Webアプリケーションおよび接続指向アプリケーションの両者に対応したシングルサインオンの実現

奥山 航平†、新城 靖††、板野 肯三††

†筑波大学理工学研究科、††筑波大学システム情報工学研究科

1 はじめに

近年、さまざまなサービスがWebアプリケーションとして提供されるようになった。Webアプリケーションで用いられているHTTPは、コネクションレスの通信である。Webアプリケーションに対して、POPなどセッションの開始から終了まで接続を張りっぱなしにするようなものを接続指向アプリケーションと呼ぶことにする。両者の間でシングルサインオンが可能になれば有用である。シングルサインオンとは、一度だけ利用者認証を済ませれば、他の許可されたサービスを再び利用者認証を行うことなく利用できるようになることである。

従来のKerberosを用いる方式では、Webブラウザが HTTPの Negotiation認証と Kerberosプロトコルに対応していなければならない。そこで、本研究では Web アプリケーションにおいて広く使われている HTTP の仕組みのみ（クッキーや Basic 認証）を用いて両者間でシングルサインオンを実現する。

2 関連研究

Kerberosは、ネットワークを用いた認証システムのである[1]。Kerberos では、認証サーバでユーザ名とパスワードによりユーザ認証を行い、TGT（Ticket-Granted-Ticket）と呼ばれる認証情報を生成する。以降、TGTを元に個々のアプリケーション毎のチケットを生成し、個々の認証を省略することができる。Apacheには、Kerberos V5に対応するためのモジュールがある。しかしながら、事前に入手した TGT を用いてシングルサインオンを行うためには、HTTP 上の Negotiation 認証が必要である。さらに、Web ブラウザも Negotiation認証と、Kerberosプロトコルまたは GSS-API に対応していなければならない。

3 Liberty プロトコルを用いた接続指向アプリケーション

本研究では、Liberty Alliance が提案するシングルサインオンの方式を接続指向アプリケーションにおいて利用可能にすることで、Webアプリケーションと接続指向アプリケーションの両者に対応したシングルサインオン を実現する。Liberty Allianceとは、Webアプリケーション間でのシングルサインオンの実現を目指すためのプロジェクトである[2]。Libertyでは、HTTPのクッキーを利用しており Negotiation 認証を利用していないので、現在多くのブラウザで利用できるという利点がある。

Liberty では利用者のアイデンティティ情報を生成、保管、管理するための特殊なWebアプリケーションをアイデンティティプロバイダ（IDP）と呼ぶ。一般のWebアプリケーションをサービスプロバイダ（SP）と呼ぶ。１つの IDP と複数の SP が信頼の輪を形成する。SP と IDP がアカウント情報を交換することを**連携（federation）**という。連携により一度 IDP で認証を行えば、信頼の輪の中にある SP ではシングルサインオンが実現される。また、IDP 同士を連携させることも可能である。

Liberty においてシングルサインオンを行うための認証プロセスは以下の通りである。Webブラウザで連携を行うと SP からクッキーが配布される。ユーザが連携を行った SP にログインしようとする時に、Webブラウザは SP にクッキーを送る。SP はそれを受け取ると、連携を行った IDP へのリダイレクト応答を返す。するとWebブラウザには IDP へのログイン画面が表示される。ここで IDP のアカウント を用いて認証を行うと Liberty プロファイルを含む SP へのリダイレクト応答を返す。Liberty プロファイルとは、認証が完了したことを示すセッション 情報を含む SAML 形式の文書である。以降は信頼の輪の中の SP、IDP であれば認証は必要ない。

本研究を用いた認証のシナリオを図 1 に示す。本研究では、Webアプリケーションと接続指向アプリケーションの両者で認証サービスを提供するための特別な SP を動作させる。これを、Gateway SP と呼ぶことにする。また、認証に用いる IDP と Gateway SP の間で個々のユーザに対応するアカウントに対して連携を行っておく。

3.1 システムへのログイン

利用者は、あるシステムにログインする時に、ログインプログラムにユーザ名とパスワードを入力する。ログインプログラムは、入力されたユーザ名とパスワードを IDP に送る。この時に、HTTP

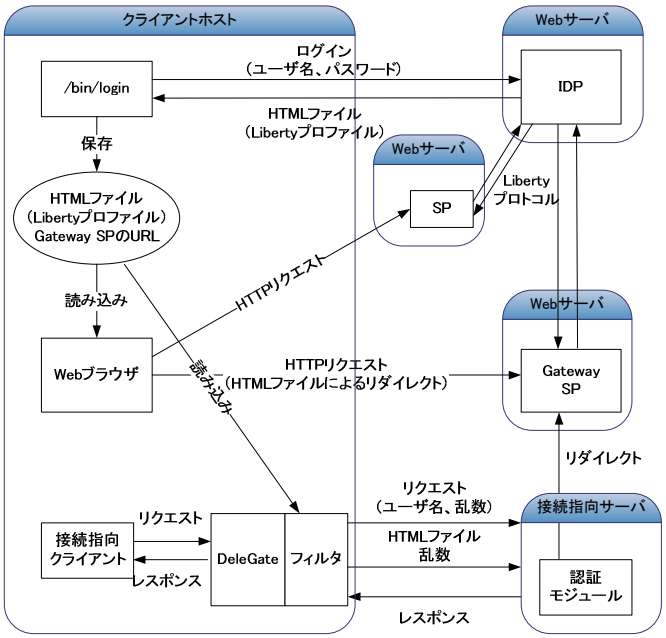

図 1　本研究を用いた認証

リクエストに Gateway SP の URL を含めて、あたかも Web ブラウザが Gateway SP を通ってアクセスしたようにふるまう。IDP は、認証が成功したら認証されたユーザの Liberty プロファイルと Gateway SP の URL を HTML 文書として応答を返す。ログインプログラム では、この HTML 文書をファイルに保存しておき、ユーザのログインを完了させる。

3.2　Web アプリケーションの利用

ログイン後に認証が必要な Web アプリケーションを利用する時は、利用者は Web ブラウザを起動して、まずログイン プログラム が保存した HTML ファイルを読む。すると JavaScript によりリダイレクトと同様の動作が行われ、自動的に Gateway SP へ Liberty プロファイルが送られる。これにより Gateway SP と IDP の間でセッション情報の確認が行われる。正しく確認が行われると、Web ブラウザは Gateway SP にログインした状態となり、以降認証が必要な Web アプリケーションは、シングルサインオンが利用できるようになる。

3.3　接続指向アプリケーションの利用

接続指向アプリケーションでは、通常クライアントがユーザ名とパスワードをサーバに送信することで認証を行う。本研究の方式では、パスワードの代わりに、IDP から返信された Liberty プロファイルを送る。これを実現するために、プロキシサーバ（DeleGate）のフィルタ機能を利用する。

フィルタでは、クライアントから送信されたリクエストに含まれるパスワードを乱数に書き換える。ここで、接続指向アプリケーションのサーバともう 1 本接続を張るようにし、同じ乱数とログインプログラムが保存した HTML 文書を送信する。サーバでは、乱数と HTML 文書を受け取る。ここで通常の接続から送られた乱数と比較し、同じであれば、HTML 文書から Liberty プロファイルを取り出し、SP に送信する。SP では Liberty プロファイルを用いて IDP と通信を行い、それが適切であれば、成功を示す応答を返す。それを受け取った接続指向アプリケーションのサーバは利用者認証が完了したとみなす。

フィルタはプロトコル 1 つにつき 1 つ必要である。これにより、クライアントプログラムは変更する必要がなくなる。

4　セキュリティ

本研究で提案する認証方式に対する脅威として、まずネットワークの盗聴によりパスワードが盗まれることが挙げられる。これは SSL 等を用いて、通信を暗号化することで対応できる。また、返ってきた HTML 文書は、ログインしようとしているユーザにしか読むことのできないパーミッションでファイルに保存する。これにより、TGT をファイルに保存する Kerberos と同程度の安全性が得られる。

5　おわりに

本論文では、Web アプリケーションおよび接続指向アプリケーションの両方に対応したシングルサインオンを実現するための利用者認証方式を提案した。その方法として、Web アプリケーション間でシングルサインオン を可能にする Liberty Alliance の方式を接続指向アプリケーションに拡張する方法を用いる。現在、ログインプログラムと POP、FTP の DeleGate 用フィルタを実装した。今後の課題は、対応できる接続指向プロトコルを増やすことと、GSS-API へ対応させることと評価を行うことである。

参考文献

[1] J. Kohl,C. Neuman：The Kerberos Network Authentication Service (V5), RFC 1510, 1993.

[2] Liberty Alliance Project, http://www.projectliberty.org/, 2002.