WebDAV システムのセキュアな設定・運用に関する調査

調査報告書

(設定・運用ガイド編)

平成 15 年 4 月

情報処理振興事業協会

セキュリティセンター

- Microsoft、MS、MS-DOS、Windows、Windows NT は、米国 Microsoft Corporation の米国および他の国における登録商標である。
- UNIX は、X/Open Company Limited が独占的にライセンスしている米国および他の国における登録商標である。
- Java 及びすべての Java 関連の商標及びロゴは、米国および他の国における米国 Sun Microsystems,Inc.の商標または登録商標である。
- その他のシステム名、アプリケーション名、製品名、会社名は、各社の商標または登録商標である。
- 本調査報告においては、(TM)、(C)、(R)などの記号は省略している。

1. 本編の概要

本調査では、セキュアな WebDAV システムについて、その利用を促進するために設定方法を総括する調査を行うとともに、ソフトウエア開発が必要と考えられるアクセス権管理機能を中心としたフィジビリティスタディ等を行い、今後の技術開発の必要性についての検討を示す。[1]

「設定・運用ガイド編」では、WebDAV システムの設定および運用手法に注目して調査および検討を行った結果を示す。以下の項目について調査を行った。:

- セキュアな WebDAV 運用モデル
  先に「現状調査編」において WebDAV 関連仕様の調査により示されたセキュリティ実現上のポイントについて検討を加え、セキュアな WebDAV 運用モデルを定義する。

- セキュアな WebDAV サーバの導入
  前項で定義したセキュアな WebDAV 運用モデルを、各種のサーバソフトウエアに適用する手法を示す。

- アクセスコントロールモデル
  仕様と各種実装に関する調査結果について検討を行い、WebDAV システムにおける適切なアクセスコントロールモデルを定義する。

- アクセスコントロールモデルの導入
  前項で定義したアクセスコントロールモデルを各種のサーバソフトウエアに適用する手法を示す。

- WebDAV クライアントの導入
  各種の WebDAV クライアントソフトウエアを適切に設定する方法を示す。

[1] 本調査は、調査主体である情報処理振興事業協会セキュリティセンター（IPA/ISEC）が、株式会社 SRA 先端技術研究所に委託して平成 14 年度に実施した。

■ 相互運用可能性

各サーバソフトウエアとクライアントソフトウエアの相互運用可能性に関する調査結果を示す。さらに日本語を含むリソースの扱いに関して、調査結果および仕様に基づき検討を行う。

■ 運用上の注意

前述したモデル設定、適用手法、および相互運用可能性調査の結果を基に、WebDAV システムをセキュアに運用するための注意点を提示する。

■ セキュア通信機構

SSL/TLS 以外のセキュア通信機構を WebDAV システムに導入する方法を示し、SSL/TLS を用いてセキュアな WebDAV システムを構築した場合と比較する。

2. セキュアな WebDAV 運用モデル

この章では、WebDAV システムをセキュアに運用するために必要な要素を整理し、運用モデルの定義を示す。

2.1. 定義

ここでは「セキュアな WebDAV システム」を以下のように定義する。

- クライアントとサーバ間で送受信されるデータが外部に漏洩しない WebDAV システム

運用モデルにおいて想定する利用形態は以下のようなものとする。

- WebDAV サーバとクライアントから構成される
- クライアントは WebDAV クライアントと HTTP/1.1 クライアントの双方を対象とする
- ファイアウオール、VPN 等は想定しない

2.2. 検討事項

WebDAV システムをセキュアに運用する上で検討すべき事項を以下に挙げる。

（1） 認証の手段

仕様から明らかであるように、WebDAV 自身は認証メカニズムを提供していないため、適切な認証の手段を必要とする。http には 2 種の認証メカニズムが存在するが、Basic 認証は単にパスワードを Base64 エンコードしただけで送信する問題があり、また Digest 認証はサポート状況に難がある。

（2） 通信路の暗号化

WebDAV における通信は http 上で行われるが、http の通信路は暗号化されていない。そのためネットワークでやり取りされるファイルの内容などが盗聴される可能性がある。また Basic 認証のパスワードや、リソースのプロパティなども盗聴される可能性については RFC2518 のクライアント認証に関する記述においても指摘されている。

通信路を暗号化する方式にはいくつかの方法が考えられる。SSL/TLS は現在 WWW において最も標準的に用いられている通信路暗号化方式であり、多くの Web ブラウザと WebDAV クライアントが対応している。またサーバソフトウエアにおいては、Apache 2、IIS 5.0、OS X の WebDAV サーバ機能も SSL に対応している。

その他には IPsec による暗号化、ssh によるポートフォワーディングを用いた暗号化当の方法がある。これらについては別途詳細を述べる。

（3） サーバ実装によるセキュリティ設定上の相違

各サーバの実装により、SSL の有効化／無効化を制御可能な単位が異なる。Apache 2 においては、SSL はサーバ（Virtual host）単位で制御される。サーバの空間にどのディレクトリを置くかによってコンテンツ単位の SSL 化を指定できる。

IIS では仮想ディレクトリ単位で SSL 機能が制御される。このため特定のディレクトリについてのみ SSL を有効にする指定が容易に行えるが、同名のエイリアスを SSL 有効と無効の両方の設定で運用できない。

2.3. 運用モデル

上の検討を踏まえて、セキュアな WebDAV システムの運用モデルを記述していく基本方針を示す。

- コンテンツの作成は、WebDAV を利用して行う。SSL/TLS を用いて暗号化された通信路においてのみ WebDAV を有効化することで、通信路からのデータの漏洩を防ぐ。認証は、暗号化された通信路の上で汎用性のある Basic 認証を利用して行う。

- 一般に向けたコンテンツの公開は、WebDAV を無効化し認証無しで行う。暗号化されていない通信路を使う。

3. セキュアな WebDAV サーバの導入

ここでは、各種の WebDAV サーバをセキュアに導入する手法を示す。

3.1. Apache 2.0

この項目では、Apache を用いた WebDAV サーバの導入方法を示す。使用するバージョンは WebDAV を標準に搭載している Apache バージョン 2 とする。ここで対象とする OS は Red Hat Linux 8.0 と Debian GNU/Linux unstable である。

WebDAV の設定の前に、これらの OS のインストールとパッケージ取得の設定は既に終わっているものとする。

3.1.1. Apache 2.0 のバージョン

Apache のインストールは、それぞれバイナリパッケージを用いて行う。各ディストリビューションに含まれる Apache のバージョンを以下に示す。

表 3-1　各ディストリビューションに含まれる Apache のバージョン

ディストリビューション	パッケージバージョン	Apache バージョン
Red Hat 8.0	httpd-2.0.40-11	Apache-2.0.40
Debian	apache2-common- 2.0.43-1 apache2-mpm-worker 2.0.43-1	Apache-2.0.43

3.1.2. WebDAV 設定の方法

Apache では、SSL 関連の設定を ssl.conf で一括して行うことが一般的である。ssl.conf 内の設定は SSL 環境にのみ適用される。本モデルでは、これを利用して ssl.conf 内のみに WebDAV 関連の設定を閉じることで、WebDAV 機能の明確な有効化・無効化を実現する。

3.1.3. Red Hat へのインストール

一般に Red Hat 8.0 環境における Apache の基本設定は、GUI ベースの設定ツールを使用して行うが、この設定ツールでは WebDAV 機能は制御できない。

WebDAV 機能の設定のためには定義ファイルを直接編集する必要がある。設定ツールを用いずに httpd.conf を編集した場合には、後に基本設定用の GUI ベースの設定ツールを使った際に以前の設定内容が上書きされて消されてしまう。

本モデルでは WebDAV 関連の設定を ssl.conf のみに行い、httpd.conf に変更を加えないので、基本設定については GUI ベースのツールを併用できる。

以下、Red Hat へのインストールを示す。

（1） パッケージのインストール

Red Hat 8.0 で採用されている Apache はバージョン 2 系統のものである。パッケージは OS インストール時に同時にインストールするか、CD に収録されているパッケージから redhat-config-packages コマンドを使ってインストール可能である。

セキュアな WebDAV サーバを構築するためには、以下のパッケージをインストールする必要がある。

- httpd
- mod_ssl
- openssl
- redhat-config-httpd

調査時点で最新の httpd パッケージは、httpd-2.0.40-11 である。インストール後は、up2date コマンドを用いてパッケージを最新の状態にすることができる。

（2） 証明書の設定

SSL に必要な証明書の設定を行う。本稿では詳細な手順については取りあげない。レッドハット社のドキュメント[2]を参照して作業を行う。

（3） 基礎設定

設定ツール redhat-config-httpd を用いて基礎設定を行う。最低限必要な設定を以下に示す。

- メインタブ
  - サーバ名
    サーバ名を FQDN（完全修飾名）で入力する
  - 管理者メールアドレス
    管理者のメールアドレスを入力する
  - 使えるアドレス
    全アドレスのポート 80

- 仮想ホストタブ
  デフォルトの仮想ホストに対して以下の編集をする
  - SSL
    「SSL サポートを有効」をチェックする

（4） WebDAV ディレクトリの作成

WebDAV コンテンツを収納するディレクトリを作成し、Apache のプロセスがアクセス可能にする。ここでは/home/webdav とする。

```
# mkdir /home/webdav
# chown apache:apache /home/webdav
```

（5） WebDAV の有効化

作成したディレクトリは、/etc/httpd/conf.d/ssl.conf において、VirtualHost ディレクティブの中で以下のように設定し公開する。

---

[2] レッドハット社のドキュメント：
http://www.redhat.co.jp/manual/Doc80/RH-DOCS/rhl-cg-ja-8.0/ch-httpd-secure-server.html

```
<VirtualHost _default_:443>
        （中略）
<Directory "/home/webdav/">
 Dav on
</Directory>
Alias /webdav “/home/webdav”
        （中略）
</VirtualHost>
```

Directory ディレクティブの中で、Dav on を指定することにより、そのディレクトリに対する https を経由した WebDAV アクセスが有効になる。上の例では、さらに Alias ディレクティブにより、WebDAV ディレクトリを/webdav に位置付けている。

（6） WebDAV ディレクトリの http による公開

HTTP/1.1 における WebDAV ディレクトリの公開の方法を以下に示す。

redhat-config-httpd では、Alias ディレクティブを使った任意のディレクトリの Web 空間へのマウントが行えない。このため、シンボリックリンクを用いる。

```
# ln –s /home/webdav /var/www/html
```

これにより、/（ルート）と同様な設定に基づく http アクセスが可能になる。なお、ルートにおいて SymLinksIfOwnerMatch をチェックし、Apache のユーザがオーナーであるシンボリックリンク以外の適用を除外しておくことが望ましい。

3.1.4. Debian へのインストール

Debian へのインストール手順を以下に示す。

調査時点（2003 年 2 月）では Debian における Apache 2 のサポートは正式なものではないため、以下の記述の詳細については今後変更される可能性もある。

（1） パッケージリストの更新

インストールに先立ち、apt-get update コマンドを使用して、パッケージリストを最新版に更新する。パッケージ入手先の設定は適切にされているものとする。

```
# apt-get update
```

（2） パッケージのインストール

apt コマンドを使用し、apache2-common パッケージと apache2-mpm-prefork パッケージをインストールする。調査時点の最新バージョンは 2.0.43-1 である。

```
# apt-get install apache2-common
# apt-get install apache2-mpm-prefork
```

パッケージの依存関係に応じて関連するパッケージもインストールされる。インストール後のデフォルトの状態で、Apache の各設定ファイルは/etc/apache2 以下に置かれている。デフォルトの設定ファイルでは/etc/apache2/mods-available 以下にモジュールの設定ファイルが置かれ、これを/etc/apache2/mods-enable からリンクすることで Apache の起動時にロードされるようになっている。

（3） 証明書の作成

以下の手順で証明書を作成する。まずサーバ鍵と証明書要求の作成を行う。

```
# mkdir /etc/apache2/ssl
# openssl genrsa –des 1024 > /etc/apache2/ssl/server.key
# openssl req –new –key /etc/apache2/ssl/server.key –out /etc/apache2/ssl/server.csr
        所在地情報、管理者メールアドレスなどを入力する
```

server.csr を CA に送信するか、自己署名する場合は、続けて以下を実行する。

```
# openssl req –new –key /etc/apache2/ssl/server.key –x509 –days 365
–out /etc/apache2/ssl/server.crt
        所在地情報、管理者メールアドレスなどを入力する
```

SSL の有効化の設定を行う。/etc/apache2/mods-available/ssl.conf に対して以下のように追加する。

```
Listen 443
<VirtualHost *>
SSLEngine on
SSLCertificateFile /etc/apache2/ssl/server.crt
SSLCertificateKeyFile /etc/apache2/ssl/server.key
CustomLog /var/log/apache2/ssl_access.log combined
</VirtualHost>
```

（4） 基礎設定

Debian の Apache 2 パッケージにおいては、サイトの基本的な設定は http.conf と /etc/apache2/site-available/default で行う。

最低限の設定として、httpd.conf に以下の記述を追加する。サーバ名は例としてあげたものであり実際は異なる。

```
ServerName www.example.com
```

また、以下を実行してロックデータベースのパーミッションを与えること。

```
# chown www-data:www-data /var/lock/apache2
```

（5） WebDAV ディレクトリの作成

WebDAV コンテンツを収納するディレクトリを作成し、Apache のプロセスによるアクセスを可能に設定する。ここでは作成するディレクトリを/home/webdav とする。

```
# mkdir /home/webdav
# chown www-data:www-data /home/webdav
```

（６） WebDAV の有効化

作成したディレクトリは、/etc/apache2/mods-available/ssl.conf において、定義した VirtualHost ディレクティブの中に以下のような Directory ディレクティブを設定し、Alias で/webdav に位置付ける。

```
<Directory "/var/www/webdav/">
 Dav on
</Directory>
Alias /webdav "/var/www/webdav"
```

Directory ディレクティブの中で、Dav on と指定することにより、そのディレクトリに対する WebDAV のアクセスが有効になる。

（７） WebDAV ディレクトリの http による公開

作成したディレクトリは、/etc/apache2/sites-available/default において、VirtualHost ディレクティブの中で以下のように設定し公開する。

```
<Directory "/var/www/webdav/">
 Dav off
</Directory>
Alias /webdav "/var/www/webdav"
```

3.2. IIS 5.0

ここでは、IIS を用いて WebDAV サーバを導入する方法を示す。

IIS はマイクロソフト社の Windows 環境において標準的に用いられる Web サーバソフトウエアである。IIS には多くのバージョンが存在するが、ここでは Windows 2000 Server に搭載される IIS 5.0 を対象とする。

3.2.1. Windows 2000 Server のインストール

IIS 5.0 のインストールについては、OS のインストール時に同時にインストールする方法と、OS のインストール後に OS とは独立にインストールする方法がある。的確なセキュリティ設定を実現する上では OS とは独立に Web サーバをインストールすることが望ましい。OS インストール時にはインストールされるコンポーネントから IIS を除外しておくこと。

また、セキュリティ上の理由から、Web で公開するコンテンツを格納するディスクパーティションはシステムとは別のパーティションにすることが望ましい。これは不正なリクエストによるシステムファイルの参照を防ぐための一般的な措置であるが、WebDAV システムにおいても当てはまる。ファイルの書き込みを行う WebDAV ではディスク空間の消費が問題となることも考えられるため、WebDAV 用の空間は専用のパーティションとして独立させるべきである。アクセス制御を実現するために WebDAV 空間は必ず NTFS でフォーマットする。

Windows のインストールは CD より行うことになるが、リリース後に発見された複数のセキュリティ上の問題を解決するために、サービスパックや Hot Fix と呼ばれる修正プログラムを適用する必要がある。適用の詳細についてはここに示さないが、マイクロソフト社の提供する情報を元に必ず問題点を修正しなければならない。

3.2.2. IIS のインストール

Windows 2000 Server への修正プログラムの適用後に、「Windows コンポーネントの追加と削除」機能を用いて IIS 5.0 をインストールする。

Windows 2000 Server の脆弱性への対策と同様に、IIS の脆弱性についても CD からのインストール直後に対策が必要となる。調査時点（2003 年 2 月）において、マイクロソフト社から IIS 5.0 について累積的な修正プログラムが提供されている。これを必ず適用すること。

Web サーバのセットアップを行う。サーバ名の設定、ホームディレクトリの設定、ログの設定などが必要である。詳細についてはマイクロソフト社のドキュメント等を参照すること。ここでは IIS 5.0 のセキュア化に関する詳細は述べない。

3.2.3. SSL のセットアップ

次に SSL に関するセットアップを行う。なおここでは CA の運用については対象としない。

（1） サーバ証明書の作成

「インターネットサービスマネージャー」から、対象となるサイトのプロパティを開く。「ディレクトリセキュリティ」タブを選択し、「セキュリティ保護された通信」の項目にある「サーバ証明書」を選択し、「サーバ証明書ウィザード」を起動する。

- 「証明書の新規作成」を選択
- 「証明書の要求を作成して後で送信する」
- 証明書の名前を入力、ビット長を選択する
- 「組織に関する情報」を入力する
- 「一般名」として、ホスト名を FQDN で入力する
- 「地域情報」として、サーバの所在地の情報を入力する
- 要求ファイル名を入力する。

これにより、指定した位置にサーバ証明書要求ファイルが作成される。

（2） 証明書の提出と取得

作成したサーバ証明書要求ファイルを、任意の証明機関に提出し、署名付きの証明書を取得する。自己署名の証明書を作成する際には、Windows 2000 Server のコンポーネントとして付属している「証明書サービス」を用いる。これらの手順の詳細については証明機関およびマイクロソフト社のドキュメントを参照すること。

（3） 証明書のインストール

署名された証明書を取得後、ふたたび「サーバ証明書ウィザード」を起動し、「保留中の要求を処理し、証明書をインストールする」を選択する。そして取得した証明書を選択しインストールする。

3.2.4. DAV の有効化

この段階までで、SSL 運用可能な Web サーバとして IIS 5.0 が設定される。IIS では、デフォルト状態で全ディレクトリに対して WebDAV 機能が有効化されているため、使用にあたって特に拡張モジュールをインストールするような作業は必要としない。

WebDAV ディレクトリを、匿名アクセス用に作成する例を以下に示す。

（1） 物理ディレクトリの作成

インストール時に WebDAV 用に作成したディスク空間に任意の物理ディレクトリ（フォルダ）を作成する。ここでは仮に anonymous というディレクトリを作成するものとする。

（2） 物理ディレクトリへのアクセス権の設定

作成した anonymous ディレクトリに対し、アクセス権を設定する。以下に示すように anonymous のプロパティを開き、Authenticated Users グループに対して、「書き込み」と「更新」を許可する。

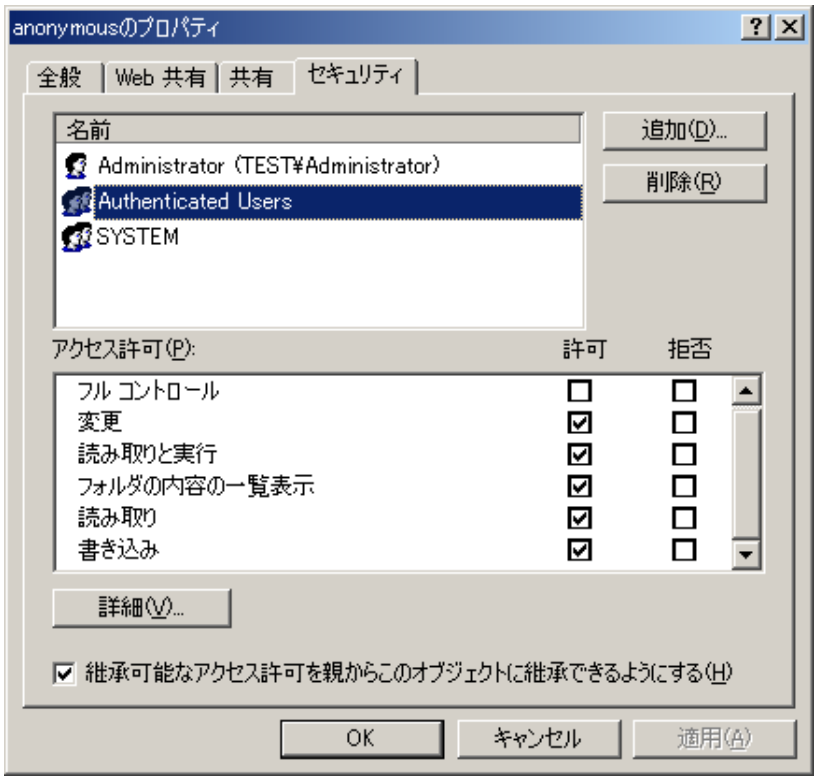

図 3-1　物理ディレクトリのアクセス権の設定

（3） 仮想ディレクトリの作成とアクセス権の設定

「インターネットサービスマネージャー」を起動し、「仮想ディレクトリの作成ウィザード」を起動する。ウィザード内で以下を設定する。

- エイリアス名　　「anonymous」
- コンテンツのパス　　anonymous ディレクトリを選択
- アクセス許可　　下図のように「読み取り」「書き込み」「参照」を許可する。

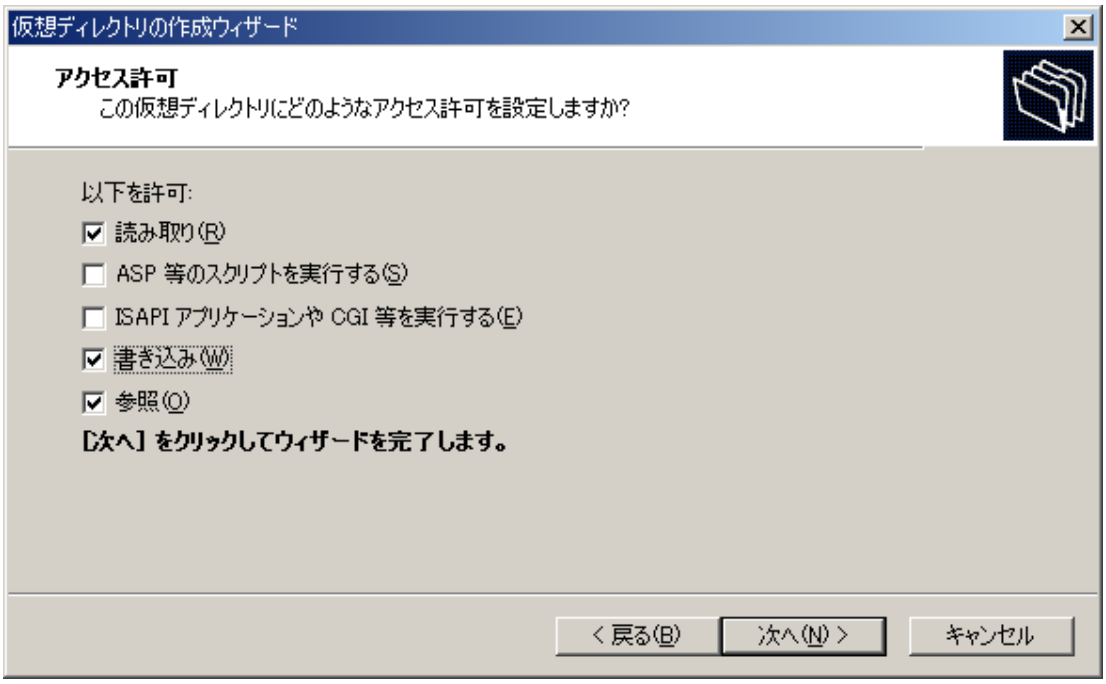

図 3-2　仮想ディレクトリのアクセス権の設定

以上により、匿名アクセスが可能な WebDAV フォルダが作成される。

3.2.5. SSL の有効化

対象の仮想ディレクトリにおいて SSL を有効に設定する。

対象となる仮想ディレクトリのプロパティから「ディレクトリセキュリティ」タブを開き、「セキュリティ保護された通信」エリアの「編集」を選択し「セキュリティ保護された通信」ウィンドウを開く。このウィンドウ内の「保護されたチャンネル(SSL)を要求する」をチェックする。これにより SSL が有効化される。下図に「セキュリティ保護された通信」ウィンドウを示す。

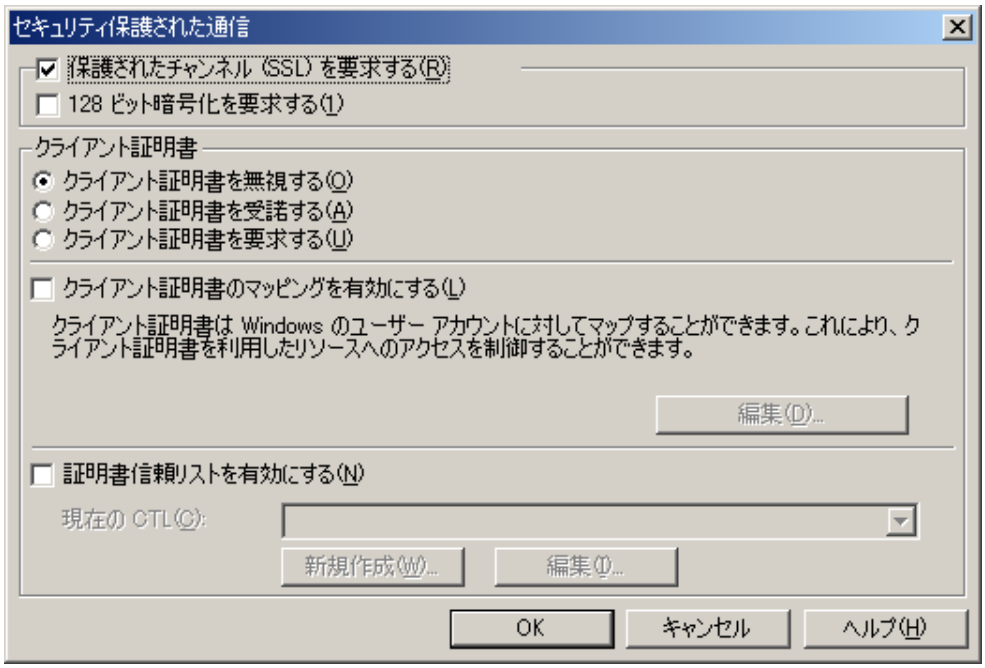

図 3-3　SSL の有効化の設定

3.2.6. WebDAV ディレクトリの http 公開

WebDAV によりコンテンツを作成し配置するディレクトリを http で公開するための設定方法を以下に示す。

コンテンツを作成するために用いるディレクトリと同じ物理ディレクトリについて、「インターネットサービスマネージャー」で異なる仮想ディレクトリを設定する。この新たな仮想ディレクトリについては、アクセス権の設定として「書き込み」のチェックを外し、WebDAV の書き込みアクセスを無効に設定すること。

3.3. OS X Server

ここでは OS X Server における WebDAV フォルダの作成方法を示す。

OS X Server では、OS に標準的に Web サーバが含まれている。Apache のバージョンは 1.3 系統のものとなるが、Apache 2 系統は基本的に他の UNIX 系 OS と同様に扱えるため、ここでは OS X Server に付属のパッケージを利用する。

3.3.1. Web サーバのセットアップ

OS X Server における Web サーバのセットアップは、Server Admin ツールを用いて行う。

（1） WebDAV フォルダの作成

サーバ上に WebDAV フォルダを作成する。ターミナルからアクセスし、パーミッションの設定を行う。手順を以下に示す。

```
# mkdir /home/webdav
# chown www /home/webdav
```

（2） サーバへ接続

Web サーバを設定するために、Server Admin ツールを起動して管理権限のあるユーザとしてサーバへと接続する。

図 3-4　サーバ接続

（3） Web サービス設定

「インターネット」タブから、「Web サービス」選択し、「Web サービスを設定」を開く。

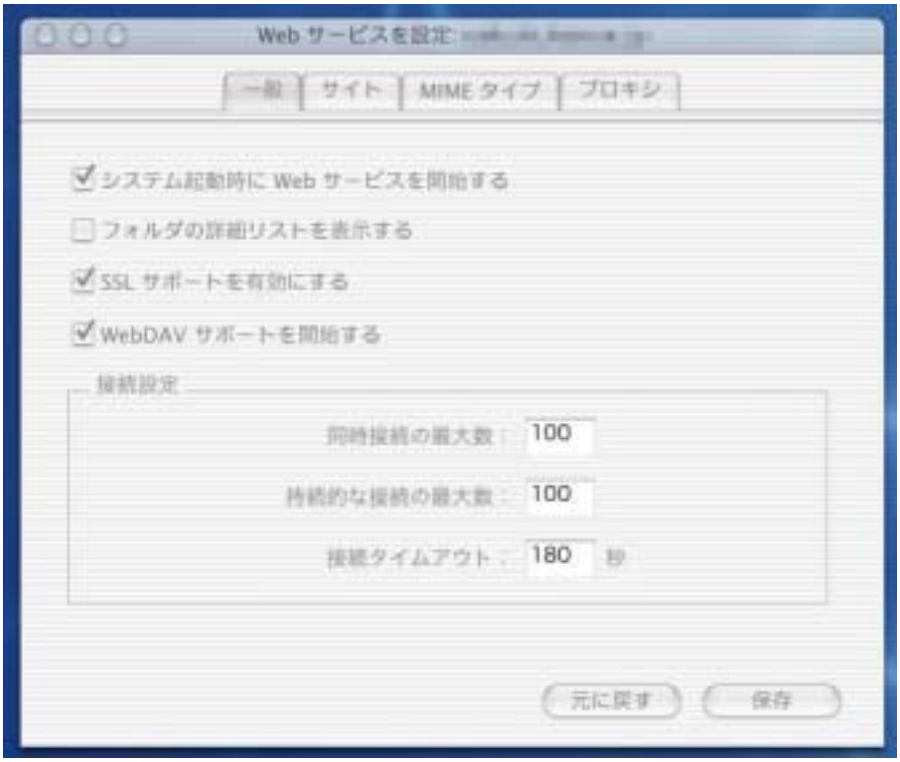

図 3-5　Web サービス設定

ウィンドウの各タブについて以下のように設定する。

- 一般

  「システムの起動時に Web サービスを開始する」、「SSL サポートを有効にする」、「WebDAV サポートを開始する」をチェックする

- サイト

「追加」を選択して、仮想ホストの追加を行う。各タブに対して以下のように設定する。ウィンドウを閉じた後、「有効」チェックボックスをチェックしてサイトを有効にすること。

■ 一般

「名前」欄にサーバ名を FQDN で、IP アドレス、使用ポート、「Web フォルダ」欄にルートとなるコンテンツ用のフォルダ、デフォルトの書類名を入力する。また、サイトオプションとして「WebDAV を許可する」をチェックする。下図に設定例を示す。

図 3-6　Web サービス設定 - サイト - 一般

■ ログ

ログファイルを指定する。

■ アクセス

「追加」を選択し、「保護領域」ウィンドウ内に「保護領域名」、「フォルダ」を入力する。さらに「アクセス」エリアでアクセス権の設定を行う。匿名の WebDAV アクセスを許可する場合は、「全員」をチェックし、「ブラウズとオーサリング可能」を選択する。下図に設定例を示す。

図 3-7　Web サービス設定 - サイト - アクセス

3.3.2. SSL のセットアップ

SSL のセットアップの手順を以下に示す。

（1） サーバ鍵と証明書の取得

OpenSSL のコマンドを用いてサーバ鍵と証明書要求を作成する。詳細は Apache 2 の場合と同様である。

作成した証明書を CA 機関に送付し、署名付き証明書と CA の証明書を取得する。

（2） SSL 化

「Web サイトの設定」から対象となるサイトを選択し、セキュリティタブを開く。

「Secure socket Layer(SSL)を仕様する」をチェックし、「証明書ファイルを編集」、「キーファイルを編集」、「ＣＡ証明書ファイルを編集」を選択し、それぞれのファイルの内容をペーストする。また鍵のパスフレーズを入力する。

3.3.3. WebDAV ディレクトリの http 公開

WebDAV によりコンテンツ作成を行うディレクトリを http で公開するためには、WebDAV ディレクトリとなっている物理フォルダを、SSL を有効化していない仮想サイトとして公開すればよい。その際にはオーサリングを無効にする点に注意が必要である。

4. アクセスコントロールモデル

この章では、WebDAV システムに対して適切なアクセスコントロールを行う方法を示す。

4.1. 定義

ここでは、「適切なアクセスコントロールが行われる WebDAV システム」を以下のように定義する。

- コンテンツの汚染が防止されている。承認されたユーザのみがコンテンツへの書き込みアクセスが可能であり、第三者がコンテンツを改変（汚染）できない。
- 情報漏洩が防止されている。セキュリティ上・プライバシー上の問題となるリソース・プロパティに第三者がアクセスできない。

4.2. 利用モデル

以下のような利用モデルを想定する。

- 作成者として許可されたユーザのみが WebDAV を用いてオーサリングを行う。

- 作成したコンテンツは基本的に一般に公開する。コンテンツの内容によっては認証手段を用いたアクセスコントロールを行う。

4.3. 検討事項

アクセスコントロールモデルに関連して検討すべき事項を挙げる。

（1） メソッドの実行制限方法

メソッド単位の実行の許可・不許可については、Apache 2.0 においては設定可能であるが、IIS 5.0 においては設定できない。代用の手段として、IIS では該当するユーザの NTFS ファイルへのアクセスを制御することで、ある程度まで類似する結果を得ることはできる。

（2） WebDAV 有効化・無効化の方法

サーバソフトウエアが、通信路の暗号化の有無により WebDAV 機能の有効化・無効化を切り替え可能かどうかが運用上の問題となる。

Apache 2.0 においては、リソースごとに WebDAV の有効化・無効化を指定可能であり、http と https により設定を分けられるので、暗号化された通信路においてのみ WebDAV を有効化できる。なお、Apache では、あるロケーションで WebDAV を有効にすると、そのサブロケーションでは全て自動的に WebDAV が有効になる。

これに対して IIS 5.0 においては、レジストリ設定を用いてサーバ全体について WebDAV を無効化することは可能であるが、ディレクトリ単位の有効・無効の制御は行えない。

（3） プロパティに対するアクセスコントロールの方法

サーバ実装により、プロパティに対するアクセスコントロールの方法が異なる。

Apache 2 においては、メソッド単位のアクセスコントロールが可能であるため、PROPFIND および PROPPATCH の実行の制限を行うことでプロパティへのアクセスを制限できる。しかし、この制限はリソース単位の制限となり、リソースに付加されたプロパティごとの制限は行えない。

IIS 5.0 においては、HTTP メソッド単位のアクセスコントロールは行えない。このため、プロパティへのアクセスは NTFS の「属性」「拡張属性」へのアクセス権でコントロールすることになるが、この 2 つのアクセス許可は HTTP とも連動しているため不都合が生じる。仮想ディレクトリへの「書き込み」権限を与えないことでプロパティへの書き込みは無効化できるが、「読み込み」権限を与えないと HTTP のアクセスも不可能になる。結果としてプロパティの読み取りアクセスを制限できない。

（4） 認証

IIS における Digest 認証は Windows ネットワークと関係しており、Windows ネットワーククライアントとしての機能を持つクライアントにおいてのみ取り扱うことができる。Apache においてはこのような制限は無く、RFC に基づくクライアントであれば Digest 認証の利用が可能である。

また、グループに対する認証を行えるかどうかは実装に依存する。Apache 2 と IIS 5.0 は共にグループ単位の認証を行うことができる。

（5） アクセス主体

サーバソフトウエアが動作時のユーザおよびその権限は、マルチユーザーOS における運用上の検討事項となる。

Apache は特権ユーザとして動かすことも可能であるが、一般にはセキュリティ上のリスクを小さくするために、起動後にユーザを Apache 専用ユーザに変更して動作させる。このため、WebDAV プロトコルを用いて作成されたファイルやディレクトリは、Apache を動作させるユーザをオーナーとして作成される。CGI 実行時にはさらに CGI プログラムの所有者の権限で動作する機構があるが、現状では WebDAV にそのような機構は用意されていない。よって、Apache の動作権限では書き込みが許可されていない空間については WebDAV による汚染から守られていると言える。

IIS においては、IIS が認証したユーザの権限でコンテンツにアクセスする際は、一時的にそのユーザの権限でアクセスを行う偽装（Impersonalization）と呼ばれるメカニズムが用いられる。このためファイルとディレクトリは認証されたユーザをオーナーとして作成される。このことよりユーザが書き込み可能な空間は WebDAV により汚染を受ける可能性がある。

（6） スクリプト編集の方法

CGI などのスクリプトを WebDAV で編集する場合には、スクリプトが実行された結果ではなくスクリプトのソースにアクセスする必要がある。しかしながら、通常このようなアクセスはセキュリティ上の重大な弱点となるため許可されていない。

Apache では、スクリプトの存在するディレクトリをソースアクセス専用の保護された空間に位置づけ、スクリプト拡張子の設定を強制的に上書きしてテキストとして扱うことでソースへのアクセスが可能になる。

IIS では、仮想ディレクトリに対し「スクリプトソースアクセス」を許可することでソースへのアクセスが可能になる。

（7） 閲覧者用アカウントの方式

WebDAV 固有の問題ではないが、閲覧にも認証によるアクセスコントロールを行う場合には、閲覧用のユーザとパスワードを閲覧者全体で共通にするか、個人ごとに別にするかを検討する必要がある。共通アカウントは管理が容易であるが情報の秘匿性は低い。個別アカウントは管理が難しい反面、情報の秘匿性は高い。

4.4. 運用モデル

以上の点を検討した結果、適切なアクセスコントロールを行う WebDAV システムの運用モデルは以下のように整理できる。

- 認証されたユーザにのみ WebDAV 関連メソッドの発行を認める。ユーザ認証は Basic 認証を用いる。

- WebDAV エリアへのアクセスについては、Apache では公開形態によって HTTP/1.1 によるアクセスは認証無しで許可するか、認証付きで許可するかを選択する。IIS ではプロパティへのアクセスの問題があるため、HTTP/1.1 によるアクセスに対しても認証を必須とする。

- スクリプトソースの編集は専用の空間で認証されたユーザにのみ許可する。

5. アクセスコントロールモデルの導入

検討したアクセスコントロールモデルを各サーバソフトウエアに導入する手順を示す。

5.1. Apache 2.0

「セキュアな WebDAV サーバの導入」において Red Hat および Debian における Apache 2.0 の設定を示した。どちらにおいても WebDAV に関する設定は ssl.conf にまとめているので、ここでは両者を一括して記述する。

5.1.1. HTTP ユーザの作成

Apache においては、HTTP 認証の対象となるユーザは OS のユーザとは別に取り扱われる。そのため、ユーザ認証を行うためにはまず HTTP ユーザを作成する必要がある。

最初に、パスワードファイルを置く場所を決める。パスワードファイルを HTTP でアクセスされないように、コンテンツとは異なる空間に置くことが望ましい。ここでは同様に取り扱われるべき Apache の設定ファイルと同じディレクトリ内に置くことにする。

Red Hat 8.0 では、/etc/httpd/auth、Debian では/etc/apache2/auth ディレクトリをそれぞれ作成し、その中に passwd という名前で作成する。

```
# mkdir /etc/apache2/auth
```

ユーザの作成、パスワードの管理は htpasswd（Debian では htpasswd2）コマンドで行う。初回のみ「-c」オプションをつけ、passwd ファイルの新規作成を行う。

```
# /usr/bin/htpasswd2 -c /etc/apache2/auth/passwd user1
New password:
Re-type new password:
Adding password for user user1
# /usr/bin/htpasswd2 /etc/apache2/auth/htpass user2
New password:
Re-type new password:
Adding password for user user2
```

グループを作成する場合は、パスワードファイルと同じディレクトリに group というテキストファイルを作成する。このファイルにはグループ名と所属するユーザを列挙して定義する。例として、user1 と user2 が WebDAV グループに含まれる場合の group の内容を以下に示す。

```
webdav: user1 user2
```

5.1.2. ユーザ認証

ユーザ認証を行うためには、まず認証方式やパスワードファイルの所在を指定する必要がある。以下のディレクティブを用いて指定する。

表 5-1　ユーザ認証に関するディレクティブ

AuthType	認証方式の指定。Basic 又は Digest。
AuthName	認証の領域（realm）を示す。
AuthUserFile	HTTP パスワードファイルを指定する

ユーザ認証によるアクセスコントロールには require ディレクティブを用いる。書式を示す。

表 5-2　require ディレクティブの書式

require user <username*>	指定したユーザを許可する
require group <groupname*>	指定したグループを許可する
require valid-user	AuthUserFile に含まれる全てのユーザを許可する

5.1.3. メソッド実行の制限

Apache では、Limit ディレクティブおよび LimitExcept ディレクティブによりメソッドの実行を制限できる。Limit は指定したメソッドの実行を制限し、LimitExcept は指定したメソッド以外の実行を制限する。条件の例を以下に示す。

- HTTP/1.1 ブラウザからのアクセスは無条件で許可
- WebDAV 関連メソッドの実行は認証が必要（WebDAV グループに対して許可）

上のような条件の場合は、LimitExcept を用いて以下のように記述できる。ただし CGI を用いる際には POST メソッドの実行も例外として無条件に許可することが必要な場合もある。

```
<LimitExcept GET HEAD OPTIONS>
        Dav on
        require group webdav
</LimitExcept>
```

コンテンツの公開に対しても認証によるアクセスコントロールを行う場合は、require の対象となるユーザー（もしくはグループ）を変えることで実現する。以下に示す例では visitor グループを用いている。visitor グループに WebDAV グループのメンバーが含まれていないと、webdav グループのメンバーによるブラウズが行えなくなることには注意を要する。

```
<Limit GET HEAD OPTIONS>
        require group visitor
</Limit>
<LimitExcept GET HEAD OPTIONS>
        Dav on
        require group webdav
</LimitExcept>
```

5.1.4. スクリプトソース編集用空間

スクリプトのソースを WebDAV から編集可能にする場合は、cgi の許可方法に特に注意する必要がある。AddHandler ディレクティブを上書きできないので、仮想ホスト全体に対して.cgi を指定した AddHandler を設定するとその仮想ホストでは ForceType ディレクティブを用いたソースへのアクセスができなくなる。（ただし、これはセキュリティ上有益な仕様である）

スクリプトは ScriptAlias ディレクティブを用いて/cgi-bin などの特定の空間に集約させることが望ましい。Red Hat と Debian にはデフォルト設定で/cgi-bin が用意されており、その実体は Red Hat は/var/www/cgi-bin、Debian は/usr/lib/cgi-bin に存在する。

Debian において、/usr/lib/cgi-bin にソース編集可能な WebDAV 空間を設定する例を示す。

```
ScriptAlias /cgi-bin /usr/lib/cgi-bin
Alias /webdav-script "/usr/lib/cgi-bin"
<Location /webdav-script>
        （略）
    Dav on
    require group webdav
    ForceType text/plain
</Location>
```

5.1.5. その他の設定

(a) メッセージボディの制限

WebDAV メソッドのメッセージボディは XML で記述されているが、その最大サイズの制限を行う方法は通常のリクエストボディと異なり、LimitXMLRequestBody ディレクティブで指定する。ただし、この制限は全ての XML メッセージボディに影響するので注意が必要である。

(b) Depth ヘッダの制限

WebDAV プロトコルには、巨大なコレクションに対して Depth ヘッダで Infinity を指定した要求を発行することにより DoS 攻撃が成立する問題が存在する。これに対応するため、Apache 2 ではデフォルト状態では Depth ヘッダに Infinity を指定できない。DavDepthInfinity を on に変更は可能だが、セキュリティ上の観点から off のままで運用することが望ましい。

5.2. IIS 5.0

ここでは、IIS 5.0 の WebDAV サーバにアクセスコントロールモデルを導入する方法を示す。

5.2.1. ユーザ・グループの作成

IIS において認証を行うためには、HTTP ユーザを Windows ユーザとして作成する必要がある。

ユーザの作成は「コンピュータの管理」から行うことができる。その際には適切なユーザ管理ポリシーに則って、ユーザ自身によるパスワード変更の可否、パスワードの期限設定などの設定を行うことが重要である。設定は Windows ネットワークの構成とも関連する。

また、グループも Windows のユーザ・グループとして作成する必要がある。以下の例では、WebDAV コンテンツの作成が可能なユーザの所属する[DAVauthors]グループ、WebDAV コンテンツの閲覧が可能なユーザの所属する[DAVreaders]グループを作成する。もし WebDAV コンテンツのアクセス単位を分割する場合は、コンテンツごとのグループを作成するとよい。

最後に、グループにユーザを所属させる。WebDAV のコンテンツ作成を行うユーザは DAVauthors と DAVreaders の双方に、閲覧可能なユーザは DAVreaders にのみ所属させる。

5.2.2. HTTP 認証の設定

WebDAV を有効化している（「書き込み」が許可されている）仮想ディレクトリについて、プロパティに対する読み取りアクセスを規制するために、全てのアクセスに認証を要求するよう設定する。

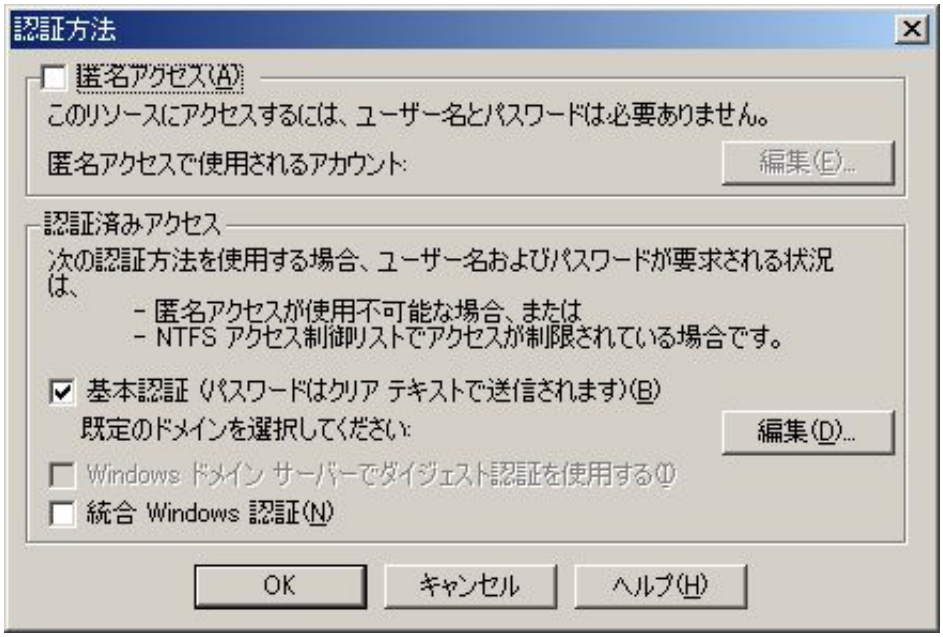

図 5-1　HTTP 認証の設定

5.2.3. WebDAV フォルダのアクセス権設定

次に、WebDAV フォルダのアクセス権の設定を行う。この作業は NTFS の機能を用いて行う。WebDAV フォルダのプロパティを開き、セキュリティタブを選択する。

（1） 継承の禁止

フォルダ独自のアクセス権を設定するために、「継承可能なアクセス許可を親からこのオブジェクトに継承できるようにする」のチェックを外し、アクセス権の継承を禁止する。

（2） [Authenticated Users]グループの削除

デフォルトの状態では、以下のユーザとグループにアクセスが認められている。

- Administrator
- Authenticated Users
- SYSTEM

このうち、[Authenticated Users]グループは全ての承認されたユーザを意味し、認証が無い場合は匿名ユーザも含まれる。[Authenticated Users]グループを削除し、詳細なアクセスコントロールを可能にする。

（3） グループの追加

作成した WebDAV アクセス用グループである[DAVauthors]と[DAVreaders]を追加する。

（4） DAVauthors のアクセス権の設定

「詳細」ボタンを押して「アクセス制御の設定」ウィンドウを開き、DAVauthors グループを選択し「編集」を押して「アクセス許可のエントリ」ウィンドウを開く。

アクセス権の詳細設定は以下のように行う。

表 5-3　DAVauthors のアクセス権設定

項目	設定値
適用先	このフォルダ、サブフォルダおよびファイル
フォルダのスキャン/ファイルの実行	許可
フォルダの一覧/データの読み取り	許可
属性の読み取り	許可
拡張属性の読み取り	許可
ファイルの作成/データの書き込み	許可
フォルダの作成/データの追加	許可
属性の書き込み	許可
拡張属性の書き込み	許可
サブフォルダとファイルの削除	許可
削除	（チェック無し）
アクセス許可の読み取り	許可
アクセス許可の変更	（チェック無し）
所有権の取得	（チェック無し）

（5） DAVreaders のアクセス権の設定

同様に、DAVreaders のアクセス権を以下に示すように設定する。

表 5-4　DAVreaders のアクセス権設定

項目	設定値
適用先	このフォルダ、サブフォルダおよびファイル
フォルダのスキャン/ファイルの実行	許可
フォルダの一覧/データの読み取り	許可
属性の読み取り	許可
拡張属性の読み取り	許可
ファイルの作成/データの書き込み	（チェック無し）
フォルダの作成/データの追加	（チェック無し）
属性の書き込み	（チェック無し）
拡張属性の書き込み	（チェック無し）
サブフォルダとファイルの削除	（チェック無し）
削除	（チェック無し）
アクセス許可の読み取り	許可
アクセス許可の変更	（チェック無し）
所有権の取得	（チェック無し）

5.2.4. IIS における問題点

前述したように、IIS にはプロパティの読み取りを防ぐ手段が無い。また、Depth ヘッダにおいて Infinity が指定された際には指定通りに動作してしまう問題がある。

これらの問題を抑制するため、IIS において WebDAV を有効化する場合には、適切な認証機構を導入することは必須と言える。また、IIS で WebDAV を用いず HTTP/1.1 を用いてリードオンリーな Web サーバを運用する場合には WebDAV 機能を無効化することが望ましい。

5.3. OS X Server

OS X Server の WebDAV サーバにアクセスコントロールモデルを導入する方法を示す。

5.3.1. ユーザ

OS X Server のユーザ管理モデルは、他の UNIX におけるユーザ管理のモデルとは大きく異なっている。OS X Server における標準的な Web サーバソフトウエアは Apache であるが、これは OS X Server のユーザ管理システムと統合されており、OS X のユーザを用いてアクセスコントロールを行う方式が取られている。このため、HTTP 認証においても OS X のユーザとグループが認証の対象となる。

5.3.2. アクセス権の種類

ServerAdmin で WebDAV 機能に対しアクセスコントロールを行う場合、以下の 3 種類の制限方法が存在する。

（1） 全員に対するブラウジングおよびオーサリング許可

匿名アクセスに対して、ブラウズとオーサリングが許可される。最もセキュリティレベルの低い選択肢である。

（2） 全員に対するブラウジング許可・認証ユーザのオーサリング許可

限られたユーザだけが WebDAV によるオーサリングを行い、匿名アクセスはブラウズのみが許可される。公開コンテンツの運営に適している。

（3） 認証ユーザに対するブラウジング許可・認証ユーザに対するオーサリング許可

オーサリングだけでなく、ブラウズにも認証が必要となる。アクセスを制限する必要のあるコンテンツの運営に適している。

ここで取り扱おうとするアクセスコントロールモデルには、（2）および（3）が適合する。公開形態に応じて選択し、ブラウズまたはオーサリング可能なユーザを正しく設定することでアクセス権の設定が適切に実現できる。

5.3.3. スクリプトソース編集用空間

OS X Server においては、各仮想ホストについて CGI の実行のオン・オフを設定する方式が取られている。そのため、スクリプトソース編集用の空間は異なる仮想ホストに別途作成する必要がある。通常の Web 利用とは異なるポートを用いる専用の仮想ホストを設定し、スクリプトソースの空間を CGI の実行をオフにしてエイリアスすることでスクリプトソース編集用の空間が構築可能である。

6. WebDAV クライアントの導入

ここでは、WebDAV クライアントの導入について記述する。以下を示す。

- Windows における Web フォルダの導入
- OS-X におけるファインダを用いた WebDAV リソースへのアクセス
- UNIX における cadaver の導入

6.1. Windows 2000/XP

Windows2000/XP における Web フォルダの導入方法を示す。

（1） コンポーネントのアップデート

Web フォルダの導入に先立ち、Windows のコンポーネントをアップデートすること。サービスパックと InternetExplorer については最新版を導入することが望まれる。

（2）「ネットワークプレース追加ウィザード」の起動

「マイネットワーク」を開き、「ネットワークプレースの追加」をダブルクリックして、ネットワークプレース追加ウィザードを起動する。

（3） URL の入力

ネットワークプレースの追加ウィザードの開始ダイアログ上の「ネットワークプレースの場所を入力してください(L):」に URL を入力し、「次へ(N)＞」を選択する。

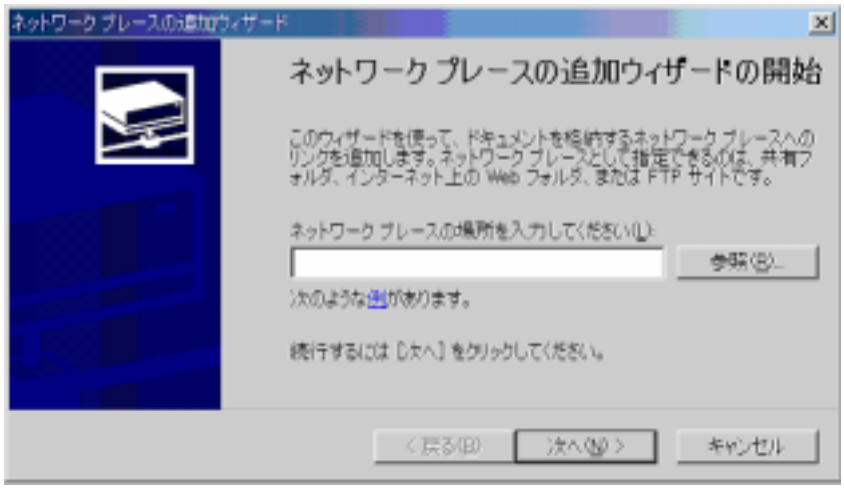

図 6-1　URL の入力

（4） 証明書の確認

初めて WebDAV を利用する場合にはセキュリティの警告ダイアログが表示される。そこで証明書を確認して問題が無ければ「はい(Y)」を選択する。

（5） パスワードの入力

WebDAV のフォルダを利用するためのパスワードを要求される。そこで「ユーザ名(U)」と「パスワード(P)」を入力して「OK」を選ぶ。また、このとき「このパスワードを保存する(S)」を選択しておけば、以降のこのフォルダを利用する際のパスワード入力が省略される。

（6） エイリアスの作成

接続が成功するとフォルダをあらわすエイリアスの設定ダイアログに移る。「このネットワークプレースの名前を入力してください(E):」へエイリアスを入力し、「完了」を選択する。

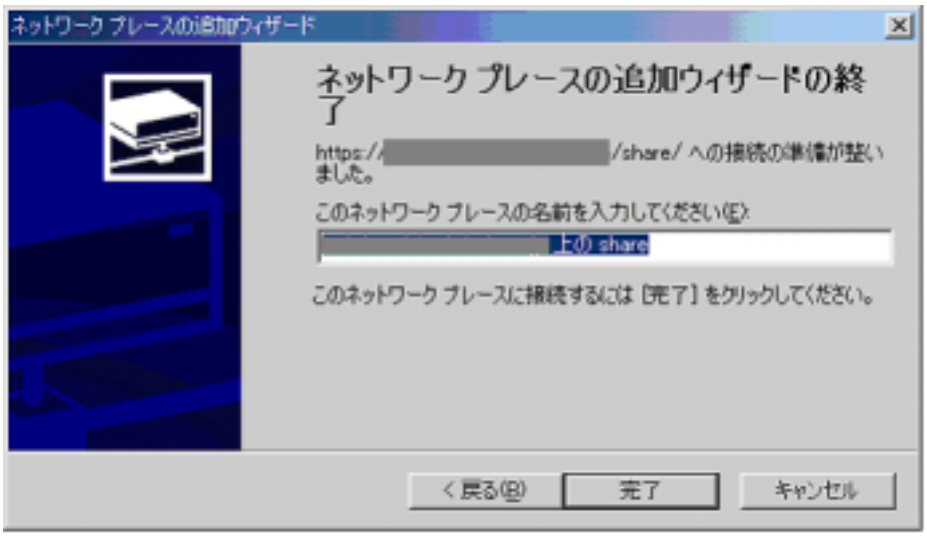

図 6-2　エイリアスの作成

次回からはマイネットワークウィンドウ内に先ほど設定した WebDAV のフォルダのエイリアスを持つアイコンが現れる。それをダブルクリックすることで WebDAV フォルダの内容の操作が可能になる。

（7） フォルダの利用

設定した Web フォルダは、通常のフォルダと同様にアクセス可能であり、ドラッグアンドドロップによるファイルのコピー、フォルダの新規作成などが行える。

（8） Web フォルダにおける問題点

読み込みを無条件で許可、書き込みに認証が必要と設定した WebDAV フォルダにアクセスする際に、ファイルのドロップによるコピーが失敗する問題がある。フォルダの新規作成を行うことで認証処理が行われ、以後の書き込みアクセスが可能になる。

6.2. OS-X

Mac OS-X において、Finder を用いて WebDAV リソースにアクセスする方法を示す。ここで取り扱うバージョンは OS-X 10.2.4 である。

（1） Finder を起動する

（2）「サーバへ接続」の選択

Finder のメニューバーから「サーバへ接続」を起動する。

図 6-3　サーバへ接続の選択

（3） URL の入力

「サーバへ接続」のアドレス欄に、WebDAV リソースの URL を入力する。

（４） WebDAV フォルダの利用

認証が要求される場合、ユーザ名とパスワードを入力するとフォルダが開かれる。

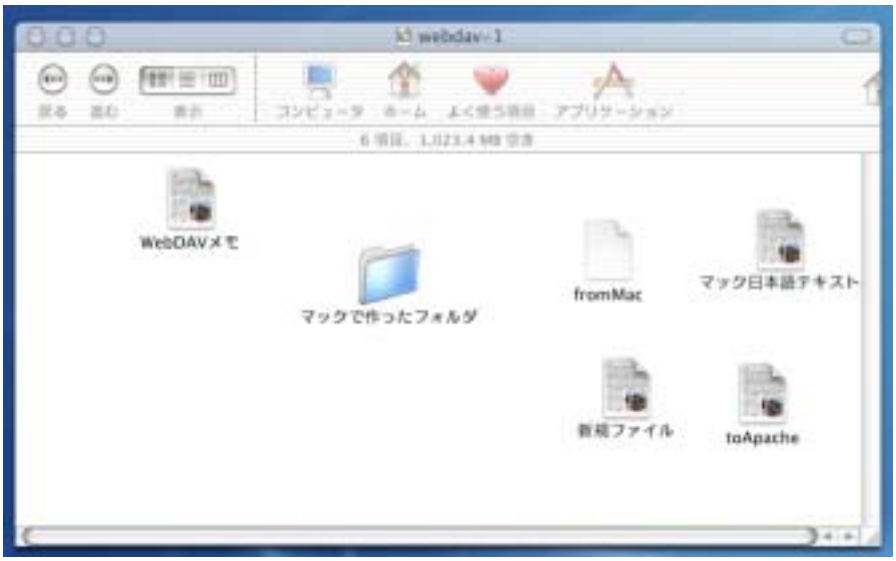

図 6-4　Web フォルダの利用

また、デスクトップ上に WebDAV サーバへの接続を示す以下のアイコンが表示される。

図 6-5　WebDAV サーバへの接続を示すアイコン

このアイコンを開けば WebDAV フォルダを開くことができる。また、このアイコンをゴミ箱へ捨てることで、WebDAV サーバとの接続を切断できる。WebDAV フォルダには、通常のフォルダと同様のアクセスが可能である。

6.3. UNIX

ここでは、UNIX 上で動作する代表的な WebDAV クライアントとして cadaver を取り上げ、そのセットアップ方法と使用方法を示す。

（1） cadaver のインストール

多くの UNIX 互換 OS で用意されているバイナリパッケージを利用することが望ましい。特にソースコードを用いて cadaver をインストールする際の手順を以下に示す。

1. ソースコードを cadaver の開発ページ[3]からダウンロードする。
2. ソースコードの展開
3. configure

   configure スクリプトを実行する。SSL を有効化し、XML パーサを付属のものを利用する場合の例を以下に示す。

```
% ./configure --with-included-expat --with-ssl
```

4. make
5. make install

（2） WebDAV フォルダへの接続

cadaver の起動時に WebDAV フォルダを指定し接続する。SSL を用いる場合は証明書の概要が表示され、認証が必要な場合はユーザ名とパスワードを入力する。

[3] http://www.webdav.org/cadaver/

（3） コマンド

接続後、以下のコマンドを用いて WebDAV フォルダの操作を行う。

表 6-1　UNIX 環境における WebDAV フォルダ操作コマンド

コマンド	概要
ls	現在の WebDAV ディレクトリのファイル一覧を取得
cd	WebDAV ディレクトリの移動
put <filename>	指定したローカルファイルをアップロードする
get <filename>	指定したリモートファイルをダウンロードする
cat <filename>	指定したリモートファイルを閲覧する
mkcol	ディレクトリの作成
rmcol	ディレクトリの削除
delete	指定したリモートファイルを削除する
copy <src> <dst>	src から dst へのコピー
move <src> <dst>	src から dst への移動（ファイル名の変更）
lock <filename>	filename のロック
unlock <filename>	filename のアンロック

7. WebDAV ソフトウエアの相互運用可能性

ここでは､各サーバソフトウエアとクライアントソフトウエア間の相互運用可能性の調査を行った結果を示す。

7.1. 相互運用可能性実験

WebDAV 環境に関して相互運用可能性を確認する実験を行った結果を以下に示す。

7.1.1. 使用したソフトウエア

実験に使用したサーバソフトウエア、クライアントソフトウエアを下表に示す。

表 7-1　相互運用可能性実験に使用したサーバソフトウエア

名称	バージョン	OS
Apache 2	Apache 2.0.40	Red Hat 8.0
IIS	IIS 5.0	Windows 2000 Server 5.00.2195 SP3
OS-X Server	Apache 1.3.26	OS-X Server

表 7-2　相互運用可能性実験に使用したクライアントソフトウエア

名称	バージョン	OS
Web フォルダ（Windows2000）	IE6.0.2800.1106	Windows 2000 SP3
Web フォルダ（WindowsXP）	IE6.0.2800.1106.xpsp1	Windows XP SP1
Finder	-	OS-X 10.2.4
cadaver	0.21.0	FreeBSD4.7

7.1.2. 実験結果

実験結果を下表に示す。表中、特に問題なく動作する場合は○、動作はするが何らかの問題を伴う場合は△、動作しない場合は×であらわした。

表 7-3　WebDAV 相互運用性 実験結果

	Apache 2			IIS5.0			OS-X Server		
	GET	PUT	MKCOL	GET	PUT	MKCOL	GET	PUT	MKCOL
Web フォルダ (Windows2000)	○*2	○*2	○*2	○	○	○	△*3	△*3	△*3
Web フォルダ (WindowsXP)	○	○	○	○	○	○	○	○	○
Finder	○	○	○	○	○	△*1	○	○	○
cadaver（3）	○	○	○	○	○	○	○	○	○

*1：フォルダの作成時にデフォルトで付加されるフォルダ名が文字化けし、
アクセス不能になる。フォルダ名を変更することによりアクセス可能になる。

*2：日本語のリソースは、サーバ側ファイルシステムに正しく作成されるものの利用できない。

*3：日本語の取り扱いはできない

結果に示されるように、サーバとクライアントの組み合わせにおいては、リソース名として英数字を用いる限りにおいては WebDAV 機能には高い相互運用可能性があると言える。しかしながら、リソース名として日本語を用いた際には、いくつかのクライアントーサーバの組み合わせにおいては表示がおかしくなる、アクセスができなくなる等の問題が発生する。

7.2. 日本語リソース名の扱い

ここでは、相互運用可能性実験で問題となった日本語リソース名の扱いについて記述する。

7.2.1. 実装における問題

まず、実験により確認されたクライアントーサーバ間の問題について、その詳細を示す。

（1） Windows 2000 と Apache 2 間における問題

Windows 2000 から Apache2 の WebDAV フォルダにアクセスし日本語フォルダの作成を行った場合、Web フォルダ上では一見正常に作成されたように見えるが、更新を行うとフォルダ名が文字化けを起こし、アクセスができなくなる。

Windows 2000 より Apache 2 の Web フォルダにアクセスし新規フォルダの作成操作を行うと、「新しいフォルダ」という名前のコレクションの作成が行われる。発行される MKCOL メソッドを以下に示す。

```
MKCOL /webdav/¥220V¥202¥265¥202¥242¥203t¥203H¥203¥213¥203_ HTTP/1.1
```

この MKCOL メソッドで指定されているターゲット URI は、RFC2396 で定義されている非 US-ASCII 文字を使った URI のエスケープ方法に従っておらず、シフト JIS のフォルダ名がそのまま用いられている。しかしながら、Apache 2 はこれを正しく受け取り、シフト JIS で名付けられたディレクトリを作成する。

次に Web フォルダの更新を行うと、PROPFIND メソッドが発行されてメンバーが取得される。その際に Apache 2 が返すレスポンスを以下に示す。

```
Content-type: text/xml; charset="utf-8"
        (中略)
<D:href>/webdav/%90V%82%b5%82%a2%83t%83H%83%8b%83_/</D:href>
```

UTF-8 によるエンコードが指定されているが、実際にはコレクション名はシフト JIS で記述されている。おそらく受け取る側の Windows 2000 がこのコレクション名をレスポンス内の Content-type で示された UTF-8 とみなして、シフト JIS への変換処理をしてしまうため、以後の文字化けが発生するものと考えられる。

このコレクションに対する Windows 2000 の操作は文字化けしたフォルダ名を元に行われるため、対象となるリソースが存在しない操作になってしまう。以後コレクションに対しては操作が不能となる。

（2） Windows 2000 と OS-X Server 間における問題

Windows 2000 の Web フォルダに接続して日本語リソースの作成を行うと、メソッドの実行が失敗する。原因は、Web フォルダがシフト JIS を直接含むメソッドを発行した際に、Apache はシフト JIS を含むターゲット URI を解釈するのに対し、OS-X のファイルシステム（UFS）が直接シフト JIS のファイルを作成できないことに依存している。

また、日本語リソースの読み込みも前項（１）の問題と同様な理由で失敗する。

（3） Finder と IIS 間の問題

初期リソース名について問題が発生する。名称変更自体と変更後のアクセスは通常と同様に行われる。おそらく Finder が持っている初期フォルダ名が WebDAV フォルダの機能により暗黙的に変換されてしまい、キャッシュの整合性がとれなくなってしまうためと考えられる。

7.2.2. インターネット標準仕様との関係

URI を定義する RFC2396 では、非 US-ASCII 文字をオクテット値としてエンコードする方式を定めており、これにより日本語を含む URI も扱うことができる。しかしこれはあくまでもオクテット値（バイナリデータ）で記述するということあり、文字コードの種別については関知しない。

WebDAV の仕様では、プロトコルを実装する際には XML を利用するため最低限 UTF-8 のサポートが要求されるが、それ以上詳細な規定は行われていない。相互運用可能性を確保するにあたっては、クライアントとサーバの双方が少なくとも UTF-8 に対応するよう心掛ける必要がある。しかしながら調査時点では主要な WebDAV 実装においても UTF-8 に対応していないものが見られた。

WebDAV で作成したコンテンツを HTTP で公開する際にも問題が存在する。URI の仕様により、日本語の部分は全てオクテット値でエンコードされなければならないが、これは人間には非常に分かりにくい。ブラウザで日本語を入力すると自動的にオクテット値に変換されてアクセスするようなメカニズムは一見有効に思えるが、これはアクセス対象のエンコーディングを事前に知らない限りアクセスできない、入力した文字のエンコーディングの種類によりアクセスできなくなってしまう現象を招く。そもそも日本語に対応していない環境ではこのようなメカニズムは利用できない。

また、URI は識別子として用いられることもある。WebDAV においても ACL の PRINCIPAL として URI を用いているが、もし PRINCIPAL に日本語を用いた場合、エンコーディングの違いによって異なる PRINCIPAL と扱われてしまう問題がある。

8. WebDAV システムの運用における注意点

ここでは、これまでの仕様の検討および運用の検討を元に、まとめとして、WebDAV システムを安全かつ相互運用可能性を保ちながら運用する指針を示す。

（1） セキュアな通信路の確保

WebDAV システム自体では暗号化の処理を行わずにネットワーク上でファイルや認証情報などを送受する。通信路上の覗き見を防ぐためには SSL/TLS などを用いて暗号化した通信路を確保するなどして、セキュアな通信路を使うべきである。セキュアではない通信路は、読み取りのみを許可した、HTTP によるコンテンツ公開に限定して使用するべきである。

（2） WebDAV コンテンツの適切なアクセスコントロール

WebDAV コンテンツに対しては、運用のために必要最低限のアクセス権のみを認めるべきである。リソースに対するアクセスコントロールだけでなく、プロパティに対するアクセスコントロールについても慎重な検討が必要となる。特に WebDAV アプリケーションがデッドプロパティとしてセキュリティ上およびプライバシー上問題となる情報を付加するかどうかについては調査し対策を施す必要がある。

（3） リソース要求の制限

WebDAV プロトコルはディスク空間や処理能力などのリソースに大きな影響を与える。WebDAV 用のディスク空間は別のファイルシステム上に確保し、ディスクフルによるシステム全体のダウンを回避する必要がある。処理能力については、Depth ヘッダによる制限や XML ボディの容量制限を検討する。

（4） 日本語の使用の限定

WebDAV をネットワークファイルサーバへのアクセス手段として用いる場合においては、リソース名に対する日本語の使用は限定されたソフトウエア構成においてのみ行うべきである。HTTP を用いて一般に公開するリソース名には日本語を用いるべきではない。

9. セキュア通信機構の調査

ここでは、SSL/TLS に相当するセキュア通信機構として IPsec および ssh によるポートフォワーディングを取り上げる。これらについて導入方法を示した上で、比較検討の結果を示す。取り上げる項目を以下に示す。

- IPsec による WebDAV システムのセキュア化
- ssh による WebDAV システムのセキュア化
- SSL/TLS、IPsec、ssh の比較検討

9.1. IPsec による WebDAV システムのセキュア化

9.1.1. 概要

TCP/IP 環境におけるセキュアな通信の実現を検討する際には、大きく分けて 2 種の選択肢が存在する。ひとつは SSL/TLS のような、TCP のコネクションを暗号化して保護する方式である。もうひとつは IP パケットそのものを暗号化する方式である。IPsec は後者のメカニズムである。

IPsec は IETF において開発され標準化された IP 層における暗号化メカニズムである。調査時点における IPsec の実装には以下のようなものがあった。

- FreeS/WAN
  Linux をターゲットにした IPsec 実装である。多くの稼動実績を持つ。

- KAME
  BSD 系 UNIX を対象にした IPv6/IPsec の実装である。BSD 系 UNIX で幅広く採用されている。

- USAGI
  比較的新しい Linux の IPv6 実装であり、最近のバージョンでは IPsec に対応している。

■ Windows 2000/XP

OS に標準的に IPsec が実装されている。

以下では、FreeS/WAN を組み合わせた Red Hat 8.0 上で WebDAV サーバを動かし、Windows 2000 の Web フォルダからアクセスする構成で、セキュアな WebDAV システムのモデルを示す。

9.1.2. ネットワーク構成

想定するネットワーク構成を下図に示す。LAN 内に Windows 2000 Professional と Red Hat Linux 8.0 サーバそれぞれ 1 台設置する。WebDAV サーバは Red Hat 上で稼動しており、Windows 2000 の Web フォルダからアクセスする。

図 9-1　想定するネットワーク構成

9.1.3. Red Hat 8.0 の IPsec 化

標準的な Linux 環境に IPsec の実装は含まれていないため、FreeS/WAN をインストールする。Red Hat 系 Linux ではあらかじめ用意された rpm がカーネルのモジュールとして動作する。

（1） 現在の Linux カーネルのバージョンを確認する

シェルから以下を入力し Linux カーネルのバージョンを確認する。

```
$ uname –a
```

（2） FreeS/WAN のダウンロード

FreeS/WAN 公式サイト[4]から現在使用している Linux カーネルのバージョンに合った rpm をダウンロードする。ただし、Windows やサードパーティの IPsec 製品と連携をとる場合は、x509 パッチされた rpm を使用する必要がある。以下のファイルをダウンロードする[5]。

- freeswan-1.99_x509_0.9.15_2.4.18_24.8.0-0.i386.rpm
- freeswan-module-1.99_x509_0.9.15_2.4.18_24.8.0-0.i386.rpm

また FreeS/WAN は libpcap パッケージを必要とするため、Red Hat 社のダウンロードページ[6]からキーワード「libpcap」で検索し、以下のファイルをダウンロードする。

- libpcap-0.6.2-16.i386.rpm

（3） パッケージのインストールと/etc/sysctl.conf の修正

ダウンロードした rpm をインストールする。手順を以下に示す。

```
$ su
# rpm -ivh libpcap-0.6.2-16.i386.rpm
# rpm -ivh freeswan-module-1.99_x509_0.9.15_2.4.18_24.8.0-0.i386.rpm
# rpm -ivh freeswan-1.99_x509_0.9.15_2.4.18_24.8.0-0.i386.rpm
```

[4] http://www.freeswan.ca/download.php
[5] http://download.freeswan.ca/freeswan-x509/RedHat-RPMs/2.4.18-24.8.0/
[6] http://www.redhat.com/apps/download/advanced_search.html

次に/etc/sysctl.conf の net.ipv4.conf.default.rp_filter の値を 1 から 0 に修正し再起動する。

```
$ su
# vi /etc/sysctl.conf
net.ipv4.conf.default.rp_filter = 0
# sync
# sync
# sync
# reboot
```

（4）インストールの確認

以下を実行し、インストールが正常に行なわれたどうかを確認する。

```
# /sbin/ifconfig
ipsec0    Link encap:Ethernet Hwaddr **:**:**:**:**:**
          inet addr:[eth0 と同じ IP アドレス] Mask:[eth0 と同じネットマスク]
          …中略
```

```
# lsmod
Module              Size      Used     by
Ipsec               239024    2
…中略
```

以下を実行し、grep を除いて 6 つのプロセスが動いていることを確認する。

```
# ps -ax | grep Pluto
```

（5） FreeS/WAN の設定

FreeS/WAN を設定するためには、以下に示すように/etc/ipsec.conf に記述する。

```
config setup
        interfaces=%defaultroute
        klipsdebug=none
        plutodebug=none
        plutoload=%search
        plutostart=%search
        uniqueids=yes

conn testvpn
        type=transport
        left=192.168.1.2
        leftid=
        leftsubnet=
        right=192.168.1.1
        rightid=
        rightsubnet=
        auth=esp
        authby=secret
        pfs=yes
        keylife=1h
        ikelifetime=8h
        auto=add
```

また/etc/ipsec.secrets には以下のように記述する。

```
192.168.1.2 192.168.1.1 : PSK “testvpn”
```

9.1.4. Windows 2000 の IPsec 化

Windows プラットフォーム（Windows 2000 Professional, Windows 2000 Server, Windows XP）においては IPsec が標準で実装されているため、新たなソフトウエアを追加する必要はない。以下に Windows 2000 における設定方法を示す。

(a) IPsec の設定 1 – ローカルセキュリティポリシーの設定

（1） [スタート]-[設定]-[コントロールパネル]-[管理ツール]-[ローカルセキュリティポリシー]をダブルクリックする。

（2） 画面左側の[ローカルコンピュータの IP セキュリティポリシー]を右クリックし、[IP フィルタ一覧とフィルタ操作の管理]をクリックする。

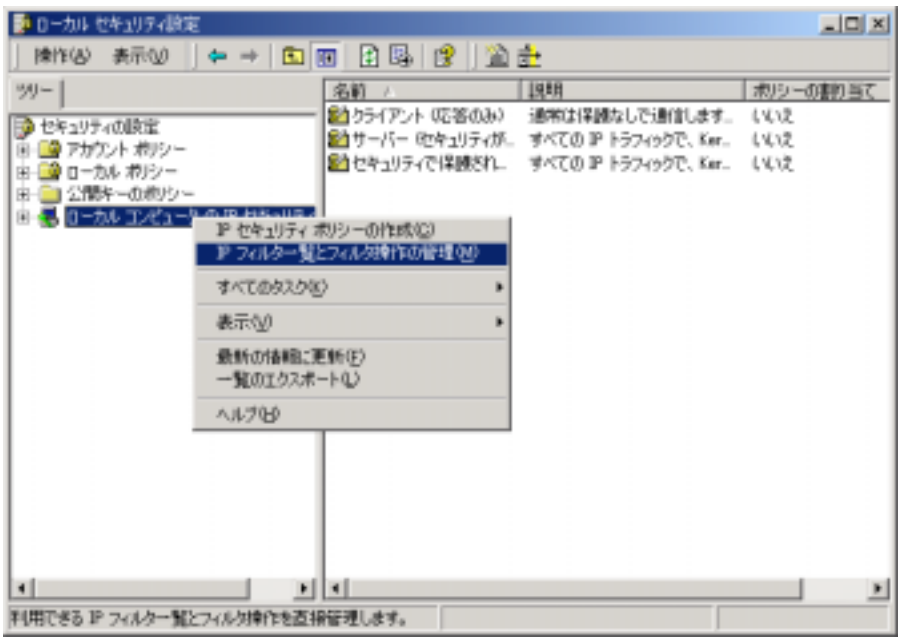

図 9-2　ローカルセキュリティポリシーの設定

（3）「IP フィルター一覧とフィルタ操作の管理」ダイアログが表示される。「IP フィルター一覧とフィルタ操作の管理」ダイアログ内の[追加]ボタンをクリックする。

図 9-3　「IP フィルター一覧とフィルタ操作の管理」ダイアログ

（4）「IP フィルター一覧」ダイアログが表示される。名前に「IP フィルター一覧（to 192.168.1.2）」を入力し、[追加ウィザードを使用]のチェックを確認し、[追加]をクリックする。

図 9-4　「IP フィルター一覧」ダイアログ

（5） フィルタウィザードが実行される。ウィザードを用いて以下を設定する。

① IP トラフィックの発信元: このコンピュータの IP アドレス

② IP トラフィックの宛先: 特定の IP アドレス（192.168.1.2）

③ IP プロトコルの種類: 任意

④ ウィザードの完了: [プロパティの編集]にチェックを付け、[完了]ボタンをクリックする。

（6） 「フィルタのプロパティ」ダイアログが表示される。「フィルタのプロパティ」ダイアログ内の[ミラー化]にチェックが付いていることを確認する。

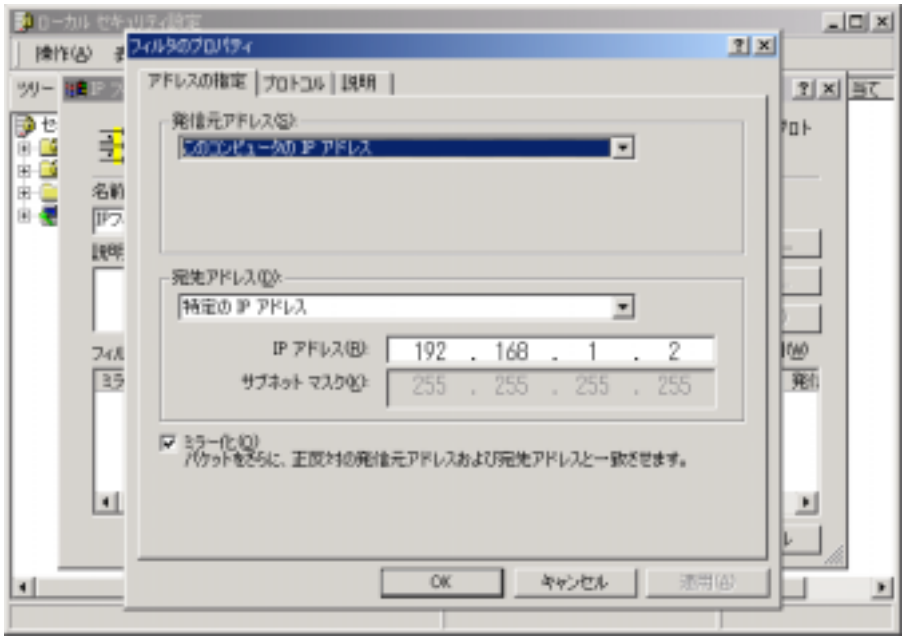

図 9-5 「フィルタのプロパティ」ダイアログ

（7） [OK]ボタンをクリックし、「フィルタのプロパティ」ダイアログを閉じる。さらに[閉じる]ボタンをクリックし、「IP フィルター一覧」ダイアログも閉じる。

（8）「IP フィルター一覧とフィルタ操作の管理」ダイアログ内「フィルタ操作の管理」タブをクリックし、[追加ウィザードを使用]にチェックが付いている事を確認し、[追加]ボタンをクリックする。

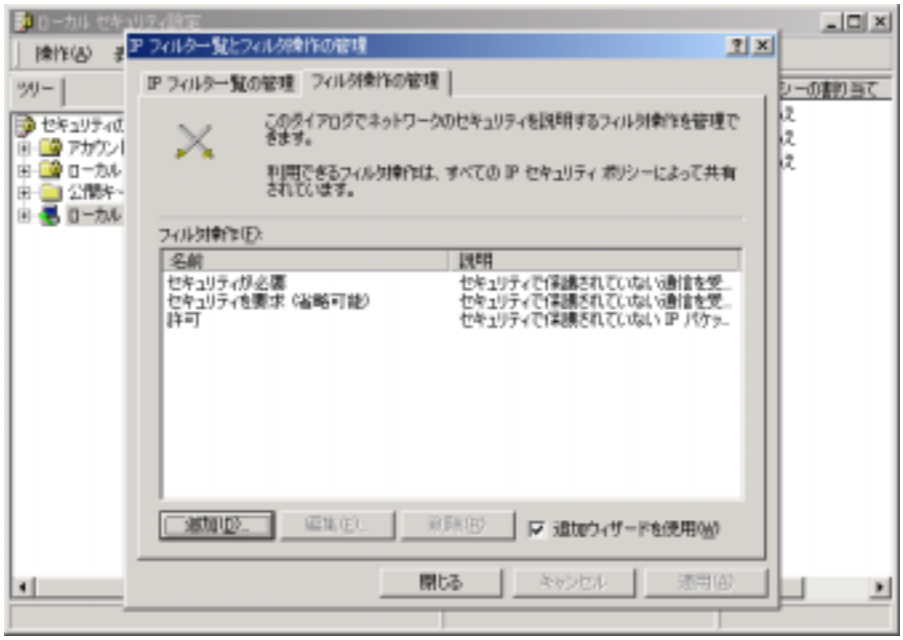

図 9-6　「IP フィルター一覧とフィルタ操作の管理」ダイアログ

（9）「フィルタ操作ウィザード」が表示される。ウィザードを用いて以下を設定する。

① フィルタ操作名:「フィルタ操作（to 192.168.1.2）」と入力する。

② フィルタ操作の全般オプション: セキュリティのネゴシエート

③ IPsec をサポートしないコンピュータとの通信: 通信しない

④ IP トラフィックセキュリティ: カスタムを選択し、[設定]ボタンをクリックする

以下に[設定]の変更項目を示す。

整合性アルゴリズム: SHA1

暗号化アルゴリズム: 3DES

新しいキーの生成間隔（G）にチェックを付け、100000KB とする

新しいキーの生成間隔（R）にチェックを付け、3600 秒とする

[OK]ボタンをクリックする

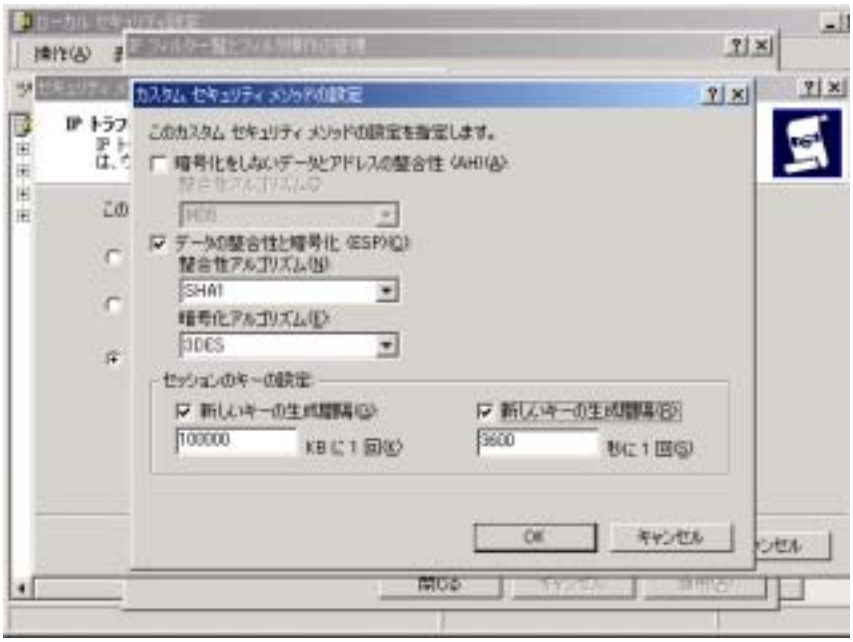

図 9-7　フィルタ操作ウィザード

⑤　ウィザードの完了: プロパティの編集にチェックを付け、[完了]ボタンをクリックする。「新しいフィルタ操作のプロパティ」が表示される。

（１０）[セッションキーの PFS（Perfect Forward Secrecy）]にチェックを付ける。[OK]ボタンをクリックする。[閉じる]ボタンをクリックし、すべてのダイアログを閉じる。

(b)　IPsec の設定 2 – ローカルセキュリティポリシーの作成

（1） 画面左の[ローカルコンピュータの IP セキュリティポリシー]を右クリックし、[IP セキュリティポリシーの作成]をクリックする。「IP セキュリティウィザード」が表示される。

① IP セキュリティポリシー名: 「IP セキュリティポリシー（to 192.168.1.2)」と入力する

② セキュリティで保護された通信の要求: [既定の応答規則をアクティブにする]のチェックを外す

③ ウィザードの終了: [プロパティを編集する]にチェックが付いていることを確認し、[完了]ボタンをクリックする。

（2） 「IP セキュリティポリシー（to 192.168.1.2）のプロパティ」が表示される。「IP セキュリティポリシー（to 192.168.1.2）のプロパティ」ダイアログ内の[追加ウィザードを使用]にチェックが付いている事を確認し、[追加]ボタンをクリックする。

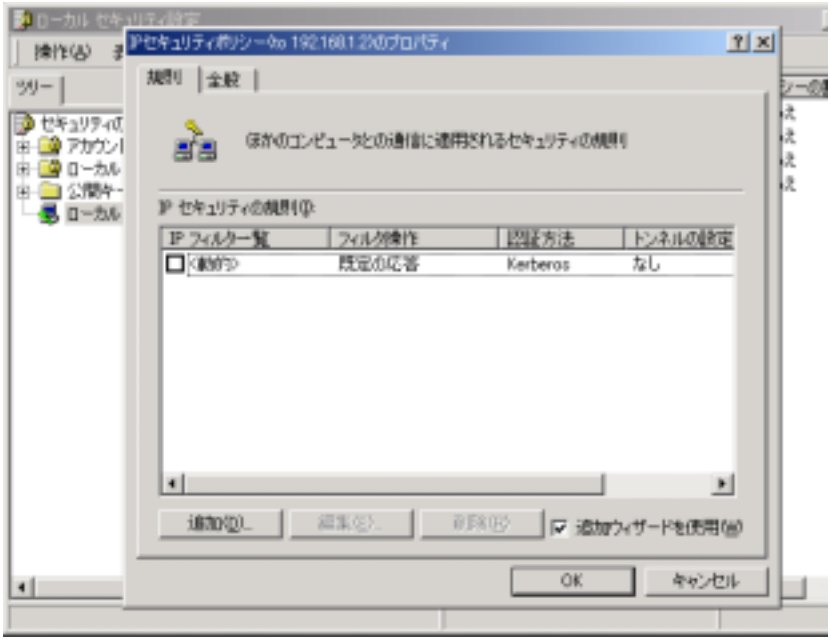

図 9-8　IP セキュリティポリシーのプロパティ

（3）「セキュリティの規則ウィザード」が表示される。以下をウィザードで設定する。

① トンネルエンドポイント: 使用しない

② ネットワークの種類: ローカル エリア ネットワーク（LAN）

③ 認証方式: [次の文字列をキー交換（仮共有キー）の保護に使う]を選択し、テキストボックスに「testvpn」と入力する

④ IP フィルター一覧: 先ほど作成した「IP フィルター一覧（to 192.168.1.2)」を選択する。（右側のラジオボタンがオンになるようにラジオボタンをクリックする。）

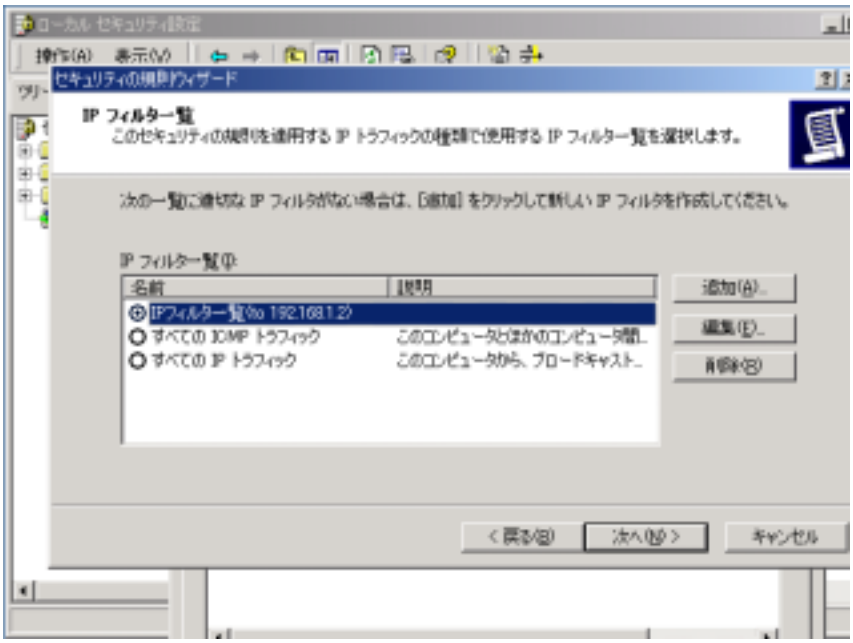

図 9-9　セキュリティの規則ウィザード

⑤ フィルタ操作: 先ほど作成した「フィルター一覧（to 192.168.1.2)」を選択する。（右側のラジオボタンがオンになるようにラジオボタンをクリックする。）

⑥ ウィザードの完了: [プロパティの編集]にチェックが付いていない事を確認し、[完了]ボタンをクリックする。「IP セキュリティポリシー（to 192.168.1.2）のプロパティ」ダイアログが表示される。

（4）「IP セキュリティポリシー（to 192.168.1.2）のプロパティ」ダイアログの[全般]タブをクリックし、キー交換のための設定[詳細]ボタンをクリックする。「キー交換の設定」ダイアログが表示される。

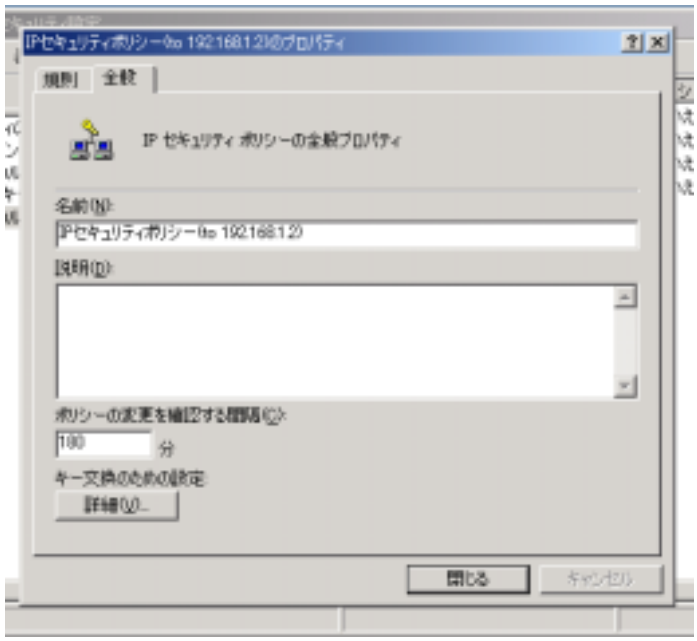

図 9-10　IP セキュリティポリシーのプロパティ

（5）　「キー交換の設定」ダイアログ内の[マスタキーの PFS（Perfect Forward Secrecy）]にチェックを付ける。次に ID の保護に用いるセキュリティメソッド[メソッド]ボタンをクリックする。「キー交換のセキュリティメソッド」ダイアログが表示される。

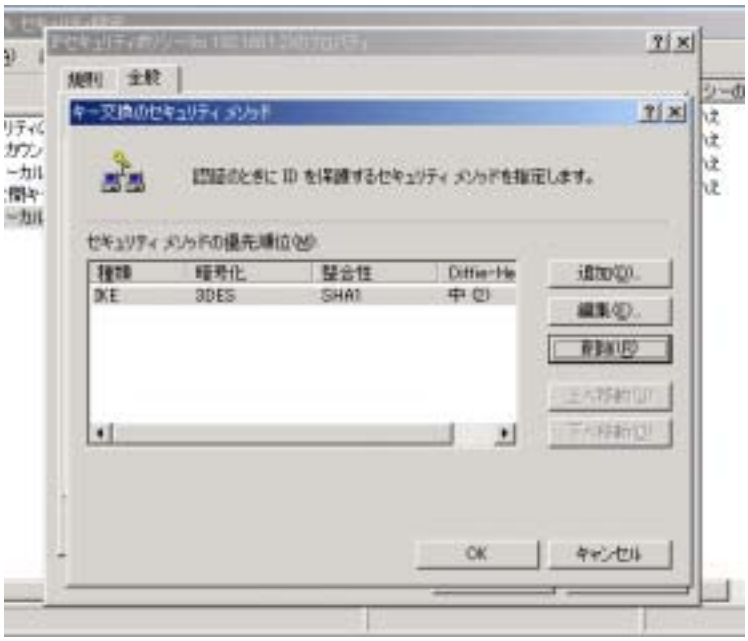

図 9-11　「キー交換のセキュリティメソッド」ダイアログ

（６） 「種類: IKE, 暗号化: 3DES, 整合性: SHA1, Diffie-Hellman グループ: 中（2)」で示されるメソッド以外のメソッドを削除する。

（７） [OK]ボタン、[OK]ボタン、[閉じる]ボタンを順にクリックし、すべてのダイアログを閉じる。

（８） 画面右側の先ほど作成した「IP セキュリティポリシー（to 192.168.1.2)」を右クリックし、[割り当て]をクリックする。画面右側の[ポリシーの割り当て]が「いいえ」から「はい」になることを確認する。うまくいかなかった場合は、作成したポリシー等を削除し、最初からこの手順をやり直す。

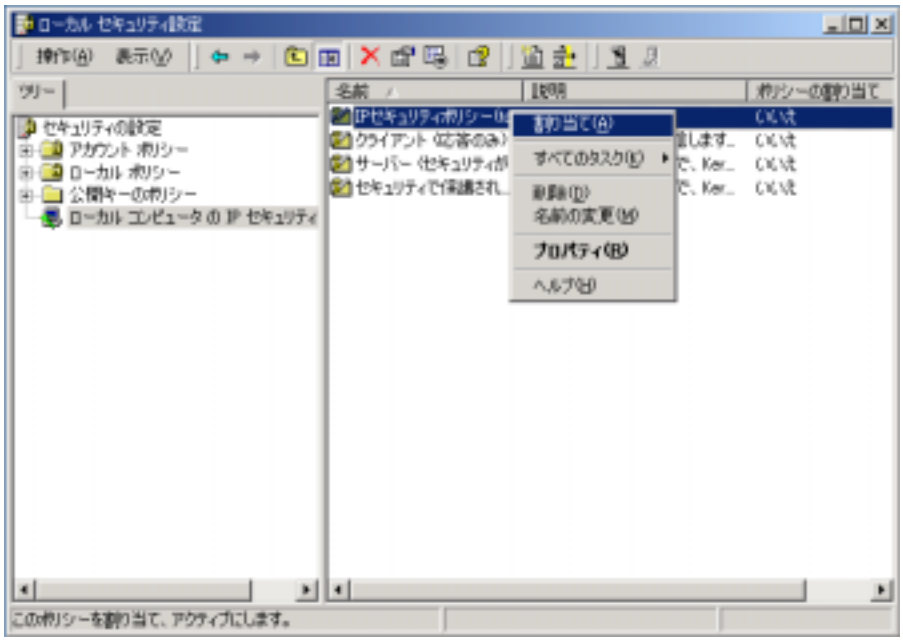

図 9-12　ポリシーの割り当て

9.1.5. 通信の確認

以下の手順で IPsec による通信が正しく行われていることを確認できる。

(a) Windows 2000 における確認の方法

（1） Ping を用いた確認

OS に標準実装されている ping コマンドを用いて確認できる。IPsec のネゴシエーションが行われた後に通信が行われていることを確認する。

```
C:¥> ping 192.186.1.2
Negotiating IP Security.

  中略…（SA の折衝が確立されていないので再度 ping を打つ）

C:¥> ping 192.186.1.2
Reply from 192.168.1.2 byte=32 time<10ms TTL=128

  以下略…
```

（2） ipsecmon

ipsecmon プログラムにより、GUI 上で IPsec コネクションの状況を確認することができる。

(b) Red Hat における確認の方法

Red Hat において IPsec が正しく動作し暗号化が行われている場合には、tcpdump や ethereal などのパケットモニタソフトウエアを用いて通信を捉えた際に、サービスタイプが不明となる。

9.2. ssh による WebDAV システムのセキュア化

ここでは、ssh を用いてセキュアな通信路を介して WebDAV システムを利用する方法を示す。

9.2.1. ssh によるポートフォワーディングの概要

ssh はセキュアなリモートアクセスを可能にするだけでなくポートフォワーディング機能を持つ。これは ssh のクライアントとサーバが作る暗号化通信路を他のアプリケーションに提供し、リモートのサービスへの安全な接続を実現する機能である。

リモートホストの ssh サーバプログラムが指定されたポートに接続し、ローカルホストの ssh クライアントプログラムが内部ネットワークに向けてリモートホストのアプリケーションポートを転送する。ssh サーバと ssh クライアントの間は ssh の機能により暗号化通信路が確保されるため、結果として、暗号化されていない部分がリモートホストおよびローカルホスト内に限定された安全な通信路が確保される。

WebDAV の通信は HTTP 上で行われるが、HTTP は常に同じポートを使用するので、ポートフォワーディングとは相性がよい。ポートフォワーディングは FTP のようにデータポートが変動するプロトコルとは組み合わせが困難である。

9.2.2. ポートフォワーディングの設定

(a) UNIX

UNIX 環境では OpenSSH を用いることで ssh によるポートフォワーディングを容易に実現できる。多くのディストリビューションで OpenSSH は標準的に搭載されている。

OpenSSH で HTTP ポートを転送する際の起動方法を以下に例示する。

```
% ssh –L 10443:www.example.com:443 www.example.com
```

この例では www.example.com の HTTP ポートを、ローカルホストの 10443 ポートに転送している。以後は、https://localhost:10443/という URL において暗号化通信路を用いたセキュアな HTTP アクセスが可能である。

(b) Windows 2000/XP

Windows 環境では、PortFowarder を用いるのが簡単である。PortFowarder はポートフォワーディングに機能を絞った ssh クライアントであり、Web 上で配布されている[7]。

PortFowarder における設定の方法を以下に示す。以下のような設定ファイルを作成する。

```
Host www.example.com
        HostName www.example.com
        User username
        LocalForward 10443 www.example.com:443
```

9.2.3. WebDAV アクセスの方法

ポートフォワーディングを設定した後は､転送を行うローカルのポートを指定することで暗号化通信路を介したセキュアな WebDAV アクセスが可能となる。HTTP を利用する場合と特に違いは無い。

9.2.4. 仮想ホストとの関係

ポートフォワーディングを用いて Web サーバにアクセスした場合、ターゲット URL は localhost 以下のものとなる。そのためデフォルト以外の名前ベースの仮想ホストへのアクセスは行えない。

[7] PortForwarder Home　http://www.fuji-climb.org/pf/JP/

9.3. セキュア通信機構の比較

ここでは、SSL/TLS、IPsec、ssh の 3 つのセキュア通信機構を比較する。

（1） 安全性

通信路の機密性については、用いる暗号アルゴリズムにもよるが 3 つのセキュア通信機構はほぼ同等の機密性を実現できると考えられる。暗号化された通信の盗聴により情報が漏洩する可能性は極めて低い。

接続先ホストに対して認証を行う上では、サーバ証明書の確認が容易に実現できる点で SSL/TLS が他の機構に比べ優れている。

（2） 利便性

不特定多数のユーザによるセキュアなアクセスが困難であるという点では ssh は他の 2 つに比べて利便性の面で劣っていると言える。ssh は接続先に事前に認証用のアカウントを設定しておく必要があるが、SSL/TLS および IPsec を利用した暗号化通信では接続先にアカウントを必要としない。IPsec は通信の際に相手ホストの認証が必要となる。Kerberos、x509 証明書などによる認証を行わなければ IPsec による暗号化通信路は確立されないが、ユーザにかかわる認証は必要無い。

また ssh のポートフォワーディング機能では名前ベースの仮想ホストへのアクセスに問題が発生する。

（3） ユーザから見た透過性

SSL/TLS と ssh はアクセス先のサーバとクライアントとの間で利用されるメカニズムであり、ユーザはこれらを意識的に使用する。これに対して IPsec は通信路の途中でユーザは特にこれを意識せずに利用することが可能である。

以上を総合する。

- SSL/TLS は不特定多数のユーザアクセスをセキュアにすることが容易に実現できる点で優れている。HTTP との相性がよい。
- ssh によるポートフォワーディングは特定のユーザのみが使用する場合の小規模な利用に適している。名前ベースの仮想ホストへのアクセスに問題が発生する。
- IPsec はユーザ環境の変更を伴わずに特定のネットワーク間の通信をセキュアにする場合に適している。

参考文献

[1].Apache HTTP サーバ バージョン 2.0 ドキュメント
http://httpd.apache.org/docs-2.0/index.html.ja

[2].Red Hat Linux 8.0 オフィシャル Red Hat Linux カスタマイズガイド
http://www.redhat.co.jp/manual/Doc80/RH-DOCS/rhl-cg-ja-8.0/index.html

[3].Debian GNU/Linux
http://www.debian.org/

[4].Overview of apache2 source package
http://packages.qa.debian.org/a/apache2.html

[5].MSDN ライブラリ 「WebDAV の発行」
http://www.microsoft.com/japan/developer/library/default.asp?URL=/japan/developer/library/jpiis/core/wcwbdav.htm

[6].セキュアな Web サーバーの構築と運用
http://www.ipa.go.jp/security/awareness/administrator/secure-web/index.html

[7].アップル - ソリューション - Web パブリッシング - WebDAV の設定 「Mac OS X Server の WebDAV を設定」
http://www.apple.co.jp/solutions/webpublishing/technology/WebDAVsetup/

[8].Locking Down WebDAV Through ACL Still Allows PUT and DELETE Requests
http://support.microsoft.com/default.aspx?scid=http://www.microsoft.com/japan/support/kb/articles/307/9/34.asp

[9].Internationalized Resource Identifiers (IRIs) (2003/8/31)
http://www.ietf.org/internet-drafts/draft-duerst-iri-03.txt

[10].FreeS/WAN Project
http://www.freeswan.org/

[11].KAME Project
http://www.kame.net

[12].USAGI Project - Linux IPv6 Development Project -
http://www.linux-ipv6.org/