセンサ・リソースの配信を実現する。また、Webサービス間のデータフォーマットとして、RSSフォーマットを利用することで、Webサービス間で統一にかつ容易にセンサ・リソースの利用が期待できる。

今後の課題としては、以下の事項が挙げられる。

- システムの実装：
Globus Toolkitを用いたシステムの実装と、RSSデータフォーマットのXML Schemaの設計を行う。

- 多種多様なセンサの利用：
より多くのセンサを用いた評価実験を行う。例えば人体情報（脈や心拍数，血圧など）や在籍情報（位置情報）といった，よりプライベートな情報をセンサすることを考える。
- キラー・アプリの創生：
MARSのセンサ・プラットフォームを利用することで，今までになかった新たな有益なサービスを創生する。

謝辞：MARSは，独立行政法人情報処理推進機構（IPA）の2006年度未踏ソフトウェア創造事業「未踏ユース」からの援助を受けて開発を行っている。

参考文献

- 1) Ian Foster Carl Kesselman and Tuecke Steven: *The anatomy of the grid : Enabling scalable, virtual organizations. International J. Supercomputer Applications 15(3)*, 2001.
- 2) National Research Grid Initiative (NAREGI) : <http://www.naregi.org/>
- 3) BioGrid: <http://biogrid.icm.edu.pl/>
- 4) Yoshio Tanaka, Hidemoto Nakada, Satoshi Sekiguchi, Toyotaro Suzumura, Satoshi Mat-suoka: Ninf-G: A Reference Implementation of RPC-based Programming Middleware for Grid Computing, *Journal of Grid Computing*, Vol. 1, No. 1, pp.41-51,2003.
- 5) M.J. Litzkow, M. Livny, and W. Mutka: Condor - a hunter of idle workstations, *Proceedings of the 8th International Conference on Distributed Computing Systems*, pp.101-111, 1988.
- 6) Foster,I. et.al.: Globus: A Metacomputing Infrastructure Toolkit, *International Journal of Supercomputer Applications*, Vol.11, No.2, pp.115-128, 1997.
- 7) The Globus Alliance: <http://www.globus.org/>
- 8) I. Foster, H. Kishimoto, A. Savva, D. Berry, A. Djaoui, A. Grimshaw, B. Horn, F. Maciel, F. Siebenlist, R. Subramaniam, J. Treadwell, and J. Von Reich. The Open Grid Services Architecture, Version 1.0. Open Grid Service ArchitectureWG, Global Grid Forum, 2005. [Online]. Available:<http://www.ggf.org/documents/GFD.30.pdf>.
- 9) K. Czajkowski, D. Ferguson, I. Foster, J. Frey, S. Graham, T. Maguire, D. Snelling, and S. Tuecke. From Open Grid Services Infrastructure to WS-Resource Framework : Refactoring & Evolution. 2004. [Online]. Available : http://www.globus.org/wsrf/specs/ogsi_to_wsrf_1.0.pdf.
- 10) K. Czajkowski, F. D. Ferguson, I. Foster, J. Frey, S. Graham, I. Sedukhin, D. Snelling, S. Tuecke, and W. Vambenepe. The WS-Resource Framework, Version 1.0. <http://www.oasis-open.org/committees/download.php/6796/ws-wsrf.pdf>, 2004.
- 11) Novotny, J., Tuecke, S. and Welch, V.: Initial Experiences with an Online Certificate Repository for the Grid: MyProxy, *Proceedings of the Tenth International Symposium on High Performance Distributed Computing (HPDC-10)*, IEEE Press (2001).