M-067

安全なアドホック情報共有のための動的 ACL 設定方式
Dynamic Security Provisioning for Ad-Hoc Information Sharing

森田 陽一郎†，中江 政行†，小川 隆一†
Yoichiro MORITA, Masayuki NAKAE, Ryuichi OGAWA

1. はじめに

近年，部門内に限らず，複数の部門にまたがったプロジェクトや，拠点間での遠隔会議などの場面で，アドホックにファイルサーバなどを共有して，ユーザ同士で情報共有を行いたいというニーズが高まっている．

共有サーバは，FW (ファイアウォール) によって保護することが一般的である． FW の ACL (アクセスコントロールリスト) 管理は，一部のネットワーク管理者によって行われており，アドホックなサーバ共有アクセスに柔軟に対応することは難しい．

そこで，リソースにアクセスできるユーザグループを表す記述であるアクセス権ポリシとして，頻繁に生滅やメンバ変動を繰り返すユーザグループ (アドホックグループ) に対応できるポリシ (CBAC ポリシ) を定義し，このポリシに指定されたタイミングで FW の ACL 設定を変更する動的 ACL 設定方式を提案する．

本稿では，アドホックグループのアクセス権記述モデルである CBAC モデル，および本モデルに基づく動的 ACL 設定方式のアーキテクチャと動作について述べる．

2. CBAC モデル

ユーザグループに対するアクセス制御モデルとしては，RBAC (ロールベースアクセス制御) [1] が一般的である．RBAC では，ユーザ集合として定義される「ロール」に対して，リソースへのアクセス権を割り当てる． これを記述する RBAC ポリシは，ロールのユーザリストと，リソースの組である．

しかし，ロールの定義において，ユーザ集合の経時的変化が考慮されないため，メンバ構成が経時的に変化するアドホックグループのアクセス権を記述することが困難である． そこで，アドホックグループに対応したアクセス制御モデルとして，CBAC モデルを提案する． CBAC は，ユーザの所属・位置などのアドホックグループを特徴づけるユーザ属性 (コンテキスト) の条件を用いて，ロール定義 (ポリシ設定) を行うよう RBAC を拡張したものである (図 1).

ここで，ポリシには，リソースと，それを共有するユーザ集合の定義を記述するものとする． ユーザ集合の定義として，RBAC ポリシでは，ユーザリストを記述するが，CBAC ポリシでは，コンテキストを用いた条件 (その時その部屋にいるメンバ，など) を記述する．

ユーザ集合が変化する場合，RBAC では，ユーザ集合を再定義するため管理者によるポリシ変更が必要となるが，CBAC では，ユーザごとのコンテキスト情報を外部から受け取り，コンテキスト条件に合致するユーザリストを逐次更新すれば，ポリシを変更せずにアクセス制御の動的な再設定が 可能である．

†日本電気（株）インターネットシステム研究所，
Internet systems res. labs., NEC Corp.

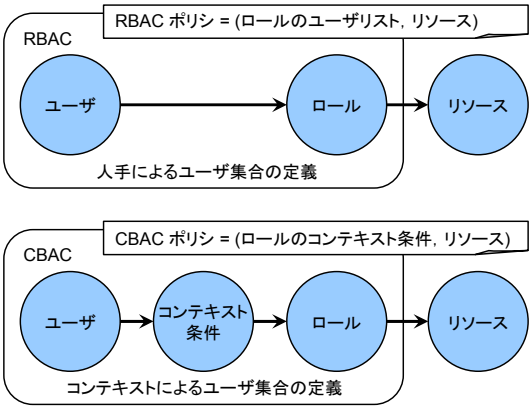

図 1： CBAC と RBAC の比較

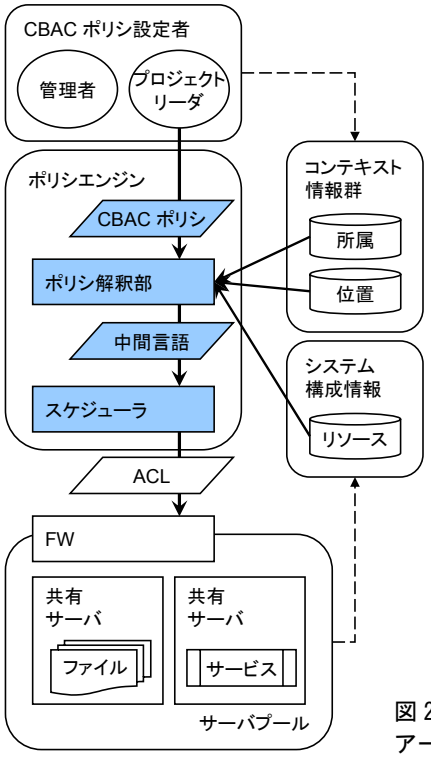

図 2: アーキテクチャ

3. 動的 ACL 設定方式

3.1.アーキテクチャ

図 2 に基本アーキテクチャを示す． ポリシエンジンは，共有リソースを保護する FW もしくは同等のアクセス制御機能に対して，管理者やプロジェクトリーダから与えられる CBAC ポリシに従って ACL の動的な生成・設定を行うことで，メンバ構成などの変化に追随して適切なアクセス制御を行う．

すなわち，コンテキスト情報群およびシステム情報から，適切なメンバ構成，ネットワーク上の位置，サーバアドレ

スを取得し，リソースの共有条件を記述した CBAC ポリシと合わせて，機種非依存の ACL 記述である中間言語を生成する． 次に，スケジューラによって，CBAC ポリシのコンテキスト条件で指定された時間通りに，中間言語から機種依存の ACL を生成し，FW に設定する．

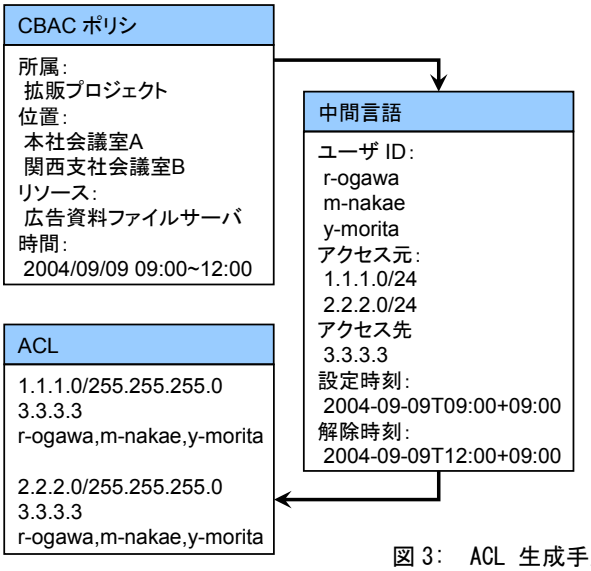

図 3： ACL 生成手順

3.2.CBAC ポリシ

CBAC ポリシ (図 3 左上) には，コンテキスト条件によるアドホックグループの定義と，そのアドホックグループに対してアクセスを許可するリソースの名称を記述する． コンテキスト条件には，ポリシ解釈部を起動すべきコンテキストの変化イベントを選択する機能と，ポリシ解釈で生成するユーザリストに含むユーザを選択する機能がある． 具体的なコンテキストとしては，以下を想定する．

- コンテキスト条件
  - ユーザの所属条件 (企業や団体，部門，役職，プロジェクトチーム，会議メンバなど)
  - ユーザの位置条件 (地域，建造物，部屋，座席など)
  - ユーザがアクセスを行う時間条件
- リソース
  - リソース名 (サービス，資料など)

3.3.ポリシ解釈部

入力された CBAC ポリシを読み，以下のような処理を行って，中間言語の内容決定・記述に必要な ACL パラメータを揃え，中間言語を生成し，スケジューラに受け渡す．

- コンテキスト情報群から，コンテキスト条件に合致する ACL パラメータを取得．
  - 所属条件から，ユーザ ID リストを取得．
  - 位置条件から，アクセス元のネットワークアドレスを取得．
- システム構成情報から ACL パラメータを取得．
  - リソース名から，アクセス先の IP アドレスを取得．
- ポリシの時間条件から，設定時刻と解除時刻を決定．

3.4.中間言語

中間言語 (図 3 右側) の役割として，ポリシ解釈と ACL 設定の時間的な非同期への対応と，多機種の FW への対応がある． 前者は，ポリシ解釈時に，すぐに ACL を生成・設定せず，中間言語をスケジューラに渡すことで実現している． 後者は，中間言語を XACML をベースにした汎用表現とし，特定の FW に依存しない形式で，具体的なアクセス制御ルールを記述することで実現している． 中間言語には，以下に挙げるようなアクセス制御ルールが，複数記述される．

- アクセス主体
  - ユーザ ID のリスト
  - アクセス元のネットワークアドレス
- アクセス対象
  - アクセス先の IP アドレス
- 条件
  - 設定時刻と解除時刻

3.5.スケジューラ

入力された中間言語の条件にある時刻通りに，アクセス主体とアクセス対象の内容から，機種依存形式の ACL (図 3 左下) を生成して FW に設定，あるいは解除を行う．ACL の生成には，XSLT による記述形式の変換を用いる．これにより，様々なアクセス制御機器に対応することが可能となる．

3.6.コンテキスト変化に伴う更新

動的更新の処理の流れは以下の通りである．

① コンテキスト情報群から，コンテキストの値に変化の有ったコンテキストの名称が通知される．
② ポリシエンジンが，変化したコンテキストをコンテキスト条件に持つ CBAC ポリシの再解釈を行う．

システム構成情報の変化についても，同様に通知を受け，更新を行う．

これにより，登録済みの CBAC ポリシを変更することなく，コンテキストの変化に応じて，必要な ACL だけを動的に再生成することができる． コンテキストを用いて，プロジェクトや遠隔会議などのアドホックグループを定義することで，メンバ構成が経時的に変化する場合でも，ユーザによるポリシ修正なしに動的にアクセス制御を変更することができる．

4. おわりに

アドホックグループのアクセス権を表現できる CBAC モデルについて述べた．また，CBAC に基づく動的 ACL 設定方式に関して，アーキテクチャと動作を述べた．

参考文献

[1] R. Sandhu et al., “Role-Based Access Control Models”, IEEE Computer, v. 29, n. 2, 1996, pp. 38-47.