

M-036

IP ストレージに対する Web ベースのアクセスモデルの提案 Proposal of Web-based Access Model for IP-Storage

武田 裕子[†]
Yuko Takeda

小口 正人[†]
Masato Oguchi

1. はじめに

クライアントが、Web サーバのストレージに保持されているデータへアクセスする際には、基本的にブラウザのみでアクセスが行われる。まずブラウザから Web サーバへ HTTP(HyperText Transfer Protocol) を用いて要求が送られ、Web サーバが要求されたコンテンツをストレージから取り出し、クライアントに結果を返す。アプリケーションの実行はアプリケーションサーバ上のみで行われ、クライアントは結果を受け取って表示するのみである。基本的にすべてのデータもサーバ内のストレージに格納される。サーバマシンとストレージ間の接続には一般に SCSI が利用され、1 台のサーバマシンに 1 台もしくは複数台のストレージが接続される。

近年ではサーバにおけるストレージ接続に、SAN(Storage Area Network) を用いることが多くなってきた。特に TCP/IP ネットワーク上で SCSI コマンドを転送する SAN である iSCSI で接続される場合も増えてきた。

SAN を利用することにより、コンピュータとストレージを N 対 N で接続することができる。しかし SAN の利用はサーバサイト内に限定されており、オープンな環境における利用は考えられていない。そこで本研究では、IP-SAN に対しインターネットのオープン環境においてアクセスできる仕組みを考え、ストレージを自由に配置できるようにする。これによりユーザの状況に応じた柔軟な IP ストレージ利用が可能となる。これを実現するためには、IP ストレージを遠隔より管理したり、クライアント認証などのアクセス制御を行ったりするメカニズムが必要になる。本研究ではそのような機能を持ち、ユーザのブラウザからアクセスして利用可能であるサーバモデルを提案し、その実装を行った。

2. 提案モデルとその特徴

本研究では、Web サーバやストレージアクセスドライバの機能を持つ IP ストレージ・アクセスコントロールサーバ(以降はアクセスサーバと呼ぶ)をオープンなインターネット環境に設置することを提案する。本モデルは、アクセスサーバがブラウザとストレージを仲介し、ブラウザへは HTTP、ストレージへは iSCSI を用いて接続する。提案モデルの概念図を図 1 に示す。

提案モデルは、アクセスサーバにおいてストレージへのクライアントからの Web ベースのアクセスを制御し、HTTP、iSCSI 共にオープンなインターネット環境を利用するという点に特徴がある。これによりアクセス制御をアクセスサーバで一貫して行うことができ、IP ストレージを柔軟に構築して利用することが可能となる。

具体的な実現方法としては、例えばアクセスサーバを

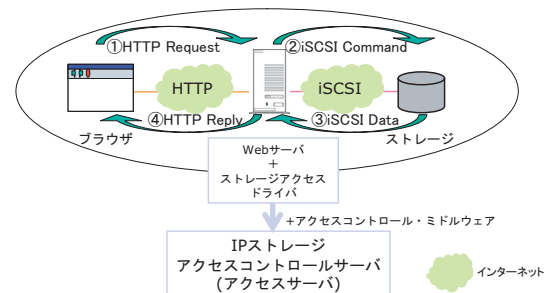


図 1: 提案モデル

LAN 上に設置し、LAN 上のクライアントのブラウザからアクセスする形が考えられる。この提案モデルの実現例を図 2 に示す。この場合ストレージは WAN を介した遠隔地に存在することも可能であり、ストレージへのアクセス制御はアクセスサーバが対応する。クライアントはストレージの位置を気にすることなく、自分のコンテンツにアクセスすることができる。

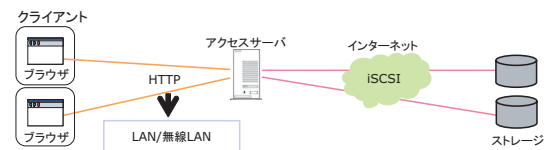


図 2: 提案モデルの実現例

また例えばアクセスサーバを無線 LAN 上に設置することにより、クライアントはモバイル環境において移動先でアクセスサーバに接続し、そのサイトまたは遠隔地のストレージ利用のサービス提供を受けることができる。このようにユーザの利用シーンに応じて柔軟な構成の IP ストレージを提供できる。

3. 実験環境

本研究では 3 台のマシンをクライアント、アクセスサーバ、IP ストレージとして用い、提案手法を実装した。これを図 3 に示す。これらのマシンは CPU として Intel Xeon 2.4GHz を用い、512MB のメインメモリ、36GB の SCSI ドライブ、Intel PRO/1000XT の Gigabit Ethernet アダプタを持つ PC サーバである。OS は Linux2.4.18-3 を使用した。アプリケーションサーバとして Tomcat5.5.4 を利用し、Tomcat を実行するための Java には J2SE5.0 を用いた。また、iSCSI は UNH IOL reference implementation ver.3[1] を使用した。

Tomcat は Jakarta プロジェクトのサブプロジェクトとして開発されているオープンソースのソフトウェアである。サーブレットや JSP を処理するアプリケーションサーバであり、単独で Web サーバとして動作すること

[†]お茶の水女子大学, Ochanomizu University

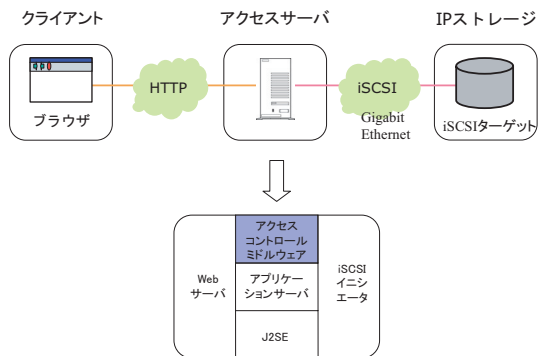


図 3: 実装環境とアクセスサーバの構成

も可能である。本研究においても Tomcat の Web サーバをそのまま使用している。サーブレットは、Java プログラミング言語で CGI プログラムに相当する Web アプリケーションを動作させるためのサービスおよびそのプログラムである [2][3][4]。

4. 提案モデルの実装

本節ではアクセスサーバの実装の詳細を説明する。図 3 にはアクセスサーバの構成図も示されている。本研究で主に実装した部分は、図中に示されるアクセスコントロール・ミドルウェアである。本実装は、以下に示す機能を実現する。

4.1 ブラウザから IP ストレージアクセスリクエストを受付

クライアントはブラウザからアクセスサーバへ HTTP を用いて接続する。アクセスサーバは、Web サーバにおいてリクエストを受付け、アプリケーションサーバへリクエストを送り、アクセスコントロール・ミドルウェアのサーブレットを起動する。

4.2 ユーザ認証を行いストレージのアクセス制御

クライアントに対し、ストレージ利用に必要なアクセス認証を行う。認証をさせたいページに移動するときはログインページを表示し、入力されたユーザ名とパスワードが一致すれば移動できる。一致しない場合にはエラーページを提示する。実装には Tomcat の FORM 認証を使用した。

4.3 ブラウザ上の操作により IP ストレージへ接続

図 3 に示されるように、アクセスコントロール・ミドルウェアの背後には iSCSI イニシエータ（ストレージアクセスドライバ）が存在する。クライアントのブラウザからリクエストを送ると、これにより起動されたミドルウェアが iSCSI イニシエータに対するコマンドを発生し、iSCSI ターゲットへの接続を行う。iSCSI の起動や終了などをクライアントのブラウザから行うことができる。

4.4 ブラウザからストレージ上のファイル进行操作

iSCSI が起動されると、クライアントはブラウザから IP ストレージ上のファイルへアクセスすることができるようになる。ミドルウェアには、ファイル操作の機能モジュールも含まれている。これによりクライアントは、

ブラウザから iSCSI を利用して IP ストレージ上のファイルの作成・削除などを行うことができる。

4.5 ブラウザから UNIX コマンドの実行

アクセスサーバにおいて UNIX コマンドを実行することにより、IP ストレージに対して高度な処理を行うことが可能となる。ブラウザから UNIX コマンドを入力すると、アクセスサーバから実行され、その結果がブラウザに表示される。

ブラウザから UNIX コマンドを実行するためのページの表示例を図 4 に示す。これは IP ストレージ上のファイル一覧を表示させた結果である。このようにクライアントのブラウザ内に入力することにより、アクセスサーバ上でコマンドが実行され、IP ストレージのデータが処理される。出力はブラウザに返される。

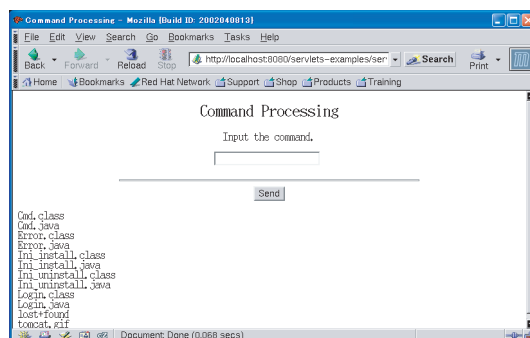


図 4: UNIX コマンド処理ページの表示例

5. まとめと今後の課題

本研究では、クライアントが IP ストレージに Web ベースでアクセスする際のモデルを提案し、アクセスサーバの実装を行った。これまで SAN はサーバサイト内に限定されてきたが、オープンなインターネット環境で利用可能とすることにより、柔軟に IP ストレージシステムを構築することができる。また、アクセスサーバにおいて IP ストレージへの Web ベースのアクセスをコントロールをすることで、アクセス制御を一貫して行うことができる。

今後はアクセスサーバにさまざまな機能を付加する予定である。さらにモバイル環境など異なる環境においても提案モデルを動作させ、その有効性を実証する。

参考文献

- [1] Inter Operability Lab, the Univ. of New Hampshire, <http://www.iol.unh.edu/consortiums/iscsi>
- [2] <http://www.hellohiro.com/servlet.htm>
- [3] 山本秀樹: Servlet & JSP 入門, CQ 出版社
- [4] 原田洋子: Java Servlet, 秀和システム