はじめに

ADSL や FTTH といったブロードバンドサービスの登場によって、以前とは比べ物にならないほど低価格に、インターネットへの常時接続環境が手に入るようになった結果、小規模なオフィスや個人ユーザーでも自サイトに設置したサーバをインターネットへと容易に公開できるようになった。安価なアクセスラインさえあれば、サーバを立てること自体は特別難しいことではない。インターネットへ常時接続された PC を用意して、サーバソフトウェア、たとえば Web サーバ（Windows XP ならば IIS）をインストールすればいいだけだ。

オフィスあるいは自宅でインターネットに接続されたサーバを運用するメリットは何だろうか。簡単に言ってしまえば、情報を 1 か所に集めておけることにある。

FTTH を利用すれば、いまや一昔前の LAN 接続に匹敵する 10Mbps〜100Mbps という通信速度が手に入る。ということはつまり、かつては机を並べなければできなかった作業が、ブロードバンドネットワークにさえつながっていれば、どこにいてもできるようになったということを意味している。在宅勤務者や遠方の支店で働く社員でさえ、その場にいるがごとくグループワークに参加できるということだ。インターネット経由のアクセスには不安があるかもしれないが、Windows XP の標準機能の 1 つである VPN を利用すれば、めったなことでは情報漏えいやクラックの心配はいらない。

もし、クライアント側にダイヤルアップ接続のようなナローバンド接続しか無くても、サーバの用途が無くなるわけではない。出張先のホテルにある無線 LAN ホットスポットを利用したり、出先から PHS を使ってインターネットに接続したりできれば、多少遅くとも、オフィスのファイルサーバに格納されたファイルを直接読み書きできるのだ。出張前

夜に、いそいそと必要なファイルだけノートパソコンにコピーするような真似はもはや必要ないし、ノートパソコンで編集したファイルをいちいちオフィスに戻ってから、サーバにアップロードする手間も不要だ。

ホームユースでも、サーバを公開すれば、おもしろいことはいろいろできる。ユーザーアカウントを管理すれば、クローズドな BBS を設置したり、仲間内で写真を公開したり、プライベートなコミュニティサイトとして利用できる。また、独自のメールサーバを設置すれば、無償でメーリングリストをいくらでも運用することができるし、好みのメールアドレスを使うことも自由にできる。もちろん、単純に技術的好奇心からサーバ構築に乗り出すのもいいだろう。最高の実験環境が手に入るはずだ。

コンピュータは、その処理速度が向上するに従って、用途を無限に広げていった。これと同じく、ブロードバンド化は転送速度の向上だけを意味するわけではない。高速化したネットワークは、その用途を大幅に広げていくはずだ。本書は、こうしたブロードバンドネットワーク環境を利用して、Windows XP Professional がインストールされた PC をサーバマシンとして運用する、実践的環境構築術について解説する。なお、作業手順等は基本的に Windows XP を対象としているが、基本的には Windows 2000 Professional や Windows 2000 Server でも、ほとんど同じことが可能だ。

2002 年 7 月
田口 景介

目次

第1章
安全で実用性の高いサーバ構築

皆さんは「サーバ」という言葉にどんなイメージを持っているだろうか。ネットワークの向こう側に「あるらしい」得体の知れない存在だろうか。しかし、インターネットには「あっち」も「こっち」もないので、サーバから見ればあなたの PC が「ネットワークの向こう側」だ。両者にたいした違いは無い。

何のことはない、サーバとは、ある種のソフトウェアが実行されているごく普通のコンピュータに過ぎないのである。Web ブラウザやメーラなど、クライアント用途に利用する PC と違って年中無休が要求されるサーバ用途のコンピュータには、電源を切らずに故障したハードウェアを交換したり、不具合が発生しても即座にバックアップシステムがフォローしたりする仕組みが用意され、簡単には停止しないように信頼性を向上させるための付加装置が加えられているが、それ以外はごく普通のコンポーネントを組み合わせて作られているのが実際である。

また、サーバだからといって、特別高性能でなければならない理由はない。もちろん一日に数十万ヒットものアクセスを処理する Web サーバにはそれなりのシステムが必要とされるだろうが、ごく少数のユーザーが情報共有のためにアクセスするだけならば、少々型落ちの PC でもサーバ運用には事足りる。オフィスアプリケーションやゲームのほうがよほどマシンパワーを要求するというものだ。

要するに、ごく普通の PC に、ごく普通の OS である Windows XP Professional をインストールすれば、あとはインターネットへの常時接続回線を用意するだけで、サーバが完成するということだ。

もっとも、ただそれだけでは、何の役にも立たないのは言うまでもない。サーバとして運

用するにはソフトウェアをインストールして、それなりのセットアップが必要だ。もっとも、ファイルサーバ機能や、サーバサイドアプリケーションの実行環境は、最初から Windows XP に付属しているので、標準ソフトウェアだけでも、かなりのことが実現可能だ。ただ、本書で解説しているサーバアプリケーションを利用すれば、想像以上に Widnows をサーバとして活用できるようになるはずだ。

また、サーバの管理には、それなりのスキルが要求される。無防備にサーバを設置すれば、インターネットを徘徊するクラッカーの餌食になり、ファイルが盗み見されたり、悪くすれば書き換えられてしまう可能性もある。それがヒミツの日記ならばともかく、顧客リストや友人一同の住所録では笑い話ではすまされない。それに、自分が被害にあっているうちは、まだマシだと言える事態すら起こりうる。たとえば、知らぬ間にあなたのサーバがスパムメールの発信基地として利用されてしまい、取引先や友人達にスパムメールをばらまいてしまったら、どうだろう。ひどく信頼を損ねることになりかねない。また、2001年の夏に流行した Code Red や Nimda のようなインターネットワームに取り付かれ、見知らぬサイトへと攻撃をしかけてしまえば、あなたは被害者でありながら加害者となってしまうのだ。そして、もし先方に重大な損害を与えてしまえば、賠償問題に発展する可能性すらありうる。これはさすがに脅かしすぎかもしれないが、サーバにセキュリティ対策をほどこすのは自分のためだけでなく、サーバ公開者の義務であり、責任である。もし、セキュリティの管理ミスから痛い目にあった経験があるなら、本書の解説が役に立つはずだ。

ところで、一般的にサーバという言葉は、ネットワーク経由でサービスを提供するソフトウェアと、主にサーバソフトウェアの運用目的で利用されるコンピュータの両方の意味で使われているが、本書では基本的にサーバソフトウェアをサーバと呼称する。

1.1　柔軟性の高いサーバ構築

どこからでもアクセス可能な情報基地がほしいだけならば、インターネットには月々数百円から数千円程度でディスクスペースをレンタルするレンタルサーバサービスが数多く提供されており、こうしたサービスを利用するのも選択肢の 1 つである。それでは、こうしたサービスを利用すれば、あえて自前の PC をインターネットに公開し、サーバとして運用する必要はなくなるのだろうか。そんなことはない。自身で管理するサーバには、柔軟性という大きな魅力があるのだ。

1.1.1 プライベートな情報の公開

どこからでもアクセスできたら便利だと思える情報ほど、個人情報であったり、機密性の高い情報であったり、えてして容易に他人には知られたくない、知られてはまずい情報であるものだ。こうした情報を便利だからというだけで、レンタルサーバに保存するわけにはいかないだろう。もちろん情報の機密性を保障するレンタルサーバ事業者はあるし、にわかネットワーク管理者が管理するサーバとネットワークスペシャリストが管理するサーバのどちらが安全であるか、考えるまでも無く明らかだ。しかしそれなりのサービスにはそれなりのコストが発生するのが当然であるし、技術的にはともかく、気分的に割り切って個人情報をレンタルサーバ上に置けるかどうかは無視できない問題だろう。その点サーバ上にデータを保管するだけならば、いざとなれば即座に公開を停止できる安心感がある。

また、たとえば mp3 ファイルに変換したミュージックデータをサーバに保存して、どこからでもネットワーク経由で個人的にストリーミングを楽しみたいとしよう。いちいちファイルをコピーすることなく、学校から自宅の mp3 データを直接聞けたら便利だろうが、これをレンタルサーバで実現しようとすれば、ユーザーアカウントによるアクセス制御を行っていたとしても法的な問題をクリアできるかは微妙な問題であるし、レンタルサーバによっては規約によって拒否される場合もあるだろう。それに曲数が増えてくれば、レンタルサーバへアップロードする手間もばかにならず、データサイズが増えれば相応に料金が発生するだろう。しかし、こうした問題の一切がサーバの活用で解決される。

1.1.2 柔軟なアクセス制御

基本的に Web ページの公開を目的としてサービスされているレンタルサーバは、不特定多数のユーザーからのアクセスを前提としているため、特定のユーザーにのみアクセスを許可することは難しい。アクセス制御をサービスしているレンタルサーバもあるが、柔軟性の点では自前で運用するサーバが遙かに上回る。

サーバを自前で運用すれば、パスワードを設けてユーザーアカウントを持つユーザーにのみアクセスを許可することはもちろん、ファイルやフォルダごとにきめ細かくアクセス可能なユーザーをコントロールすることが可能だ。またユーザーごとにファイルの読み取り、書き込み、作成、削除などの権利を設定することもできる。さらに、複数のユーザーをグループにまとめ、グループごとにアクセス制御を行うこともできる。この柔軟性は多くのレンタルサーバでは得られないものだ。

1.1.3 機能性

CGI スクリプトを利用できるレンタルサーバは珍しくないが、そこで利用できるプログラミング言語は、perl とシェルスクリプト程度に制限されているケースが一般的だ。また、サーバに過大な負荷をかける CGI スクリプトをレンタルサーバで走らせていると、アクセスが制限されたり、悪くすれば使用停止を求められる可能性もある。その点、自前で運用するサーバならば、C 言語だろうが Java だろうが好みのプログラミング言語を利用することができるし、ほかのプロセスに気兼ねすることなく CPU パワーを存分に利用できる。単純に Web ページを公開したり、ファイルサーバとして利用するだけならばレンタルサーバでも事足りるかもしれないが、CGI や ASP などを利用した Web アプリケーションを使いたければ、サーバの自由度の高さは魅力的だ。

また、Web サーバの公開ばかりがサーバの用途ではない。たとえば、メールサーバを運用する場合を考えてみよう。ISP のメールサーバを利用する場合は、1 通あたりのサイズや、メールボックスのサイズに上限が設けられているのが一般的なので、巨大なファイルをメールに添付しようとしてもはじかれてしまうが、自前で運用するサーバならばディスクスペースに余裕があれば事実上メールサイズに上限は無い。さらに、メールが届くと同時に、サーバのディスク上に保管されるため、瞬時にメールの着信を知ることができる。より高度に活用すれば、メーリングリストを主催することも可能だ。

メールサーバと同時に IMAP サーバを設置して、メールを集中管理すれば、さらに便利なメール環境が手に入る。POP の代わりに IMAP を利用してメールボックスからメールを読み出せば、POP のように受信したマシンにメールが分散することがなくなり、すべてのマシンから、同じように既読メールを読めるようになる。

このように、単なるファイルサーバ以上のサービスを実現できるのが、自前でサーバを運用することの大きなメリットだ。

1.1.4 暗号化

データの漏洩を防ぐには、サーバにパスワードを設定するだけでは不十分だ。サーバ自身が攻撃対象にならなくとも、インターネット上をデータが流れれば、どこかで盗み見される可能性を考慮しなければならない。しかし、Windows XP が持つ VPN（Virtual Private Network）サーバの機能を利用すれば、サーバとの通信内容をすべて暗号化し、データの漏洩や改ざんを防ぐことができる。また、Windows XP が持つ Web サーバにサーバ証明書をインストールすれば、通信内容がすべて暗号化されるセキュアサーバ（URL に https://～を指定してアクセスする）として運用することも可能だ。

1.2 具体的な活用例

自前でサーバを運用することがいかに自由度に優れ、可能性を秘めているかおわかりいただけただろうか。しかし、サーバをどれだけ活用できるかは、サーバ管理者の実力しだいであり、TCP/IP ネットワークや Windows XP の理解がそれなりに求められるのは事実である。そこで本書では、次のような利用法を、バックグランドとなる知識の解説を交えながら、具体的な設定方法を解説していく。

●Web サーバを設置して、プライベートな写真などをユーザーアカウントを持つ友人だけに公開する

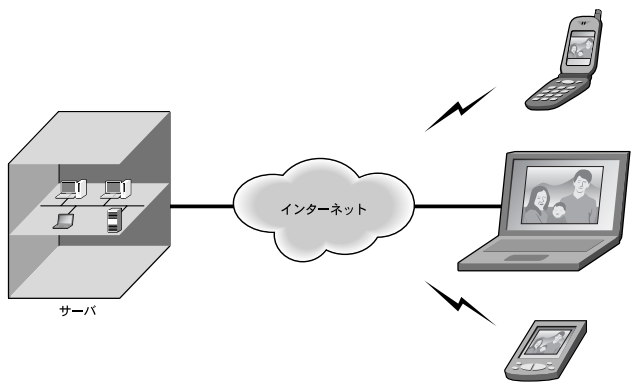

図 1-1　活用例 1

●学校やオフィスなどからインターネットを通じて、サーバ上のファイルを読み書き可能に

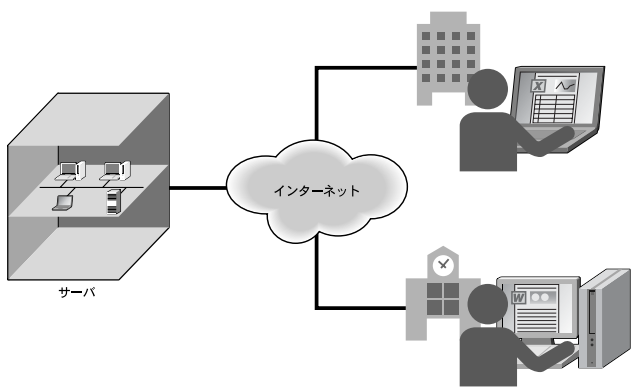

図 1-2　活用例 2

●メールサーバを設置して、自由にメールアドレスを選べるようにする。またメーリングリストを管理する

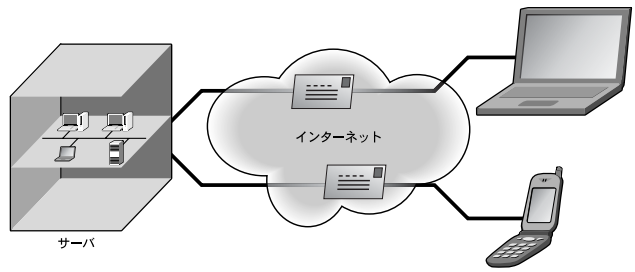

図 1-3　活用例 3

●サーバ上でブックマークを集中管理する

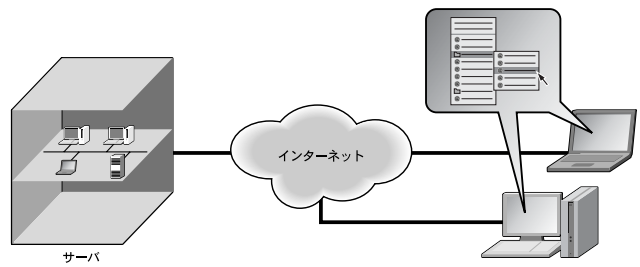

図 1-4　活用例 4

●サーバ上のドキュメントを検索エンジンで高速に検索する

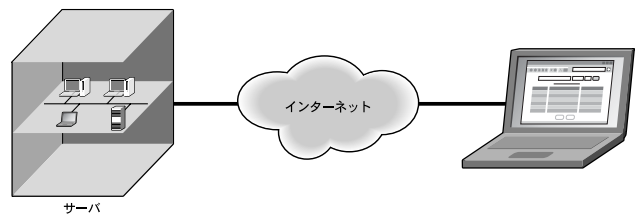

図 1-5　活用例 5

●サーバをメディアサーバに仕立て。オーディオデータやビデオデータをストリーミングで配信する。また検索エンジンを使って、ID3 タグやファイル名から mp3 ファイルを検索する

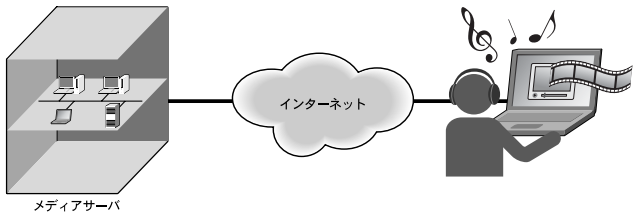

図 1-6　活用例 6

1.3　サーバ公開にはグローバルアドレスが必須

こうしたサーバを構築するには、常時接続サービスを契約するだけではダメで、サーバソフトウェアを PC にインストールする必要があるが、Windows XP Professional には、PC をサーバとして運用するために必要な機能のほとんどが搭載されている。さらに、インターネットで配布されているフリーウェアやシェアウェアを駆使すれば、さらに高度なサーバを構築することも可能だ。常時接続環境を手に入れたら、ぜひとも使いこなしてみよう。ただ、残念ながら Windows XP Home Edition には、サーバソフトウェアが付属していないため、サーバを構築することができない。Windows XP Professional へのアップグレードが必要だ。

また、サーバを構築するには最低限、ISP からグローバル IP アドレスが割り当てられている必要がある。もし ISP から割り当てられた IP アドレスが**表 1-1** に示す範囲にあれば、残念ながらサーバをインターネットに公開することはできない。どうしてもサーバをたてたければ、ISP の変更を検討しなければならない。ただ、現在ではごく一部の CATV ネットワークサービスを除けば、ほとんどの ISP がグローバル IP アドレスを配布しているので、特に問題はないはずだ。

表 1-1　サーバを公開できない IP アドレス（プライベート IP アドレス）

IP アドレス
10.0.0.0～10.255.255.255
172.16.0.0～172.31.255.255
192.168.0.0～192.168.255.255

また、インターネットへの接続に利用するルータにも、サーバ公開用の機能が必須である。一般的にブロードバンドルータと呼ばれている、コンシューマ向けのルータ製品を使ってサーバを公開するには、ポートフォワーディングやバーチャルサーバ、または静的 IP マスカレードと呼ばれる機能が備わっている必要がある。これらの機能は、名前は違えど、ほとんど同じ機能と考えてよい。なおルータを使わずにインターネットへと接続している場合は、容易にサーバを公開可能だが、セキュリティを確保しながらの公開には高度な知識が要求されるため、本書では対象としない。本書では、NAT（Network Address Translation）あるいは IP マスカレードと呼ばれる機能を備えたルータを使ってインターネットへ接続している環境を想定して解説を行う。

それから意外と馬鹿にならないのが電気代である。PC のスペックや運用形態によって大きく上下するため一概にはいえないが、24 時間通電したまま PC を運用した場合、1 月あたり 2000 円～3000 円程度の電気代を見込んでおきたい。これは PC のパフォーマンスが高いほど、また接続されているハードディスクなどの周辺機器が多いほど高くなる。

1.4　セキュリティの確保

自前でサーバを運用することは非常に魅力的だが、おそらく誰もがセキュリティの心配をしているだろう。インターネットにサーバを公開すれば、なんらかの被害に遭う可能性が少なからず増すことは確かである。

2001 年の夏に大流行した Code Red や Nimda のようなインターネットワームは今後も登場し、管理の甘いサーバを荒らして回るはずだし、世の中にはいたずら半分でサーバへのアタックを繰り返す愉快犯が五万といる。あまりに緩い管理をしていれば、こういったユーザーからサーバ上のファイルを読み書きされてしまったり、悪くすればシステムを破壊されてしまったりする可能性が無いとはいえない。

ただし、こうした輩の手口はきわめて単純で、アタックとすら呼べないようなアクセスがほとんどだ。彼らの大部分は簡単にアクセスできる、まったくセキュリティが考慮されていないサーバをあてずっぽうに探しだし、いたずらする程度のレベルでしかない。特に Web サーバに攻撃を仕掛けるインターネットワームの類など、Web サーバへのアクセスに

ユーザー認証を要求するように設定するだけで、ほぼ完全にシャットアウトできる。もし非常に高い技術力を持つクラッカーに目を付けられ、本気で攻撃されれば、多少のセキュリティ対策は破られてしまうかもしれないが、その可能性は非常に低いと言えるだろう。もし本格的な攻撃を受けたとしても、ISP への接続を数十分程度切断しておけば、次に接続したときには IP アドレスが変わっているはずなので、クラッカーが再度あなたのサーバを見つけるのは至難の業だ。

サーバを守るために必要な作業は、ユーザー認証のようなごく簡単なセキュリティ対策と、こまめなログのチェックである。記録したアクセスログに目を通して、怪しげなアクセスを見つけたら、ひとまずルータの設定を変更してサーバへのアクセスを遮断してしまえばよい。それで被害を被る可能性が 0 %になるわけではないが、その程度でも大部分の攻撃は交わせるはずだ。車の窃盗犯はシートの被せられた車は狙わないというが、それと同じだと考えればよい。

セキュリティについて詳しくは第 8 章でまとめて解説しているが、各サーバに依存する対策については、随所で触れていくことにするので、よく目を通してほしい。

冒頭で述べたように、サーバが機能するには、インターネットから常時アクセス可能であること、そしてサーバソフトウェアが適切に構成されていることが最低条件となる。ここまでの解説ですでに常時アクセス環境が整っているはずなので、次にサーバソフトウェアの構成を進めていく。

本書では、主にサーバを次の用途で活用することを目的としている。

●Web サーバ
●ファイルサーバ
●メールサーバ

以上のうち、Web サーバとファイルサーバを支えるのが、Windows XP に付属する統合サーバソフトウェアである IIS（Internet Information Services）だ。本書で解説する大部分の内容に IIS がからむと考えてもらってよい。とはいえ、本書は IIS の解説書ではないし、IIS のすべてを使いこなす必要も無い。IIS は Web サーバ、FTP サーバ、SMTP サーバなどが統合されたサーバソフトウェアだが、本書では Web サーバとして以外は利用しない（FTP サーバの利用法だけは解説する）。

3.1 IIS

Web サーバとして利用すると言っても、IIS を含めて現在の Web サーバは単純に HTML ファイルを配信するだけのプログラムではないので、本書を読み進めていくと、多少 Web

サーバのイメージが変わるかもしれない。たとえば、IIS の ASP（Active Server Pages）と呼ばれるアプリケーションサーバ機能を利用すれば、CGI に比べてより高度な Web アプリケーションの開発が可能だ。また、IIS には WebDAV と呼ばれる、Web サーバを通してファイルアクセスを行うためのプロトコルが実装されているため、IIS をファイルサーバとして運用することができる。今や Web ページの配信は Web サーバが担う仕事のほんの一部に過ぎないのだ。結局のところ、本書におけるサーバの使いこなしとは、IIS の使いこなしであると言っても過言ではない。

そこで、本書では IIS 以外のサーバソフトウェアも利用するが、まずは IIS のセットアップから始めることにする。なお、本節の指示に従って IIS をセットアップしても、次節で解説するように、ルータでポートフォワーディングの設定を行わなければ、まだインターネットからサーバ（＝ IIS）へアクセスすることはできないので、安心して作業を進めてほしい。

なお Windows XP をインストールした直後は IIS がインストールされていないので、まずはコントロールパネルの［プログラムの追加と削除］を開き、［Windows コンポーネントの追加と削除］から［インターネットインフォメーションサービス（IIS）］をインストールし、それから続きを読み進めてほしい。IIS をインストールするとコントロールパネルの「管理ツール」に「インターネットインフォメーションサービス」が追加され、ここから IIS を構成できるようになる（以後これを IIS の管理ツールと呼ぶ）。

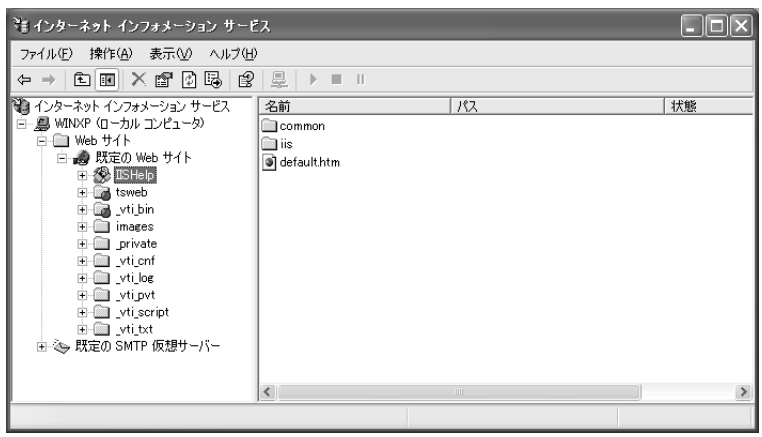

図 3-1　IIS の管理ツール

3.1.1 IIS による Web ページの公開

今でこそ Web サーバは複雑な処理系を背後に抱えた巨大なサーバソフトウェアに成長しているが、Web サーバ本来の姿は URL で指定されたファイルを Web ブラウザへ送り返す、一種のファイル転送サーバでしかない。

URL とサーバ上のファイルをマッピングする仕組みも極めて単純だ。ルートディレクトリとして指定されたディレクトリを基準として、そのサブディレクトリの構成がそのまま URL に反映される。IIS をインストールした直後は「¥inetpub¥wwwroot」がルートディレクトリとして設定されているので（ドライブは Windows XP のシステムドライブと同一）、URL に「http://＜ホスト名＞/home/server/index.htm」を指定すれば、「¥inetpub¥wwwroot¥home¥server¥index.htm」がそのまま何の加工もされずに Web ブラウザへと送信される。つまり IIS で Web ページを公開したければ、「¥inetpub¥wwwroot」以下にファイルをコピーするだけでよい。

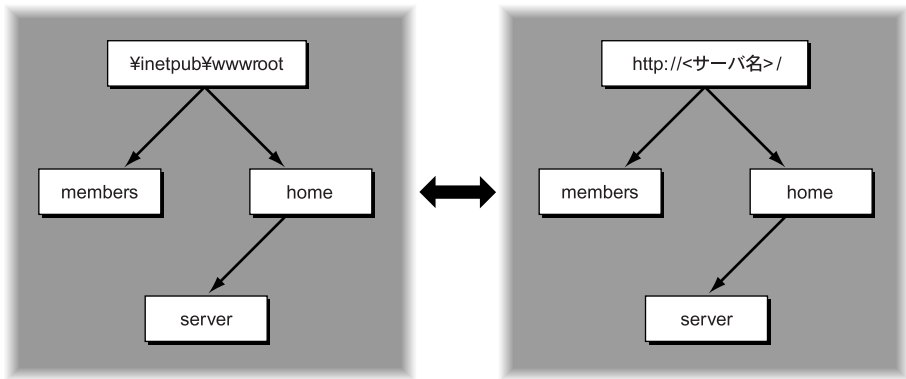

図 3-2　ディレクトリツリーと URL パス

ルートディレクトリを変更する

次の手順で、IIS のルートディレクトリを「¥inetpub¥wwwroot」から変更することができる。

1. まず IIS の管理ツールで、「既定の Web サイト」のプロパティを開く。
2. ［ホームディレクトリ］タブを選択して、［ローカルパス］に新しいルートディレクトリを入力する。

こうしてルートディレクトリを変更すると、「¥inetpub¥wwwroot」以下に格納されていたファイルにはアクセスできなくなるが、仮想ディレクトリは維持されるため、そちらに影響はない。

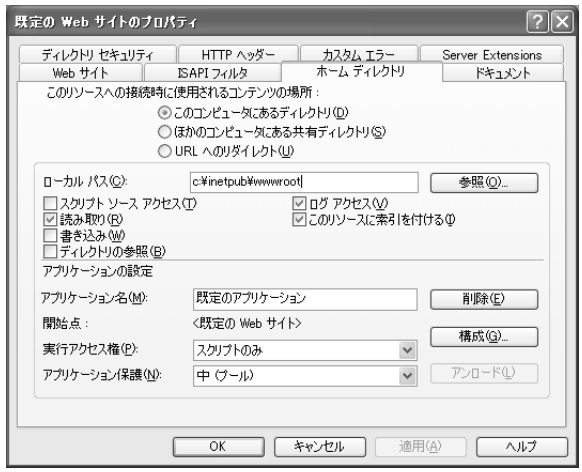

図 3-3　ルートディレクトリの変更

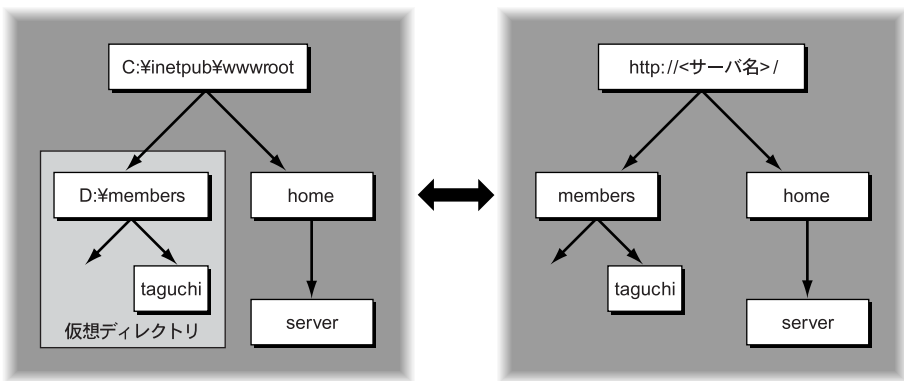

図 3-4　仮想ディレクトリ

IIS ではルートディレクトリ以下のファイルが公開されると述べたが、それだけではいささか都合が悪いこともあるはずだ。たとえば、公開したいファイルが普段はシステムドライブ以外のドライブに保存されている場合や、システムドライブの空き容量が残り少ない

のでほかのドライブにファイルを保存したい場合などだ。このような場合には、「仮想ディレクトリ」を作成することで対応できる。仮想ディレクトリを作成すると、指定した任意のディレクトリがルートディレクトリ以下にある「ふり」をして、ルートディレクトリ以下にないディレクトリを IIS で公開できるようになる。たとえば、「D：￥members」を仮想ディレクトリ「/members」として指定すれば、URL に「http://＜ホスト名＞/members/」が指定されると、「￥inetpub￥wwwroot￥members」ではなく、「D：￥members」以下のファイルがアクセスされるようになる。なお仮想ディレクトリを作成すると、サブディレクトリを含めて公開対象となる。

具体的に仮想ディレクトリを作成する手順を示す。

1. IIS の管理ツールを使って、仮想ディレクトリを作成したい位置をディレクトリツリーから選択する（ルートディレクトリ直下に作りたければ［既定の Web サイト］を選択する）。
2. メニューから［操作］－［新規作成］－［仮想ディレクトリ］を実行する。
3. ウィザードが起動されるので、［エイリアス］に仮想ディレクトリの名前を、［ディレクトリ］に物理的なディレクトリパスを、それぞれ入力する。

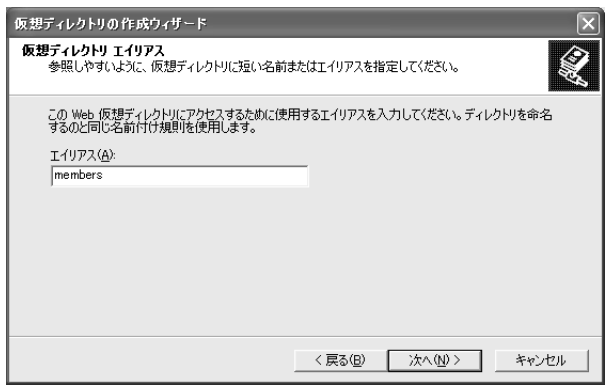

図 3-5　［エイリアス］の指定

4. ［アクセス許可］の設定があるが、ここではデフォルトの［読み取り］と［ASP などのスクリプトを実行する］が選択された状態のまま［次へ］をクリックして、仮想ディレクトリの作成を終える。細かなアクセス制御については、追って解説する。

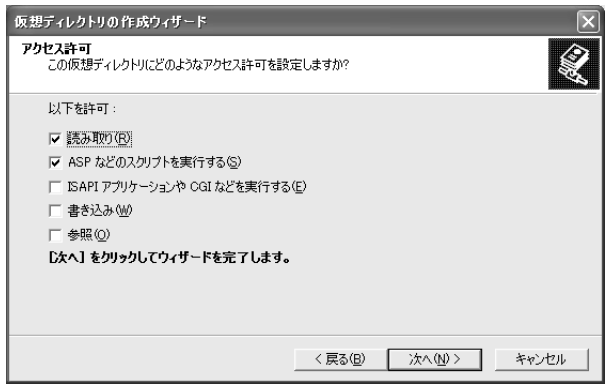

図 3-6　［アクセス許可］の設定

仮想ディレクトリを削除したければ、IIS の管理ツールで目的の仮想ディレクトリを選択し、メニューから［操作］－［削除］を実行すればよい。こうして仮想ディレクトリを削除しても、仮想ディレクトリに指定されていたディレクトリ以下のファイルまで削除されることはない。

3.1.2　ディレクトリのアクセス制御

このようにデフォルトの状態では、ルートディレクトリと仮想ディレクトリ以下に格納されたすべてのファイルが、自動的に公開される仕組みになっている。これはファイルをコピーするだけで手軽に公開できる反面、都合の悪いファイルまで公開されてしまうことも意味している。

Web サーバを管理していれば、あるディレクトリ以下を公開したいが、その中の一部のディレクトリ、たとえば書きかけの HTML ファイルを格納しているディレクトリだけは除外したいと思うことが必ずあるはずだ。こうした目的のため、IIS では各ディレクトリごとに、細かく公開、非公開の設定が可能になっている。

指定したディレクトリ以下を非公開にしたければ、次の手順で作業を行う。まず、IIS の管理ツールで目的のディレクトリを右クリックして、メニューから［プロパティ］を実行する。すると、**図 3-7** のダイアログボックスが開く。ここでは、個々のディレクトリに独立して適用される、アクセス制御などさまざまな設定が可能だが、ディレクトリを非公開にするには、［読み取り］のチェックを外せばよい。

図 3-7　ディレクトリのプロパティ

ファイルのプロパティ

あまり使い道は無いかもしれないが、IIS ではファイル単位で公開、非公開を設定することもできる。

そのためには IIS の管理ツールから、右側のペインで目的のファイルを選択して、［プロパティ］ダイアログボックスを開く。するとディレクトリのプロパティと同様にダイアログボックスが表示されるので、［読み取り］のチェックを外せば、非公開に設定できる。

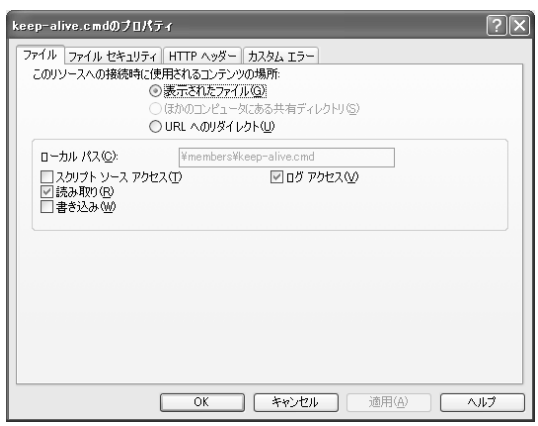

図 3-8　ファイルのプロパティ

以上の操作で、指定したディレクトリ以下のファイルをすべて非公開に設定することができるが、ここで注意してほしいのは、ディレクトリプロパティの設定は、指定したディレクトリだけでなく、そのサブディレクトリにも継承されるということだ。たとえば、ディレクトリ「/home」を読み取り不可に設定すると、自動的に「/home/server」も読み取り不可になるということだ。IIS では、特別指定されていない設定項目については、すべて「親ディレクトリの設定に従う」仕組みになっているからだ。IIS をインストールした直後は特殊なディレクトリを除いて、すべてのディレクトリはデフォルト状態にあるため、ルートディレクトリの設定を変更すれば、それがすべてのディレクトリに反映される。ただし、個別に設定されたディレクトリがあれば、そのディレクトリ以下にルートディレクトリの設定が反映されることは無い。

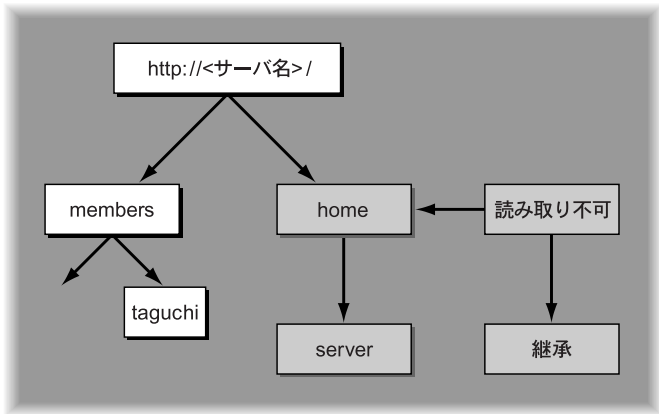

図 3-9　ディレクトリプロパティ

この仕組みは読み取り設定だけに限らず、ディレクトリごとに設定可能なすべての設定に適用される。たとえば、「ログアクセス」の設定も同じくサブディレクトリへと継承される。IIS はクライアントからのアクセスをログに残すことができるが、「ログアクセス」のチェックを外したディレクトリへのアクセスはログから除外できる。また「ディレクトリ」タブ以外でも、ディレクトリプロパティダイアログボックスで設定される項目はすべて、サブディレクトリへと継承される。

画像ファイルは別ディレクトリに格納すべし

デフォルトの設定では、IIS はすべてのアクセス記録をログに残すように設定されているため、Web ページ（HTML ファイル）だけでなく、ページにインライン表示される画像ファイルなどもログに記録されてしまう。つまり、1 ページに 10 個のインライン画像があれば、1 ページがアクセスされただけでも、11 件のアクセス記録がログに残されることになる。多くの場合、これは無駄なログでしかないだろう。

そこで、こうしたインライン画像は別ディレクトリ（例：images ディレクトリ）に格納し、images ディレクトリの［ログアクセス］プロパティのチェックを外しておけばよいだろう。

図 3-10　ログアクセス

IIS の継承モデルはごく少数のディレクトリで設定を集中管理できるスマートな仕組みだが、同時に見通しの悪さも併せ持っている。IIS 管理ツールのプロパティダイアログボックスでは、現在の設定値を確認することしかできず、それがディレクトリで個別に設定されているのか、親ディレクトリの設定を継承しているのか、まったくわからないからだ。このため、無計画に各ディレクトリの設定を変更していくと、後で必ず大変な混乱を招くはめになるので、十分注意が必要だ。設定を変更したときには、あとでわかるようにメモを残しておいたほうがよいだろう。

ところで、ディレクトリのプロパティを変更すると、**図 3-11** のダイアログボックスが表示されることがある。ここで［子ノード］に一覧表示されているのは、サブディレクトリ

の中にある、個別設定されたプロパティを持つディレクトリである。ここで選択したディレクトリは、個別設定が取り消され、親ディレクトリの設定、すなわち今設定を変更したプロパティ値を継承する状態に戻される。選択しなかったディレクトリは、個別設定がそのまま維持される。

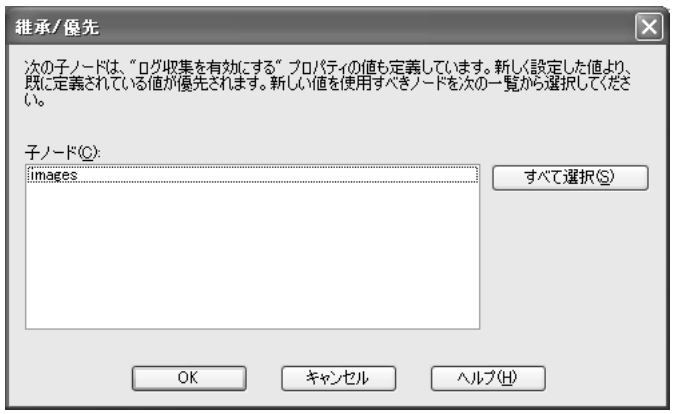

図 3-11　個別設定の取り消し

Metabase Editor

本文で解説したように、非常に見通しの悪いシステムとなっている IIS の継承システムだが、以下の URL から入手できる Metabase Editor と呼ばれるツールを利用すれば、どのディレクトリが個別に設定されたプロパティ値を持っているのか、簡単に調べることができる。

```
http://support.microsoft.com/default.aspx?scid=kb;en-us;Q232068
http://download.microsoft.com/download/iis50/Utility
 /5.0/NT45/EN-US/MtaEdt22.exe
```

ただし、Metabase Editor はいわば IIS 用レジストリエディタのようなものなので、不用意に設定を変更すると、正常に IIS が機能しなくなる危険性があるので、利用には十分な注意が必要だ。

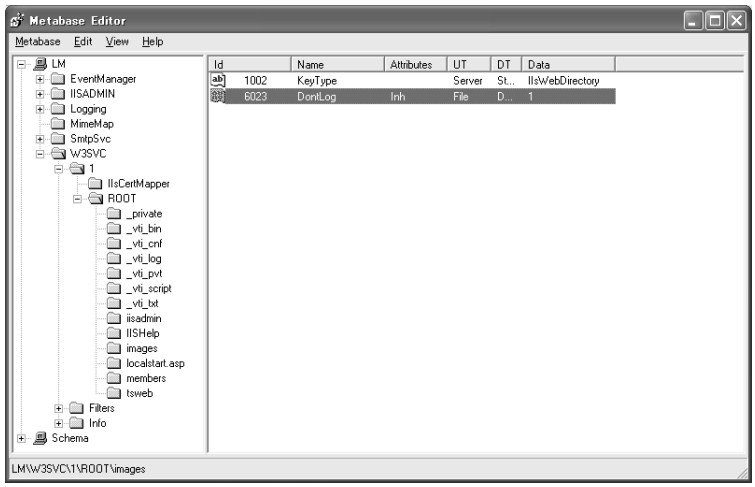

図 3-12 Metabase Editor

3.1.3 デフォルトドキュメントとディレクトリの参照

WWW の世界に親しんでいれば、「http://hostname/home/server/」のようにファイル名を省いて URL を指定すると、index.html のようなデフォルトドキュメントが Web サーバから送り返されることはよくご存知のことだろう。IIS にももちろんデフォルトドキュメントの仕組みは実装されており、初期状態では default.htm がデフォルトドキュメントとして扱われるように設定されている。ただし、IIS では複数のデフォルトドキュメントを優先順位をつけて指定することが可能で、default.htm が見つからなかった場合には、default.asp がアクセスされるように設定されている。

しかし、インターネットの世界では、どちらかと言えばデフォルトドキュメントのファイル名は index.html のほうが一般的で、HTML 形式で配布されているドキュメントをサーバで公開したいときなど、デフォルトドキュメントを index.html に変更しておいたほうが便利なこともある。

デフォルトドキュメントのファイル名を変更したいのであれば、ディレクトリプロパティの［ドキュメント］タブで設定が可能だ。ここで index.html を最上位に追加すれば、default.htm が同じディレクトリにあっても、index.html が優先してアクセスされるようになる。なお、デフォルトドキュメントのファイル名もアクセス制御と同じく、ディレクトリプロパティの一部として格納されるため、サブディレクトリに継承される。したがって、

サイト全体のデフォルトドキュメントを変更したければ、ルートディレクトリの設定を変更すればよい。

図 3-13　デフォルトドキュメントの変更

それでは、デフォルトドキュメントが見つからなかった場合は、どうなるのだろうか。それは、ディレクトリプロパティの「ディレクトリの参照」の設定に依存する。ここにチェックが入っているディレクトリがアクセスされた場合には、IIS によって**図 3-14** に示すファイル一覧が表示される。チェックが外されていれば、「このページの表示が認められていません（HTTP エラー 403 - アクセス不可）」が Web ブラウザに表示される。デフォルトでは、すべてのディレクトリで「ディレクトリの参照」は無効に設定されているので、明示的に設定しない限りは、ファイル一覧が表示されることはない。なお、デフォルトドキュメントが格納されているディレクトリならば、「ディレクトリの参照」にチェックが付けられていても、ファイル一覧が表示されることはない。

ディレクトリの参照を利用すれば、いちいちリンク集のような Web ページをデザインすることなく、Web サーバ上のファイルを公開できるため、アーカイブファイルの公開などには便利な機能だが、不用意に利用すると、予期しないファイルまで公開されてしまうので注意が必要だ。また、これまでに解説したディレクトリプロパティと同じく、「ディレクトリの参照」もサブディレクトリに継承されることを忘れてはならない。

ところで、理解して使えば、手軽で便利なディレクトリの参照だが、ページのデザインや機能は、いまひとつだ。そこで、第 6 章では、より高機能で見栄えのよいファイル一覧を表示する Web アプリケーションを紹介する。

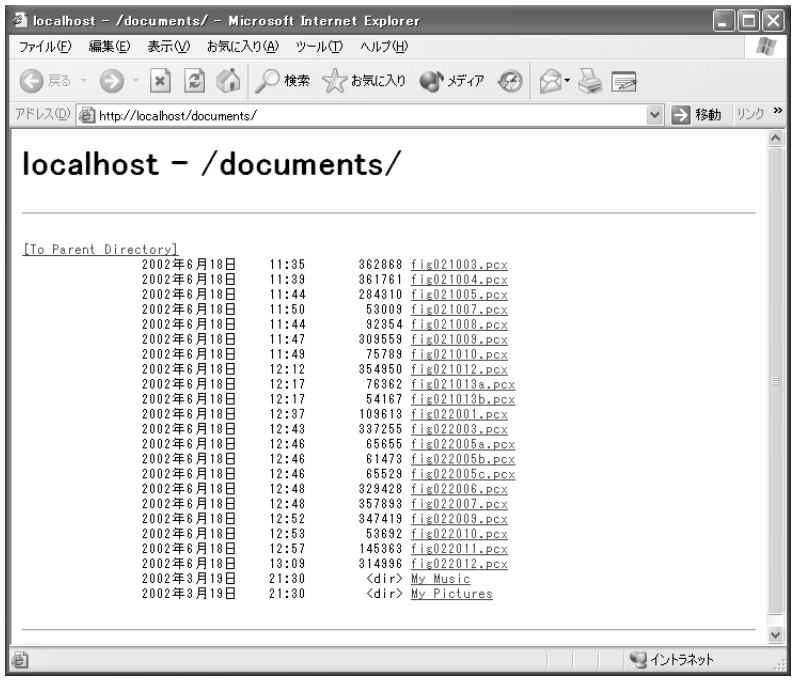

図 3-14　ディレクトリの参照

3.1.4　ルートディレクトリのデフォルトドキュメント

IIS のデフォルトドキュメントは基本的に default.htm に設定されているが、初期状態ではルートディレクトリに default.htm が用意されていない。それでも、「http://localhost/」へアクセスすると Web ページが表示されるのは、ルートディレクトリに、デフォルトドキュメントとして「iisstart.asp」が指定されているからだ。この Web ページを開くと、ページ中に記述されているスクリプトによって、ローカルホストからアクセスされたときには IIS のリファレンスマニュアルが、リモートホストからアクセスされたときには「作成中」のページが表示される仕組みになっている。

ただ、ルートディレクトリでも、より優先順位の高いデフォルトドキュメントとして default.htm が指定されているので、このファイル名でトップページを用意すれば、iisstart.asp がデフォルトドキュメントとしてアクセスされることはなくなる。

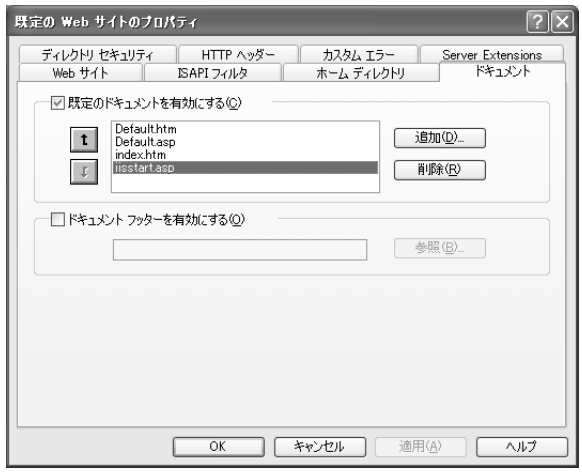

図 3-15　ルートディレクトリのデフォルトドキュメント

3.2 ユーザー認証

IIS をインストールした直後は、ユーザー認証を要求しない、匿名アクセスが可能な状態に設定されている。自前のサーバを ISP などの Web スペースの代わりに、自由度の高い Web サーバとしてパブリックに公開するのであれば、デフォルト設定のままでも構わないが、不特定多数のユーザーにアクセスを許可するとどうしてもセキュリティが甘くなりがちで、サーバへの侵入を許すことになりかねない。特別な理由がない限りは匿名アクセスを無効にし、特定のユーザーだけがアクセスできるクローズドなサーバとして運用することをお勧めする。ユーザー認証を設定しておけば、Code Red や Nimda のようなインターネットワームや、遊び半分で探りを入れてくる不埒な輩からの攻撃にはかなり有効であるし、今後 IIS にセキュリティホールが発見されたとしても、いきなり被害にあう可能性ははるかに低くなるはずだ。

さらに、ユーザー認証を行えば、アクセス可能な Web ページやファイルをユーザーごとに細かく制御できるようになる。たとえば、特定のプロジェクトの情報を掲載したページは、メンバにだけアクセスを許可するような設定が可能だ。前述したディレクトリプロパティやファイルプロパティでも公開、非公開は設定できたが、ユーザーを特定したアクセス制御を行うには、ユーザー認証が必須である。ただし、ユーザー認証後のアクセス制御は、Web ページが NTFS パーティション上に格納されていることが条件となるので、以降の解説は NTFS を前提として進めていく。

3.2.1 認証方式

Windows XP Professional の IIS では、認証方式を次の 2 種類から選択して、設定することができる。

●基本認証
●統合 Windows 認証

基本認証は HTTP の一部として定義されている、WWW の標準的な認証方式である。そのため、どんなクライアントでも利用可能だ。ただし、パスワードがほとんどクリアテキストに近い状態でネットワークを流れるため、パスワードが盗み見られる可能性がある。

一方、統合 Windows 認証は Microsoft 独自の認証方式であり、基本的に InternetExplorer など Microsoft 製のクライアントでしか利用することができない。その代わりパスワードが漏洩する心配はほとんどない。また、Windows へのログオン時に入力したユーザーアカウントとパスワードが自動的に IIS へのアクセス時にも利用されるため、基本認証のように、アクセスのたびにユーザーアカウントを入力する手間を省くことができる。ただし、クライアント側にプロキシサーバが設置されていると、正常に認証できなくなるという欠点も持ち合わせている。また、クライアントとサーバの両方で、アカウント名とパスワードを一致させておかなければならないなど、その運用には制限もある。

セキュリティを考慮すれば当然統合 Windows 認証を選択したいところだが、プロキシサーバを通過しないのでは、ほとんどのオフィスからはアクセス不能になってしまうだろう。プロキシサーバ経由のアクセスを考慮せずに済むならば統合 Windows 認証を選択し、プロキシサーバがあれば基本認証を選択することになろう。なお、基本認証でもサーバ証明書を入手して、SSL を導入すれば、パスワードの漏洩を心配することなくアクセスできるようになる。この方法については、「第 8 章 セキュリティ」で解説する。

認証方式の設定はルートディレクトリで行い、サブディレクトリに継承させるのがいいだろう。そうすれば、サイト全体がユーザー認証を受けなければアクセスできないように設定される。

認証方式を設定するには、IIS の管理ツールでルートディレクトリのプロパティを開き、［ディレクトリセキュリティ］タブを選択し、［編集］ボタンをクリックする。すると初期状態では［匿名アクセス］と［統合 Windows 認証］にチェックが入ってるはずなので、［匿名アクセス］のチェックを外し、［基本認証］または［統合 Windows 認証］のどちらか利用する認証方式にチェックをつける。

以上の作業で、サイト全体が指定した認証方式で守られるようになる。なおすでに述べたように、特定のディレクトリに［匿名アクセス］を指定すれば、そのディレクトリ以下

は認証を要求されること無くアクセスできるようになる。

Windows XP Professional では使えないダイジェスト認証とは？

IIS では、基本認証と統合 Windows 認証に加えてもう 1 つ、ダイジェスト認証と呼ばれるユーザー認証方式が用意されている。このダイジェスト認証を利用すれば、統合 Windows 認証と同じくパスワードの漏洩を防ぐことができるうえ、HTTP 1.1 で正式に定義されている標準認証方式であるため、最近の Web ブラウザやプロキシサーバならば、ベンダを問わずにアクセスすることができる。ただ残念ながら、Windows XP Proffesional に搭載されている IIS では利用することができず、Windows 2000 Server や今後発売が予定されている Windows .NET Server でなければ、利用することができない。なお、ダイジェスト認証はドメインコトローラでのみ使用できる。

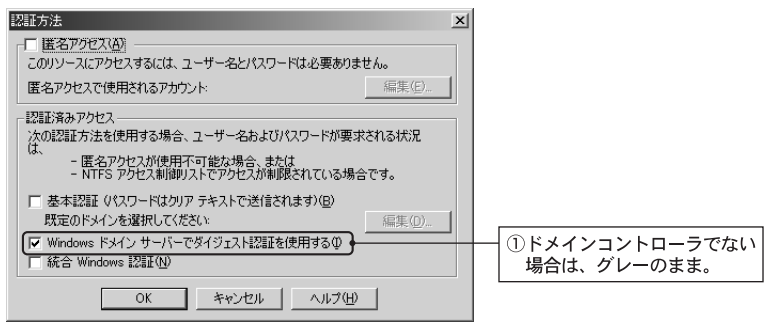

図 3-16　ダイジェスト認証

3.2.2 ユーザーアカウントの管理

基本認証または統合 Windows 認証を設定すると、IIS へアクセスしたとき、ユーザー名とパスワードの入力が要求されるようになる。このユーザー認証に用いられるユーザーアカウントは、Windows XP へのログオンに用いるものと同じものだ。IIS は Windows XP のユーザー認証システムと密接に統合されているため、ユーザー認証とその後のアクセス制御、それに第 6 章で解説する Web アプリケーションの実行時権限など、IIS を通して行われるすべてのアクセスには、Windows XP のユーザーアカウントが用いられる。このため IIS 自身には、ユーザーアカウントを管理する機能が備わっていない。

このため、IIS へのアクセスを許可するユーザーアカウントを作成するには、コントロールパネルの［ユーザーアカウント］、または［管理ツール］－［コンピュータの管理］を利

用する。作成したユーザーアカウントは、ひとまず Users グループのメンバとして登録しておけばよいだろう。これは［コンピュータの管理］でユーザーアカウントを作成したときのデフォルトの動作である。［ユーザーカウント］で作成するときには、［制限付きアカウント］を選択すればよい。

ユーザーアカウントが所属するグループについては、アクセス制御に深く関係するので、おって解説していくことにする。

図 3-17　ユーザーアカウント

ローカルログオンを拒否する

IIS の認証システムと Windows の認証システムが統合されていることで、ファイルシステムを利用したアクセス制御が実現しているわけだが、弊害もある。それは、IIS へのアクセスだけを許可したいユーザーでも、コンソールからログオンする権限が与えられてしまうことだ。セキュリティを重視するのであれば、これは大きな問題だ。

この問題は解決可能だが、統合 Windows 認証を選択していることが条件となる。その場合は、コントロールパネルの［管理ツール］－［ローカルセキュリティポリシー］を実行して、［ローカルポリシー］－［ユーザー権利の割り当て］－［ローカルでログオンを拒否する］に、ユーザーを登録すればよい。そうすれば、登録されたユーザーは、IIS へアクセスできるが、ローカルにログオンすることはできなくなる。

なお、基本認証や匿名アクセスを利用している場合は、ローカルログオンの権利がないと、IIS にアクセスすることもできないので、この手は使えない。

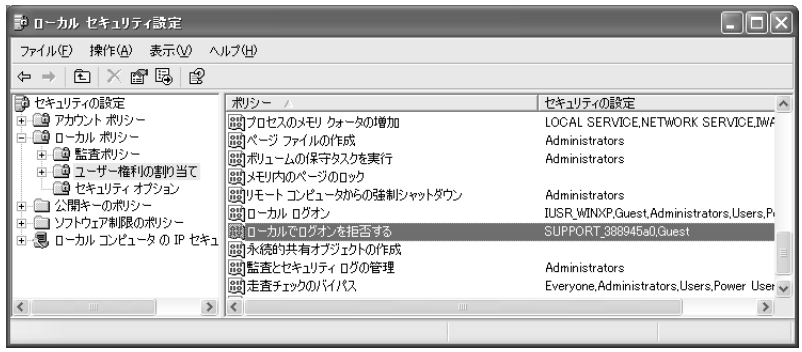

図 3-18　ローカルログオンを拒否する

IUSR_＜ホスト名＞アカウント

匿名アクセスが許可されている IIS ではユーザー認証が行われないため、誰がアクセスしているかわからないわけだが、この場合アクセス制御には無条件でユーザーアカウント「IUSR_＜ホスト名＞」が用いられる（例：ホスト名が WINXP ならば、IUSR_WINXP）。このユーザーアカウントは IIS をインストールすると自動的に作成されるもので、Guests グループのメンバとして登録されている。

3.3　ユーザー認証に基づくアクセス制御

ここまでに IIS へのアクセスを制限する 2 つの要素を解説した。ユーザー認証と IIS のディレクトリプロパティである。IIS へのアクセスが行われると、まずユーザー認証が行われ、これに成功するとサイト全体へのアクセス権が与えられる。次に、指定された URL に対応するディレクトリのプロパティが調べられ、読み出し許可が与えられていれば、そのディレクトリに格納されたファイルへのアクセス権が与えられる。このディレクトリへのアクセス権は、ユーザー認証の結果とは無関係に、ディレクトリプロパティの値のみを元に判断される。

こうしてサイトへのアクセス権とディレクトリへのアクセス権を手に入れても、まだ許可は下りない。これから解説するファイルシステムへのアクセス権を手に入れれば、ついに URL で指定されたファイルへのアクセスが許可され、ブラウザへと送信される（**図 3-19**）。

とても複雑に感じられるかもしれないが、ユーザー認証さえ成功すれば、その後は好きにアクセスしてもらって構わない、というポリシーで運営するのであれば、こうしたアクセス制御の仕組みを意識する必要は無い。初期状態では、ユーザー認証以外はすべてオー

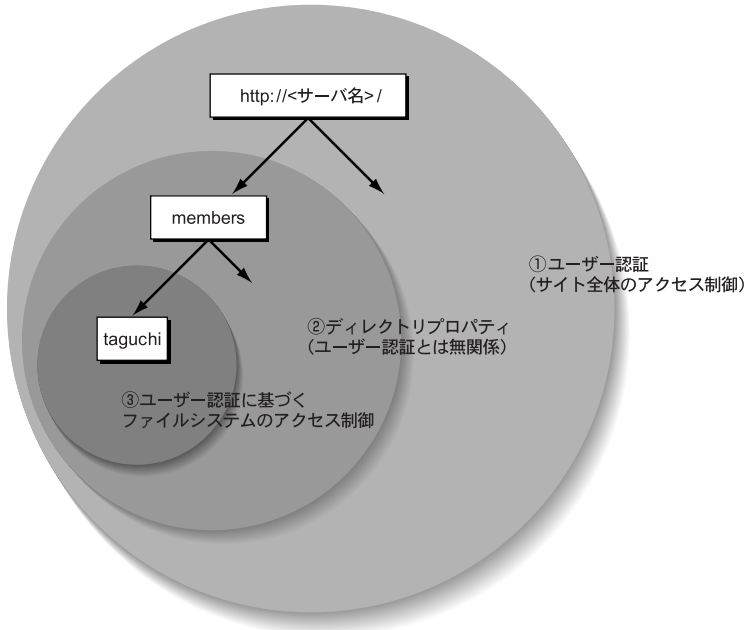

図 3-19　IIS のアクセス制御

プンに設定されているからだ。これ以降の解説は、ユーザー認証を行い、ファイル単位あるいはディレクトリ単位でアクセスに制限を加える必要がある場合にのみ、読み進めればよい。また、IIS のルートディレクトリが FAT 上に設定されている場合には、ユーザー認証に基づくアクセス制御は設定できないため、以降は NTFS 上にファイルが格納されていることを前提に解説する。

なお、以降の解説は IIS に特化したものではなく、Windows XP の NTFS パーティションにおけるアクセス制御についての解説そのものである。したがってすでに NTFS に精通している読者は読み飛ばしても構わない。

3.3.1 ユーザーアカウントとグループ

ユーザー認証に基づくアクセス権の検証は、二段階に分けて行われる。まず対象のユーザーにアクセス権が与えられているか検証される。もし与えられていなければ、今度はそのユーザーが所属するすべてのグループについて、アクセス権が与えられているか検証される。こうしてユーザーとグループのどちらか一方にでもアクセス権が与えられていればアクセスは許可され、ブラウザに Web ページが表示される。どちらにもアクセス権が与えられていなければ、Web ブラウザには「ページを表示できません」とだけ表示されること

になる。

ここで呼ぶグループとは、複数のユーザーを一括管理するために利用するユーザーグループのことである。ユーザーがごく少数であれば、ユーザーを個別に管理することも可能だが、ユーザー数が増えてくれば、同種のユーザーをグループにまとめ、グループに対してアクセス権を与えるほうがはるかに効率的な管理が可能になることは想像がつくだろう。こうした仕組みが用意されているため、Windows XP では通常アクセス権はまずグループへ与えられる。そしてアクセス権を与えたいユーザーアカウントをグループのメンバとして登録することによって、アクセス権のコントロールを行う。

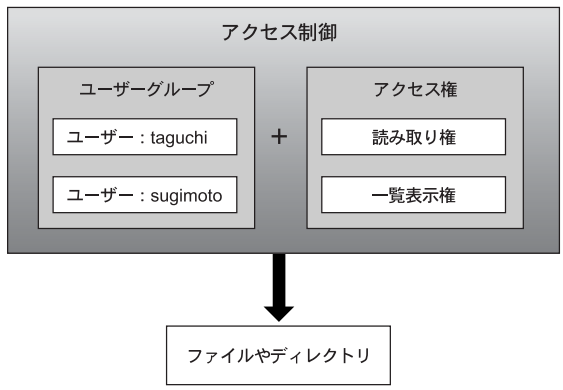

図 3-20　グループとアクセス制御

このように、Windows XP では「アクセス権の取得＝グループへの参加」と考えてよい。もしどこのグループにも所属していないユーザーアカウントがある場合、そのアカウントでは、ローカルログオンや IIS を通したアクセスはもちろん、いかなるアクセスも認められない。また、ユーザーは同時に複数のグループに参加することができるため、容易にユーザーにアクセス権を与えたり、取り除いたりできる。

そこで、アクセス権の設定方法を解説する前に、まずグループについて解説しておく。グループは大きく分けて、ビルトイングループと通常グループの 2 種類に分類できる。ビルトイングループは、システムに最初から用意されているグループである。ビルトイングループには、権利があらかじめ割り当てられているグループや、所属するメンバが自動的に決まる特殊なグループなどがある。もう一つの通常グループは、ユーザーによって作成されるグループである。通常グループのメンバは自由に編集できるが、初期状態では一切の権利を持っていないので、グループのメンバに登録しても何の効果もない。したがって通常グループを作成したら、まずグループに対してなんらかの権利（ファイルやディレクトリへのアクセス権など）を割り当てなければ意味がない。ただし、一般的にはユーザー

を適切なビルトイングループのメンバとして登録すれば、それだけで十分なアクセス制御が可能だ。

3.3.2 ビルトイングループ

Windows XP にはあらかじめ**表 3-1** に示すビルトイングループが用意されている。これらのグループは、コントロールパネルの［管理ツール］－［コンピュータの管理］－［システムツール］－［ローカルユーザーとグループ］で自由にメンバを追加したり、取り除いたりできる。

表 3-1　ビルトイングループ（メンバを編集可能）

Administrators
Power Users
Users
Guests
Backup Operators
Remote Desktop Users
Replicator

このほかにも**表 3-2** に示す、メンバを編集できない特殊なビルトイングループも用意されている。これらのグループのメンバは、ユーザーの状態によって自動的に決まる。たとえば、ネットワーク経由でアクセスしようとしているユーザーは、NETWORK グループのメンバとして自動的に登録される。また、あらゆるユーザーは Everyone グループのメンバとして登録される。

表 3-2　ビルトイングループ（メンバの編集不可）

ANONYMOUS LOGON
BATCH
CREATOR GROUP
CREATOR OWNER
Everyone
INTERACTIVE
LOCAL SERVICE
NETWORK
REMOTE INTERACTIVE LOGON

SERVICE
SYSTEM
TERMINAL SERVER USER

このようにたくさんのビルトイングループがあるが、ごく基本的なアクセス制御を行うだけであれば、Administrators グループと Users グループだけに注目すれば十分である。この 2 つのグループは、コントロールパネルの［ユーザーアカウント］では、それぞれ［コンピュータの管理］（Administrators グループ）と［制限付きアカウント］（Users グループ）と呼ばれている。

この 2 つのグループは、ビルトイングループのなかでも特別なポジションに位置づけられているグループである。Administrators グループはその名前のとおりコンピュータの管理者が所属するグループとして用意されいて、Administrators グループのメンバには全権が与えられる。したがって、Administrators グループのメンバは、あらゆるセキュリティチェックをパスして、アプリケーションやデバイスドライバのインストール、それにあらゆるファイルの読み書きが許可されている（一部例外を除く）。なお、Windows XP のインストーラで作成されるユーザーアカウントは、自動的に Adminnistrators グループのメンバに登録されている。

一方 users グループのメンバが行えるのは、アプリケーションの実行とマイドキュメントフォルダへのファイルの保存、それにインターネットへのアクセスなど、ユーザーが一般的に行う作業だけに制限されている。したがって、ドライバをインストールしたり、システムフォルダのファイルを編集したり、システムを不安定にするかもしれない作業のほとんどは実行することができない。このため、Designed for Windows XP 認定プログラムに対応していないアプリケーションの中には、システムフォルダへ設定ファイルを書き込もうとするなどの理由から、Users グループのメンバでは正常に実行できないものがあるので注意が必要だ。

この 2 つのグループを使った最も基本的なアクセス制御のシナリオは、サーバのオーナーだけを Administrators グループに登録し、それ以外のユーザーをすべて Users グループに登録するというものだ。基本的に Users グループのメンバは特別に権利を与えられない限り、ファイルを読み出すことはできても、編集することはできないので、意図せずにサーバの状態を変更されるような危険を犯さずに、サーバを公開することが可能だ。

もう少しユーザーごとに細かくアクセス制御を行いたければ、通常グループを利用することになる。たとえば、通常グループ「friends」を作成し、次に解説する ACL を適切に設定しておけば、グループのメンバ以外は/home/servers/friends フォルダ以下にアクセスできないようにすることが可能だ。

3.3.3　NTFS ファイルシステムのアクセス制御

ユーザーアカウントとグループの解説を終えたところで、再びファイルシステムのアクセス制御に話題を戻そう。

Windows 9x/Me で使われる FAT ファイルシステムでは、アクセス制御といえば、せいぜいがファイルに「読み取り専用」属性をつけるぐらいが関の山だったが、Windows XP の NTFS ファイルシステムにはまったく異なるアクセス制御システムが実装されている。すべてのファイル、すべてのフォルダに対してアクセス可能なユーザーまたはグループを指定可能であり、それぞれに**読み取り**のみ、**読み取り**と**実行**、**読み書き**および**実行**など、きめ細かな制御が可能だ。しかもそれを最小限の手間で設定できる。ただ、自由度が高いぶん難易度は高い。もっとも、実際には Windows XP の GUI が不必要に難易度を高めているだけで、仕組みそのものはそれほど複雑ではないので心配は無用だ。なお、以降の解説は NTFS ファイルシステムでフォーマットされたディスクを前提としている。Windows XP でも FAT ファイルシステムでフォーマットされたディスクの扱いは Windows 9x/Me と変わらないので、ここでは除外して解説を進めていく。

ACL（Access Control List）

NTFS 上に作成されたフォルダとファイルは、それぞれ個別に ACL（Access Control List）と呼ばれるアクセス制御情報を持っている（**図 3–21**）。この ACL は複数の ACE（Access Control Entry）から構成され、各 ACE にはユーザーアカウントまたはグループと、それに対するアクセス権のペアが記録されている。そして、ACL に登録されていないユーザーからのアクセスは、すべて拒否される。これが NTFS におけるアクセス制御の基本である。

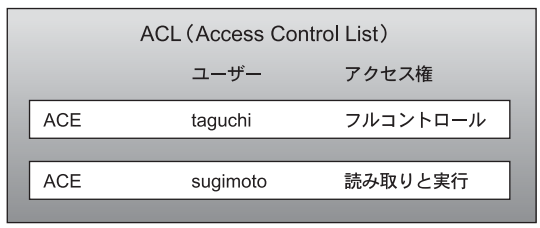

図 3–21　ACL

たとえば ACE には「Everyone グループに読み取りのみ許可する」「Administrators グループにフルコントロールを許可する」などの情報が記録されている。そしてこれら ACE が ACL に束ねられ、1 つのファイルまたはフォルダに割り当てられる。今例に挙げた 2 つの ACE を含む ACL が割り当てられたファイルのアクセス許可は、「誰でも読めるが、書

き込んだり削除したりできるのは Administrators グループのメンバだけ」となる。

各 ACE に設定できるアクセス許可タイプはさまざまだが、基本的には「フルコントロール」「変更」「読み取りと実行」の 3 タイプから選択して指定する（**図 3-22**）。後者ほどアクセス権は弱く、前者は後者を含む上位アクセス権を持つ。

図 3-22　基本的な ACE の設定

「読み取りと実行」を設定すれば、ファイルの読み取りと実行、またはフォルダの一覧表示だけが可能になる。「変更」を設定すれば、「読み取りと実行」の権利に加えて、ファイルの編集と削除、それにファイルやフォルダの新規作成が可能になる。「フルコントロール」を設定すれば、「変更」に加えて、アクセス許可の変更が可能になる。アクセス許可を変更できるということは、アクセス制御を自由にコントロールできることを意味しているので、最強の権利である。また、「変更」許可しか持っていないユーザーがフォルダを削除するには、フォルダに含まれるすべてのファイルとサブフォルダにも「変更」許可が必要だが、「フルコントロール」許可を持っていれば、無条件でサブフォルダを含めて削除できてしまう。このように「フルコントロール」は使い方を間違えると危険なので、取り扱いには十分注意が必要だ。

また、より細かく設定したければ、「フォルダの内容の一覧表示」「読み取り」「書き込み」の任意組み合わせでアクセス許可を設定することも可能だ。これを利用すれば、「書き込み」許可だけを持つ、「ファイルやフォルダを作成できるが、一覧表示できないフォルダ」を作ることができる。

アクセス許可の継承

アクセス制御を行うために、各ファイルやフォルダは ACL を持っているわけだが、多くの場合、個々の ACL を直接設定する必要はない。基本的にすべてのファイルとフォルダは、親フォルダの ACL を継承する仕組みになっているからだ（**図 3-23**）。すなわち、明示的に指定しない限りは、ルートディレクトリに設定された ACL がすべてのファイルとディレクトリにそのまま継承されるということだ。

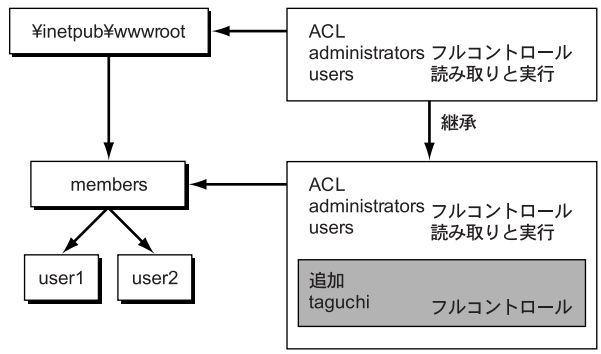

図 3-23　ACL の継承

このように NTFS では、明示的に ACL が設定されていないファイルやディレクトリは、明示的に ACL が設定された、最も近い親ディレクトリの設定を参照する仕組みになっている。そして、参照先ディレクトリの ACL が変更された場合には、自動的に継承しているディレクトリやファイルにも反映される。この参照モデルによる継承の仕組みは、ファイルシステムの管理にきわめてマッチしている。一般的に、あるフォルダを基点として、その下に格納されるファイルやフォルダには、同一のアクセス許可を与えるものだからだ。たとえば、¥Program Files 以下にはアプリケーション関連のファイルが格納されるので、アプリケーションのインストールを許可するユーザーにのみ書き込み許可を与え、それ以外のユーザーには読み取りと実行許可だけを与えればよい。また、ホームディレクトリにはそのオーナーだけがフルコントロールできる許可を与えればいいだろう。こうした基点となるフォルダでは親フォルダからの継承を行わず、独自のアクセス許可を与えれば、自動的にその下に格納されるファイルやフォルダには、基点フォルダと同じアクセス許可が行き渡る。こうしておけば、後でアクセス許可を変更したいときにも基点フォルダだけを操作すればよいので、管理ポリシーの変更にも柔軟に対応できるという利点がある。

ただし、親フォルダの ACL は必ず継承されるわけではなく、特定の ACE のみ拒否する設定を加えたり、また継承した ACL に新たな ACE を加えることも可能だ。ただし、こうした継承元となるアクセス許可が増えるに従って、ファイルシステムの管理は複雑化して

いくので、不用意にアクセス許可の修正は行わず、できるだけシンプルな設定を心がけるべきである。

また、ファイルやディレクトリを同一ドライブ内で「移動」したときには、移動先の親ディレクトリから ACL の影響を受けないことに注意しておこう。たとえば、「マイドキュメント」ディレクトリに作成したファイルを「¥inetpub¥wwwroot」に移動したとしよう。通常「マイドキュメント」ディレクトリでは、所有者にのみフルコントロールが与えられているため、そのファイルを読み書きできるのは、ファイルの作成者だけだ。そして、それはファイルを Users グループに読み出し権利が与えられている「¥inetpub¥wwwroot」ディレクトリに移動しても変わらない。つまり、相変わらず所有者以外はアクセスできないということだ。このような無用なトラブルを避けるためにも、異なる ACL が設定されたディレクトリ間での移動は、できるだけ避けたほうがいいだろう。

なお、異なるドライブへ移動したり、コピーしたときは、移動先ディレクトリの ACL を継承するため、問題はない。

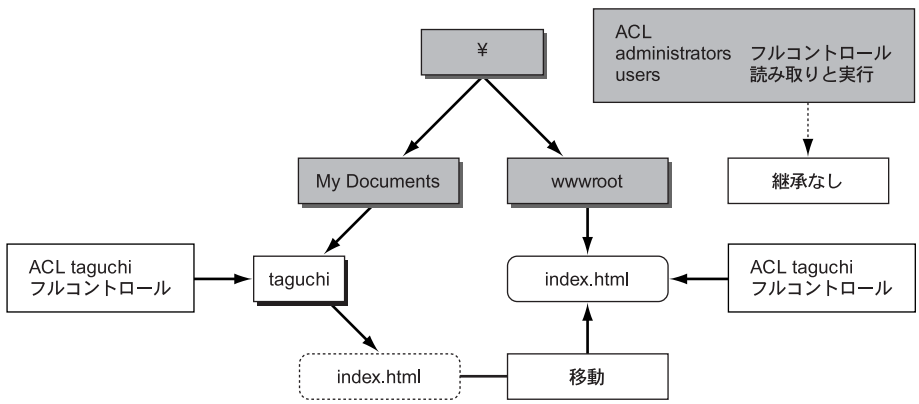

図 3-24　移動時の ACL

CREATOR OWNER、CREATOR GROUP

ファイルやディレクトリが新規作成されたとき、その ACL はどうなるのだろうか。これは NTFS の仕様に従って、親ディレクトリの ACL が継承される。これは、ファイルの作成者が誰であるかにかかわらず、親ディレクトリの ACL にのみ、アクセス許可は依存するということだ。もっとはっきりいえば、ファイルの所有者（＝作成者）であっても、ACL で明示的にアクセス許可が与えられていなければ、ファイルを読み書きすることはできないということだ。つまり NTFS 上では、ファイルやディレクトリの所有者だからといって、特別待遇されることはないのだ。ほかの OS に慣れていると、不思議に思えるかもしれな

いが、NTFS 上ではファイルの所有者に大きな意味はないのである。

ただし、それでは不便なので、特殊なビルトイングループ CREATOR OWNER が用意されている。この擬似グループは非常に特殊で、フォルダの ACL に対してのみ指定することが可能であり、また ACL を設定したフォルダ自身には何も影響を与えない。CREATOR OWNER が効果を発揮するのは、ファイルやサブフォルダを作成したときである。新規作成したファイルやディレクトリが ACL を継承するとき、この CERATOR OWNER はファイルやディレクトリの作成者に対する ACE へと、自動的に置き換わるのだ。つまり、CREATOR OWNER にフルコントロールが与えられているフォルダにユーザー ascii がファイルを作成すると、そのファイルにはユーザー ascii にのみフルコントロールを与えるアクセス許可が設定される。これを利用すれば、作成者のみ読み書き可能となるアクセス許可を容易に実現できる。逆に言えば、CREATOR OWNER を利用しなければ、作成者にフルコントロールを与えることはできないのだ。

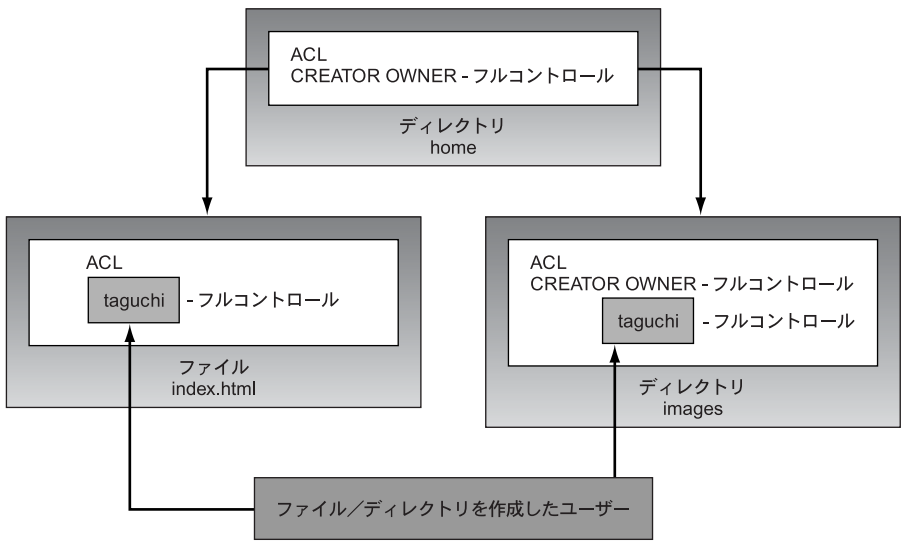

図 3-25　CREATOR OWNER

なお、CREATOR OWNER と同種の擬似グループとして CREATOR GROUP も用意されている。こちらはご想像のとおり、ファイルを作成したユーザーが属するグループにアクセス許可を与える擬似グループである。

アクセス許可の基本パターン

こうして解説に目を通しても、いきなり ACL を一から設定するのは難しいだろう。そこで、まずは Windows XP のインストール時に作成されたディレクトリの ACL をながめて、基本的な ACL のパターンに慣れていくといいだろう。

Windows XP をインストールした直後の初期状態では、ほとんどのフォルダに次の ACL がセットされている。この ACL はほとんどの場合うまく機能する、アクセス許可の基本パターンだと言える。

●Administrators グループに［フルコントロール］
●SYSTEM グループに［フルコントロール］
●CREATOR OWNER グループに［フルコントロール］
●Users グループに［読み取りと実行］

図 3-26　基本 ACL

Administrators グループ（コンピュータの管理者）はシステム管理を行うため、基本的には、すべてのフォルダでフルコントロールを与えておいたほうがよい。また、SYSTEM グループはシステム（Windows XP）やサービスがファイルアクセスを行うときの権限として利用されるため、フルコントロールを与えておかないと思わぬ不具合に遭遇する可能性が高い。CREATOR OWNER グループを使って、ファイルの所有者にフルコントロールを与えるのは妥当な設定であろう。そして Users グループ（制限付きアカウント）によって、基本的にすべてのユーザーに対して、書き込むことはできないが、読み取りと実行が許可される（作成されたユーザーはデフォルトで Users グループのメンバに登録される）。

以上が基本パターンだが、すべてのファイルやフォルダにこのアクセス許可を設定したのでは、管理者以外はまったくファイルを書き込むことができなくなってしまうので、初期状態では［マイドキュメント］や［デスクトップ］、それに［お気に入り］など、ユーザーの個人情報が格納される¥Documents and Settings¥＜ユーザー名＞フォルダに、各フォルダの所有者に対してフルコントロールが与えられている。つまり、Users グループのメンバ（制限付きアカウント）は、［デスクトップ］と［マイドキュメント］にしかファイルを書き込むことができないのである。

以上が基本設定だが、いくつかのディレクトリでは特殊な ACL が設定されている。

たとえば、ルートディレクトリには基本パターンに加えて、Users グループのメンバにディレクトリの作成権が与えられている。ファイルの作成権は与えられていないので、ディレクトリしか作成することはできない。なお、以前「明示的に指定しなければ、ルートディレクトリの ACL が継承される」と述べたが、Windows XP をインストールした直後、ルートディレクトリには 4 つのディレクトリ（「Documents and Settings」「Inetpub」「Program Files」「Windows」）しか存在せず、これらのディレクトリにはそれぞれ固有の ACL が設定されているため、ルートディレクトリの特殊な権利は、ユーザーが作成したディレクトリにしか継承されることはない。興味があれば、これら 4 つのディレクトリの ACL にも目を通しておくとよいだろう。また、Windows ディレクトリのサブディレクトリには、さらに細かな設定が行われている。

3.3.4 特定のユーザーにのみ Web ページを公開する

以上で NTFS におけるアクセス制御の仕組みの解説を終えたので、具体的に特定のユーザーにのみ Web ページを公開する作業について解説しよう。意外に思われるかもしれないが、ここでの作業は IIS の管理ツールだけではなく、エクスプローラと「コンピュータの管理」も利用する。これは前述したように、IIS のアクセス制御がファイルシステムのアクセス制御と統合されているためだ。したがって、ここでは、コンソールからログオンしたユーザーに対して、ファイルシステムのアクセス制御を行う場合とまったく同じ作業を行うことになる。

まず、公開したいユーザーだけが所属する、新しいグループを作成する。これには「コンピュータの管理」を利用して、次の手順で作業を行う。

1. コントロールパネルの「管理ツール」－「コンピュータの管理」を起動する
2. ［システムツール］－［ローカルユーザーとグループ］－［グループ］を選択する
3. メニューの［操作］－［新しいグループ］を実行する
4. ［グループ名］に「friends」を入力し、［追加］ボタンをクリックして、目的のユー

ザーをすべて登録する

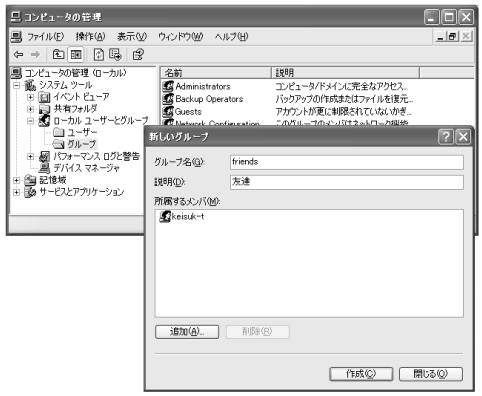

図 3-27　グループ friends の作成

以上でグループ friends が作成されたので、次に目的のフォルダにアクセス許可を設定する。ここでは限定して公開したい仮想ディレクトリ「/home/server/friends」が「c:¥inetpub¥wwwroot¥home¥server¥friends」に対応しているものとし、このフォルダ friends に対してエクスプローラから設定を行う。

エクスプローラでフォルダ friends のプロパティを開き、［セキュリティ］タブを選択する。そして次の作業を行い、グループ friends にアクセス許可を与える。

1.　フォルダ friends のセキュリティタブを開く

図 3-28　friends のアクセス許可を与える

2. ［追加］ボタンをクリックする
3. ［選択するオブジェクト名を入力してください］に friends を入力し、［OK］ボタンをクリックする

図 3-29　friends のアクセス許可を与える

以上の作業を行った後、セキュリティタブを見ると、フォルダ friends のセキュリティ一覧にグループ friends が追加され、アクセス許可として「読み取りと実行」「フォルダの内容の一覧表示」「読み取り」が与えられていることが確認できる。

これでグループ friends のメンバが/home/server/friends へアクセスできるようになったが、デフォルトではグループ Users に対して読み取り許可が与えられているので、このままでは、friends のメンバ以外にも、Users のメンバもアクセスできてしまう。そこで、次にフォルダ friends からグループ Users のアクセス許可を取り除く。この作業は単に Users を選択して、［削除］ボタンをクリックすればよいだけに思えるかもしれないが、実はそうはいかない。なぜならグループ Users へ与えられたアクセス許可は、c:¥inetpub¥wwwroot から継承した物だからだ。こうした、フォルダに直接設定されたものではなく、親フォルダから継承したアクセス許可から部分的にアクセス許可を取り除くには、まず「継承を断ち切る」必要がある。断ち切る方法にはいくつか選択肢があるが、親フォルダのアクセス許可をコピーした後継承を断ち切るのが一般的だ。こうすれば、現在の状態から目的のアクセス許可だけを取り除いた状態を簡単に作ることができる。

このようにして、フォルダ friends からグループ Users に対するアクセス許可を取り除くには、次の手順で作業を行う。

1. フォルダ friends のセキュリティタブを開く
2. ［詳細設定］ボタンをクリックする

3.　［子オブジェクトに適用するアクセス許可エントリを親から継承し、それらをここで明示的に定義されているものに含める］のチェックを外す

図 3-30　セキュリティの詳細設定

4.　［コピー］ボタンをクリックする

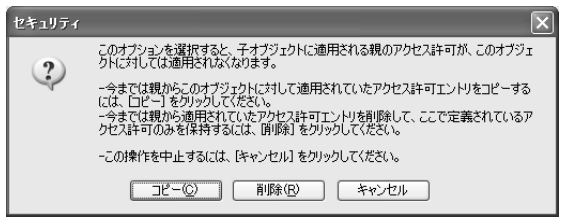

図 3-31　ACL の継承

5.　［アクセス許可エントリ］からグループ Users を削除する

以上の作業を終えると、/home/servers/friends 以下のファイルやフォルダには、グループ friends のメンバ以外はアクセスできなくなる。設定を終えた後に、アクセスを許可するユーザーを追加したり削除したりするには、グループ friends のメンバを編集すればよい。

Windows XP のセキュリティタブ

Windows XP をインストールした直後は、プロパティダイアログボックスの［セキュリティ］タブが隠蔽され、ディレクトリごとに詳細なアクセス制御を設定できないようになっている。これを解除して、［セキュリティ］タブを表示するには、エクスプローラのメニュー［ツール］－［フォルダオプション］を実行して、フォルダオプションダイアログボックスを開く。次に［表示］タブを選択して、［詳細設定］の［簡易ファイルの共有を使用する］のチェックを外す。これで［セキュリティ］タブが表示されるようになる。

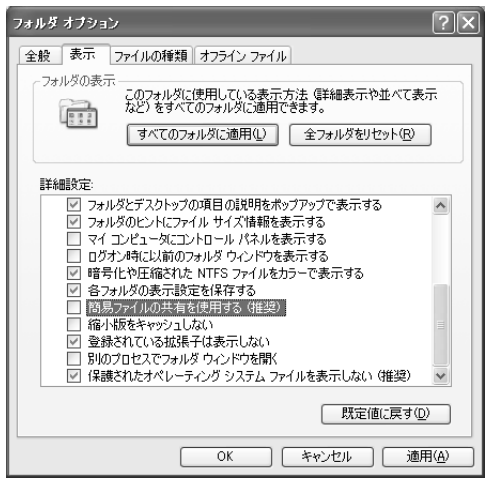

図 3-32　簡易ファイルの共有を使用する

3.4 インデックスサービス

インターネットユーザーであれば、google や wisenut などの全文検索エンジンは欠かせない必携ツールとなっているはずだ。ところで、Windows XP にもこの全文検索エンジンが搭載されていることはご存知だろうか。これはインデックスサービスと呼ばれ、IIS と連携して、IIS が公開するファイルの全文検索サービスを提供する機能を備えている。第 6 章ではこのインデックスサービスを利用して、サーバで公開されている Web ページを検索するシステムを紹介するので、ここでインデックスサービスの利用法について解説してお

こう。

3.4.1　インデックスサービスの機能

インデックスサービスを利用すると、ファイルの内容をインデックス化したデータベースを作成し、検索キーワードにマッチするファイルを高速に探し出すことができる。この点ではインターネット上の全文検索エンジンと変わりは無いが、相違点も少なくない。

まず、一般的なロボット型全文検索エンジンは、Web ページに含まれるリンクをたどって世界中の Web ページを収集し、インデックス化を行うが、インデックスサービスにこのような機能は備わっていない。インデックスサービスがインデックス化するファイルは、自サイトに格納されているファイルだけだ。インデックスサービスは、指定したディレクトリ以下に格納されているファイルをリストアップし、インデックス化を行う。このため、どこからもリンクされていないファイルであっても、指定ディレクトリ以下に格納されていれば、インデックス化の対象となる。

また、インターネット上の全文検索エンジンは、ファイルに含まれるテキストデータだけをインデックス化するのが一般的だが、インデックスサービスではファイル名や更新日時、ファイルサイズ、それにプロパティなど、ファイルに付随するありとあらゆる情報がインデックス化される。

さらに、HTML ファイルだけでなく、それ以外の多彩なファイルタイプに対してもインデックス化が可能だ。たとえば Microsoft Word の.doc ファイルや Microsoft Excel の.xls ファイルなど、バイナリ形式のデータファイルからテキストデータを抜き出し、インデックス化することができる。また、mp3 ファイルから ID3TAG を抜き出し、曲名やアーティスト名、コメントなどをインデックス化することもできる。つまり、インデックスサービスを利用すれば、簡単にミュージックライブラリの検索データベースを構築できるというわけだ。こうした多彩なファイルタイプに対応するため、インデックスサービスにはインデックスフィルタと呼ばれるインターフェイスが用意されている。このインターフェイスに対応した検索モジュールを追加することで、さまざまな種類のファイルをインデックス化できる仕組みが用意されている。

3.4.2　インデックス化するディレクトリを制限して運用すべし

このように非常に強力な検索機能を備えるインデックスサービスだが、うまく活用するにはちょっとしたコツが必要だ。このコツを知らずにインデックスサービスを起動すると、システムに多大な負荷をかけてしまうことになりかねないので、まずはインデックスサービスの設定方法と基本的な動作についての理解が欠かせない。

インデックスサービスを理解するには、インデックスサービスの管理ツールを使ってみるのが手っ取り早い。管理ツールは［コンピュータの管理］に含まれているほか、スタートメニューの［ファイル名を指定して実行］に「ciadv.msc」を実行することでも起動できる。

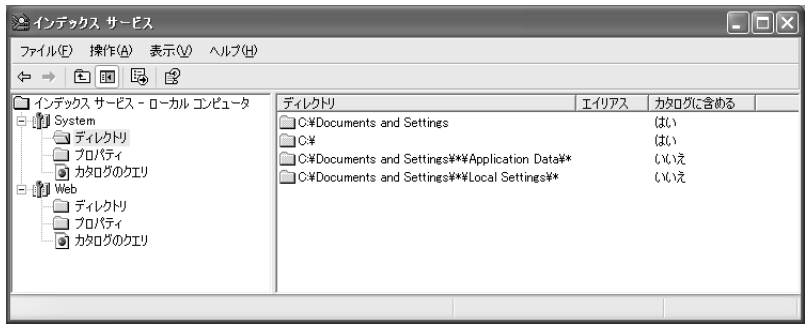

図 3-33　インデックスサービスの管理ツール

管理ツールのコンソールツリーには、System と Web が項目として表示されている。この 2 つの項目はカタログと呼ばれ、各カタログにはインデックス化するディレクトリが登録されている。インデックスサービスに検索を要求するときには、検索キーワードと同時にカタログを指定することができるため、カタログを複数用意することで、検索対象となるディレクトリセットを切り替えることができるようになっている。カタログに含まれるディレクトリは追加したり、削除したりできるほか、必要に応じてカタログを新規作成することも可能だ。

これらデフォルトで用意されているカタログには、それぞれ目的にそったディレクトリが、あらかじめ登録されている。

カタログ System は、エクスプローラで行われるファイル検索を高速化するために利用される。インデックスサービスが起動されていると、エクスプローラで行われたファイル検索はインデックスサービスのカタログ System にリクエストされる仕組みになっている。このため、初期状態ではシステムに接続されたすべての固定ディスクのルートディレクトリが登録されている。

またカタログ System には、共有フォルダとして公開されているディレクトリが自動的に登録されるようになっている。これはリモートホストからの検索にインデックスサービスが特に有効に働くからだ。リモートホストで公開されている共有フォルダから、目的のキーワードを含むファイルを検索するには、通常ならば共有フォルダに格納されているファイルをすべてローカルホストへ転送しなければならない。これでは、共有されているファイルがよほど少量でなければ、現実的な時間内で目的のファイルを見つけることは難しい。

しかし共有フォルダを公開しているリモートホストでインデックスサービスが起動されていれば、検索キーワードを送信し、検索結果であるファイル一覧を受け取るだけで検索処理が終了する。共有フォルダのインデックス化が検索時間を劇的に短縮することがわかるだろう。

一方カタログ Web は、Web サイトの検索を目的として用意されたもので、初期状態では IIS で公開されているすべてのディレクトリが登録されている。このためカタログ Web は IIS と連動し、IIS に仮想ディレクトリを追加すると、自動的にカタログ Web にも反映されるようになっている。

インデックスサービスを Web サイトの検索目的で利用するのであれば、カタログ Web の初期設定は適切な状態にあるが、問題はカタログ System である。初期状態のままインデックスサービスを起動すると、ファイルシステム全体をインデックス化するために、長時間にわたって CPU パワーが消費され、なおかつディスクアクセスが続くことになり、システムのパフォーマンスを大幅に低下させることになってしまう。さらに、インデックスサービスが作成するインデックスのサイズは、インデックス付けされるファイルの 15 %〜30 %程度になるとされている。この割合はファイルの種類によって大きく異なるが、1GB のファイルをインデックス化すれば、150Mbytes〜300Mbytes のディスクスペースがインデックスを格納するために消費されることになる。いくら高速検索が可能になるとはいえ、インデックス化のためにシステムのパフォーマンスを落としたのでは本末転倒だ。このままではあまりにも消費されるリソースが大きすぎるので、カタログ System に含まれるディレクトリを編集し、マイドキュメントなど、データファイルが格納されたディレクトリだけがインデックス化の対象となるように修正すべきだろう。あるいは、Web サイトの検索さえ高速化されればよいのであれば、カタログ System を停止して、カタログ Web だけを利用することも可能だ。インデックス化するファイルが少なければ、インデックス化も短時間で完了し、消費するディスクスペースもわずかで済む。こうしてカタログに最小限のディレクトリだけを登録するのが、インデックスサービスを利用するコツである。

3.4.3 管理ツール

カタログ System に含まれるディレクトリを編集するには、管理ツールのコンソールツリーから［System］－［ディレクトリ］を選択し、右側のペインで作業を行う。カタログ System をまったく利用しないのであれば、インデックスサービスを起動した後で、カタログ System を選択して、メニューから［操作］－［すべてのタスク］－［停止］を実行すればよい。

カタログの準備を終えたら、コンソールツリーで［インデックスサービス－ローカルコ

ンピュータ］を選択し、メニューの［操作］－［開始］を実行して、インデックスサービスを起動する。すると CPU の空き時間を利用して、バックグラウンドでインデックス化が開始される。登録されたディレクトリ数やファイル数にもよるが、すべてのインデックス化を終えるまでには、数十分から数時間の時間を要するはずだ。なお、ユーザーがコンソールで作業を行っている間は、一時的にインデックス化作業は中断されるため、できれば深夜などに実行するとよいだろう。インデックス化の進行状況は、管理ツールで［インデックスサービス－ローカルコンピュータ］を選択すれば、右側のペインで確認することができる。

インデックス化が終了すると、［状態］カラムに「開始」が表示され、インデックスサービスを利用したファイル検索が可能になる。この状態になれば、カタログに登録されたディレクトリに対してファイルを新規作成したり、編集したり、削除したりすれば、自動的にインデックスサービスがこれを検出し、インデックスの更新を行う。このためいったん起動してしまえば、基本的にインデックスサービスはメンテナンスフリーで動き続ける。ただし、編集されたファイルのインデックス化は、やはり CPU の空き時間を利用して行われるため、即座に反映されるとは限らない。また、初期状態ではインデックスサービスは自動起動されないように設定されているので、このままでは Windows XP を再起動したとき、インデックスサービスが起動されない。そこで、**図 3-34** に示すようにコントロールパネルの［管理ツール］－［サービス］で、［Indexing Service］の［スタートアップの種類］を［自動］に設定しておこう。

図 3-34　Indexing Service の自動起動

システムの復元には要注意

Windows XP には、緊急時にシステム復旧を行うためのツール「システムの復元」が付属している。このツールを使うと、あらかじめ作成しておいた復元ポイントまでシステムの状態を復元させることができるのだが、復元作業を行うと、その副作用としてインデックスサービスが作成したデータベースが無効化されてしまうようだ。しかも、その後自動的に復旧されることはないので、次のように手作業で復旧作業を行わなければならない。

まずインデックスサービスの管理ツールを起動して、インデックスサービスを停止する。次に復旧したいカタログを選択して、メニューから［操作］－［すべてのタスク］－［カタログを空にする］を実行する。こうして壊れたデータベースを破棄したら、再度インデックスサービスを起動すれば、あとは自動的にインデックス化が開始される。

こうしてインデックス化を終えると、インデックスサービスが利用できるようになるわけだが、具体的にはどうやって利用すればよいのだろうか。カタログ System は前述したように、エクスプローラから利用が可能だが、実はカタログ Web を利用するためのユーザーインターフェイスは用意されていない。正確に言えば、インデックスサービスの管理ツールにある［カタログのクエリ］を利用すれば検索は可能だが、これを利用するにはインデックスサービスの高度な知識が要求されるため、日常的に利用するものではない。それに、Web サイトの検索が Web ページから行えないのでは、まったく意味が無い。そこで本書の第 6 章では、Web ページからインデックスサービスを利用する、2 種類の Web アプリケーションを紹介する。