

## Browsing Mac OS X》

### NetInfo でアカウントを一元管理

Mac OS X では個々のマシンにそのアカウントを登録するが、Mac OS X Server を使えば、アカウント情報をサーバ上に一元化できる。また、ホームディレクトリにはサーバ上の公開フォルダを利用できることから、どのパソコンからログインしても、文書ファイルや初期設定としてはネットワーク上の同じディレクトリを使うことができ、どのマシンでもつまりは同じ作業環境になるということも実現できる。その方法とポイントになるところを解説しよう。

## ユーザ認証と NetInfo

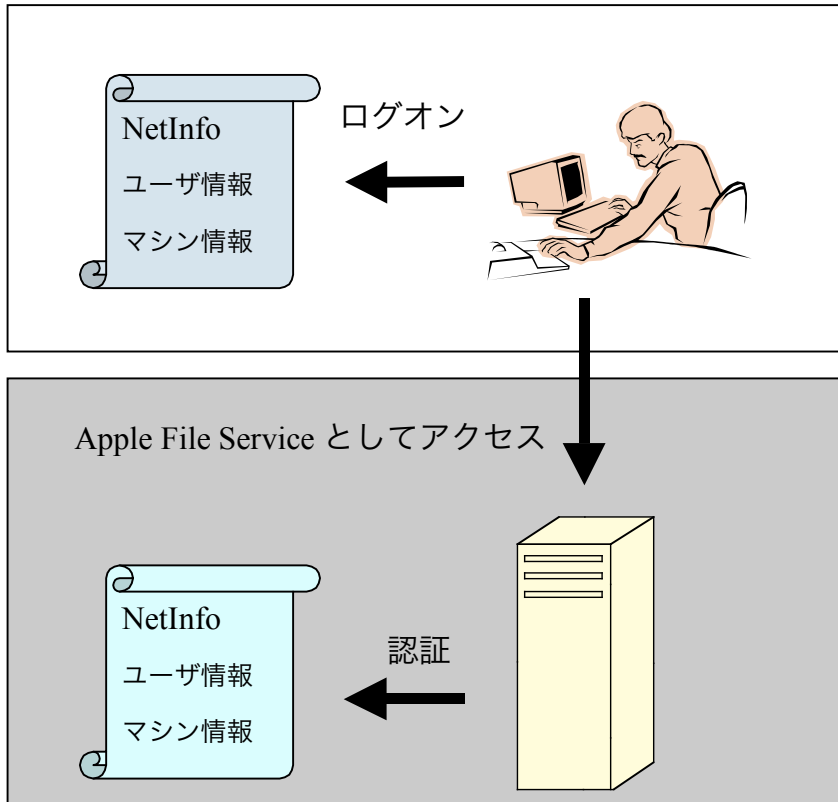
Mac OS X には、NetInfo というディレクトリサービスが組み込まれている。組み込まれているというか、OS の一部であると言って良いだろう。Mac OS X のもとになった OPENSTEP あるいは NeXTSTEP から使われているディレクトリサービスであるが、システムのさまざまな機能が NetInfo をベースにして稼働することもある。あえて Mac OS X では排除はしなかったようだ。ディレクトリサービスは、要はさまざまな情報を管理するのであるが、非常に大きな仕事が「アカウントの管理」という作業である。もちろん、セキュリティに配慮された環境で、アカウントとそのパスワードを記録し、認証等で利用できるようになっている。ただ、Mac OS X の場合 NetInfo が重要だとは言え、すべての設定が NetInfo でなされているわけではない。NetInfo が使われている場合もあれば、NetInfo が使われていない場合もある。意外にいろいろな設定は NetInfo のディレクトリに収められているものの、逆に思いのほか利用されていないものもある。単に何らかのデータを記憶するならファイルでもかまわないわけだが、ディレクトリサービスを利用する理由は、ネットワークでさまざまな情報を共有管理するというのが 1 つの大きな目的だ。さらに、そのときに階層化をはじめとして柔軟

なディレクトリを構成するといった機能が要求されるのである。なお、ディレクトリサービスのための代表的なプロトコルが LDAP であり、Windows のディレクトリサービスである Active Directory は LDAP にも対応している。LDAP をサポートした製品はけっこうたくさんあり、たとえば、認証情報を複数のサーバで一元化できるといったメリットを示して販売もされている。また、オープンソースでの LDAP サーバもある。なお、Mac OS X では LDAP での認証にも対応しているのであるが、今回は NetInfo だけを使った方法を紹介しよう。NetInfo は LDAP との互換性はなく独自のプロトコルである。

Mac OS X を普通に使う上では、実は NetInfo についてはそれほど意識する必要はない。言い換えれば、なるべく一般のユーザの目には触れないようにことごとく隠している。もちろん、システムの深部の話になるので、それはそれでいいのであるが、ネットワークを少し複雑に処理しようとした場合には、どうしても NetInfo の処理が必要になってしまうのである。ここでは、通常とおりにインストールしたところからの話を始めよう。

まず、Mac OS X をインストールすると、そのマシンで単独で NetInfo が稼働しており、そのマシンにだけにサービスを提供するローカルなディレクトリサービスが稼働している。サーバ機能に相当するデーモンなどが稼働しているのであるが、まずはイメージとして、Mac OS X にはディレクトリサービスが必ず動いている。このディレクトリサービスには、マシンを動かすための諸設定が記録される。そして、インストーラでインストールしたユーザとそのパスワードも、NetInfo のディレクトリに記録されるのである。また、システム環境設定の「ユーザ」で登録したユーザも、やはり NetInfo に記録される。一般的な UNIX では/etc/passwd ファイルに記録…というのがあるが（もはやそれすら一般的と言えないと言う話もあるけど…）、Mac OS X ではそうした記録はできない。唯一、そのマシンの NetInfo にユーザ情報が記録され、ログイン時にその NetInfo のディレクトリから情報を読み取って認証を行う。

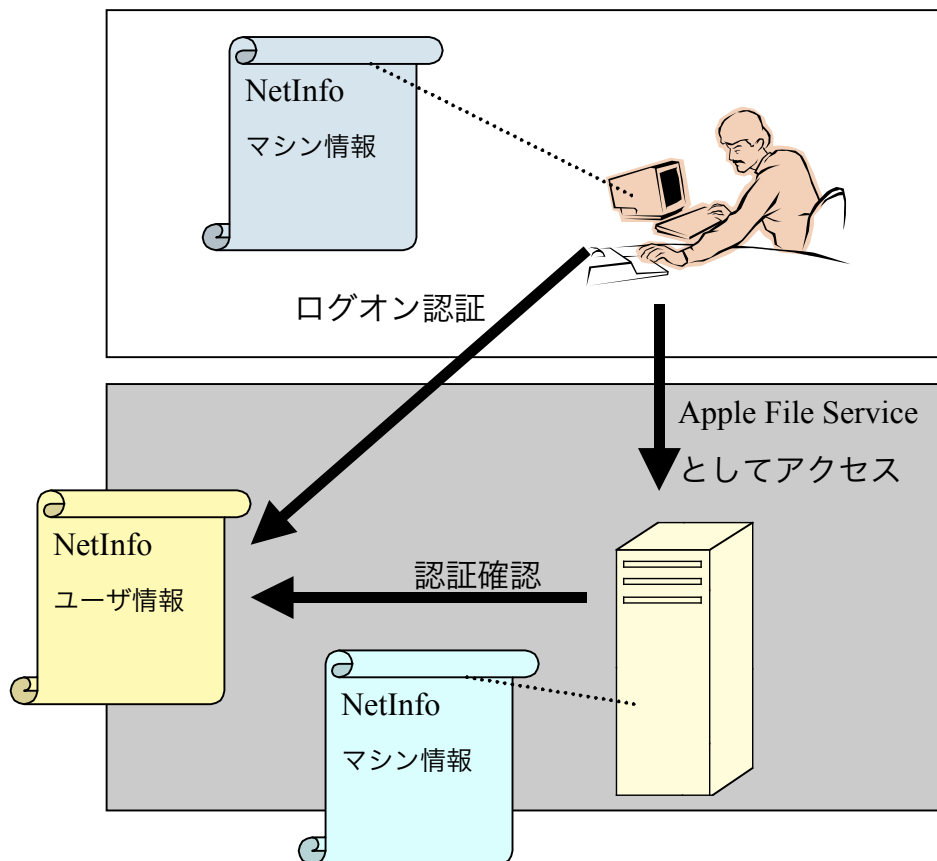
## 個々のマシンで認証を行った場合



ここで、Mac OS X Server で稼働している Apple File Service (AppleShare サーバ) に接続して、共有フォルダをマウントしてファイル共有をしたいとする。そのときにいちばん手軽な方法は、Mac OS X Server 側のローカルな NetInfo のディレクトリにユーザ登録を行う。つまり、サーバ側のマシン自体の情報やアカウントを記録している NetInfo に登録を行うのである。一般には、Mac OS X Server では Server Admin アプリケーションを使って登録を行う。サーバが、Mac OS X ならやはりシステム環境設定の「ユーザ」を使って行う。いずれにしても、前の図のように、サーバにアクセスするときのユーザ名とアカウントをサーバに登録する。だから、クライアントとサーバで共通のユーザ名とパスワードである必要もない。クライアントへのログオンと、サーバへのログインとは別々のものと認識される。

こうした共有の状況をさらに進化させることができる。クライアントのログインの認証を、そのマシンではなく、他のマシンで稼働しているディレクトリを通じて行うようにすることで、クライアントとのログイン、さらにはサーバのログインのいずれにも、共通のアカウントとパスワードを使うことができる。つまり、マシン個々に行うユーザ管理は必要がなくなるというわけだ。ユーザ管理はサーバで一元化できる。

## ユーザ管理を一元化させた場合



このとき、ユーザが利用するホームフォルダをネットワーク上に確保しておけば、たとえば LAN 内にある 10 台の Macintosh のどれを使っても、同じ名前でログインすれば、いつでも、ホームフォルダの Documents の中身は同じものを利用できるという具合になる。また、初期設定フォルダなども、マシンとは独立してサーバで一元化されているため、どのマシンを使ってもいつも同じデスクトップ画像を出すということもできる。NetInfo を使った Mac OS X ではこうした使い方まで想定しているのである。ここで、最初の状態では NetInfo のディレクトリは、マシンごとに用意されたものを使っているが、アカウント情報などの一元化を図るために、共通に使えるディレクトリを別途確保するというのが、NetInfo の基本的な設計となっている。前の図のように、サーバマシン上で、共通に使えるディレクトリを新たに用意し、クライアントはもちろん、サーバもそのディレクトリを使うようにする。そうした各マシンから共通に使うディレクトリは、ある意味ではクライアント/サーバ式の処理でもあるのだが、NetInfo はこうしたディレクトリ階層を 2 階層ではなく多階層構成することまでもできる。したがって、サーバよりも「上位」「親」といった言い方をするのが適切なようだ。親

のディレクトリを作成して運用する機能は、Mac OS X Server しか用意されていないので、こうした使い方をするには、Mac OS X Sever は必須となるだろう。以下、一元化した NetInfo のディレクトリでのログインに必要な操作を説明しよう。

その前に、稼働させた環境をまとめておく。ちょっとイレギュラーな環境であるが、その点は自宅ということでお許し頂きたい。

サーバ：Mac OS X Server 10.1.3 (Power Macintosh G3/300MHz/256MB/6G+15GB RAID)

Ethernet ポート 1：YahooBB に接続 (DHCP 接続)

実験時の IP アドレス：43.238.194.28

実験時のドメイン名：yahoobb238194028.bbtec.net

Ethernet ポート 2：内部ネットワークに接続 (DHCP サーバ、DNS など)

IP アドレス：192.168.1.1

ドメイン名：msyk.dyndns.org、right.msyk.dyndns.org (内部のみ有効)

Apple ファイルサービスサーバ名：right

クライアント：Mac OS X 10.1.3 (PowerBook G4/500MHz/512MB/20GB)

IP アドレス：192.168.1.101 (DHCP により配給)

## NetInfo ディレクトリの親を作る

---

まず、Mac OS X Server で、NetInfo の上位のディレクトリを新たに作成する必要がある。/Applications/Utilities フォルダにある、NetInfo Domain Setup をダブルクリックして起動する。NetInfo のところに現在の設定が見えるが、「ローカルの NetInfo のみを使用します」となっている。また、変更ができないようになっている。

## NetInfo Domain Setup を起動する



ここで、左下かの鍵の部分をクリックして、変更できるようにする。ここでは、admin のパスワードの入力に続いて、root のパスワードの入力まで行わないといけない。通常は、root のパスワードは、インストーラで最初に設定した管理ユーザのパスワードと同じになっている。

### 鍵をクリックした後、管理者とルートのパスワードが必要



こうして認証を終えると、設定を変更できるようになる。ここで、「お使いのコンピュータ」のところを「は NetInfo の上位層です」という選択肢に変更する。そして、「NetInfo の上位層のアドレス」には、このサーバマシンの IP アドレスを手で入力する必要がある。もちろん、このマシンに割り当てられている IP アドレスの指定が必要だ。サーバなので IP アドレスが固定されても問題はないと思うが、そうした設定が必要なのはポイントとなるだろう。NetInfo サーバタグは自動的に「network」になるものと思われる（実は申し訳ないのだが、いろいろ設定をした後なので、初期状態からの変化が確認できなかった。もしかすると、最初は手入力しないといけないのかもしれない）。

### NetInfo の上位層になるように設定する



設定を行うと、「保存」ボタンをクリックする。すると、警告のシートが出てくるので、OK ボタンをクリックする。保存後、もちろん、Mac OS X Server を再起動する必要がある。NetInfo だけの再起動はできないのかと思うところでもあるが、ここは指定とおり再起動してしまうのが確実だろう。

## 保存後再起動する必要があることがメッセージされる



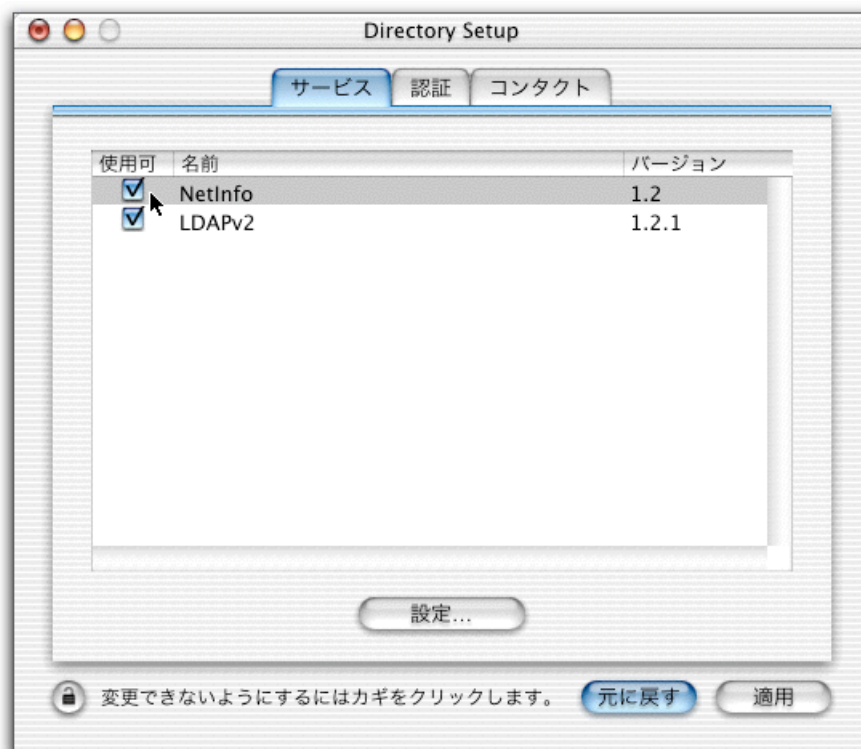
以上の設定により、Mac OS X Server は NetInfo の親ディレクトリを稼働させることができるようになった。

## クライアントを親ノードに認識させる

クライアント側は、そのままでは、自分のマシン内で稼働している NetInfo のディレクトリしか見ることができない。そこで、Mac OS X 自体が、ネットワーク上にある別のマシンの NetInfo ディレクトリを見に行くように設定を変更しなければならない。そうするには、クライアントのマシンで、/Applications/Utilities にある Directory Setup というアプリケーションを起動する。このアプリケーションも最初は設定を変更できなくなっているため、左下の鍵のアイコンをクリックして、管理者のアカウントとパスワードを設定する。そして、「サービス」のタブで、NetInfo のチェックをオンにする。

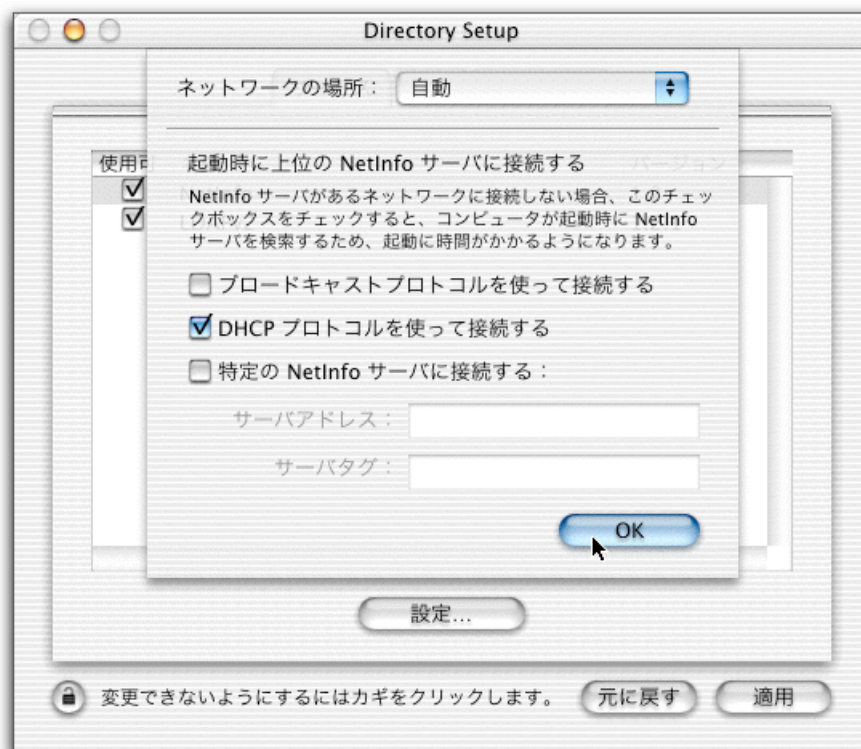


## NetInfo を利用するように設定する



さらに、NetInfo の項目を選択して、「設定」ボタンをクリックする。そうすると、次のようなシートが出てくるので、ここで NetInfo の上位のディレクトリが稼働しているマシンを探す適切な方法を、指定してやらなければならない。

## 上位のディレクトリを探す方法を指定する



ここで、「ネットワークの場所」というのは、システム環境設定の「ネットワーク」で指定する「場所」である。つまり、場所の設定に応じて、上位のディレクトリに接続する方法を選ぶことができるということだ。たとえば、社内で使うときにはチェックを入れるけど、モバイル時には場所を変更して、そちらではチェックが入っていないようにして、上位のディレクトリを利用しないといった使い方ができる。

そして、3つのチェックボックスがある。これらはそれぞれ、上位のディレクトリを探す手段である。1つだけでも、3つともチェックを入れてもかまわない。最初のブロードキャストは、Mac OS X を起動時にネットワークにブロードキャストをしてサーバを探すという手段である。これだと、もちろん、サブネット内とか、ルータを超えることができないなどサーバとクライアントの位置は物理的な制約がある。なお、NeXT時代はこのブロードキャストによるサーバ検索しかできなかったそうだ。

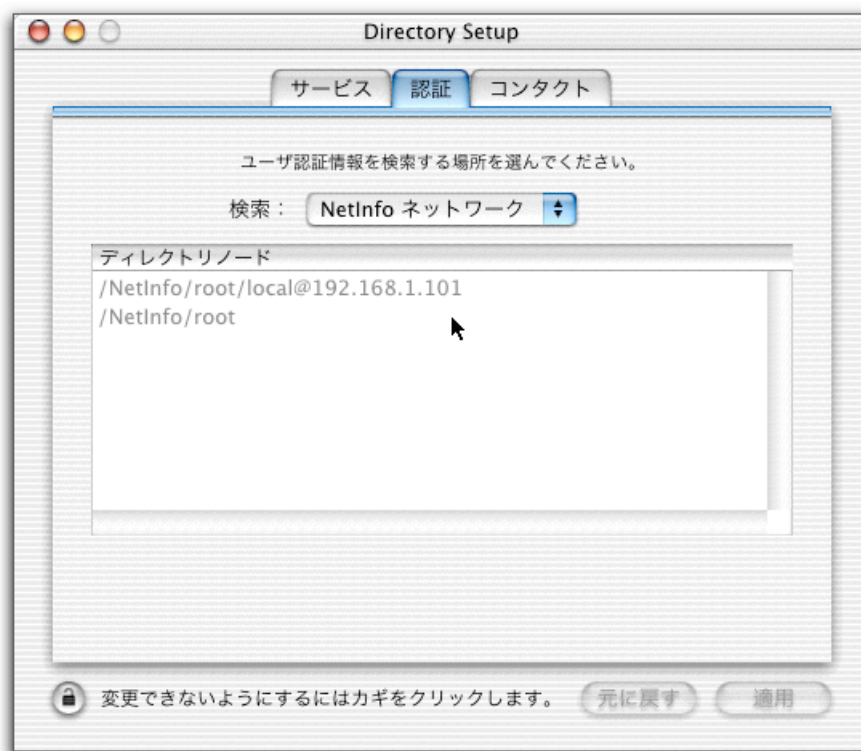
一方、DHCP を使うものは、DHCP での IP アドレスのリース時に、NetInfo の上位ディレクトリが存在するサーバの情報も伝達しようというわけである。DHCP からクライアントに対して上位のディレクトリ稼働しているサーバの IP アドレスと、サーバタグが伝達される。ちなみに、Mac OS X Server を DHCP サーバにしていれば、この NetInfo 情報を加えることは後で説明するように簡単に行える。

「特定の NetInfo サーバに接続する」をチェックした場合には、サーバアドレスとして、IP アドレスを、そしてサーバタグとして通常は「network」を指定する。この場合は、つまりは NetInfo のサーバの場所を手入力しているに他ならない。

いずれを選択するかは、ネットワークの事情によって異なるだろう。

こうして、NetInfo のチェックを入れて設定を行った後、「認証」のタブを見てみよう。新たに /NetInfo/root というノードが加わっているが、これが、別のサーバにある上位のディレクトリである。以下の図では、自分のマシンのディレクトリは、/NetInfo/root/local@192.168.1.101 という方である。後者の方には local というディレクトリが付加されている形になる。

### 上位のディレクトリが見えている

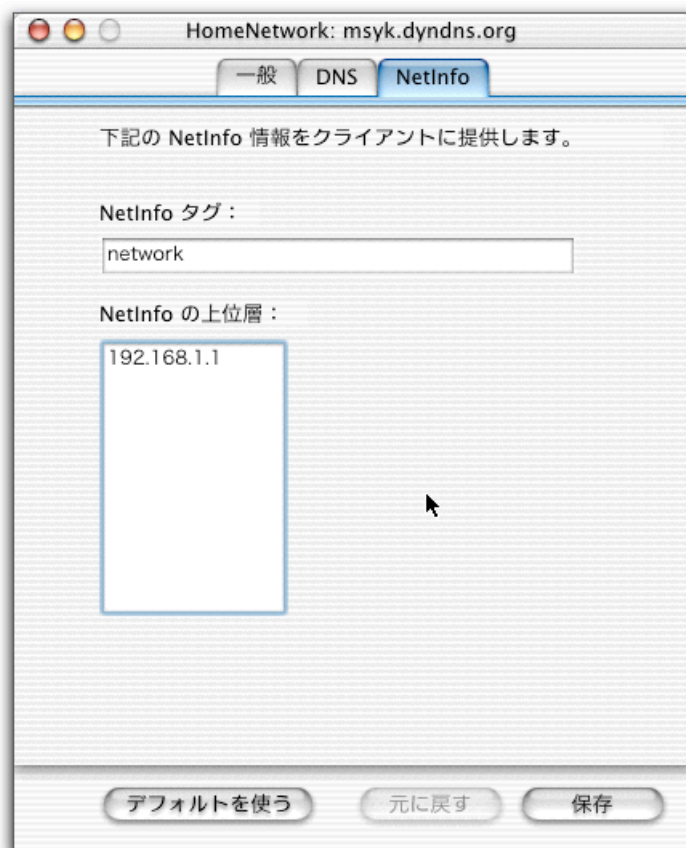


なお、ここで、認証に使うディレクトリを指定できるようになっているが、こうした小規模の構成ではあまり変更のしようがない。検索のポップアップメニューで「NetInfo ネットワーク」が選択されていると、まず、ローカルの NetInfo のディレクトリを検索し、続いて、上位の NetInfo サーバに接続して、そちらのディレクトリを検索するようになるというわけだ。

## DHCP サーバの設定

Mac OS X Server の DHCP サーバの設定は、Server Admin を使って行うことができるのであるが、アドレス範囲などに加えて、NetInfo の設定項目もある。NetInfo の上位ディレクトリ検索を DHCP で行う場合にはこの設定を行う必要がある。DHCP の設定ウインドウを呼び出し、サブネットの設定を開くと、NetInfo というタブがある。NetInfo タグは、通常は network と指定する。そして、NetInfo の上位層というところには、上位のディレクトリが稼働しているマシンの IP アドレスを入力する。

### DHCP で NetInfo の上位ディレクトリに関する情報を設定



なお、それでは Mac OS X Server 以外の DHCP サーバを使うときにどうしたらいいのかということも気になるかもしれないが、よく考えてみると、そうした使い方はどちらかと言えば特殊なのかもしれない。通常は、NetInfo の上位ディレクトリを稼働させるのだから、Mac OS X Server も動いているということになるだろう。

## 上位ノードをクライアントから参照する

クライアント側のマシンで Directory Setup で上位のディレクトリが見えるようになると、その上位のディレクトリをクライアント側から参照し、変更することもできる。ここでの上位のディレクトリとはサーバ側にあるディレクトリであるが、前に説明したように、サーバ側では 2 つの NetInfo ディレクトリが稼働している。サーバマシンの設定があるディレクトリではなく、認証のために公開されているディレクトリの方はクライアントから参照できる。

クライアント側で、NetInfo Manager を起動する。最初は、クライアントマシンの NetInfo のディレクトリが見えているはずだ。そのとき、上部の左から 4 つ目のボタンをクリックできるようになっている。「上位のドメインを開く」というヒントが出て来るように、このボタンをクリックすることで、新たにウィンドウが開いて、上位のつまりサーバ側の NetInfo のディレクトリを参照できる。

### NetInfo Manager で上位ドメインを開くボタンが使える



上位のドメインのウィンドウが開くと、タイトルバーを確認しよう。最初の network はタグで指定した文字で、続いてサーバのドメイン名が見えている。最後にノードが記載されているが、ルートだから、スラッシュだけである。

### 上位ドメインが参照できる



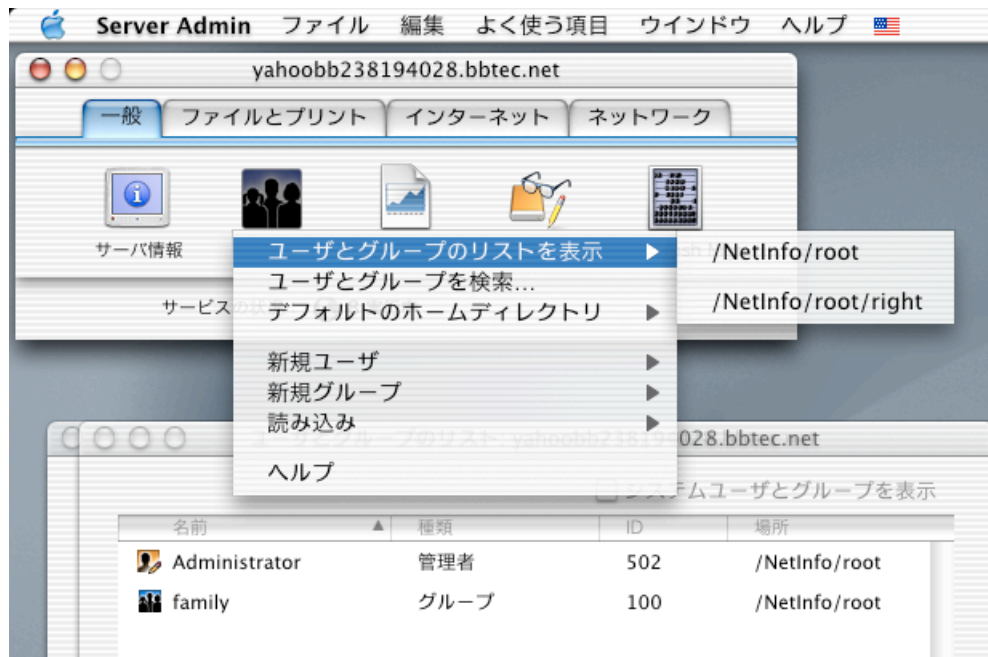
ここで、上位ドメインの変更を行ないたい場合には、もちろん、左下の鍵のアイコンをクリックする。そして、アカウントとパスワードを入力するが、サーバ側の管理権限のあるアカウントとパスワードを入力しなければならない。いずれにしても、管理作業は、クライアントからもできるということである。

## 上位ノードへのユーザ登録

新たに上位のノードを作成した場合、Server Admin の様子が違ってくる。次の図のように、Server Admin の U&G のアイコンにある「ユーザとグループのリストを表示」などの項目が、これまではたんなる 1 つの項目だったのが、階層メニューに変わって

しまい、ディレクトリの数だけメニュー項目が増えている。ここで、/NetInfo/root が公開されているディレクトリで、/NetInfo/root/right はこのマシンを管理するディレクトリである。（Server Admin をサーバ側で稼働させているため、ネットワークの外側のドメイン名が見えている。）

### Server Admin ではディレクトリごとにメニューが用意される



ここで、ユーザとグループのリストを確認したい。まず、サーバのマシンを管理しているノードについては、NetInfo Domain Setup でサーバを上位のディレクトリとして設定する前から存在している。ここでは、Apple File Service 等で使うためのユーザやグループをいくつか追加している。



## サーバマシンにもとからあったディレクトリのユーザ情報



一方、NetInfo Domain Setup で上位のディレクトリを設定して登場した/NetInfo/root のディレクトリのユーザ情報は次のようになっている。これは初期状態だ。ユーザ情報がコピーされるのかと思ったら、全部コピーはされていない。

## 新しく作られたディレクトリのユーザ情報



単に、「ユーザはコピーされない」だけなら、おそらくは Administrator もコピーされないし、family というグループもコピーされないだろう。どうも、新たに作成したユ



ーザはコピーされていないという規則にも見える。これについては後で解析するが、Server Admin で追加したユーザの全てが NetInfo でのログイン可能なユーザとは限らないようなのである。

## ホームディレクトリの自動マウント

---

続いて、ユーザのホームディレクトリ部分が、クライアントで自動的にマウントされるようにしたい。そうしないと、起動時に、ホームディレクトリが存在しないことになってしまい、その後の作業に支障をきたすことになる。

Server Admin で「共有」のアイコンから、「ディスクと共有ポイント」を選択する。そして、共有されている項目のリストに、Users は最初から用意されているはずだ。その Users を選択して、「アクセス権」ボタンをクリックする。そして、「自動マウント」のポップアップメニューを選択する。ここに「この項目を次のドメインのクライアントに自動マウントする」というチェックがある。これをオンにするのだが、ポップアップメニューの右側にある鍵のアイコンをクリックして、やはり管理者の認証を行わないといけない。ポップアップメニューはおそらく /NetInfo/root の 1 つだけだろう。

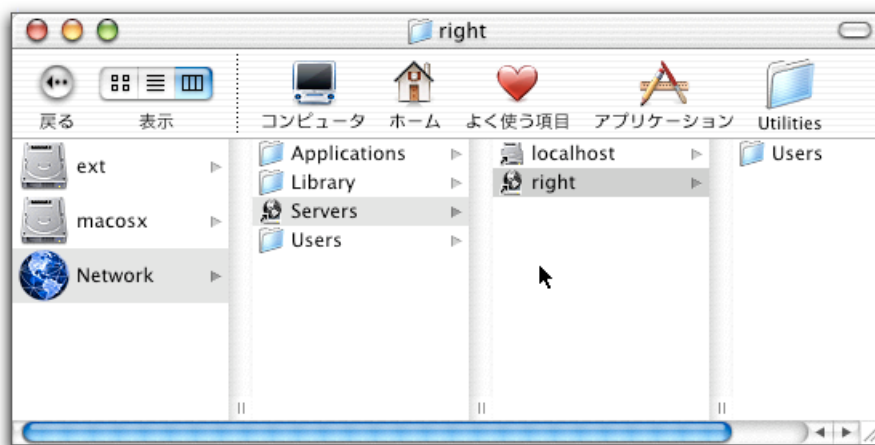
## 自動マウントの設定を行う



ここで、動的マウント、静的マウントの項目があるが、これは動的を選んでおこう。静的にマウントするというのは設定が記録されないようだが、機能的にサポートされていないのかもしれない。それから、Apple File Service 以外に NFS で Export していれば、プロトコルを選択するラジオボタンもアクティブになる。

こうして自動マウントをオンにしておき、クライアント側で上位のディレクトリを参照するようにしておくと、クライアントではルートにある Network という項目の下に、自動的にファイルサーバの公開フォルダがマウントされるようになる。これは、サーバにログインをしなくても、ローカルのクライアントからログインをするだけでも、とりあえずサーバのディレクトリはマウントされるようだ。ただし、ディレクトリの中身はアクセス権がないのでその場合は見えない。サーバのアカウントでログインしないと、マウントしたディスクの中身は見えないのである。

## 自動的にマウントされたサーバの Users フォルダ

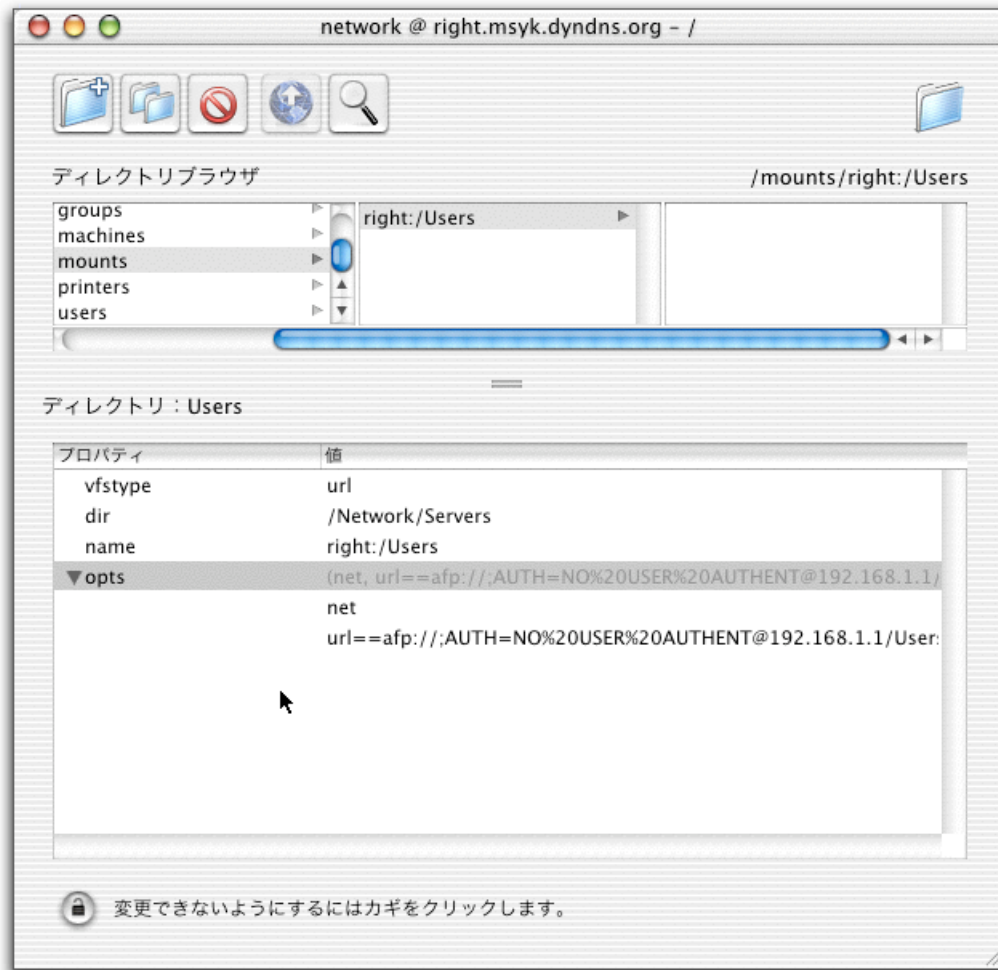


## NetInfo の設定を検討する

---

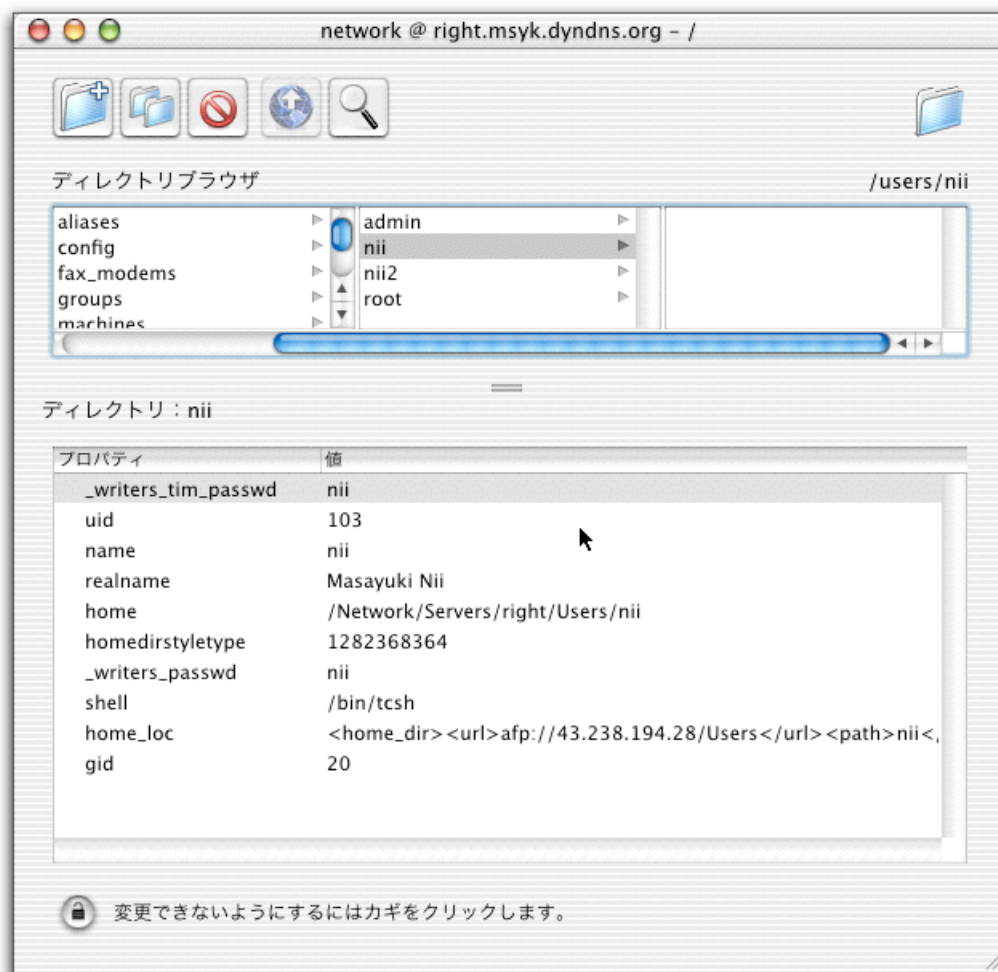
自動マウントの設定は、NetInfo に記録される。上位のノードの mounts というプロパティに記録されるが、場合によってはここを書き換える必要があるかもしれない。というのは、クライアント側の /Network/Servers というノードにマウントされるのはいいとして（設定としては dir のプロパティに対応）、その Servers の次の名前が、サーバのドメイン名の最初の単語になるようになってしまう。したがって、Apple File Service でのコンピュータ名と違う名前になってしまうこともある。そういう場合には、この mounts の name プロパティを書き換えてしまうのが 1 つの手のようなのだ。ここで、right:/Users と書き換えたが、そうすると、クライアント側で /Network/Servers の下に right という項目を作り、さらにその下にサーバ側の共有フォルダの Users が見えるようになるという具合である。

## マウントされる項目名を確認する

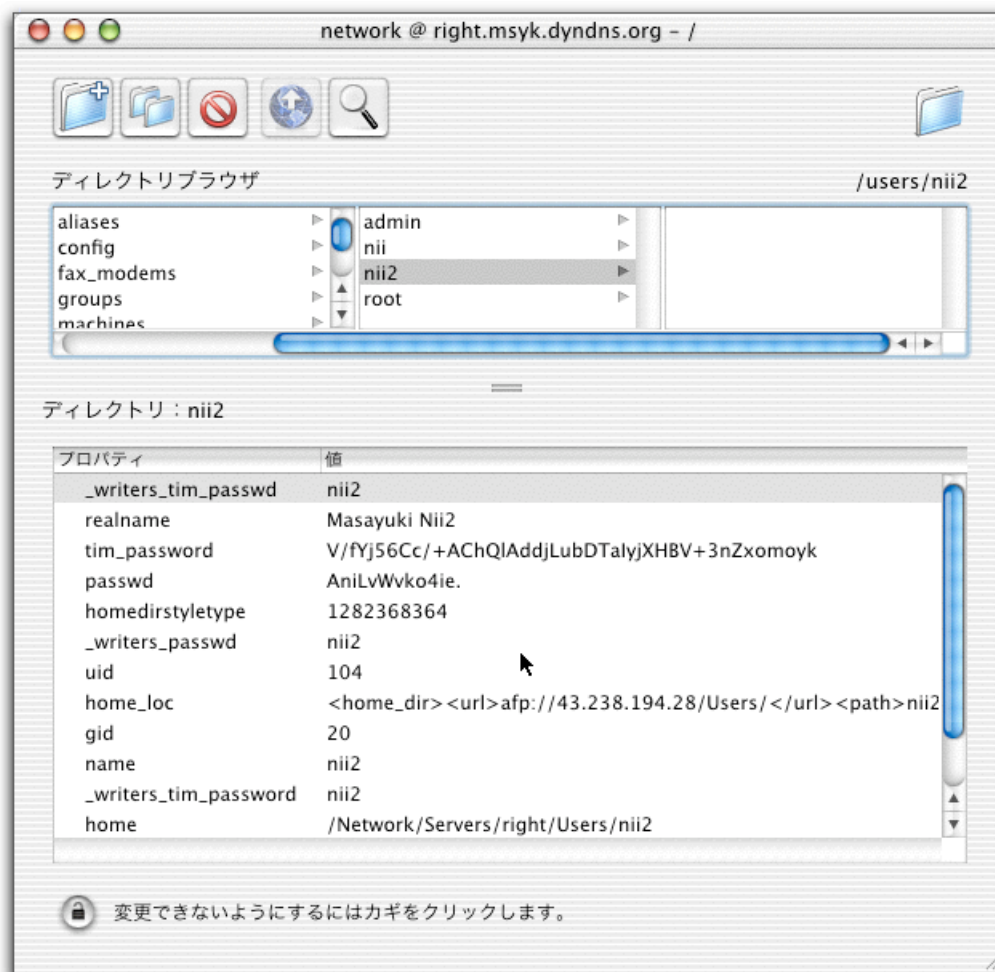


続いて、それでは、サーバにログイン可能なアカウントを、Server Admin で、/Network/root の方のノードに追加したとしよう。もちろん、U&G のアイコンから、「新規ユーザ」を選択して作成するが、そうして作ったアカウントでは、なぜかログインができなかった。結果的には、admin ユーザの設定をコピーして作ったアカウントでログインができたのだが、NetInfo でやはり設定された内容を見比べてみた。Server Admin で作ったアカウントが nii、admin アカウントをコピーして設定を変えたのが nii2 である。

## Server Admin で作ったアカウント



## 既存の Admin アカウントをもとに作ったアカウント



明らかに項目数が違うのはすぐに分かるとしても、違いをよく見ると、tim\_password というプロパティの有無がある。そして、Mac OS X Server には、tim というプロセスが起動時に立ち上がっている。どうやら、認証のための処理をここでさせているのではないかと思うのだが、単に Server Admin で登録したものでは、この tim\_password が ないため認証ができなくなっている模様だ。これが、バグなのか仕様なのかは定かではないが、とにかく、アカウントを登録したい。そこで、次のように作業を行なう。まず、/Network/root のユーザー一覧を表示して、そこにある Administrator を選択し、「書き出し」ボタンをクリックする。これで、ユーザ登録情報が、XML ファイルに書き出される。この XML ファイルの詳しい解説は、Mac OS X Server のマニュアルの 305p にあるので、詳細はそれを参照してもらいたい。最初にフィールド定義部分があるが、それ以降の部分で実際のユーザ定義部分がある。それを、たとえば、以下のように変更する。（←より右は解説なのでファイルには記入しない）

```

<uglist>
  <user
    loginEnabled = "canLogin"    ←ログイン可能という意味でそのまま
    isAdminUser = "notAdmin"    ←管理者でなくするにはこのキーワード
    uid = "505"                  ←この数値は自分で存在しないもの書き換え
    gid = "20"                   ←これもちゃんと調べて書いた方がいい
    shell = "/bin/tcsh" >      ←ログインシェル
    <nameList>
      <name
        text = "msyk_nii" />   ←ユーザ名
      <name
        text = "Masayuki Nii 3" /> ←フルネーム
    </nameList>
    <pass
      format = "clearText"      ←このキーワードにすれば
      text = "testtest" />     ←パスワードは記述通りになる
    </user>
  </uglist>

```

こうして XML ファイルを修正して、Server Admin の U&G のアイコンから「読み込み」を選択し、さらに /NetInfo/root を選択してファイルを指定する。そうすれば、上記で言えば、msyk\_nii というアカウントが作成されて、ホームディレクトリなど一連のファイルやフォルダが作成される。複数のユーザを登録するときには、上記の user タグの部分の人数分用意すれば良い。

こうして登録したユーザを Server Admin で見てみる。ユーザの一覧からユーザの情報を出して、「詳細」のポップアップを選択する。ここでホームディレクトリという設定がある。サーバは IP 番号になっているが、共有ポイントとして Users が設定されている。

## ホームディレクトリをしてみる



このユーザに関する情報を、NetInfo Manager で見てみる。もちろん、network がタグのルートの方であるが、users の下に、ここでは msyk\_nii というユーザ名と同じ項目があり、その下に各種のプロパティがある。ここで、home\_loc という項目は、XML っぽいデータで、上記のダイアログボックスでの指定結果に近いものがある。これに加えて、home というプロパティは、/Network/Servers/right/msyk\_nii となっている。

いろいろやってみた結果、home\_loc は実際の動作には関係ないようだ。つまり、上記のダイアログボックスで見えているホームディレクトリのアドレスは、確かにそのアドレスなのであるが、Finder で上記のアドレスを指定して接続した場合と、実際にログインした結果は違っている。Finder で上記のアドレスを指定すると、ボリュームはハードディスクと同じレベルに並べられるけど、NetInfo での自動ログインを行うと、/Network/Servers にサーバ名を伴ってマウントされる。

考え方としては、共有フォルダの自動マウントの設定で、クライアント側の自動マウントの処理は終わっているものと思われる。そして、NetInfo の上位ディレクトリを参照してログインが成功すると、ここでの home プロパティがクライアントに渡されて、その home に記載された絶対パスが、ログインしたユーザのホームディレクトリとなるというわけである。おそらく、home\_loc は設定の記録用であり、その home\_loc の設定から、自動的に home を構築しているのだろう。つまり、/Network/Servers までは

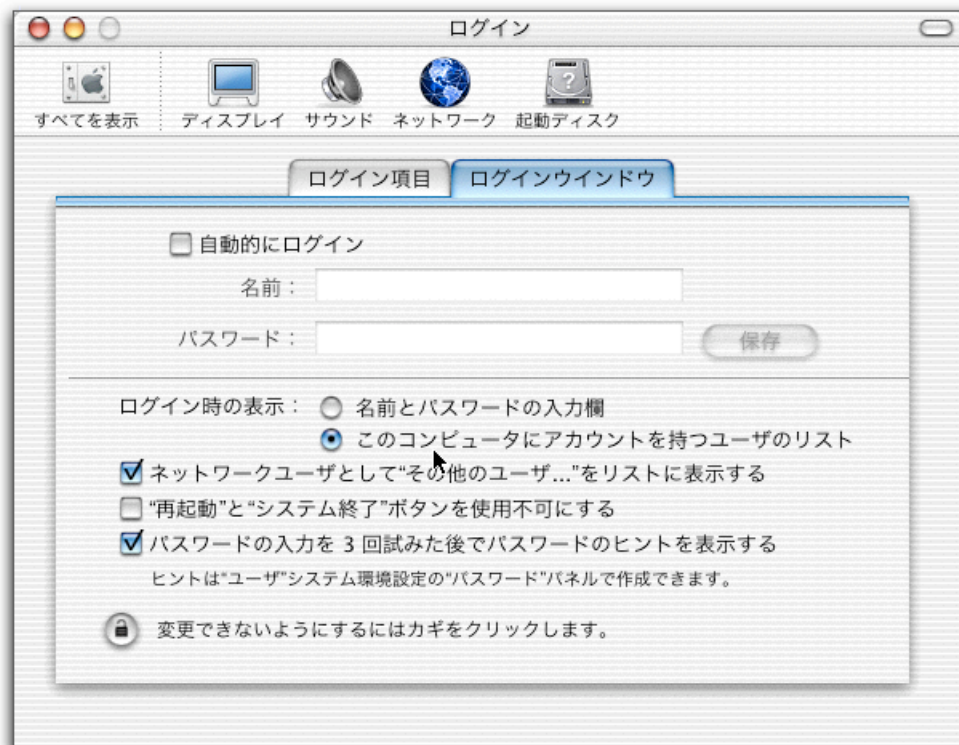


決められたところであり、サーバ名としてはサーバ側で認識した名前を入れる。そして、共有ポイントの Users が来て、さらにアカウント名の msyk\_nii がくれば、home プロパティが出来上がるということではないかと思われる。従って、サーバ名などが意図した通りにならない場合には、NetInfo で home の設定を変えてしまうことで、対処できるようだ。

## クライアントからのログイン

実際にクライアントからログインするためには、クライアントの設定の一部を変更する必要がある。システム環境設定の「ログイン」にある「ログインウインドウ」のタブで、「自動的にログイン」は基本的にはチェックを入れないことになるだろう。そして、ログイン時の表示で「名前とパスワードの入力欄」を選択するか、あるいは「このコンピュータに…」を選択した場合には、「ネットワークユーザとして“その他のユーザ…”」をリストに表示する」のチェックを入れておく。いずれにしても、アカウント名とパスワードを入力する枠がログイン時に表示されるようにしておく。

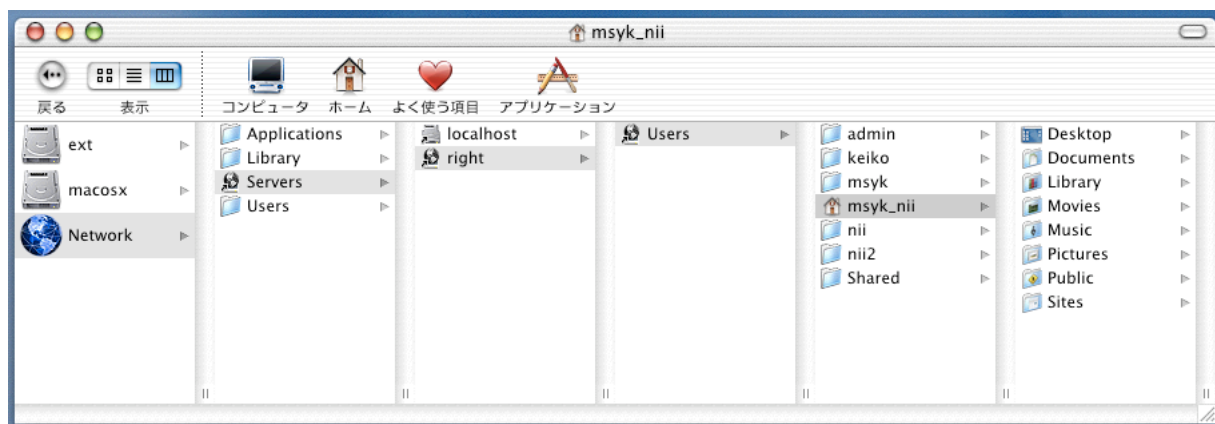
### 「ログイン」の設定を確認しておく



こうしてログインすると、ホームディレクトリは、ネットワークの先のサーバにある

自分のディレクトリとなる。以下のように、home プロパティで指定した通り、Network からたどっていったところがホームになる。

### ログインしたクライアントのホームディレクトリ



Documents をはじめとして、自分で使っている各種のフォルダが、すべてサーバにあるわけだ。結果的に、ネットワークのどの Macintosh からログインしても、ホームディレクトリは同じ環境となるというわけだ。

## ネットワーク構築の実際

こうした NetInfo の需要はあると言えはあるわけだが、さすがにどちらを見ても Macintosh があまり売れていない世界だ。たとえば、学校での利用が考えられるが、Macintosh を講義で使うような場合には、学生が座る席は一定ではないし、台数以上の利用者がいるのが一般的だから極めて有効な手法だろう。Windows NT/XP では、移動プロファイルという方法で、ログイン時に初期設定などのプロファイルのダウンロードを行う方法で、異なるパソコンでログインしても自分の環境を常に呼び出せるということは可能であり、学校教室での利用は一般的になっている。これと同等なことは NetInfo できるが、初期設定なども含めてネットワークアクセスがある。やはり多人数が同時に使う世界だけに、ネットワークのトラフィックについては設計時に神経を払うべきところだろう。

また、オフィスでの利用でもこうした使い方はあり得るものの、むしろ、1 人に 1 台のパソコンが割り当てられるとなると、かえて、個人個人でそのマシンを使うようにする方が意味では管理は楽な場合もありうる。また、デザイン系のユーザだと扱うデータのサイズが大きいいため、ファイルをネットワーク越しに開くようなことが体

感的にはなじめないだろう。ハードディスクが高価な時代はともかく、今では数十ギガが当たり前なので、コスト的なメリットもあまりないかもしれない。ただ、いずれにしても、中央で管理するという必要性がある場合には有効な方法だろう。

仮に、200人の学生のアカウントを Mac OS X Server で管理ということになったら、1人あたり 500MB としても 100GB のディスクが必要だ。そうすると、少なくとも、起動ディスクとは別のディスクにホームディレクトリを作成したいだろう。それに、まともに使うとなると 500MB/人では済まないだろう。複数のサーバに分散させないといけないことにもなる。こうした場合の設定については、以下の Knowledge Base に文書が掲載されているので、それを参照していただくといいだろう。1台の上位ディレクトリのサーバに、3台の下位のディレクトリを構築して、その3台に各学生のホームディレクトリを設定するという想定で、詳しく設定する方法が記載されている。

Mac OS X Server: Setting Up Multiple Servers to Host User Home Directories

<http://www.info.apple.com/kbnum/n106571>

ただ、問題としては、1ユーザあたりの使用量を制限すると言った機能が、標準では用意されていないことだ。実際にいろいろな使われ方がまだされていない Mac OS X Server だけに、具体的なニーズが出てきていないのかもしれない。

ログインの認証を Mac OS X Server にさせて、ホームディレクトリの利用はローカルにするというふうにしてはどうかという考え方もできる。そうすれば、/Network 以下に、サーバが自動的にマウントできる。もちろん、その通りなのではあるが、だったら、Mac OS X Server 側で認証はさせないでローカルで認証をさせ、普通に Finder からファイルサーバに接続する方が素直な使い方だろう。マウントしたサーバボリュームのエイリアスを作るなどしておき、それをシステム環境設定の「ログイン」の起動項目に入れておけば、起動時にそのサーバを自動的にマウントする。キーチェーン等の設定をしておけばすべて自動で行える。どちらが容易かは一概には言えないかもしれないが、わざわざ NetInfo の設定をいろいろいじるほどのメリットは薄いと感じる。

クライアントを上位のディレクトリに接続するようにしたとき、クライアントと上位ノードに同じアカウントを設定した場合どうなるかであるが、基本的にはクライアント側しか参照しない。たとえば、同じアカウントで異なるパスワードを設定しても、上位のノード側のアカウント情報ではログインはできない。認証するディレクトリの

順序を決めることが Directory Setup では可能だが、ローカルなディレクトリは最優先で使われる設定に変更はできない。このことを考えれば、たとえば、何台かの Macintosh で Mac OS X を使いはじめて、それからしばらくして Mac OS X Server の NetInfo をベースにして使うといったことへの移行はけっこう大変だということが想定される。サーバに設定するアカウントを意図的に変えるということをするなどの工夫が必要になる。

アプリケーションの「NetInfo 対応」ということも気になるところだ。ソースコードに「/Users/ユーザ名」的なパスの直書きがあると、上位のノードにログインしたときの動作は保証されないだろう。もちろん、システムからホームディレクトリを得て、その結果をもとにパスを構築するような形態になっていないと、問題が生じる。また、アプリケーションについても、同様に NetInfo の上位ノードにアクセスして共通に使うということもあり得るわけだが、その場合も、絶対パスでの動作が組み込まれていないようにする必要がある。今はは NetInfo ベースで使うユーザはほとんど見られないにしても、将来は分からない。開発者としては、こうした環境もあり得ることは頭に入れておくべきだろう。

~~~~~この項、以上~~~~~[新居雅行]~~~~~