Webページに対するケーパビリティを用いたアクセス制御

松井慧悟[†] 新城靖[††]
佐藤聡[††] 板野肯三[††]

1. はじめに

現在，多くのWebページに対するアクセス制御は，アクセスコントロールリストに基づいたものが使われている．アクセスコントロールリストとは，オブジェクトごとにあり，そのオブジェクトに対してどの主体がどの操作を行えるかを記述したものである．ここでのオブジェクトとはWebページであり，主体とは多くの場合別途管理されているユーザまたはそのグループである．Webページの管理者はWebページに対するアクセス制御を行うために，このリストに対する変更，および，ユーザ管理を行う．

Webページのアクセス制御にアクセスコントロールリストを用いた場合の問題点は，管理者への負担が重くなってしまうことである．負担とは具体的には，閲覧を許可するユーザを増やしたり，ユーザ情報の管理やアクセスコントロールリストへの変更を行うことである．

本研究は，ケーパビリティという概念をWebページのアクセス制御に用いることにより，この問題を解決する．ケーパビリティとはオブジェクトへの操作を行う権利を含んだ参照のことである．あるユーザがあるオブジェクトに対するケーパビリティを所持している場合，そのユーザはそのオブジェクトに対する権利を有することになる．ケーパビリティの性質として，ユーザが自由に他人に譲渡可能であることがあげられる．すなわち，ユーザが所持するケーパビリティを他人に譲渡すると，譲渡されたユーザも同様にそのオブジェクトに対する権利を有することになる．また，ケーパビリティの他の性質としては，あるケーパビリティからそれより弱い権利を持ったケーパビリティを作成可能な点があげられる．本研究では，ケーパビリティをWebページのアクセス制御に適応し，ユーザによるオブジェクトに対する権利の管理を可能にすることで，管理者の負担を軽減する．

2. プロキシによるアクセス制御の実現

Webページに対するケーパビリティを用いたアクセス制御を実現する方法としては，次の3つの方法が考えられる．

(1) HTTPを変更し，ケーパビリティを扱えるようにする．Webブラウザ，および，クライアントを拡張する．

(2) 現在のHTTPを変更せず，その上でケーパビリティの概念を実現する．このときに，Webサーバを変更をする．

(3) (2)においてWebサーバも変更せず，プロキシを用いて実現する．

3つ目のプロキシを用いる方法ならば，既存の技術に変更を加えず実現可能であり，複数のWebサーバで使えるので応用範囲が広い．よって本研究では，プロキシを用いた方法を採用する．

本研究で提案するシステムをユーザとWebサーバの間でユーザのアクセス制御を行うプロキシサーバ，ケーパビリティを管理するためのデータベース，そしてケーパビリティを発行するサーバから構成する(**図1**)．ケーパビリティは乱数により予測不可能な文字列を含むURLとして表現する．URLについて詳しくは3章で述べる．データベースには，ケーパビリティ，Webページの URL，Webサーバにアクセスするためのユーザ名とパスワード，アクセス回数，アクセス許可期間等の情報を格納しておく．Webサーバではアクセス制御の対象となるページをBasic認証やIPアドレス等を用いてプロキシにのみアクセス可能にする．

本研究では，次の2種類のユーザを扱う．

Webページの管理者：Webページの内容を作成する．初期ケーパビリティを作成する．

閲覧者：Webページを閲覧する．弱いケーパビリティを作成することもできる．

Webページの管理者は，アクセス制御を行いたいWebページを作成して，その URL を発行サーバに送る．このときにBasic認証のユーザ名やパスワード等のアクセス制御に必要な情報を同時に送る．発行サーバが受け取った情報を元にケーパビリティを作成する．そして渡された情報をケーパビリティをキーとしてデータベースに格納する．最後に，発行サーバが作成したケーパビリティを含むURLをWebページ

† 筑波大学第三学群情報学類
†† 筑波大学システム情報工学研究科コンピュータサイエンス専攻

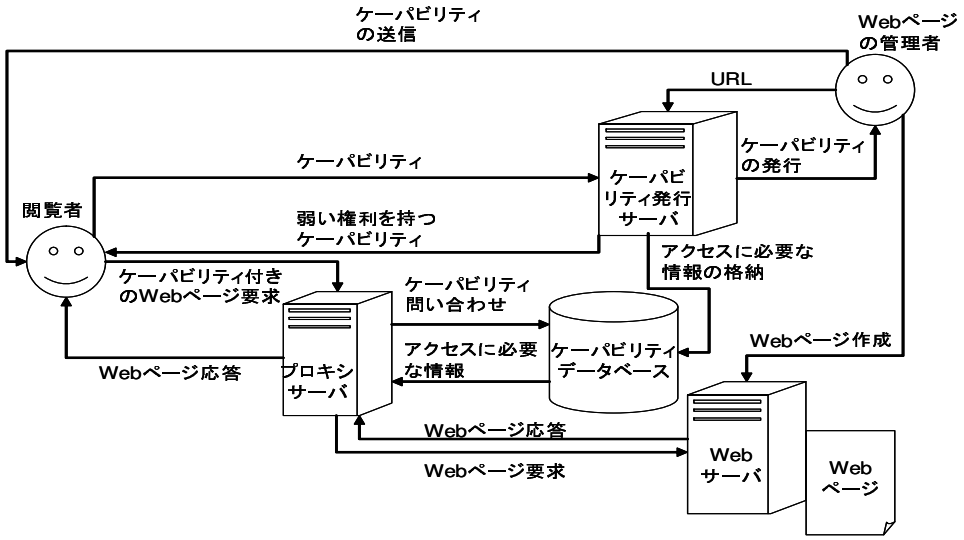

図 1 ケーパビリティを用いたアクセス制御のプロキシによる実現

の管理者に返す．Web ページの管理者はその URL をメールなどで配布して閲覧者に Web ページの閲覧を許可する．

閲覧者は Web ページを閲覧するときに自分が所持するケーパビリティを含む URL をプロキシに送信する．プロキシは送信されたケーパビリティをキーとしてデータベースに問い合わせを行う．そしてデータベースから閲覧できる Web ページ，閲覧のアクセス回数やアクセス許可期間等の情報を取得し，その情報をもとに要求された Web ページを閲覧させるか判断する．閲覧させる場合には Web ページを取得して転送する．

閲覧者は，弱いケーパビリティを作成したいときに自分が所持するケーパビリティをプロキシに送信する．弱いケーパビリティとはアクセス回数やアクセス許可期間が元のケーパビリティよりも制限されたケーパビリティである．プロキシは元のケーパビリティ，および，どのように制限を行うかの情報から弱いケーパビリティを作成して閲覧者に返す，

3. ケーパビリティを表す URL

ケーパビリティを表す URL は次のような形式になる．なお `random` は乱数により生成したケーパビリティを表す文字列である．

元の URL：

```
http://dom/dir/index.html
```

ケーパビリティを含む URL：

```
http://proxy/random/index.html
```

HTML では，内部リンクや画像を扱う必要がある．本研究では，まずケーパビリティを部分木に対して発行することとし，部分木内へのリンクや画像については自動的にアクセス可能にする．これを実装するためには，相対形式の URL に関しては，書き換えをせずにプロキシの URL に相対形式の URL をつなげてプロキシ自身へ要求が行くようにする．たとえば `index.html` のケーパビリティを持っていたとする．このケーパビリティは `http://proxy/random/index.html` と表現される．`index.html` に `a.html` への相対形式の URL があるとすると，Web ブラウザは，自動的に `http://proxy/random/a.html` と解釈する．プロキシは，`index.html` に加えて `a.html` についても中継を行う．

絶対形式の URL については Internet Archive[☆]が採用している方法を参考することを検討している．Internet Archive では過去の Web ページをアーカイブしており，リンク情報などを当時のページのまま保存している．過去のページのリンクを参照する場合，Internet Archive にある当時のページを参照するように JavaScript によりリンクの参照先を Internet Archive 内にあるページへ変更している．

4. おわりに

本研究では，プロキシを用いて Web ページに対するケーパビリティを用いたアクセス制御を実装する．プロキシ部分の実装については，Java サーブレットを用いている．また，Apache Jakarta Project[☆☆] の HttpClient を用いている．データベースは HSQLDB[☆☆☆] を使用している．今後の課題は，提案した機能を実装すること，および，安全にケーパビリティを受け渡すしくみを提供することである．

☆ http://www.archive.org/
☆☆ http://jakarta.apache.org/
☆☆☆ http://hsqldb.org/