2U-1

分散環境におけるセキュアな協調作業のための グループ管理システム

百田 信 † 甲斐 啓文 †† 伊東 栄典 ‡

† 九州大学システム情報科学研究府情報理学専攻 †† 三菱電機 ‡ 九州大学情報基盤センター

1 はじめに

インターネットが拡大, 普及し, 世界規模のインフラとなった現在では, インターネットを利用した情報資源（リソース）の共有は, 新しい知見や意思決定, 強調問題解決, 製品やサービスの構築のための重要な手段である.

分散環境でのリソース共有を考えた場合, リソース提供者は利用者の本人性を確立する必要がある（利用者認証）. しかし, 認証に必要な情報やアクセス権限はリソースを提供するサーバ毎に管理されているのが現状である. そのため, 既存のシステムでは, 分散したリソースにまたがってアクセスする場合, サーバ毎に個別に認証を得なくてはならない. そこで一度の認証手続きで複数のサーバへのアクセスを可能にする仕組み（Single Sign-On）が必要とされている. また, リソース提供者と認証された利用者との間で決められたルールによって共有が制御されなければならない. そのような共有のルールによって定義された, リソース提供者と利用者の集合は, 仮想的なリソース共有領域を形成する. 本研究では, そのような仮想共有領域を,「グループ」と呼ぶ.

我々はこれまで利用者認証について調査してきた [1, 2, 3, 4]. 近年では, 数多くのオンライン認証システムの開発がなされており, その中で SSO 機能を備えたものも多い. 代表的なものに Internet2 による Shibboleth[2] が挙げられる. Shibboleth では, SAML （Security Assertion Markup Language）[5] をもとに利用者認証を集中的に行うサーバ Identity Provider（IdP）とのリソースを提供するサーバ Service Provider（SP）を切り分けることで SSO を実現している. しかし, 認可機構はうまく機能していない.

本研究では, ID Federation の実現を目的とし, WWW 上で必要に応じて動的にグループを生成し, 容易に管理できるようなシステム Group Management System (GMS) を設計・試作した.

2 Group Management System (GMS)

2.1 システム概要

Group Management System（GMS）の概要を図 1 に示す. GMS では, GMS サーバと SP の Web サーバに組み込んだ GMS プラグインによって分散環境における仮想的なグループの形成を実現している. SP は GMS プラグインを組み込むことで GMS サーバを信頼しリソースを提供することになる.

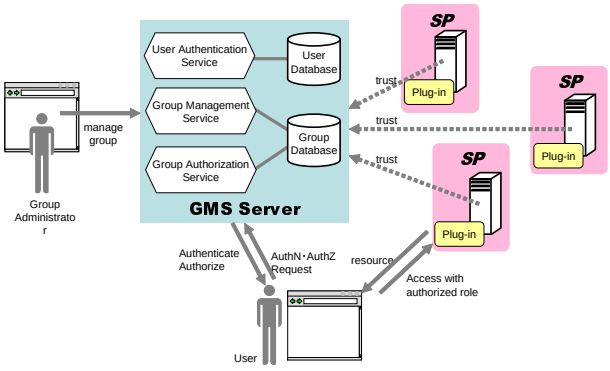

図 1: GMS Overview

2.2 GMS におけるグループ

本稿では, グループをインターネット上の仮想的なリソース共有領域として定義している. グループはメンバーとリソースからなり, メンバーは Web クライアントである利用者と一対一に対応し, グループ内におけるロール（Role）と呼ばれる属性を持つ. メンバーは各自与えられたロール属性でリソースにアクセスする. リソースは必要に応じて複数の Web サーバから提供される. 一方, グループはグループ管理者によって管理される. グループ管理者は, どのメンバーがどのロール権限でグループリソースが利用可能であるかといったグループのポリシーを決定する. 現実の組織と同様にグループの構成及びポリシーの変化は頻繁に起こると考えられる. GMS ではグループ管理に必要なグループ情報を柔軟で拡張性の高いデータ構造を持つ XML によって記述している.

Group Management System for secure distributed collaborative work

† Makoto MOMOTA (makoto.momota@i.kyushu-u.ac.jp)

†† Hirofumi KAI

‡ Eisuke ITO (itou@cc.kyushu-u.ac.jp)

Dept. of Informatics, ISEE, Kyushu Univ. (†)

Mitsubishi Electric Corporation (††)

Computing and Communications Center, Kyushu Univ. (‡)

2.3 アクセス制御機構

グループのリソースは, そのグループのメンバー以外の利用者によるアクセスから保護されなければならない. GMS では, 利用者の本人性と参加グループにおけるロールに基づいたアクセス機構を備えている. GMS では, グループリソースへアクセスするために, 利用者の身元を確認する利用者認証プロセス及びそのグループにおけるロールを付与するグループ認可プロセスを通過する必要がある. これらのプロセスを通過した利用者は SP 側で認可プロセスによって与えられたロール基づき, リソースへのアクセスを許可される. 認証・認可が既になされたかどうかは, セッションの有無に基づいて判断される. GMS では, 認証及び認可システムが独立した GMS サーバ上で稼動しており, SP とは分散しているため, GMS サーバと SP 側の両方で認証・認可セッションを生成するために SSO 機能を提供している.

2.4 処理の流れ

GMS の処理の流れを図 2 に示す. SSO は, 図 2 のように各コンポーネント間でメッセージをやりとりすることにより実現されている. リソースにアクセスした利用者はまず GMS サーバにリダイレクトされ利用者認証を求められる. これを通過すると, リソースへのアクセス権限を得るために, 今度はグループ認可のプロセスを要求される. そこで認可されると, 利用者認証と同様に GMS サーバとリソース側で認可セッションが生成されリソースへのアクセスが可能となる. 一度利用者認証を通過した利用者が複数のグループにまたがってアクセスする場合, グループ認可のみを要求され再び利用者認証を行う必要はない.

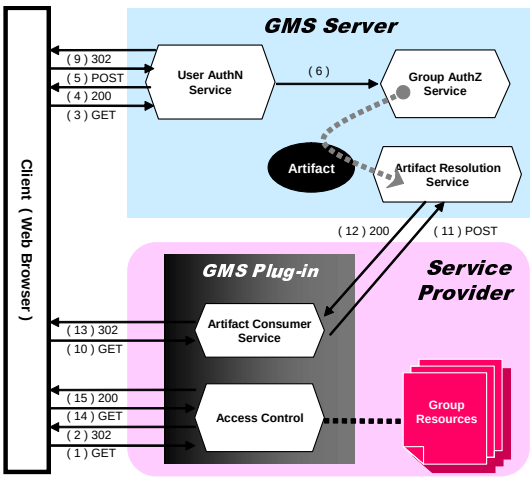

図 2: GMS Processing flow

3 GMS の試作

先述した GMS について試作システムを開発した. 試作システムはグループ管理及び認証・認可サービスを提供する GMS サーバと GMS サーバと連携するために SP へ組み込む GMS プラグインからなる. これらのプログラムはすべて Java で実装している.

GMS は SSO 機構とグループ管理機能を提供する. SSO 機構は SAML[5] の Browser/Artifact Profile を参考にし, SAML は用いず独自のデータ形式でデータのやりとりを行っている.

4 ３つの利用形態

具体的な情報システムで認可を実現するために GMS を適用する場合, SP 内蔵型, IdP 内蔵型, 独立型の 3 つの利用形態が考えられる.

SP 内蔵型は, リソースを保持する SP の中で, GMS システムを動作させる形態である. 新たに GMS サーバ機を必要としない. リソース管理者が一つの SP の中のリソースを認可したい場合には適している.

IdP 内蔵型は, GMS を IdP の中に内蔵する形態である. IdP の管理者と SP の管理者が同一人物であり, かつ, 管理する SP の運用ポリシーが均一な場合に適している.

独立型は, GMS を IdP と SP とは独立に運用する形態である. 柔軟な運用が可能になるが, 複雑な制御をするための管理コストが大きくなる.

5 おわりに

本稿では, 分散リソースを安全に共有するための GMS を開発した. 現状では GMS が IdP として機能しているが, 将来的には GMS から分離する予定である. 分散している複数の IdP の中からユーザが属する IdP を発見するためのサービスとして WAYF（Where Are You From ?）サービスも必要である. また, 今後は利用者の属性データを保持する Attribute Authority とも協調可能なシステムに拡張する予定である. .

参考文献

[1] のぎ田めぐみ, 笠原義晃, 伊東栄典, 鈴木孝彦 : “利用者認証に用いる識別子の決定方法に関する考察”, 信学技報 ISEC2006-112, pp.67-72, Dec.13, 2006.

[2] Shibboleth Project : http://shibboleth.internet2.edu/

[3] VOMS : http://hep-project-grid-scg.web.cern.ch/hep-project-grid-scg/voms.html

[4] Liberty Alliance Project : http://www.projectliberty.org/

[5] OASIS Security Services (SAML) TC : http://www.oasis-open.org/comittees/security/