鹿児島純心女子短期大学研究紀要　第35号，1－10　2005

NISを利用したPHPアプリケーションにおける認証方式の検討

吉　田　尚　史

A Study on the NIS Based Authentication System in PHP Applications

Naofumi Yoshida

ウェッブアプリケーションにおいて、ユーザー名やパスワードといった利用者情報を用いて利用者を認証する場合、アプリケーションが各々情報を登録・管理を行うことになると、利用者側･管理者側双方で運用に関するコストの上昇を招く可能性がある。これに対してネットワーク上でNISなど既存のネームサービスが運用されている場合には、その情報を認証に利用することでこのような問題に対応することができる。本稿ではPHPアプリケーションからNIS情報を利用し認証を行う実装とその有効性に関して検討する。

Key words：PHP、NIS、ウェッブアプリケーション、認証

（Received September 16 , 2004）

1　はじめに

ネットワーク上で提供されるメールの受信やファイル共有など諸サービスにおいて、その内容にアクセスしたり機能を利用するために必要な利用者の認証においては、ユーザー名やパスワードなど利用者が入力する文字情報が一般的に用いられている。これらのサービスが増加し利便性が増す一方で、それぞれのサービスが独立して情報を管理している場合、利用者が管理する情報も増加し、また管理者側も利用者情報の登録やサポートなどにかかる総体的な管理コストの増加を招く。これはウェッブサーバーのCGI、あるいはスクリプトエンジンモジュールの機能を利用し､サーバーサイドで実行されるウェッブアプリケーションの場合も同様である。

ホストコンピュータの増加におけるこのような問題の解決にはNISやNIS+、LDAPといったネームサービスあるいはディレクトリサービスが利用されている。これらのサービスはネットワーク上に複数設置されたホストで、同一の利用者情報を保持、参照することを可能としている。ウェッブアプリケーションの場合にもNISやLDAPの情報を利用することが可能であるが、独自のデータベースでアプリケーションごとに利用者情報を管理している場合が少なくない。本稿においてはウェッブアプリケーションの利用、運用のコストの増加を防ぐ観点から、既存のNIS認証情報が利用可能な場合、ウェッブアプリケーションの認証における実装とその評価を検討課題とする。サーバーサイドスクリプティングによるウェッブアプリケーションでは

＊　鹿児島純心女子短期大学生活学科こども学専攻（〒890-8525 鹿児島市唐湊４丁目22番地１号）

Perl、Java、Cなど様々なスクリプト言語が利用されるが、本稿ではウェッブアプリケーションの作成においてHTML言語との親和性の高さや拡張性の高さが評価されているPHPを利用した場合を検討する。

2　検討の対象

a）ウェッブアプリケーションにおける認証方式

ウェッブアプリケーションにおいて、特定の利用者に対してのみ情報を公開する場合や、特定の機能を利用させる場合、ユーザー名とパスワードの入力を求め、それらの情報による認証を行う場合が多い。利用者の認証方式としては

・Basic認証、Digest認証などウェッブサーバーの認証方式
・アプリケーションで独自に実装した認証方式

が考えられ、どのような情報に対して認証し、どのような認証方式をとるかについても

・. htaccessなどのウェッブサーバーの認証方式（Basic認証など）
・アプリケーションで独自に実装した認証方式

がある。また認証情報の保存方法としては

・ウェッブサーバーのアクセス設定で指定した認証情報ファイル（. htpasswdなど）

表１　認証方式・認証設定・認証情報の保存

<table>
<tr><td rowspan="2">認証方式
認証設定
保存方法</td><td colspan="2">ウェッブサーバー実装の認証方式
（認証機構はサーバー側で実装）
（UI*はブラウザーのダイアログ）</td><td>アプリケーション
（UIはHTMLによる独自実装）</td></tr>
<tr><td>設定ファイル</td><td>アプリケーション</td><td>アプリケーション</td></tr>
<tr><td>認証ファイル
（. htpasswdなど）</td><td>ファイル･ディレクトリに対するアクセス設定によるBasic認証</td><td>スクリプト内でhttp-headerを利用しBasic認証を要求</td><td>・認証機構は独自に実装
・スクリプト内で認証ファイルを検索</td></tr>
<tr><td rowspan="2">ネームサービス</td><td>同上</td><td>同上</td><td rowspan="2">・ 認証機構は独自に実装
・スクリプトから関数を利用しネームサービスを検索</td></tr>
<tr><td colspan="2">（mod_z_auth、mod_ldapなどのサーバーモジュールを利用し認証ファイルではなくネームサービスにアクセスして認証を行う）</td></tr>
<tr><td rowspan="2">データベース</td><td>同上</td><td>同上</td><td rowspan="2">情報管理・認証とも独自に実装しSQLデータベースなどを利用して情報を保持、PHPLibなどの既存のライブラリを利用</td></tr>
<tr><td colspan="2">（mod_auth_db、mod_authn_mysqlなどのサーバーモジュールを利用しデータベースに保存された情報で認証を行う）</td></tr>
</table>

＊UI：User Interface、ここでは利用者がユーザー名などを入力するインターフェースの意味で使用

・データベースエンジンなどを利用し、アプリケーション独自に管理
・NIS、LDAPなどネームサービスの利用

などの方法が考えられる。以上を表にすると表１のようになる。

本稿で取り上げたPHPアプリケーションにおけるNISの利用は表中の、認証方式、認証設定がアプリケーション、保存方法がネームサービスの場合になる。

ｂ）PHP

PHPはスクリプト言語の一種であり、HTMLファイル内に記述することで、ウェッブサーバーが稼動するホストコンピュータ上で何らかの処理手順を記述したスクリプトを実行し、テキストなどの実行結果を利用者のウェッブブラウザに対して出力することができる。

通常、HTTPを利用してサーバーサイドでスクリプトを実行する場合にはCGI（Common Gateway Interface）の形式でスクリプトの解釈と実行を行うスクリプトエンジンを呼び出すことで処理されるが、ウェッブサーバーに拡張モジュールを組み込むことにより、処理速度を高速化し、ホストコンピュータのプロセスの増加を抑え、負荷を低減することができる。

PHPは Unix系システム、Linux、Windows、Mac OS X、Novell NetWare、OS/２、RISC OS、SGI IRIX 6.5.x、AS/400といった多種のOSで動作し、またCGIが利用できるウェッブサーバで稼動させることができる。特にApacheやMicrosoft Internet Information Serverなどのウェッブサーバーではcgi経由ではなくサーバーの拡張モジュールとして動作することが可能である。

PHPの出力はテキストだけではなく、画像やPDFファイル、Flash形式のデータを生成することができる。また多くのデータベースとの連携をサポートしており、データベース機能を用いたウェッブページの作成が可能である[1)]。

ｃ）NIS

NIS（Network Information Service）は、限定されたネットワーク上のすべてのコンピュータでログイン名やパスワードなどの情報を共有するサービスである。NISのようなサービスを利用しない場合、ネットワーク上の各マシンごとにこれらの情報が独立して存在することになり、それぞれの情報の同一性を保持することは困難になる。NISを開発したSun Microsystems社によればNISは「分散型ネームサービスであり、ネットワーク上のオブジェクトおよびリソースを識別し、探索するメカニズム」と定義され、また「ネットワーク全体の情報に関する一様な記憶領域と検索方法を、トランスポートプロトコルやメディアに依存しない形式で提供」するものと定義される[2)]。NISが扱う情報は複数のNISマップに保存される。デフォルトでは表２のようなマップが用意される。

表２　デフォルトのNISマップ

bootparams	netmasks.byaddr
ethers.byaddr	networks.byaddr
ethers.byname	networks.byname
group.bygid	passwd.adjunct.byname
group.byname	passwd.byname
hosts.byaddr	passwd.byuid
hosts.byname	protocols.byname
mail.aliases	protocols.bynumber
mail.byaddr	rpc.bynumber
netgroup.byhost	services.byname
netgroup.byuser	services.byservice
netgroup	ypservers
netid.byname	

各マップには

・ブート時にクライアントが必要とするファイルのパス名
・マシン名
・Ethernetアドレス
・IP アドレス
・ネットワークマスク
・ネットワークプロトコル
・インターネットサービス
・RPCのプログラム番号と名前
・NISサーバー
・ユーザー名
・パスワード
・グループ名
・エイリアス
・メールアドレス

の情報が含まれる[3)]。認証等でNISを利用する場合にはこれらのマップを検索し取り出した値と利用者が入力した情報との整合性を確認することになる。

3　PHPアプリケーションにおけるNIS認証の実装

a）ユーザー名とパスワードによる認証

PHPスクリプトでNISの情報を用いて認証を行うには、PHPのYP/NIS関数を利用し、NISの

passwd. bynameマップの情報を得て、利用者が送信したユーザー名、パスワードの情報と比較する。手順の概要は

⑴デフォルトのNISドメインを取得
⑵利用者が入力したユーザー名をyp_cat関数でpasswd. bynameマップから検索
⑶利用者が入力したパスワードとyp_match関数で得られたパスワードを比較
⑷認証に通過した場合は認証済みであることおよびユーザー名をセッション変数にセット

となる。以下のリスト１は上記の手順部分の抜粋である。

リスト１

```
<?php
 $domain = yp_get_default_domain() ;                                    //(1)
 $user_array = explode(":", yp_match ($domain,
                    "passwd. byname", $_REQUEST['uid'])) ;              //(2)
 if(!strcmp((string)crypt($_REQUEST['passwd'],
                        $user_array[1]), $user_array[1])){              //(3)
    // パスワードが一致しなかった場合の処理
 } else {
    // パスワードが一致した場合の処理
    $_SESSION['auth']=true ;                                            //(4)
    $_SESSION['uid']=$_REQUEST['uid'] ;                                 //(4)
 }
?>
```

これらの処理を経て、ユーザー名とパスワードを入力した利用者は特定のページへのアクセスを認められることとなる。閲覧されるページ側では次のリスト２のような処理を行う。

リスト２

```
<?php
  session_start();
  if ( !isset($_SESSION['auth']) || $_SESSION['auth'] !=true){
     //認証を通過していない場合の処理
  } else {
     //認証済みの利用者に対する処理、もしくは表示する内容
?>
    <body>
   <div align = "center">
   <h1>Welcome !</h1>
   <hr>
```

```
    You are <?php= $_SESSION['uid'] ?> <br>
  </div>
  </body>

<?
  }//if clause end
?>
```

またページ単位でなく、ページ内において認証済みの場合と認証を通過していない場合に表示内容を変更することも可能である（リスト３）。

リスト３

```
<?php
  session_start();
  if (!isset($_SESSION['auth']) || $_SESSION['auth'] !=true){
    //認証を通過していない場合の処理
    $auth = false;
    $str = "ゲスト";
  }else{
    //認証済みの利用者に対する処理、もしくは表示する内容
    $auth = true;
    $str = "$_SESSION['uid']";
    $for_authuser = "認証済みの利用者向け表示";
  }
?>

 <body>
 <div align = "center">
 <h1>Welcome !</h1>
 <hr>
 You are <?php= $str ?> <br>          // 認証済みの場合ユーザー名が、
                                // それ以外の場合には「ゲスト」と表示
 <?php if($auth){ echo $for_authuser ; ?>
                                // 認証済みの場合に特定の内容を表示
 </div>
 </body>
```

ｂ） グループ情報を用いた認証

前項のようなユーザー名とパスワードを用いた認証の場合、利用者がNISに登録されているか否かの認証は可能であるが、その利用者がどのようなグループに属していて、すなわちどのような利用者権限を有しているかを判断することができない。あるグループに属している利用者にだけ閲覧を許可するページやウェッブアプリケーションの機能がある場合には、NISに登録されたグループ情報を利用することで利用者がどのグループに所属しているかの情報を得ることができる。

PHPのスクリプトで特定の利用者のグループ情報を得るには上記の手順に加え、

⑴yp_cat関数を利用しNISドメイン内のグループ情報を取り出す

⑵全グループ情報の配列からユーザー名を検索

⑶一致した場合所属グループ配列にグループ名を収納

⑷所属グループの配列をセッション変数にセット

という手順が必要となる（リスト４）。

リスト４

```
<?php
 $domain = yp_get_default_domain() ;
 $user_array = explode(":", yp_match ($domain,
                    "passwd.byname", $_REQUEST['uid'])) ;
 $pw =(string)crypt($_REQUEST['passwd'], $user_array[1]) ;
 if(strcmp($pw, $user_array[1]) !=  0 ){
     // パスワードが一致しなかった場合の処理
 } else {
     // パスワードが一致した場合の処理

     $mbr_group_array = array() ;
     $group_list_array = yp_cat($domain, "group.byname") ;                    //(1)
     while (list ($key, $val) = each ($group__list__array)) {
        $group_array[$key] = split(", ", $val) ;
        $rt = array_search ($_SESSION['uid'], $group_array[$key], true) ;
                                                                              //(2)
        if($rt){
           array_push($mbr_group_array, $key) ;                               //(3)
        }
     }
     $_SESSION['auth']=true ;
     $_SESSION['uid']=$_REQUEST['uid'] ;
     $_SESSION['group']=$mbr_group_array ;                                    //(4)
 }
?>
```

グループ情報を利用して認証を行う側のページでは、リスト５のように

⑴許可するグループを指定
⑵利用者の所属グループと許可グループを比較し一致した場合に認証を通過

となる。

リスト５

```
<?php
  session_start();
  if (!isset($_SESSION['auth']) || $_SESSION['auth'] !=true){
     //認証を通過していない場合の処理
     $auth = false;
     $str = "ゲスト";
  }else{
     //認証済みの利用者に対する処理、もしくは表示する内容
     $auth = true;
     $str = "$_SESSION['uid']";
     $for_authuser = "認証済みの利用者向け表示";

     $group = $_SESSION['group'];
     $allow_group = array("agroup","bgroup","cgroup");                    //(1)

     if(count(array_intersect($group, $allow_group)) >=  1 ){             //(2)
       // 許可されたグループに属している場合の処理
       $gauth = true;
       $allow_str = "あなたはこの部分を読むことができます。" ;
     }else{
       // 許可されたグループに属していない場合の処理
     }
   }
       <body>
       <div align = "center">
       <h1>Welcome !</h1>
       <hr>
       You are <?php= $str ?> <br>
       <?php if($auth){ echo $for_authuser ; } ?>

       <?php if($gauth){ echo $allow__str ; ?>

       <hr>
       <p>この部分はグループ情報でアクセスが制限された部分です。</p>
       <hr>

       <?php } ?>

       </div>
       </body>
?>
```

4　まとめ - NIS情報による認証の評価

以上のリストに示したように、PHPアプリケーションでは容易にNIS情報を利用して認証機構を実装し、またその情報を利用してアプリケーションの設計を行い、表示されるページのデザインや内容を設定することが可能である。従って、すでにNISにユーザー名、パスワード、利用者の所属グループなどの情報が登録され、実際に運用されているネットワーク環境であれば、管理・運用コストの増加を抑えるだけでなく、利用者側に新たなユーザー名、パスワードの利用を強いるような負担なしに、アプリケーションを稼動することが可能である。

ページ単位、ディレクトリ単位でのアクセス制限や認証が必用な場合によく利用される.htaccessファイルと独自のパスワードファイルを利用する方式でも、これらの認証・制限を実現することが容易であるが、PHPアプリケーションにおける独自の認証の実装は、利用者に必要な情報の入力を促す認証画面を柔軟に設定することが可能であり、ユーザー名や所属グループの情報によってページ単位あるいはページ内の情報の一部に認証・制限をかける場合や、所属するグループによって表示される情報を変更する、あるいは使用できる機能を変更するといった場合に有効であると考えられる。

5　おわりに

本稿ではウェッブアプリケーションにおいて既存の認証情報を利用した認証方式について検討したが、PHPアプリケーションの場合、認証機構を実装する場合に外部ライブラリである「PHP Base Library (PHPLib)[4]」を利用する場合も多い。このライブラリはセッション管理、ユーザー認証、アクセス制限の機能を提供するものである。これを利用したアプリケーションでNIS認証を用いる場合についての検討も必要であるが、これについては稿を改めたい。

また本稿の目的とは違っていたため認証過程のセキュリティの問題については論じなかったが、今回取り上げた認証過程ではパスワード情報がネットワーク上を読み取れる形で送受信されることになるため、SSL等を利用し通信経路を暗号化するか、利用者のウェッブブラウザー側で暗号化したパスワード情報を送信するなどの措置を検討する必用があると思われる。

注

1）現在PHPでは以下のようなデータベースエンジンを直接利用することができる。Adabas D，Ingres，Oracle，dBase，InterBase，Ovrimos，Empress，FrontBase，PostgreSQL，FilePro（読込みのみ），mSQL，Solid，Hyperwave，Direct MS-SQL，Sybase，IBM DB 2，MySQL，Velocis，Informix，ODBC，Unix dbm

2）Sun Microsystems 社、『Solaris 9　8/03 System Administrator Collection - Japanese 』、「Solaris のシステム管理（ネーミングとディレクトリサービス：DNS、NIS、LDAP 編）パート III NIS の設定と管理、第 7 章 ネットワーク情報サービス（NIS）（概要）」、http://docs.sun.com/db/doc/817-2463/6mi4gjiil?a=view

3）Sun Microsystems 社、『Solaris 9　8 /03 System Administrator Collection - Japanese 』、前掲、表 7 － 3 、http://docs.sun.com/db/doc/817-2463/6mi4gjiip?a=view#anis 1 -12508

4） http://phplib.sourceforge.net/