

Wen An

Implementing Secure Transactions With PHP and MySQL

Bachelor's thesis

Information Technology Programme


May 2010



MIKKELIN AMMATTIKORKEAKOULU

Mikkeli University of Applied Sciences

DESCRIPTION

| | | |
|--|---|--|
|  MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences | | Date of the bachelor's thesis 10 May 2010 |
| Author: WEI AN | Degree programme and option Information Technology | |
| Name of the bachelor's thesis Implementing Secure Transactions with PHP and MySQL | | |
| <p>Abstract</p> <p>Network security is as important aspect in the information security field. With the wide applications of network, the importance of network security is standing out every day. Network security has become an increasingly important component in many areas.</p> <p>Nowadays, e-commerce is no longer a concept speculation, it has become the most popular consumption method. Therefore, the network security is more important for personal privacy and preserving an account number. We can implement preserving data with data encryption, for example MD5.</p> <p>In my thesis, I paid attention to implementing secure transactions with PHP and MySQL, and building an authentication system based on PHP and MySQL. I set up PHP environment (WINDOWS+PHP+Apache+MySQL), and edited the user interface based on HTML. In the interface page, I added two functions, one to log in, another to create new users. All the data are stored in MySQL database. In order to preserve the information, I used MD5 to implement encryption. Thus, the personal data can be more safe.</p> | | |
| Subject headings,(keywords) PHP, MySQL, MD5, Interface, Password | | |
| Pages 64 p. | Language English | URN NBN:fi:amk-201005139227 |
| Remarks,notes on appendices | | |
| Tutor Timo Mynttinen | Employer of the bachelor's thesis Mikkeli University of Applied Sciences | |

LIST OF ABBREVIATIONS

| | |
|------|----------------------------|
| PHP | Hypertext Preprocessor |
| MD5 | Message Digest Algorithm |
| HTML | HyperText Mark-up Language |
| XML | Extensible Markup Language |
| CGI | Common Gateway Interface |
| SSI | Server Side Include |
| SSL | Secure Socket Layer |
| GPG | GnuPG |
| PGP | Pretty Good Privacy |
| SHA | Secure Hash Algorithm |

ACKNOWLEDGEMENT

I would like to thank all people who helped me and inspired me during my diploma research study.

First of all, I would like to give my deepest thanks to my nice supervisor Timo Mynttinen. Without his attitude support, I cannot finish the thesis much soon. His patience and hardworking impressed me a lot. I learnt not only the way how to study new things independently, but also do the things efficiently and carefully. That is the reason promoted me to find ways to resolve the problems and correct mistakes which I faced. I will keep it in mind, and I think this will make me life benefit.

Second, I would like to express my sincere appreciation to my kind teacher Matti Koivisto. He is always willing to help me and give me support. I learnt new knowledge of the principles for this thesis from his courses.

Lastly, I take this opportunity to express my profound gratitude to my beloved parents. They always encourage me and give me spiritual supports from the distance, which promoted me to study hard.

TABLE OF CONTENTS

| | |
|--|----|
| 1 INTRODUCTION..... | 1 |
| 2 THE BASIC CONCEPT..... | 2 |
| 2.1 PHP..... | 2 |
| 2.1.1 What Is PHP?..... | 2 |
| 2.1.2 PHP3 and PHP4..... | 2 |
| 2.1.3 What Is New in PHP 5.0?..... | 3 |
| 2.1.4 The Advantages of PHP..... | 3 |
| 2.2 MySQL..... | 4 |
| 2.2.1 What Is MySQL?..... | 4 |
| 2.2.2 What Is MySQL Query Browser..... | 4 |
| 2.2.3 The Advantages of MySQL..... | 5 |
| 2.3 Apache..... | 6 |
| 2.3.1 What Is Apache?..... | 6 |
| 2.3.2 The Properties of Apache..... | 7 |
| 3 PHP ENCRYPTION TECHNOLOGY..... | 9 |
| 4 MY OWN APPLICATION..... | 16 |

| | |
|--|----|
| 4.1 Install the PHP environment..... | 16 |
| 4.1.1 Install Apache..... | 16 |
| 4.1.2 Configure the Apache Server..... | 21 |
| 4.1.3 Install PHP and Integrate Apache with PHP..... | 22 |
| 4.1.4 Install MySQL and Integrate With PHP and Apache..... | 28 |
| 4.2 Create User Interface..... | 41 |
| 4.3 The Background Operation of Log-in..... | 44 |
| 4.4 The Interface of Creating New User..... | 48 |
| 4.5 The Background Operation of Creating New User..... | 51 |
| 5 CONCLUSION..... | 54 |
| 5.1 Challenge of My Thesis Research and Solution..... | 54 |
| 5.2 The Future of MD5..... | 55 |

BIBLIOGRAPHY

APPENDICES

1 INTRODUCTION

In the contemporary world, people pay more attention to the information security. Specially, the personal information will leak in the transmitting way when people create passwords in the web pages. So it is important for us to preserve the individual privacy. As an engineer, we should be thinking about this. In the thesis, I will implement secure transactions with PHP and MySQL.

The aim of the study is to implement the secure authentication.

The practical aim of the thesis is to use MD5 to set up the connection between web page based on PHP and MySQL.

The structure of the study is as follows.

Firstly, in Chapter 2, I will illustrate the four main aspects of implementing secure transactions, PHP, MySQL and Apache which are widely used in many areas. Then I introduce some advantages about them and how they work. After this in Chapter 3, I will introduce the main encryption methods MD5 and GPG in the world, some basic aspects of them and what the algorithms of them are. In Chapter 4, I will discuss creating an authentication with MD5. From building the PHP environment to implement an authentication with MD5. It can log in the web page with your own name and password, and create new users. Finally in Chapter 5, final conclusions and the future of MD5 are identified.

2 THE BASIC CONCEPT

2.1 PHP

2.1.1 What Is PHP?

PHP is the abbreviation of Hypertext Preprocessor. PHP is a language of HTML embedded and a scripting language of embedded HTML document which is implemented on the web server, the style of the language is similar to C language, which is most widely spoken. As of August 2004, it was installed on more than 17 million domains worldwide, and this number is growing rapidly.[1] You can see the current number at <http://www.php.net/usage.php>



Figure 2.1 The logo of PHP [1]

2.1.2 PHP3 and PHP4

Because PHP 3 was integrated with the Apache server, it is constantly updated and new features are added and supported all databases as well as it had high speed execution efficiency, the station based on PHP 3 had been used widely.

More than 150,000 in 1999. Since its source codes are full disclosure, it is the backbone of this area now. [1]With constantly adding new libraries and updating, PHP

has more new features in UNIX, LINUX or Windows platforms . It provides a wealth of functions and enables the design of the program more easy

PHP 4.0 is more efficient and reliable than dynamic web development tools. The operating speed is faster than PHP 3.0 in most situations. The better performance of PHP 4.0 is due to PHP script redesigned engine.

2.1.3 What Is New in PHP 5.0?

We can see from the PHP official web site that the version of PHP has changed from PHP 4.0 to PHP 5.0. The Zend engine beneath PHP has been rewritten for this version. Major new features are as follows:

Better object-oriented support set up around a absolutely new object pattern
Exceptions for scalable, maintainable error handling Simple XML for easy handling of XML data[2]

2.1.4 The Advantages of PHP

PHP specific grammar mix with C, Java, Perl and PHP grammars. It can be more quickly to execute dynamic pages than CGI or Perl. Compared with other programming language, PHP is a program embedded into HTML document to execute, the efficiency of the execution is much higher than completely generating HTML marked CGI, PHP can also perform compiled code,code can achieve encryption and optimize the operation of code, making the code running faster. PHP has very

powerful functions, it can achieve all the functions of CGI . Besides, PHP can support mostly all databases and operating systems.[3]

2.2 MySQL

2.2.1 What Is MySQL?

MySQL is a small relational database management system. Developer is the Swedish MySQL AB company which was acquired by Sun on April 16, 2008. While in 2009, SUN was purchased by Oracal.[4] For Mysql's future, no one has any optimistic attitude. Now MySQL is widely used in small and medium websites. Because of its small size, high speed, low cost, in particular the characteristics of open source, many small and medium websites choose MySQL as the web database. The official web site of MySQL is www.mysql.com.



Figure 2.2 The logo of MySQL[5]

2.2.2 What Is MySQL Query Browser

MySQL Query Browser is a visual toolset for creating, implementing and optimizing queries of MySQL database, it is like to an integrated drag tool embedded in the web server that provides the users with a more convenient, productive way to access and analyze information stored in the MySQL database server.

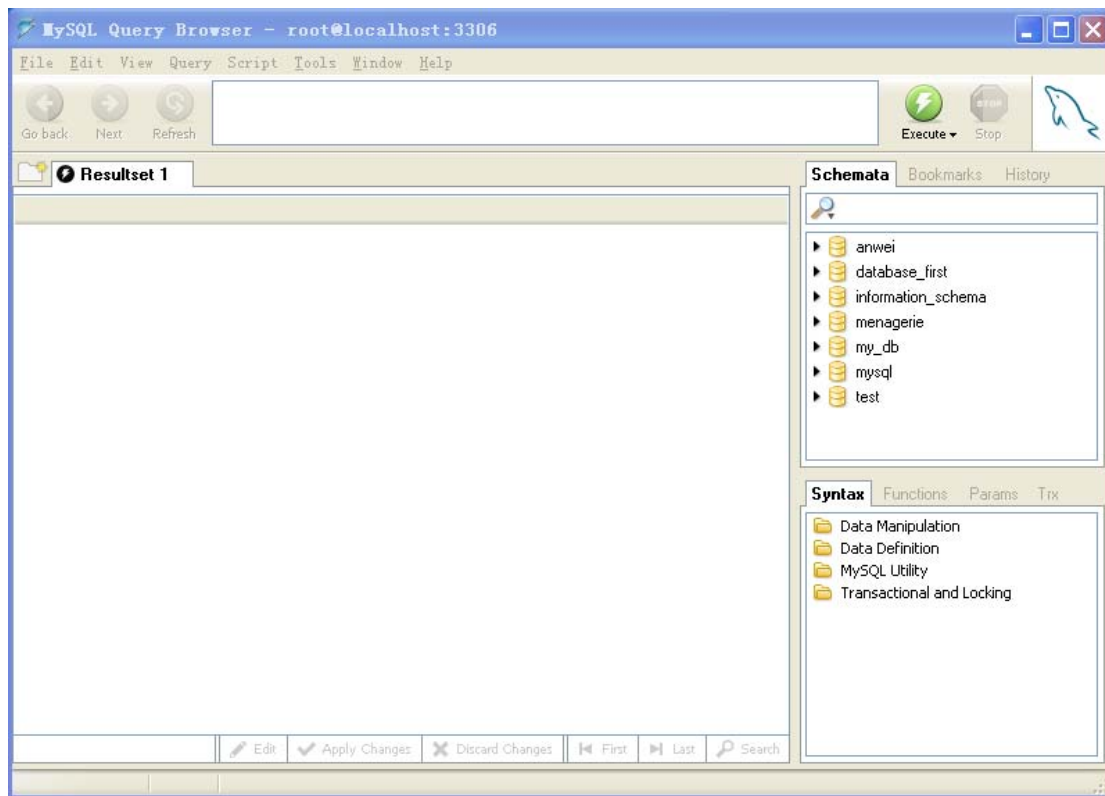


Figure 2.3 Interface of MySQL Query Browser

2.2.3 The Advantages of MySQL

MySQL's main competitors are PostgreSQL, Microsoft SQL Server, and Oracle.

MySQL has many strengths, including the following:

High performance, Low cost, Ease of configuration and learning, Portability, Availability of source code, Availability of support(PHP and Mysql)[4].

2.3 Apache

2.3.1 What Is Apache?

Apache is the world ranking Web server software. It can run on almost all widely used computer platforms. Because its cross-platform and security are widely used, Apache is one of the most popular Web servers.



Figure 2.4 The logo of Apache[6]

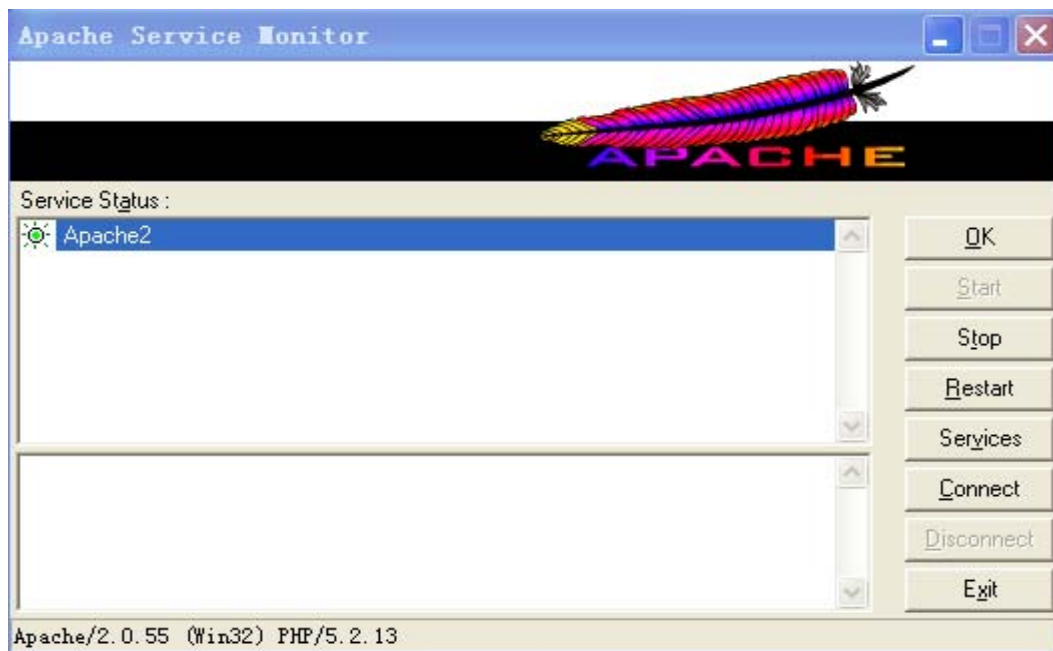


Figure 2.5 Interface of Apache

2.3.2 The Properties of Apache

Apache web server software has the following features:

Support the latest HTTP/1.1 protocol

Have a simple but powerful file-based configuration process

Support the Common Gateway Interface

Support for virtual host based IP and domain name

Support a variety of ways' HTTP Authentication

Integrated Perl processing module

Integrated proxy server module

Support real-time monitoring server status and customizing server logs

Support server-side include directive (SSI)

Support Security Socket Layer (SSL)

Provide the track of the processing of user conversation

Support for FastCGI

Through third-party modules to support Java Servlets[7]

3 PHP Encryption Technique

Nowadays PHP is installed with MD5 or Salt, they use strong 12-bits salt. But you must know what value the system uses. You can use the below PHP code to check the configuration of the server.

```
<?php echo "System salt size: ". CRYPT_SALT_LENGTH; ?>
```

The return answer will be 2, 9, 12 or 16, which tells your system what the value is.

Using MD5 or salt, you can apply crypt() and md5() in salt parameters, in order to achieve random data. MD5() can hash reactive every string, then making the string to be a fixed-length 32-bits string.

Using crypt() and md5() to encrypt password

```
<?php

$user = strip_tags(substr($_POST['user'],0,32));

$pw = strip_tags(substr($_POST['password'],0,32));

$cleanpw = crypt(md5($pw),md5($user));

$sql = "insert into users (username,password)

values('".mysql_real_escape_string($user)."',

'".mysql_real_escape_string($cleanpw)."');

//.....etc.....

?>
```

In the present age, the database has its encrypted password. However there is no way to decode the encrypted password. How to use it?

There is an easier method:

Every incoming password provided by user uses the same encrypting method. Then the result is compared to the stored password.

Revisit verify.php

```
<?php
$user = strip_tags(substr($_POST['user'],0,32));

$password = strip_tags(substr($_POST['password'],0,32));

$cryptedpw = crypt(md5($password),md5($user));

$sql = "select user,password from users
where user='". mysql_real_escape_string($user)."'
and password='". mysql_real_escape_string($cryptedpw)."'
limit 1";

$result = mysql_query($sql);

if (mysql_num_rows($result)){

//we have a match!

}else{

//no match

}
```


?>

For example, if the stored encrypted password is i83Uw28jKzBrZF, the encryption saves the incoming password and the stored encrypted password will be compared with the stored password. The unique method to destroy encryption from attackers is to use a special long string to compare with encrypted password until finding the match. This is also known as the dictionary attack, so your password should not be the string name of password and Star Trek especially your name. To ensure your password you should have long bits and include majuscules, numbers and special strings such as ! or \$, This will improve the difficulty of guessing your data.

My study case uses MD5() to encrypt. However, if you want to send a message to someone and provide some methods of decoding information, what should you do? Please use GPG (PGP) .

“GnuPG is the GNU project's complete and free implementation of the OpenPGP standard as defined by RFC4880 . GnuPG allows to encrypt and sign your data and communication, features a versatile key management system as well as access modules for all kind of public key directories. GnuPG, also known as *GPG*, is a command line tool with features for easy integration with other applications. A wealth of frontend applications and libraries are available. Version 2 of GnuPG also provides support for S/MIME.[8]”

GnuPG is Free Software. It can be freely used, modified and distributed under the terms of the GNU General Public License .

GnuPG comes in two flavours: 1.4.10 is the well known and portable standalone version, whereas 2.0.15 is the enhanced and somewhat harder to build version.

Project Gpg4win provides a Windows Version of GnuPG. It is nicely integrated into an installer and features several frontends as well as manuals in Germany.

Project Aegypten developed the S/MIME functionality in GnuPG 2.

(<http://www.gnupg.org/>)

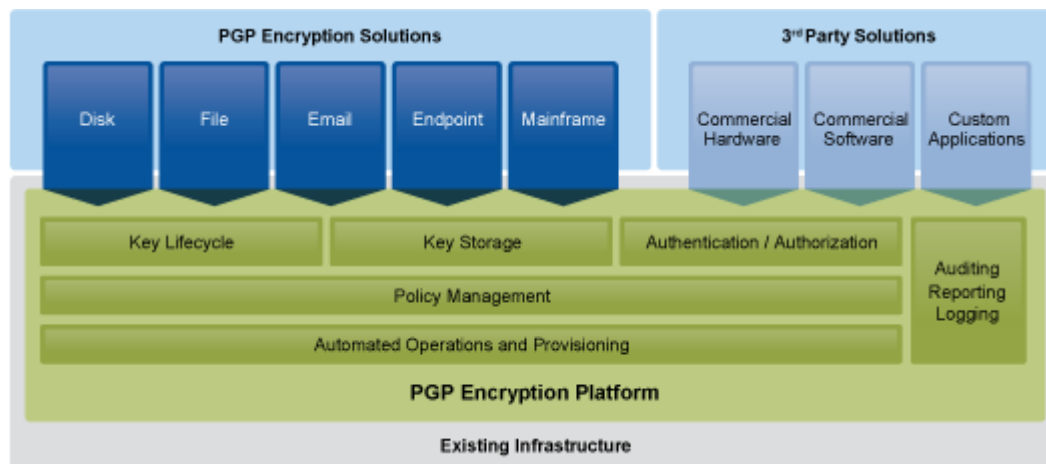


Figure 3.1 The Process of PGP Encryption[9]

Use GPG

```
<?php
```

```
//set up users
```

```
$from = "webforms@example.com";
```

```
$to = "you@example.com";
```

```
//cut the message down to size, remove HTML tags
```

```
$messagebody = strip_tags(substr($_POST['msg'],0,5000));
```

```
$message_body = escapeshellarg($messagebody);
```

```
$gpg_path = '/usr/local/bin/gpg';
```

```
$home_dir = '/htdocs/www';
```

```
$user_env = 'web';
```

```
$cmd = "echo $message_body HOME=$home_dir USER=$user_env  
$gpg_path" .
```

```
"--quiet --no-secmem-warning --encrypt --sign --armor " .
```

```
"--recipient $to --local-user $from";
```

```
$message_body = ` $cmd `;
```

```
mail($to,'Message from Web Form', $message_body,"From:$from\r\n");
```

```
?>
```

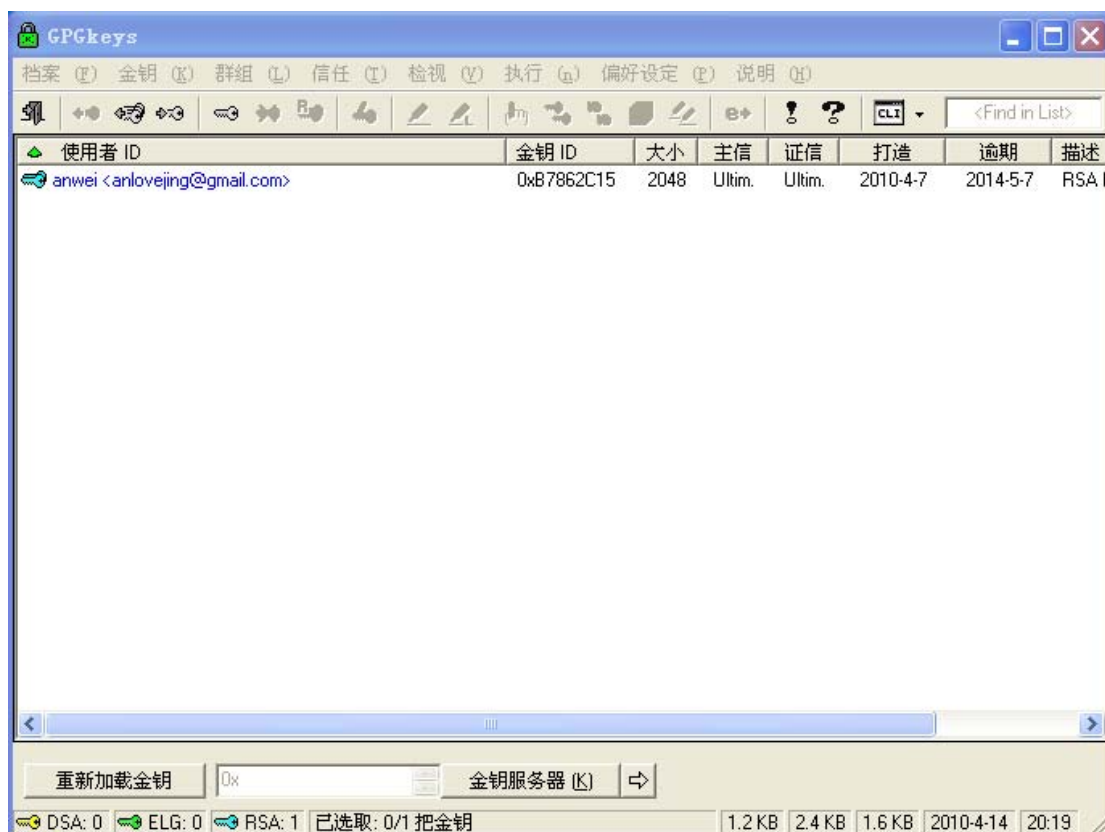


Figure 3.2 Interface of GPG

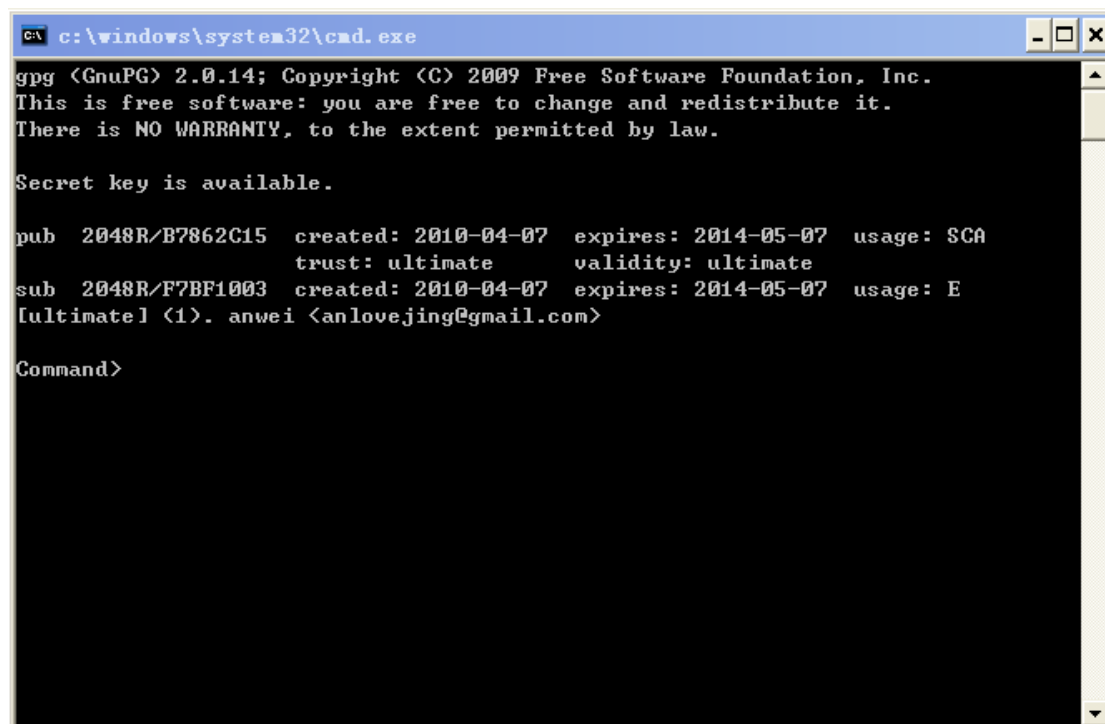


Figure 3.3 The Background Operation of GPG



Figure 3.4 The Interface of Key Generation

In this example, PHP calls /usr/local/bin/gpg(this location will be changed by server), in order to use the sender'private key and receiver'key to encrypt the information.In addition, we can configure the environment variable of HOME and USER to inform GPG where to find keyring.

Other marked functions below:

--quiet and --no-secmem-warning Restrain the warning from GPG

--encrypt

--sign

--armor Produce non-binary ASCII output so as to be sent by email.

4 My OWN APPLICATION

4.1 Install the PHP environment

Nowadays, the most fashionable website frame mode is WAMP
(WINDOWS+Apache+MySQL+PHP)

Download those softwares:

Apache official download address: apache_2.0.55-win32-x86- no_ssl.msi.

PHP official download address: php-5.0.5-Win32.zip.

MySQL official download address: mysql-4.1.14-win32.zip

4.1.1 Install Apache

Run downloaded “apache_2.0.55-win32-x86-no_ssl.msi”,show this interface:

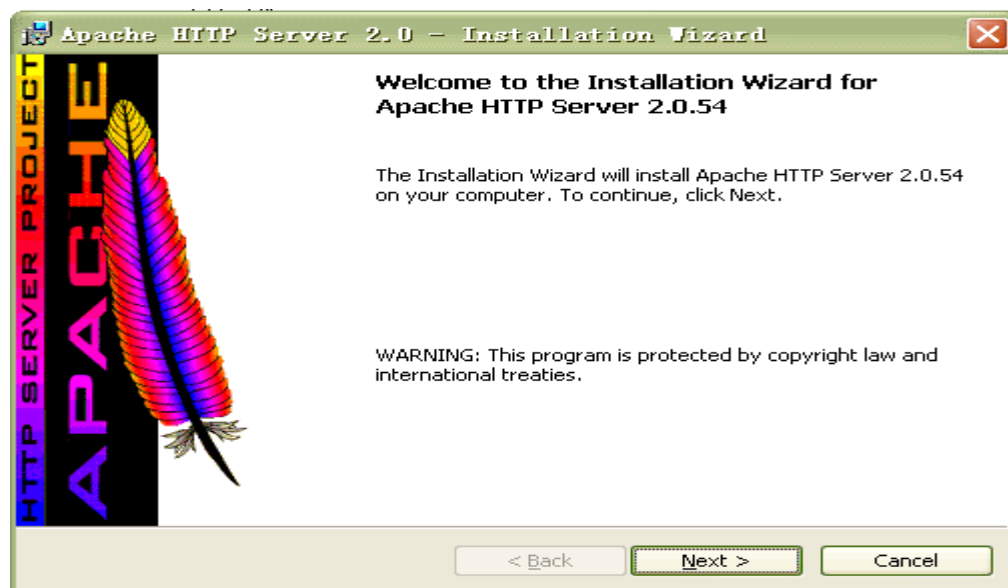


Figure 4.1 Interface of Installation Wizard 1

Show Apache HTTP [Server](#) 2.0.55 Installation Wizard, click“Next”

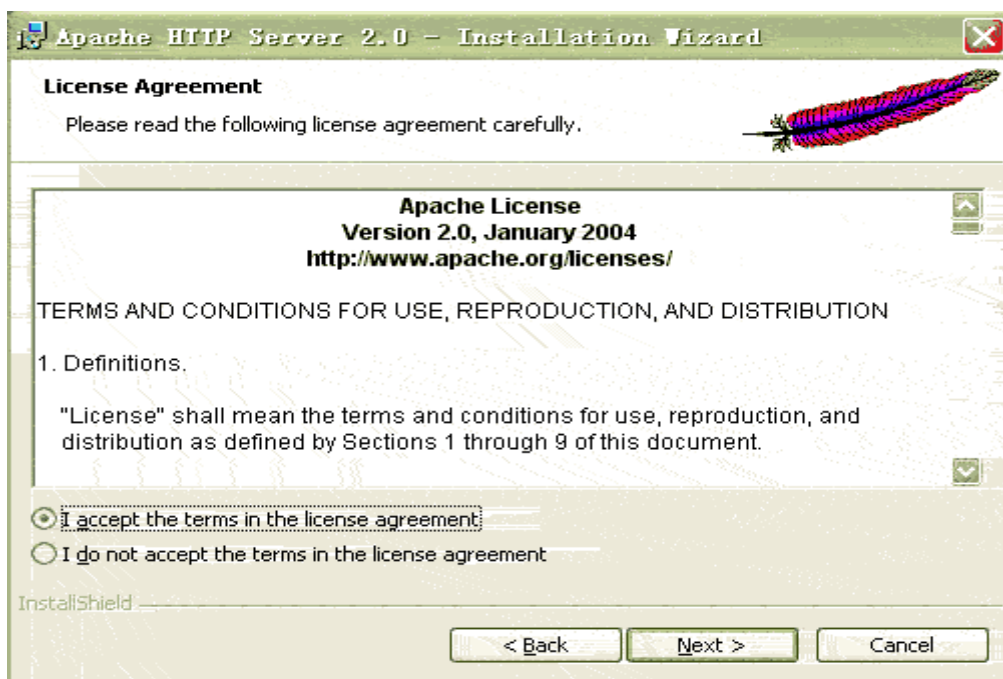


Figure 4.2 Interface of Installation Wizard 2

Choose “I accept the terms in the license agreement”, click “Next”

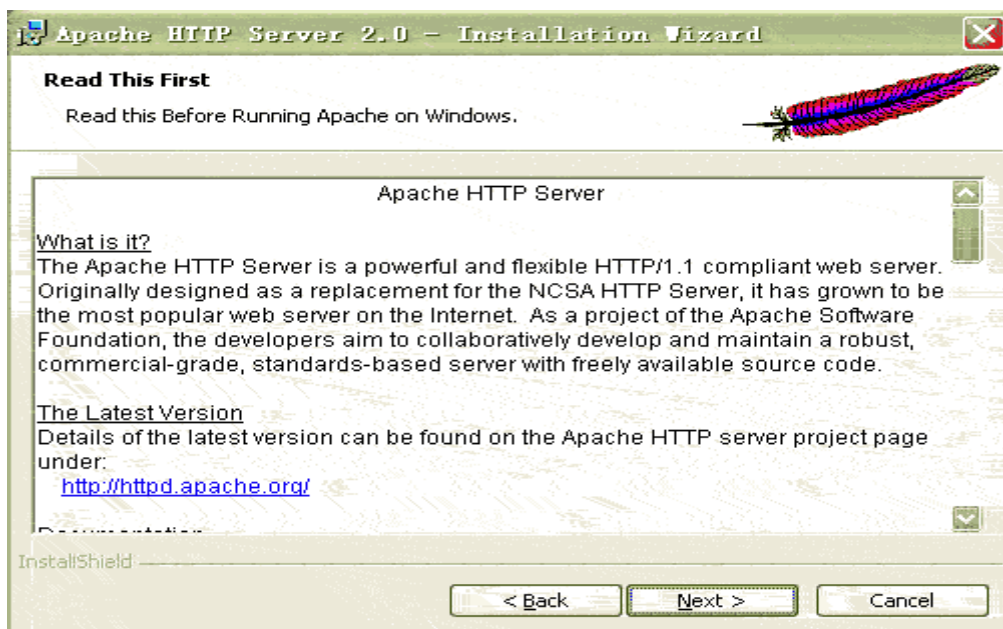


Figure 4.3 Interface of Installation Wizard 3

Complete the words, click “Next”

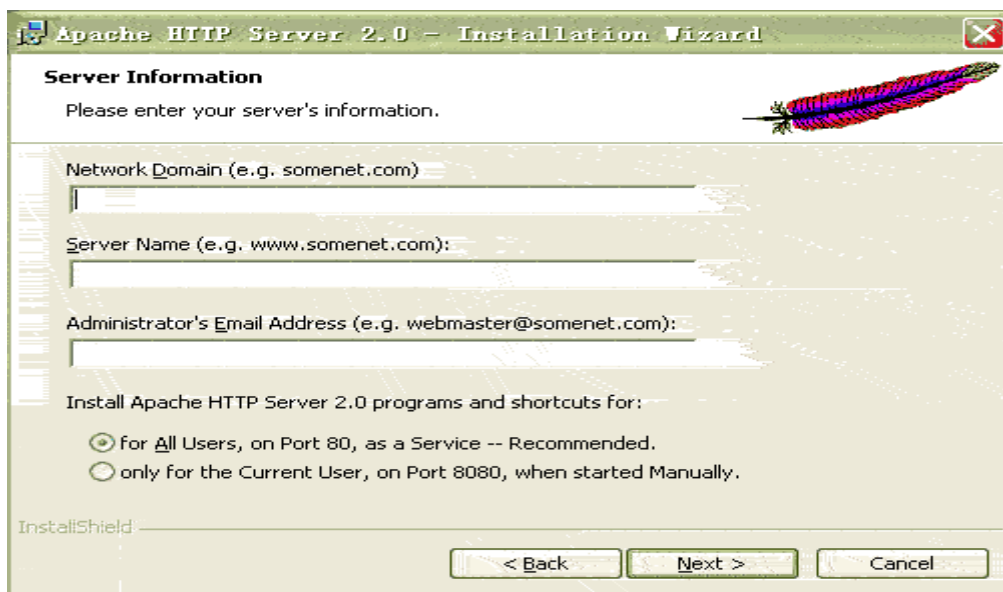


Figure 4.4 Interface of Installation Wizard 4

Configure the system information, fill-in your domain name in Network Domain, fill in your server name in your Server Name, fill-in System Administrator' email address in Administrator'Email Address, choose "for ALL Users, on Port 80, as a Service – Recommended."

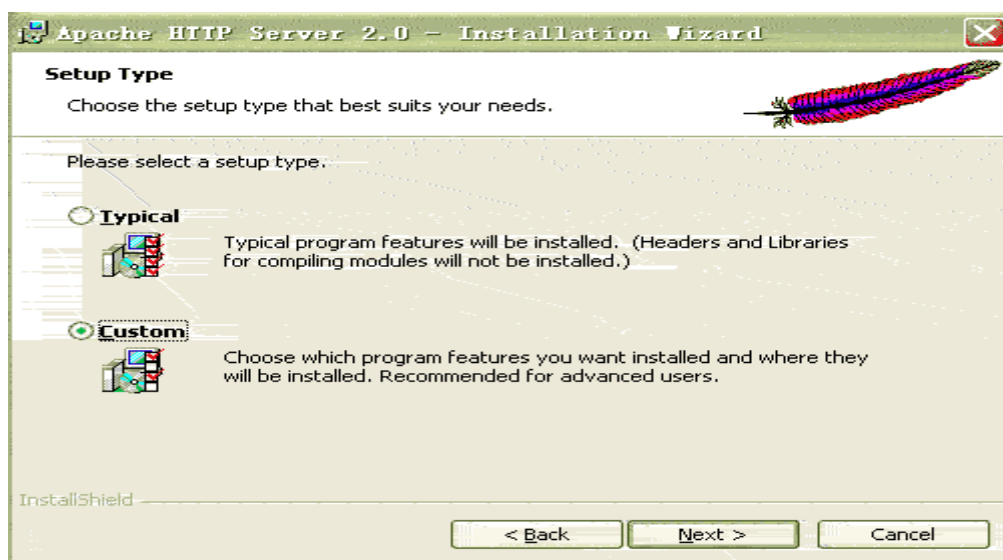


Figure 4.5 Interface of Installation Wizard 5

Choose the installing style, Typical is the default setup. Here we choose the Typical.

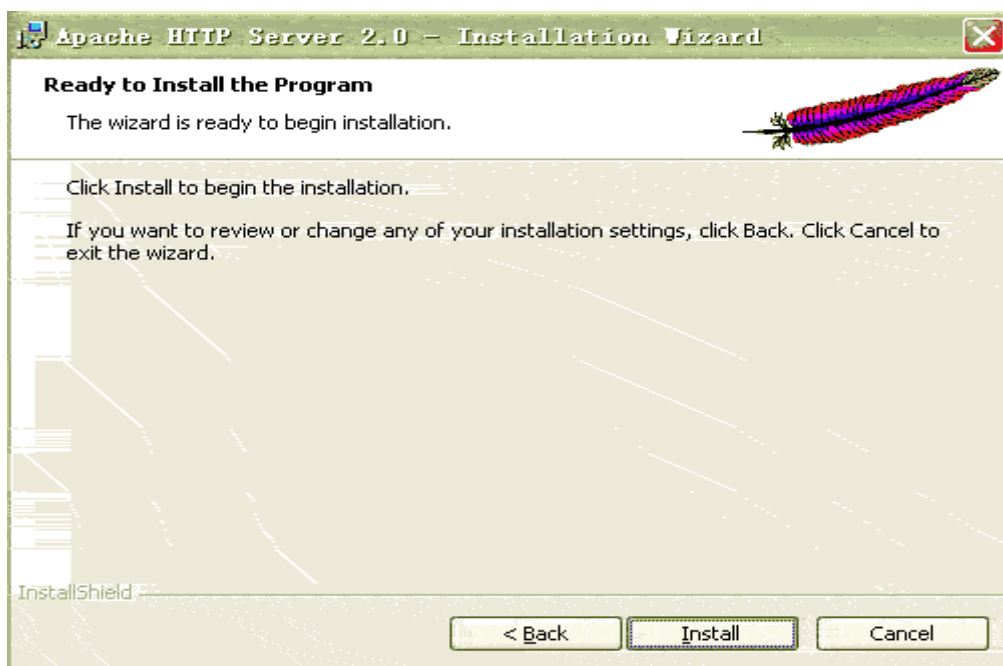


Figure 4.6 Interface of Installation Wizard 6

Click 'Install'

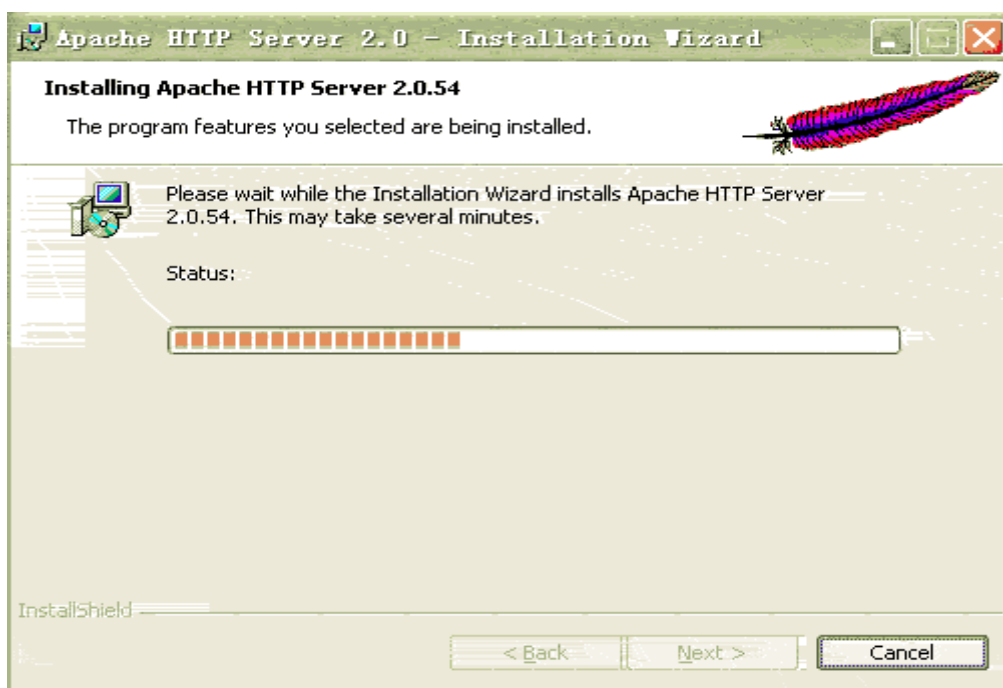


Figure 4.7 Interface of Installation Wizard 7

Having installed the interface, please wait patiently until this picture.

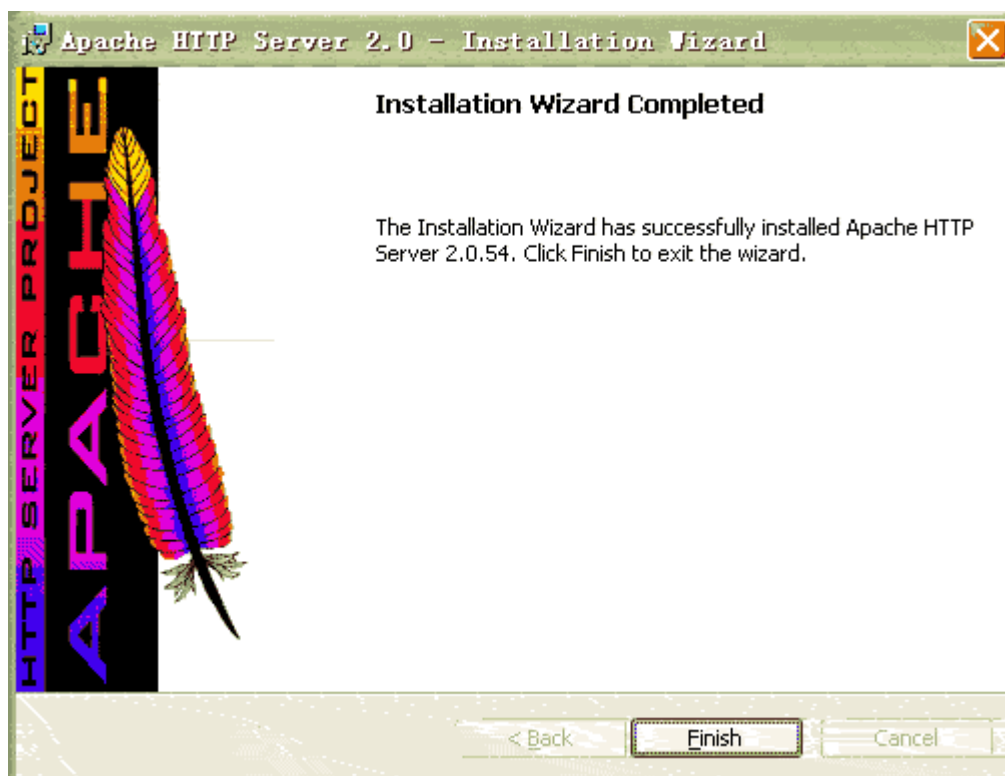


Figure 4.8 Interface of Installation Wizard 8

The installation is finished. By now a status bar will appear this green logo, showing that the Apache server has been started, click 'Finish'



Figure 4.9 Shortcutkey of Apache

Now we can test the configuration, fill in "http://127.0.0.1", click 'go to'. If you see this page, The installation of Apache has been successful.

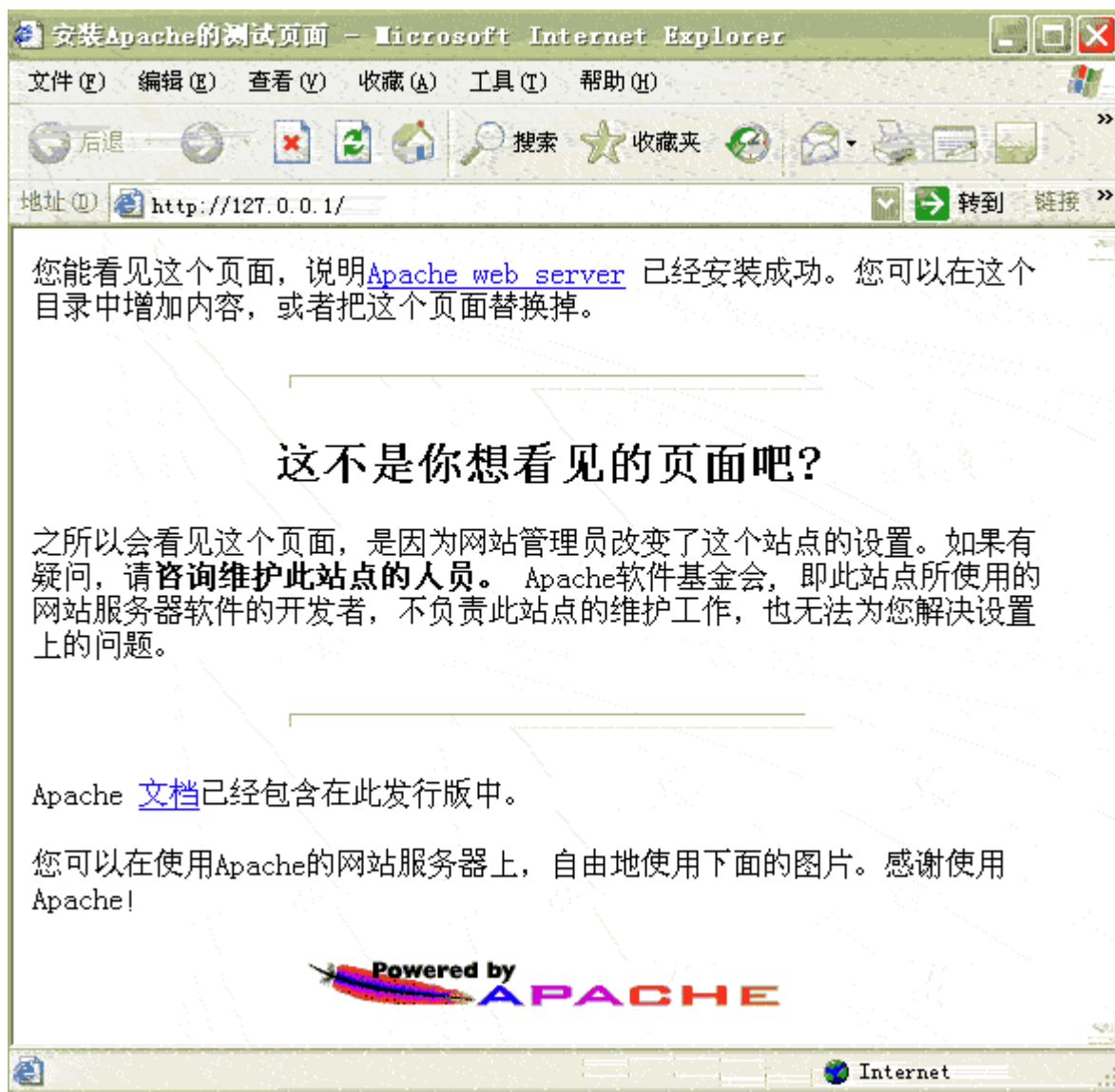


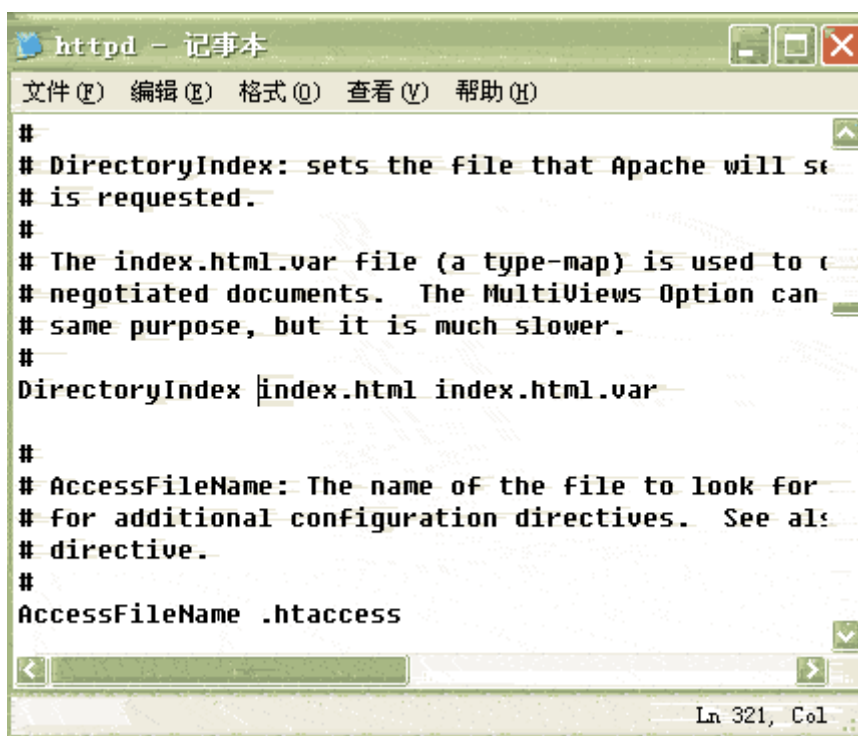
Figure 4.10 Test Apache Web Server

4.1.2 Configure the Apache Server

Now we start to configure the Apache server. Actually, if you don't configure, Apache2\htdocs under the installing directory is the default directory, where you can put files.

Open D:\Apache2\conf\httpd.conf file.

Find 'Ln321',DirectoryIndex. Add index.php behind DirectoryIndex index.html
index.html.var



```

httpd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
#
# DirectoryIndex: sets the file that Apache will serve if no
# file is requested.
#
# The index.html.var file (a type-map) is used to serve
# negotiated documents. The MultiViews Option can be used for the
# same purpose, but it is much slower.
#
DirectoryIndex index.html index.html.var

#
# AccessFileName: The name of the file to look for when the
# file does not exist. See also the AllowOverride
# directive.
#
AccessFileName .htaccess
Ln 321, Col 1

```

Figure 4.11 The file of httpd

OK,the simple Apache configuration is now finished. Next you can restart the Apache server. Your site will become a site's server.If you add firewall,please open 80 or 8080 port or permit Apache to access to network,otherwise no one can access. If you have a public IP, you can invite all your friends to visit <http://your IP address>.(For example, <http://www.goodwaiter.com> quering content is IP address). If you don't have public IP,you can tell your LAN IP address to other users in the same LAN. And they can access your site through <http://your LAN IP address>.

4.1.3 Install PHP and Integrate Apache with PHP

Download php setup file :[php-5.0.5-Win32.zip](#) and click decode.

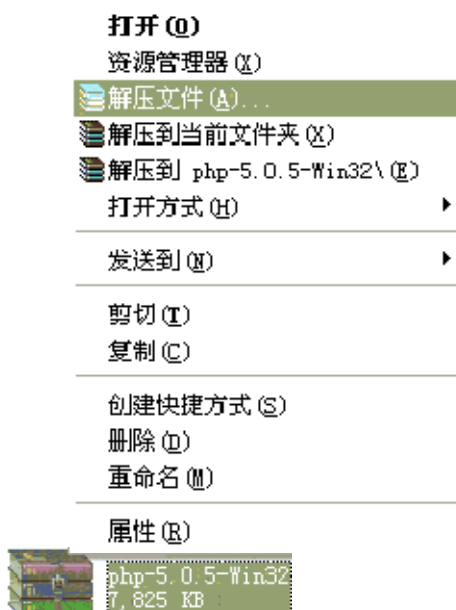


Figure 4.12 Interface of Uncompress

Appoint decoding location 'D:\php'



Figure 4.13 Configure The Uncompress Path

Look at folder after decoding, find 'php.ini-dist' file, change the name to 'php.ini'. Then move 'php.ini' to 'C:/WINDOWS/' Find php5ts.dll in PHP installing directory and

move it to 'system/system32'. Find php_gd2.dll php_mysql.dll php_mystring.dll in 'php/ext' and move it to 'system/system32'. Another place need to edit, now we make php to call other module directly such as access mysql, Look the below picture, Ln563, choose the loaded module and clear off front ';', save the changes and close.



```

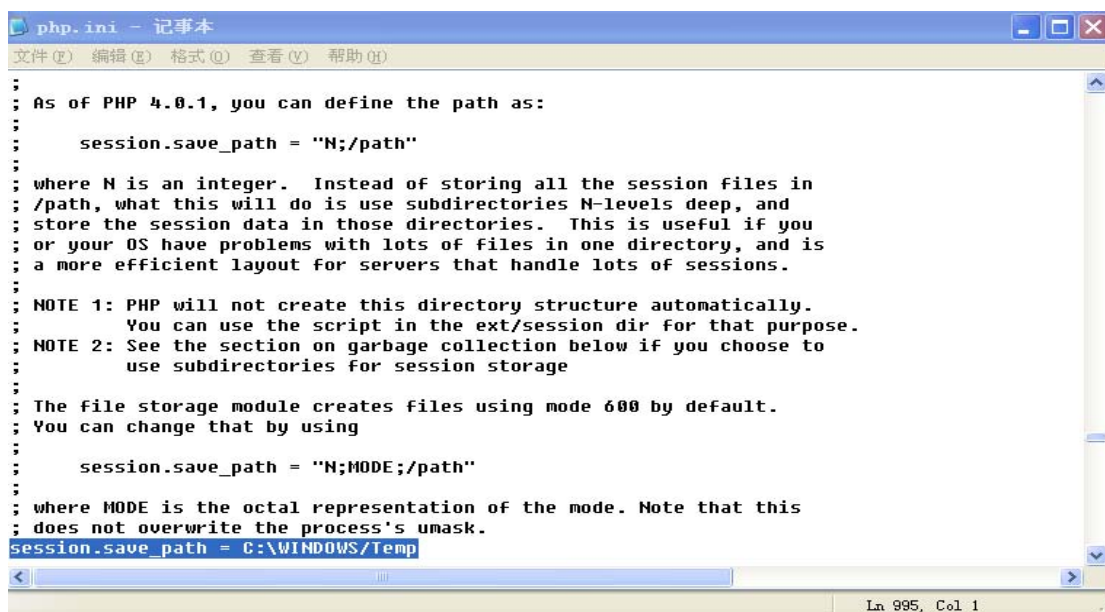
extension=php_mbstring.dll
;extension=php_bz2.dll
extension=php_cpdf.dll
extension=php_curl.dll
extension=php_dba.dll
extension=php_dbase.dll
extension=php_dbx.dll
extension=php_exif.dll
extension=php_fdf.dll
extension=php_filepro.dll
extension=php_gd2.dll
extension=php_gettext.dll
;extension=php_ifx.dll
;extension=php_iisfunc.dll
extension=php_imap.dll
extension=php_interbase.dll
;extension=php_java.dll
extension=php_ldap.dll
;extension=php_mcrypt.dll
extension=php_mhash.dll
extension=php_mime_magic.dll
extension=php_ming.dll
extension=php_mssql.dll
extension=php_mysql.dll
;extension=php_oci8.dll
extension=php_openssl.dll
;extension=php_oracle.dll
;extension=php_pdf.dll
extension=php_pgsql.dll
extension=php_shmop.dll
extension=php_snmp.dll
extension=php_sockets.dll
;extension=php_sybase_ct.dll
extension=php_tidy.dll
;extension=php_w32api.dll
extension=php_xmlrpc.dll
extension=php_xsl.dll
;extension=php_yaz.dll
;extension=php_zip.dll

```

Figure 4.14 The file of php.ini 1

Find ';session.save_path = "/tmp"' and clear away ';',

Change it into 'session.save_path= C:WINDOWS\Temp'



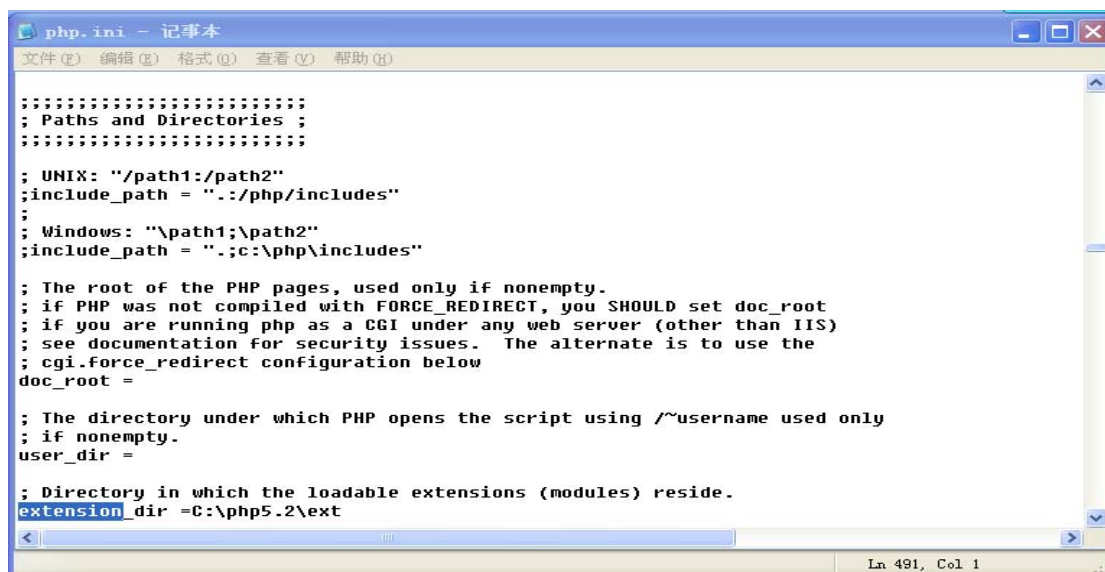
```

;
; As of PHP 4.0.1, you can define the path as:
;
;   session.save_path = "N;/path"
;
; where N is an integer.  Instead of storing all the session files in
; /path, what this will do is use subdirectories N-levels deep, and
; store the session data in those directories.  This is useful if you
; or your OS have problems with lots of files in one directory, and is
; a more efficient layout for servers that handle lots of sessions.
;
; NOTE 1: PHP will not create this directory structure automatically.
;         You can use the script in the ext/session dir for that purpose.
; NOTE 2: See the section on garbage collection below if you choose to
;         use subdirectories for session storage
;
; The file storage module creates files using mode 600 by default.
; You can change that by using
;
;   session.save_path = "N;MODE;/path"
;
; where MODE is the octal representation of the mode. Note that this
; does not overwrite the process's umask.
session.save_path = C:\WINDOWS/Temp

```

Figure 4.15 The File of php.ini 2

Find 'extension_dir = "./'" and change it to 'extension_dir = C:\php5.2\ext



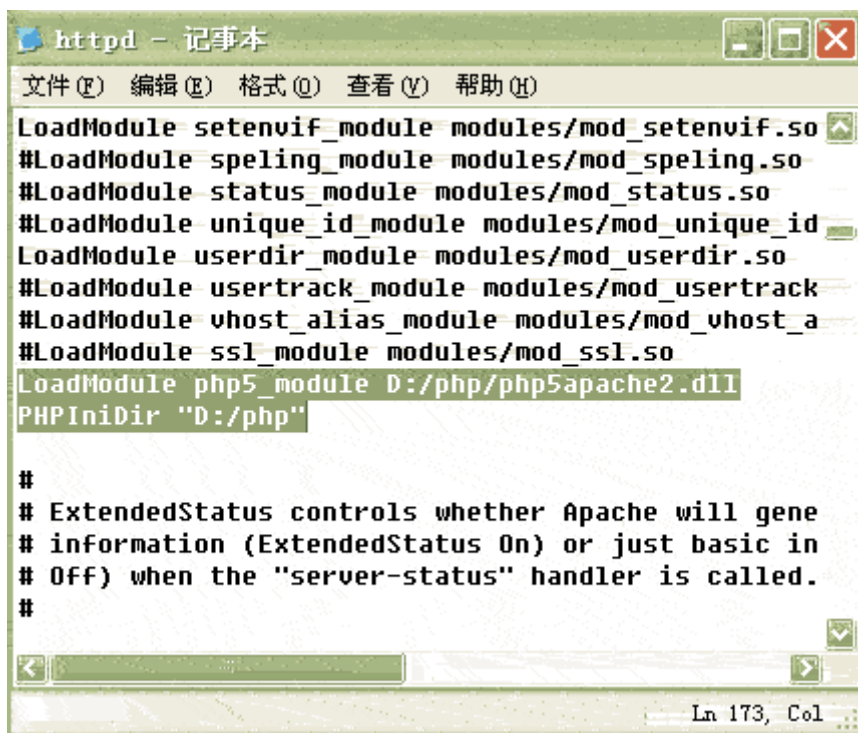
```

;;;;;;;;;;;;;;;;;;;;;;;;;
; Paths and Directories ;
;;;;;;;;;;;;;;;;;;;;;;;;;
; UNIX: "/path1:/path2"
;include_path = "./:/php/includes"
;
; Windows: "\path1;\path2"
;include_path = ".;c:\php\includes"
;
; The root of the PHP pages, used only if nonempty.
; if PHP was not compiled with FORCE_REDIRECT, you SHOULD set doc_root
; if you are running php as a CGI under any web server (other than IIS)
; see documentation for security issues.  The alternate is to use the
; cgi.force_redirect configuration below
doc_root =
;
; The directory under which PHP opens the script using /~username used only
; if nonempty.
user_dir =
;
; Directory in which the loadable extensions (modules) reside.
extension_dir = C:\php5.2\ext

```

Figure 4.16 The File of php.ini 3

Now integrate php and Apache and make php into Apache. Open the Apache' configuration file, Ln 173. Add 'LoadModule php5_module D:/php/php5 apache2.dll' behind '#LoadModule ssl_module modules/mod_ssl.so'




```
httpd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
LoadModule setenvif_module modules/mod_setenvif.so
#LoadModule speling_module modules/mod_speling.so
#LoadModule status_module modules/mod_status.so
#LoadModule unique_id_module modules/mod_unique_id
LoadModule userdir_module modules/mod_userdir.so
#LoadModule usertrack_module modules/mod_usertrack
#LoadModule vhost_alias_module modules/mod_vhost_a
#LoadModule ssl_module modules/mod_ssl.so
LoadModule php5_module D:/php/php5apache2.dll
PHPIniDir "D:/php"

#
# ExtendedStatus controls whether Apache will gene
# information (ExtendedStatus On) or just basic in
# Off) when the "server-status" handler is called.
#
Ln 173, Col 1
```

Figure 4.17 The File of php.ini 4

Find Ln 757 and add 'AddType application/x-httpd-php .php' 'AddType application/x-httpd-php .html' behind 'AddType application/x-gzip .gz .tgz'.



```

#AddEncoding x-compress .Z
#AddEncoding x-gzip .gz .tgz
#
# If the AddEncoding directives above are commented
# probably should define those extensions to indica
#
AddType application/x-compress .Z
AddType application/x-gzip .gz .tgz
AddType application/x-httpd-php .php
AddType application/x-httpd-php .html

#
# AddHandler allows you to map certain file extensi
# actions unrelated to filetype. These can be eithe
# or added with the Action directive (see below)
#

```

Figure 4.18 The File of php.ini 5

Aforesaid directory default file can change because now we add php as result some files can be saved as '.php'. We also can make 'index.php' as default index file. You can decide to arrange the priority. I put it on the first place here. Save and close.



```

#
# DirectoryIndex: sets the file that Apache will s
# is requested.
#
# The index.html.var file (a type-map) is used to
# negotiated documents. The MultiViews Option can
# same purpose, but it is much slower.
#
DirectoryIndex index.php index.html index.html.var

#
# AccessFileName: The name of the file to look for
# for additional configuration directives. See al
# directive.
#

```

Figure 4.19 The File of Apache httpd

Integrating php and Apache is completed. Restart Apache and your Apache server can support PHP.

4.1.4 Install MySQL and Integrate With PHP and Apache

Open downloaded MySQL file 'mysql-4.1.14-win32.zip' and click 'decode'. Running 'setup.exe'. Look this picture:



Figure 4.20 Interface of MySQL Setup Wizard 1

Click 'Next'

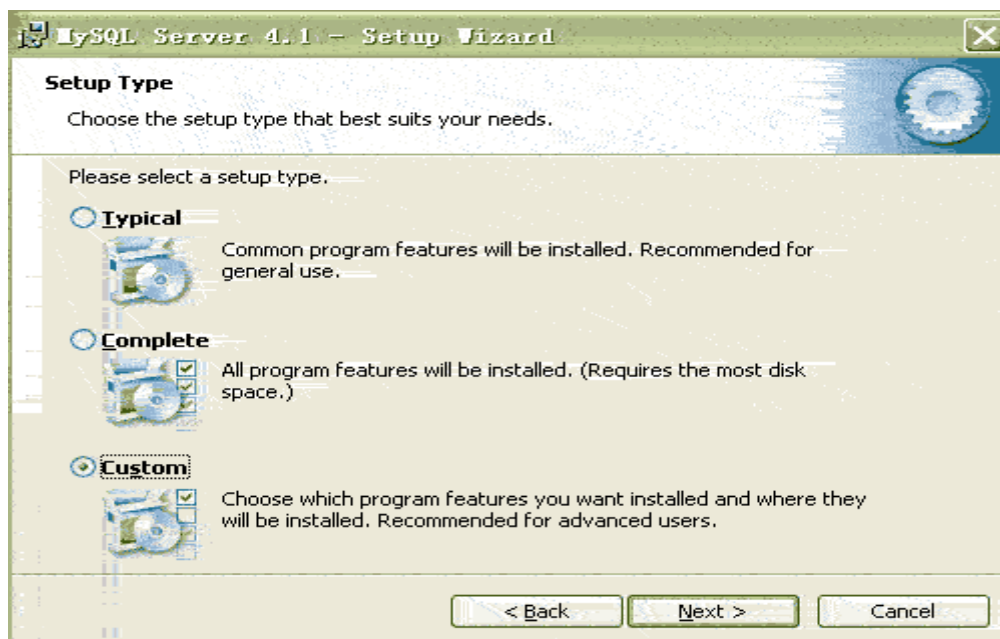


Figure 4.21 Interface of MySQL Setup Wizard 2

Here we choose 'Custom'.



Figure 4.22 Interface of MySQL Setup Wizard 3

Click 'Developer Components' and choose 'This feature, and all subfeatures, will be installed on local hard drive.' The same command is to 'MySQL Server', 'Client Programs', 'Documentation'. Then click 'Change...' to appoint installing directory.

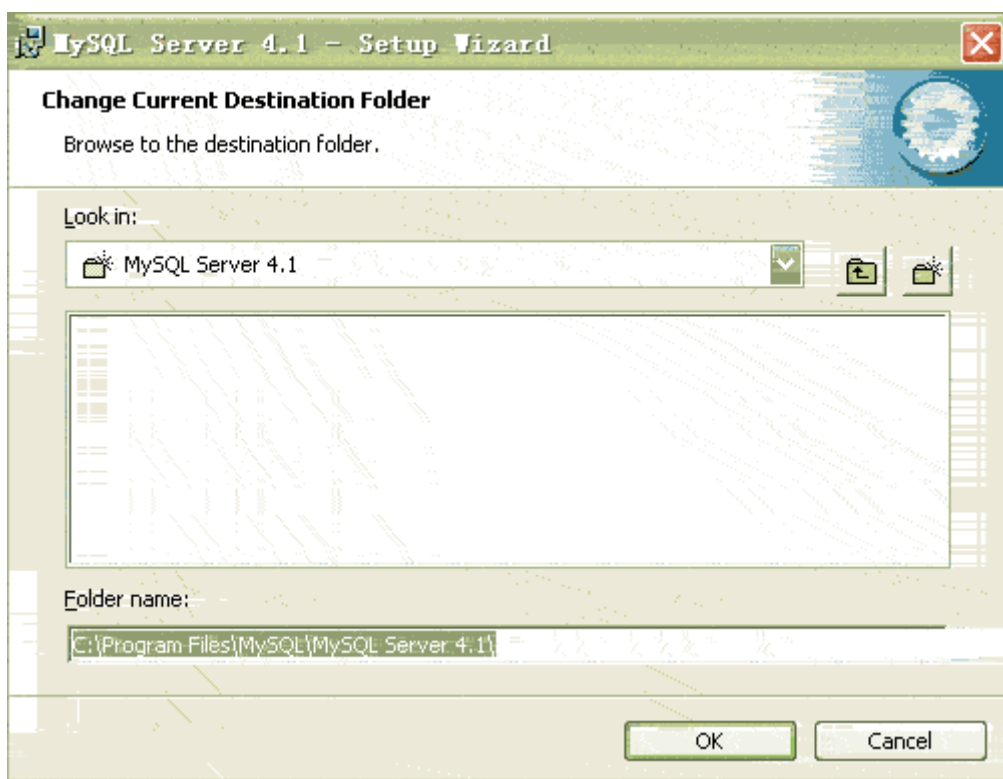


Figure 4.23 Interface of MySQL Setup Wizard 4

Add installing directory, I use 'D:\mysql'. Better not to put the folder in the system districtly because we can preserve the file when the system reset.

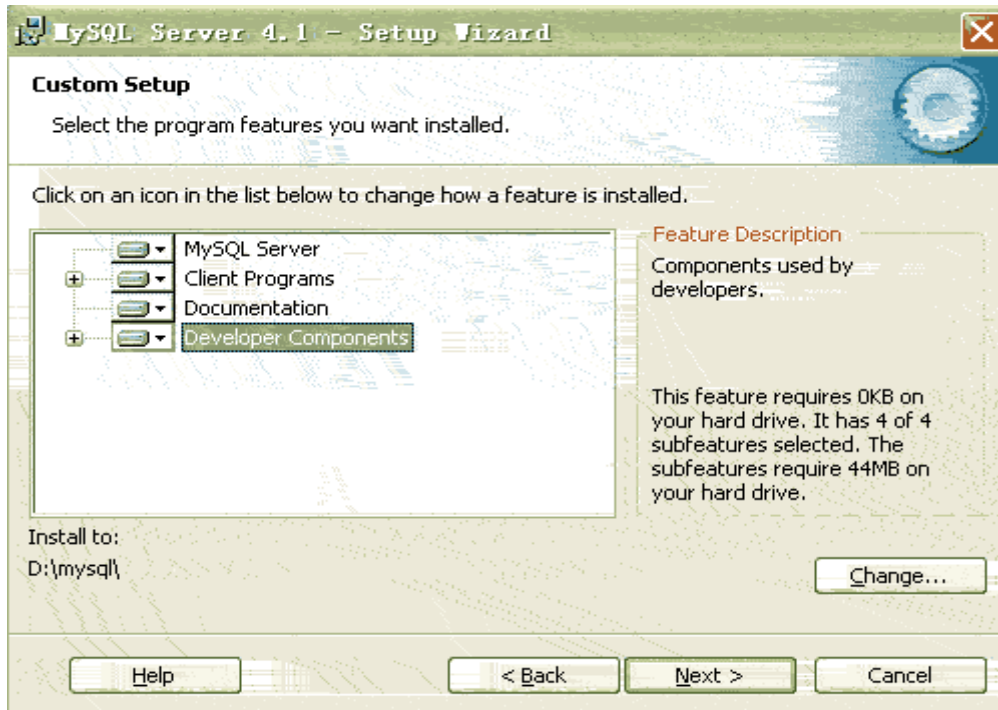


Figure 4.24 Interface of MySQL Setup Wizard 5

Click 'Next'

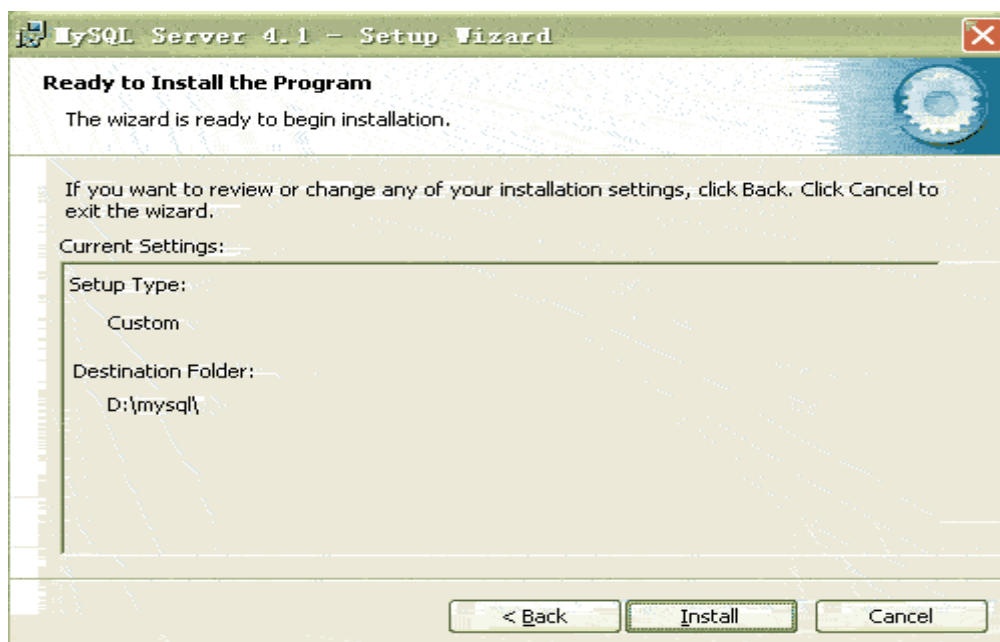


Figure 4.25 Interface of MySQL Setup Wizard 6

Ensure that the former configuration is correct, otherwise go 'Back'. Choose 'Install'.

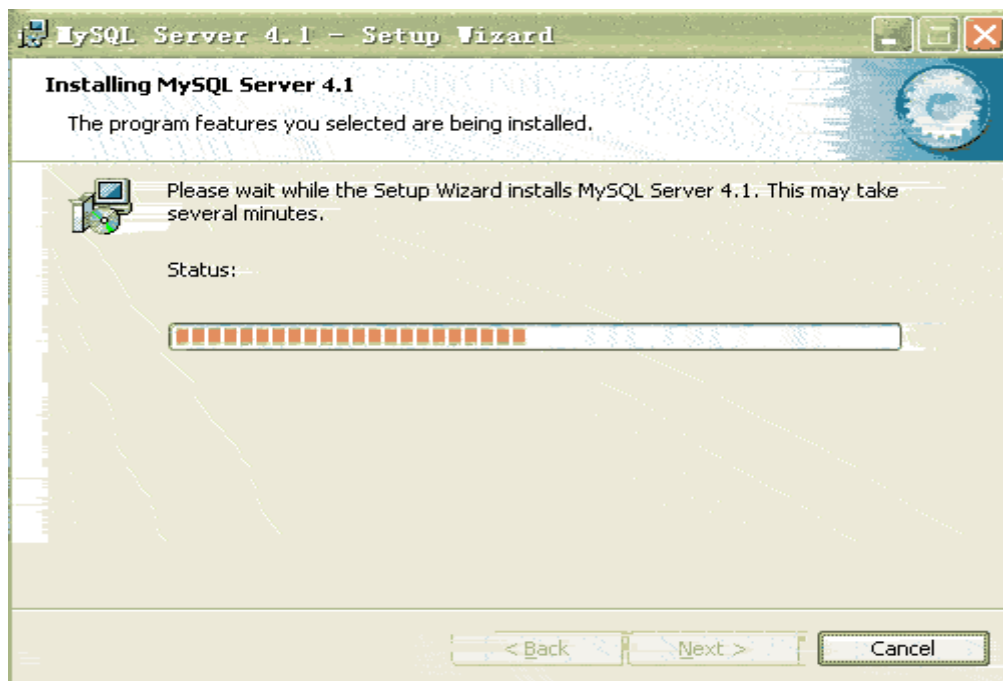


Figure 4.26 Interface of MySQL Setup Wizard 7

Waiting..

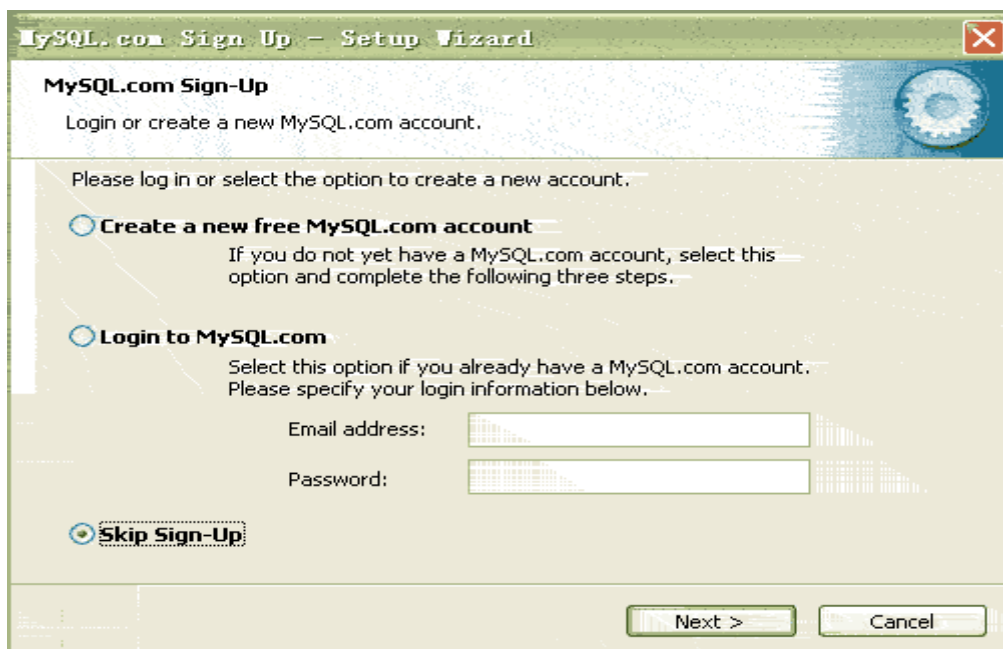


Figure 4.27 Interface of MySQL Setup Wizard 8

Here you must to ask whether you need a mysql.com'ID,or use existing ID to log in mysql.com. Commonly,we don't to create a ID. So click 'Skip Sign-Up', choose 'Next'.



Figure 4.28 Interface of MySQL Setup Wizard 9

Now the installation is completed. There is a very nice function—mysql configuration. We don't configure 'my.ini' as before. Choose 'Configure the Mysql Server now', click 'Finish' and start mysql configuration guide.

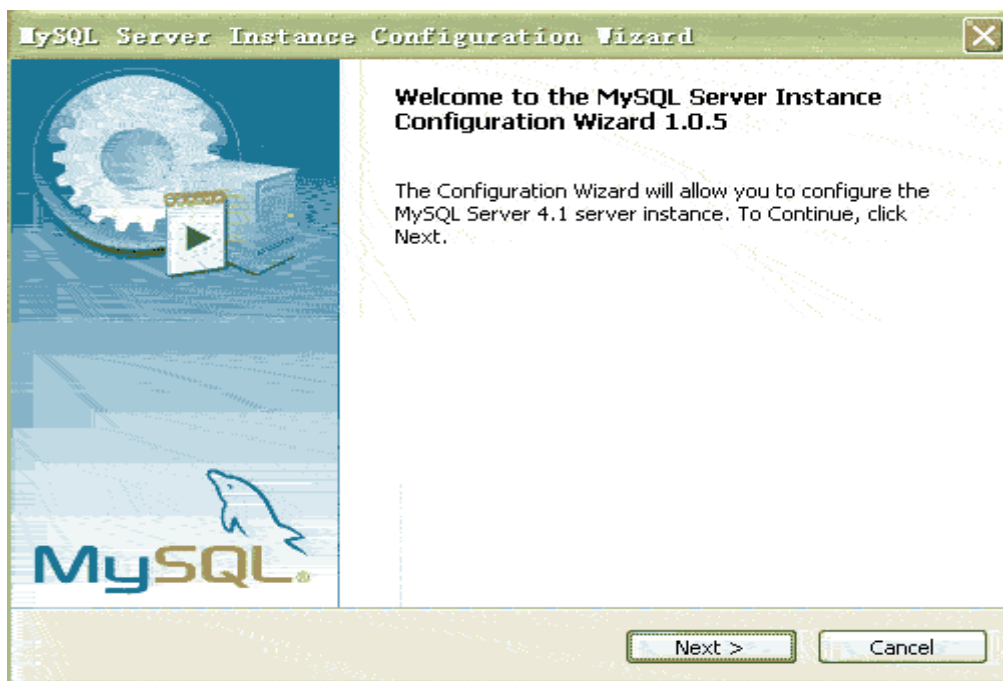


Figure 4.29 Interface of Configuration MySQL 1

Click 'Next'

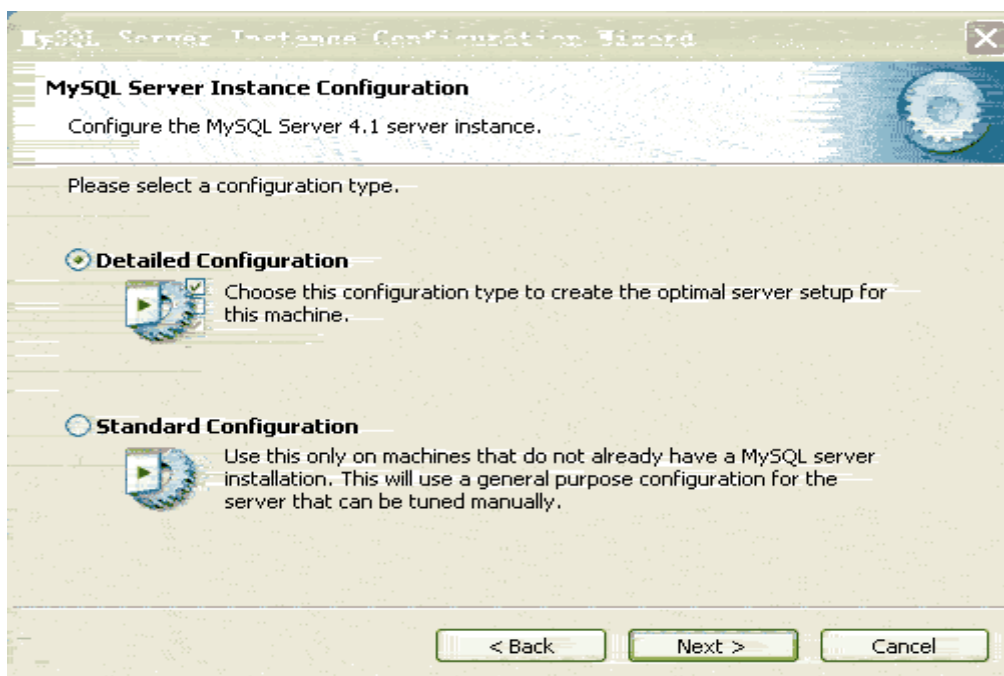


Figure 4.30 Interface of Configuration MySQL 2

Choose configuration method 'Detailed Configuration'.



Figure 4.31 Interface of Configuration MySQL 3

Here we choose 'Server Machine'.

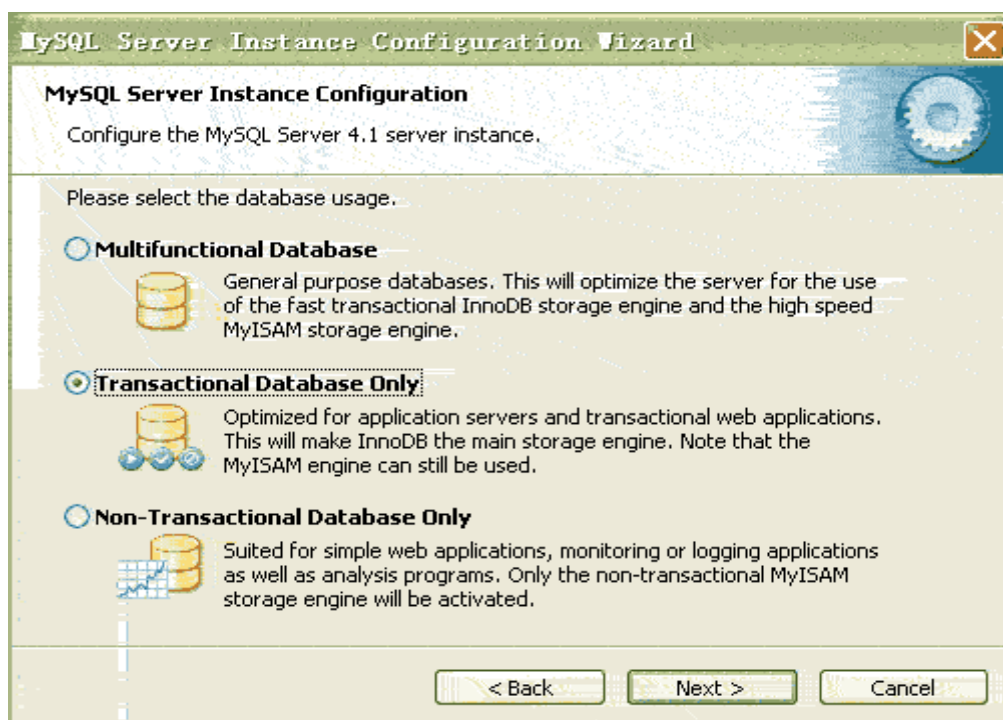


Figure 4.32 Interface of Configuration MySQL 4

On this page you must to choose the purpose of mysql database, 'Multifunctional Database' 'Transactional Database Only' 'Non-Transactional Database Only'. I choose 'Transactional Database Only', and click 'Next'.

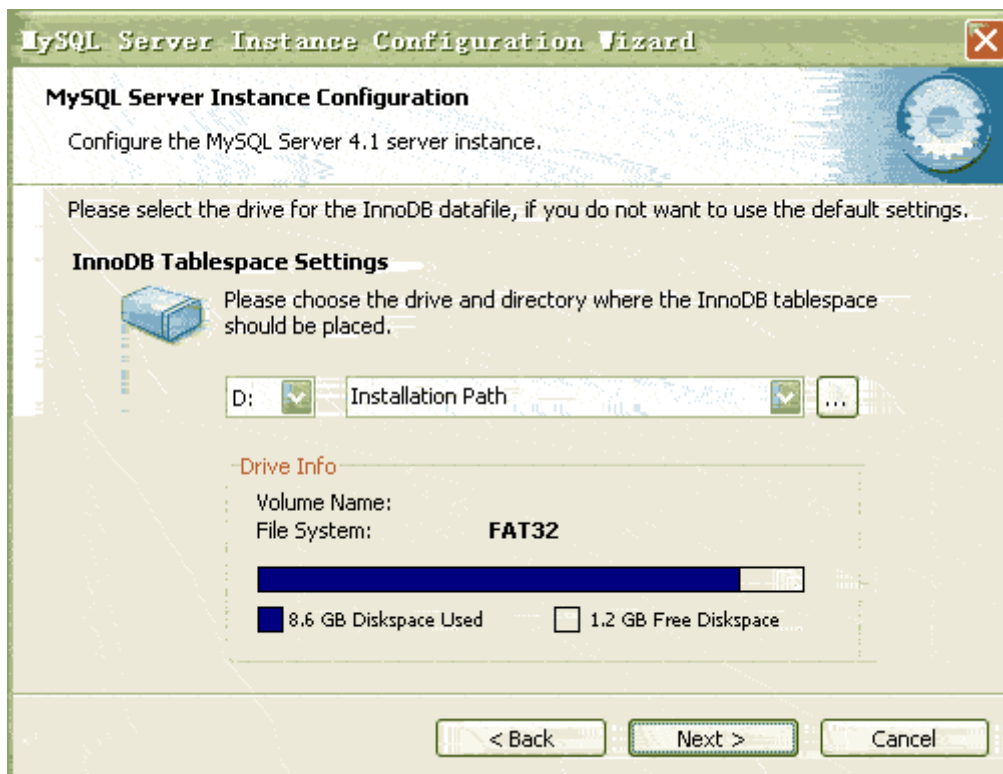


Figure 4.33 Interface of Configuration MySQL 5

Configure InnoDB Tablespace and choose a stored space. If you change it, please remember this directory. Once reset MySQL, we must choose the same place, otherwise MySQL database will be broken. Certainly, it is no problem when the database is backed up. Here I don't change it and use the default directory. Click 'Next'.

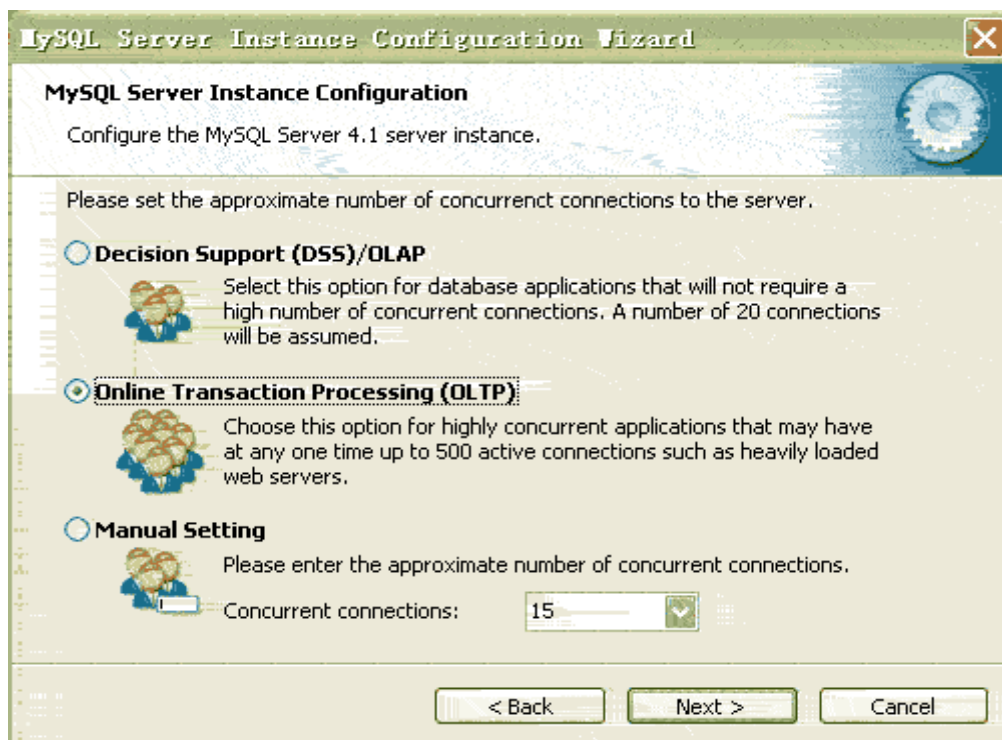


Figure 4.34 Interface of Configuration MySQL 6

Choose 'Online Transaction Processing(OLTP)' and click 'Next'.

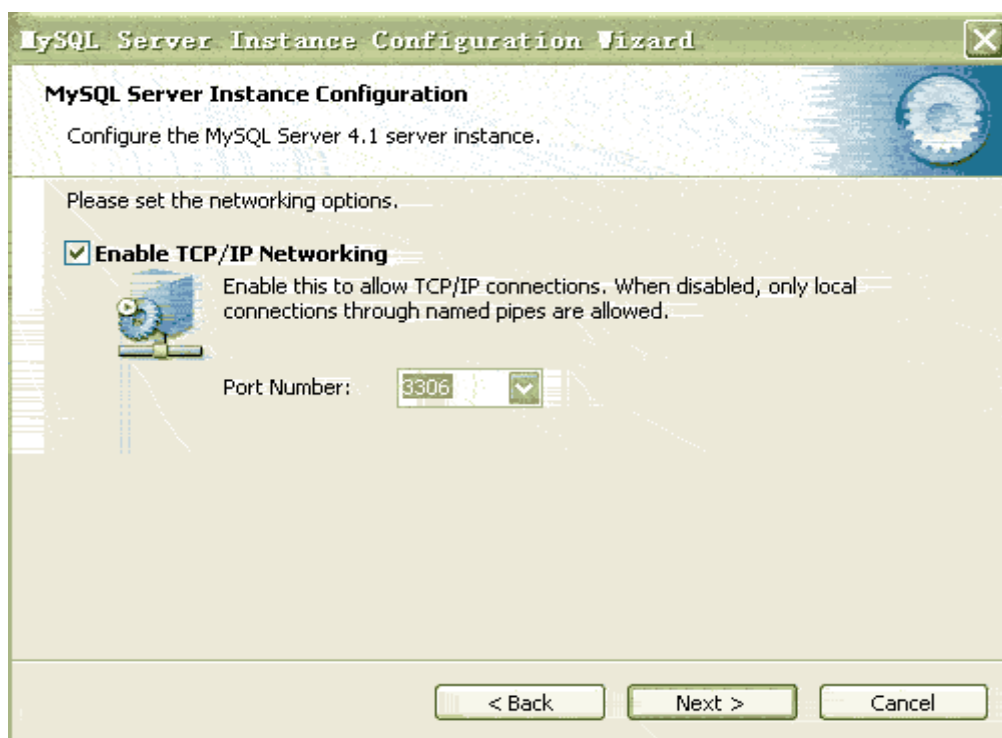


Figure 4.35 Interface of Configuration MySQL 7

Click 'Next'.

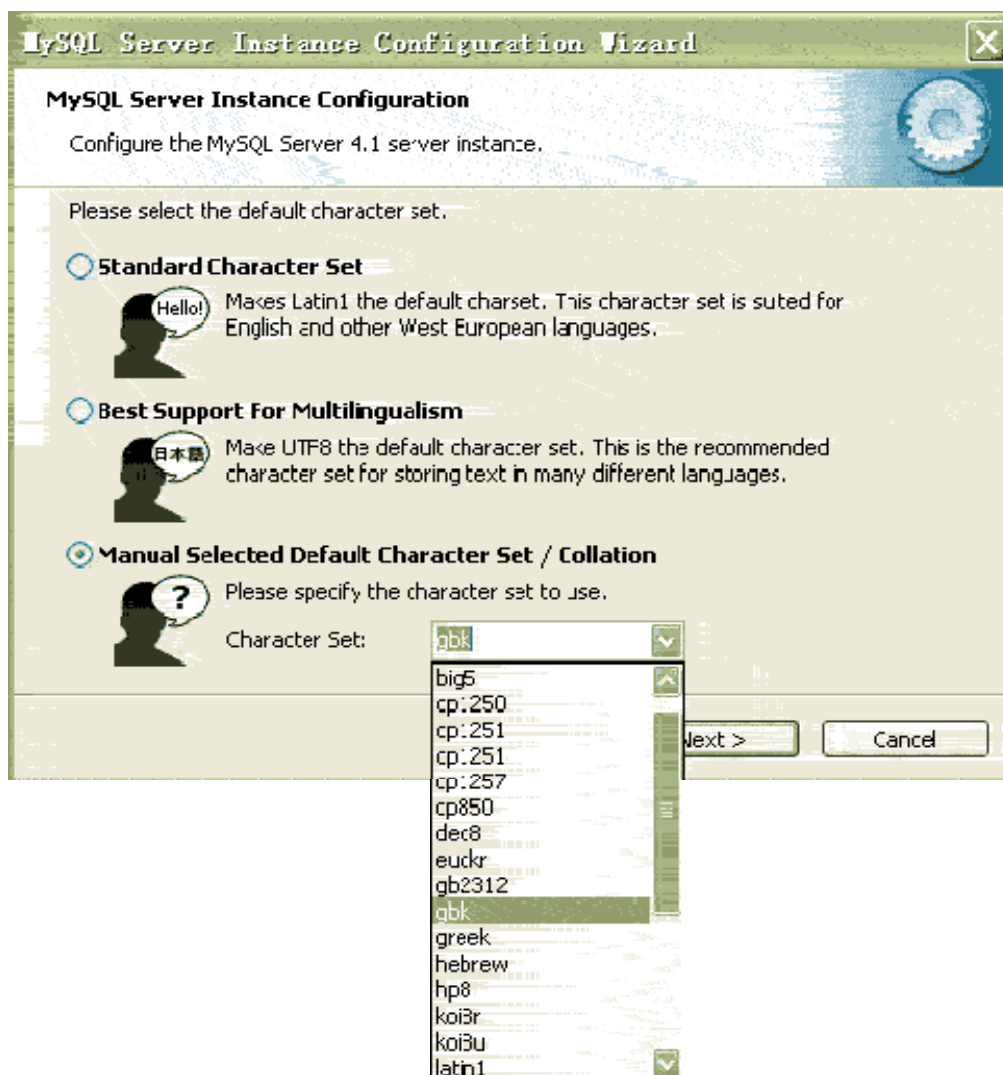


Figure 4.36 Interface of Configuration MySQL 8

Click 'Next'.



Figure 4.37 Interface of Configuration MySQL 9

Here we decide whether mysql installation serves for windows and appoint Service Name. click 'Next'.



Figure 4.38 Interface of Configuration MySQL 10

Choose 'Modify Security Settings'. This step is to ask you to alter default root user. Fill-in new root password and confirm new password. Click 'Next'.

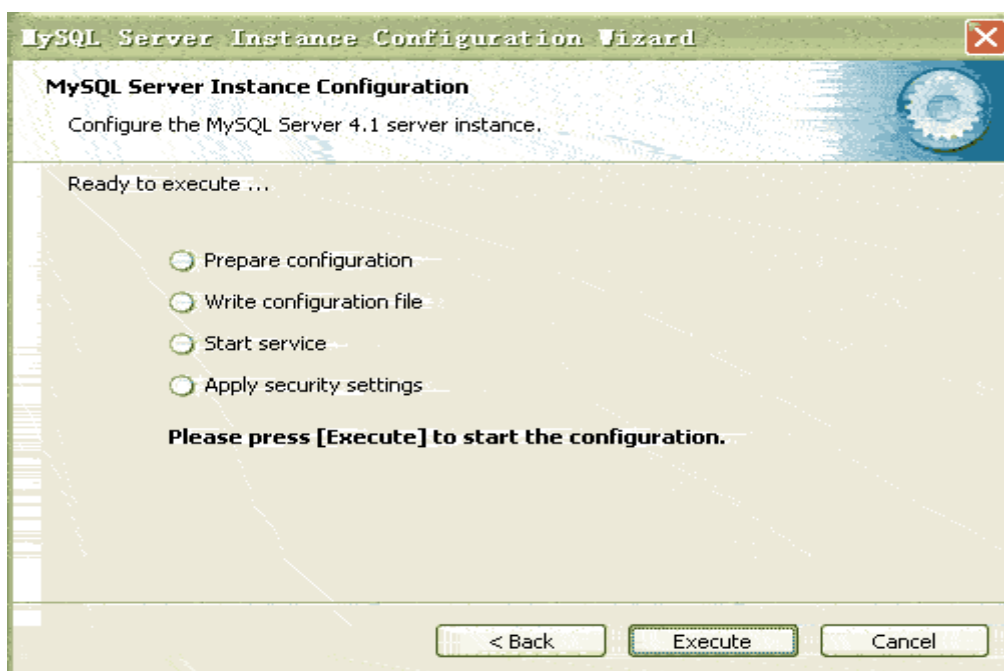


Figure 4.39 Interface of Configuration MySQL 11

Click 'Execute'.

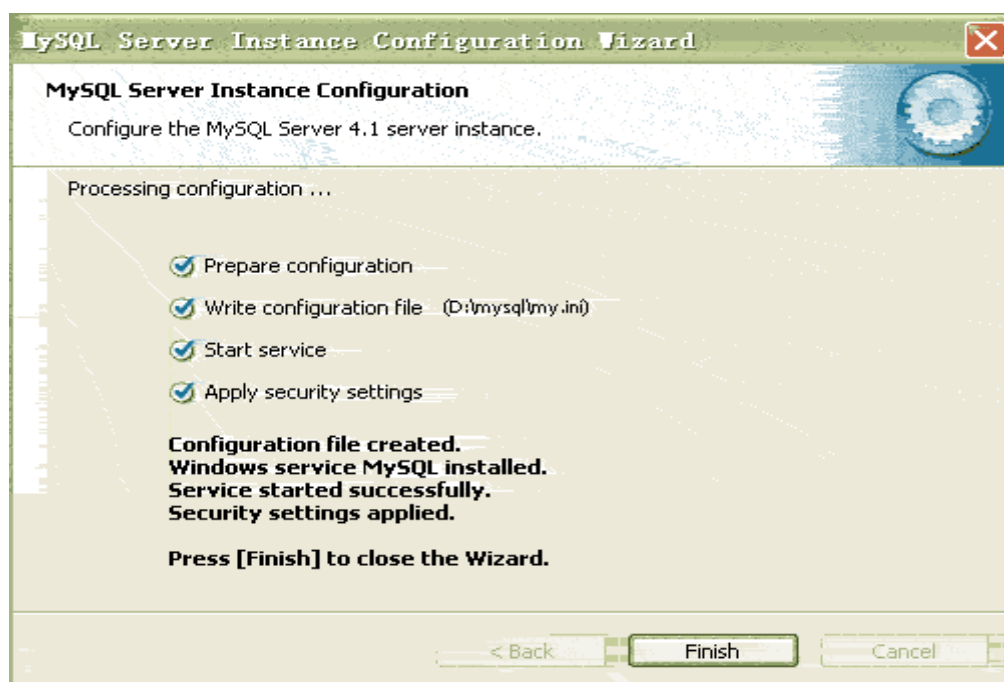


Figure 4.40 Interface of Configuration MySQL 12

Congratulations! We have completed all PHP environment (WINDOWS +PHP+Apache+MySQL).

4.2 Create User Interface

In order to implement a user authentication with MySQL and PHP, the first step I should create user interface for visitors. So I designed user interface based on HTML. It has two functions: one is to log in as a registered visitor, another is to new user to create a new name and password as a new user. The code is shown below:

3.php

```
<html>
```

```
<body>
```

```
<form action="XIAO.php" method="post">
```

```
NAME: <input type="text" name="NAME" />
```

```
PASSWORD: <input type="password" name="PASSWORD" />
```

```
<input type="submit" name="login" value="login" />
```

```
</form>
```

```
<form action="ANAN.php" method="post">
```

```
<input type="submit" name="submit" value="new user">
```

```
</body>
```

```
</html>
```

In this code, ‘<form action="XIAO.php" method="post">’ and ‘<form action="ANAN.php" method="post">’ are to call background operation. In the background operation, I add some commands to the code based on PHP. I will introduce the background operation in the next chapter. Now we look at the web page of the user interface. Open the Apache server . In the web browser, we fill in ‘http://localhost/anwei/interface/3.php’ (it is based on your own saved path.) and the following page appears:

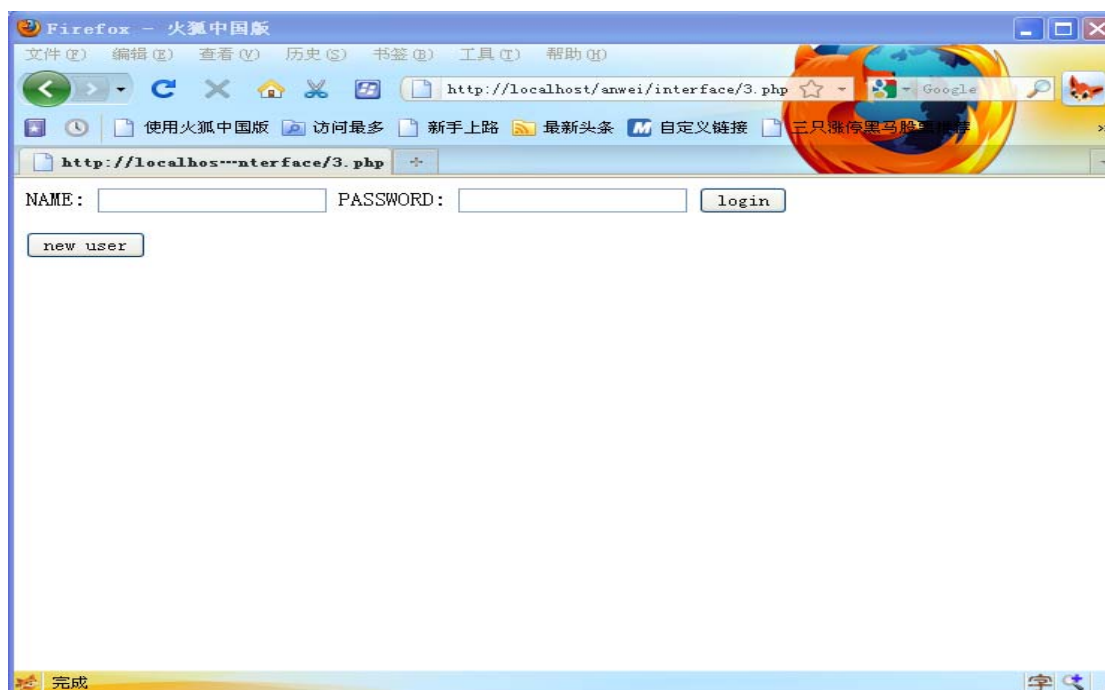


Figure 4.41 Interface of Log-in Web Page

We can see from this page that has two functions. If you are a registered visitor, you fill in your name and password, and click ‘login’.

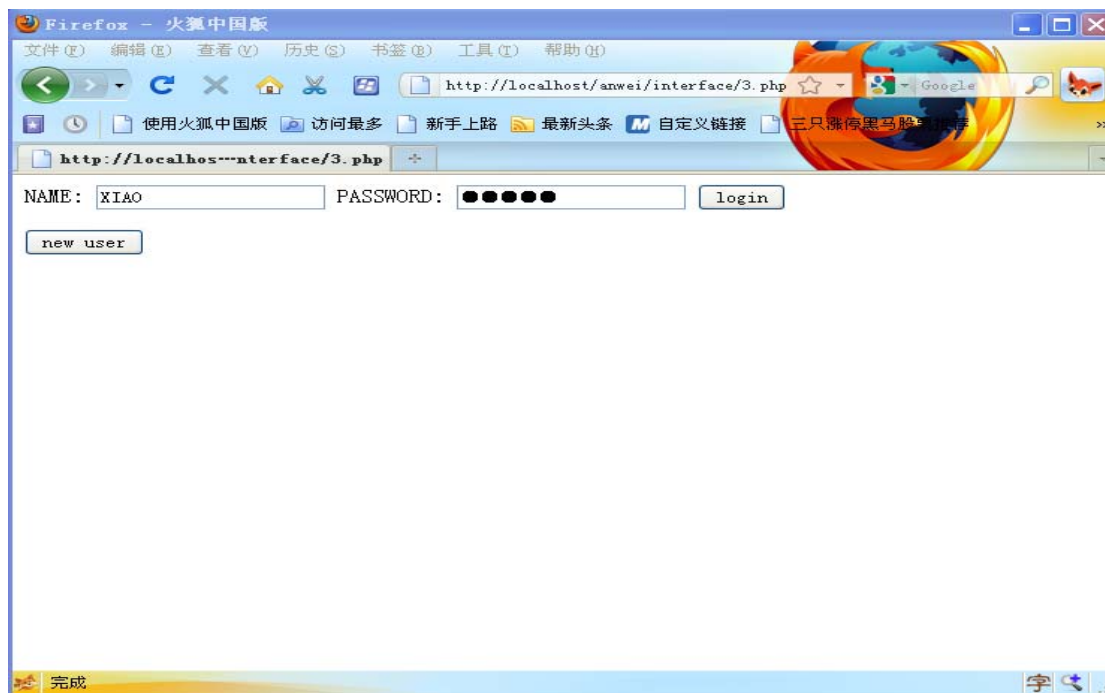


Figure 4.41 Input Values

If your input is correct, the page will look like this:

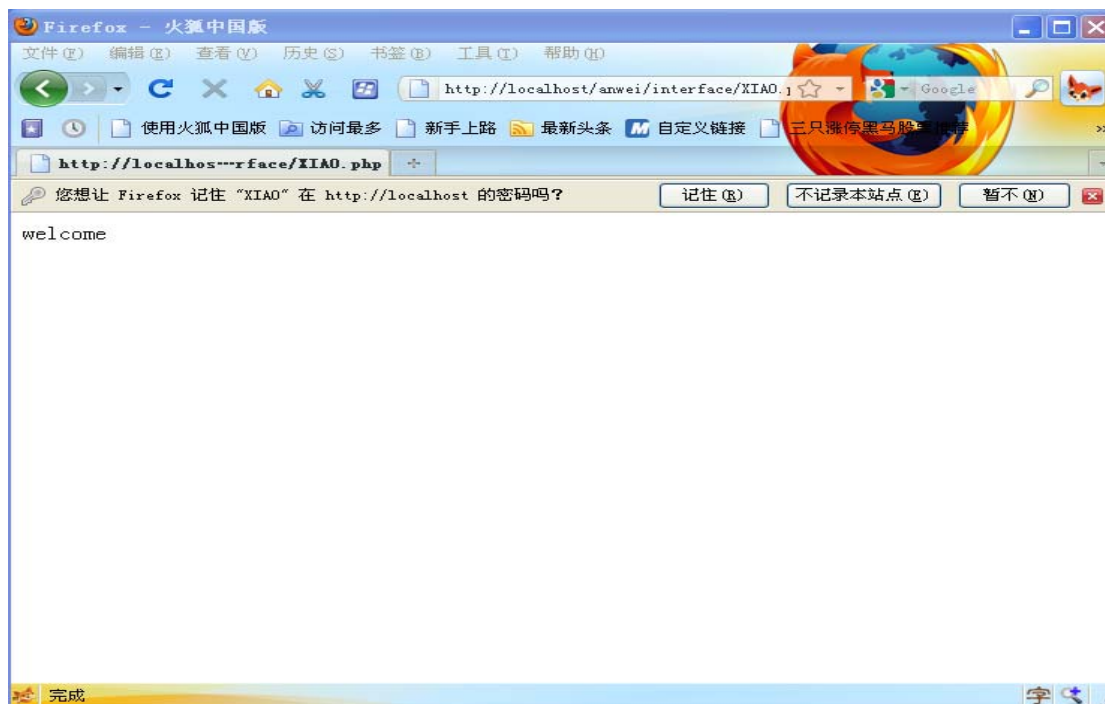


Figure 4.42 Correct Log-in

If your input is incorrect, the page will like this:

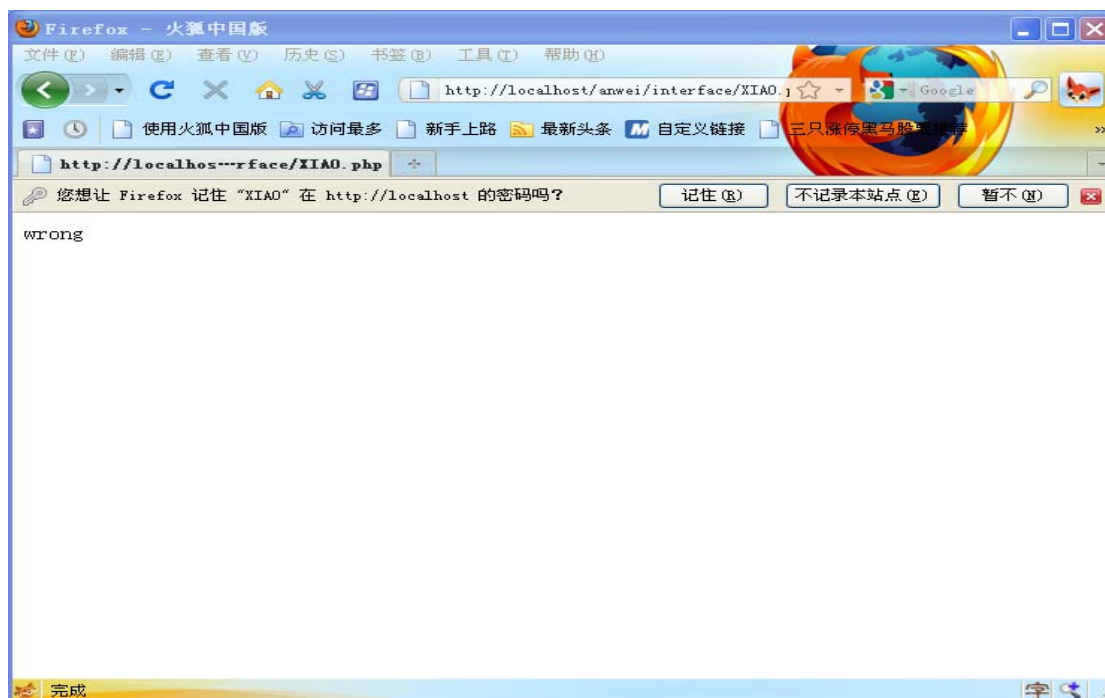


Figure 4.43 Incorrect Log-in

The login page has another function creating a new user. The function of creating new user will be introduced in Chapter 4.4 and 4.5. Next I will recommend the background operation of log-in to you in 4.3

4.3 The Background Operation of Log-in

Now I will introduce the background operation of log-in to you. Maybe you will say that ‘Your application is to implement secure transactions with PHP and MySQL but where is ‘MySQL’ and how do you implement secure transtions’. I add all this commands to background codes. Let is see the whole code:

```
<?php
```

```
$con = mysql_connect("localhost","root","19870824");

if (!$con)

{

    die('Could not connect: ' . mysql_error());

}

mysql_select_db("anwei", $con);

$str=$_POST[PASSWORD];

$str1= md5($str);

$result=mysql_query("SELECT * FROM anan

WHERE NAME='$_POST[NAME]' AND PASSWORD='$str1' ");

if($row = mysql_fetch_array($result))

{

    echo "welcome";

    echo "<br />";

}

else{

    echo "wrong";}
```

A web site has many visitors. So a web site needs MySQL database to save information. Here we use the command

`'$con=mysql_connect("localhost","root","19870824");'` to connect database. The 'root' is MySQL user's name and '19870824' is its password.

Here I will introduce 'md5(\$str)' in this code. This is my secure transaction for implementing authentication.

What is MD5?

Message Digest Algorithm MD5

In the present age, MD5 is commonly used in web sites. MD5 has been employed in many security applications and it is also commonly used to check the integrity of files. However, it has been shown that MD5 is not collision resistant. For example, MD5 is not suitable for applications like SSL certificates or digital signatures that depend on this property. A 32-digit hexadecimal number expressed an MD5 hash. The inventor is Ron Rivest, MD5 is to replace an earlier hash function.[10]

Algorithm of MD5

The MD5 algorithm is a 128-bit state, divided into four 32-bit words, denoted A, B, C and D. A, B, C and D are initialized to some fixed values. Then the algorithm operates on each 512-bit message block in turn, each block modifying the state. Four similar stages are included in the processing of a message block. Each round consists of 16 similar operations based on a non-linear function F, modular addition, and left rotation. Following figure illustrates one operation within a round. Four possible functions F are here. Each round uses a different one:

$$F(X, Y, Z) = (X \wedge Y) \vee (\neg X \wedge Z)$$

$$G(X, Y, Z) = (X \wedge Z) \vee (Y \wedge \neg Z)$$

$$H(X, Y, Z) = X \oplus Y \oplus Z$$

$$I(X, Y, Z) = Y \oplus (X \vee \neg Z)$$

$\oplus, \wedge, \vee, \neg$ denote the XOR, AND, OR and NOT operations respectively.

Figure 4.44 Four Possible Functions F [10]

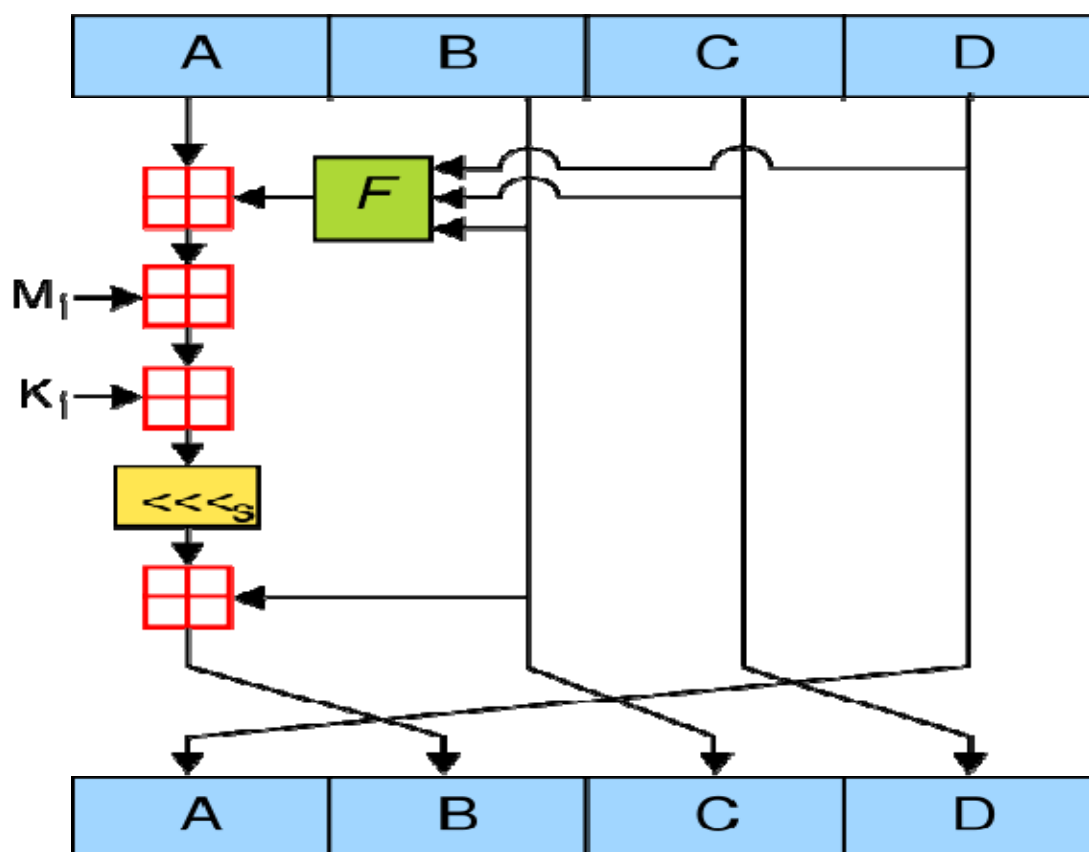


Figure 4.45 One MD5 Operation.[10]

MD5 includes 64 of these operations, grouped in four rounds of 16 operations. One function is used in each round. F is a nonlinear function; K_2 denotes a 32-bit constant, and M_2 denotes a 32-bit block of the message input, different for each operation.

Now I will give you some examples of encryption by MD5. If your password is 12345. The information will be stored in MySQL as 827ccb0eea8a706c4c34a16891 f84e7b.

| | |
|----------|----------------------------------|
| anan | anwei |
| anan6731 | 781d3b53c8b5028dc6896bf8818b0884 |
| hanna | 014dbbd9fbb4740ecaac3750e9e6d584 |
| yanliang | b0baee9d279d34fa1dfd71aadb908c3f |
| yan | 827ccb0eea8a706c4c34a16891f84e7b |
| 2b | b0f8b3e58f093359fe1af416b5ea8ed6 |
| wushen | 827ccb0eea8a706c4c34a16891f84e7b |
| timo | 827ccb0eea8a706c4c34a16891f84e7b |
| aaaa | 827ccb0eea8a706c4c34a16891f84e7b |

Figure 4.46 The Data Stored In MySQL Database

Then the code ‘`$result=mysql_query("SELECT * FROM anan WHERE NAME ='_POST[NAME]' AND PASSWORD='$str1' ");`’ to compare password with the data stored in MySQL database. And use ‘`if($row = mysql_fetch_array($result))`’ to judge. If the judgment is founding , it means your log-in is successful.

4.4 The Interface of Creating New User

In chapter 4.1, I introduced the interface of log-in. However there is no name and password stored in the background database. So we must add a function creating a new user. Now I will show the interface of creating a new user. The code of creating a new user is based on HTML.

```
<html>
```

```
<body>
```

```
<form action="insert.php" method="post">
```

```
NAME: <input type="text" name="NAME" />
```

```
PASSWORD: <input type="password" name="PASSWORD" />
```

```
<input type="submit" name="INSERT" value="INSERT" />
```

```
</form>
```

```
</body>
```

```
</html>
```

The button of 'new user' is on the same page with 'log in'. the picture will like this:

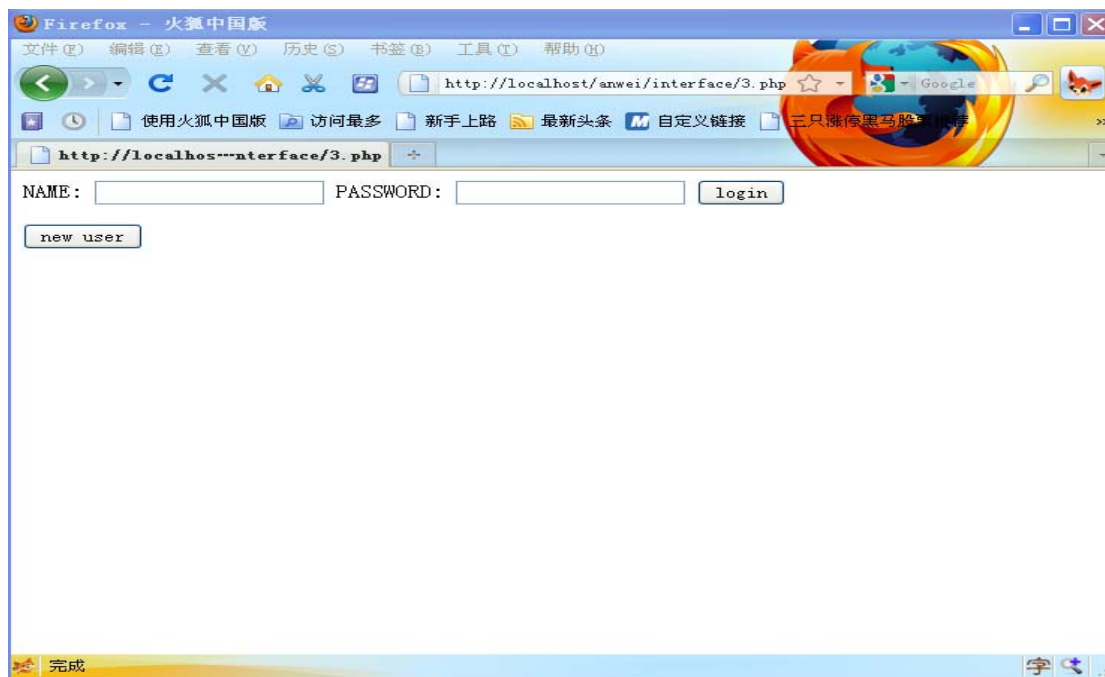


Figure 4.47 Interface of Log-in Web Page

When you click 'new user', you will get this page:

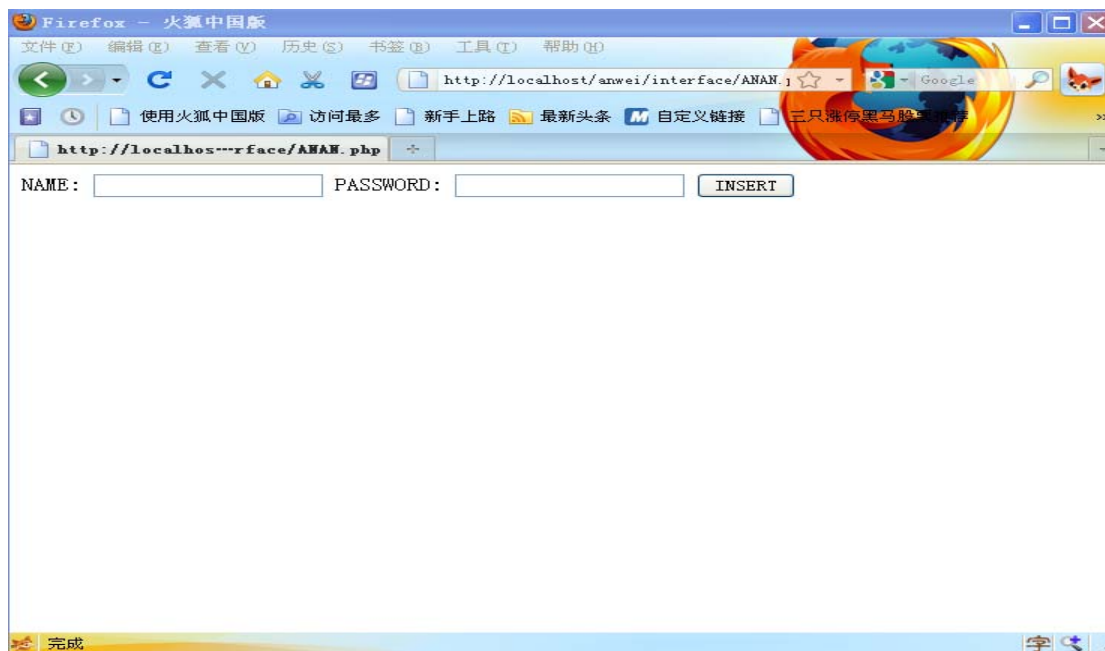


Figure 4.48 Interface of INSERT Web Page

This is to guide you to create a new name and new password. Then you can click 'INSERT'.

If your creation is successful, the page will be like this:

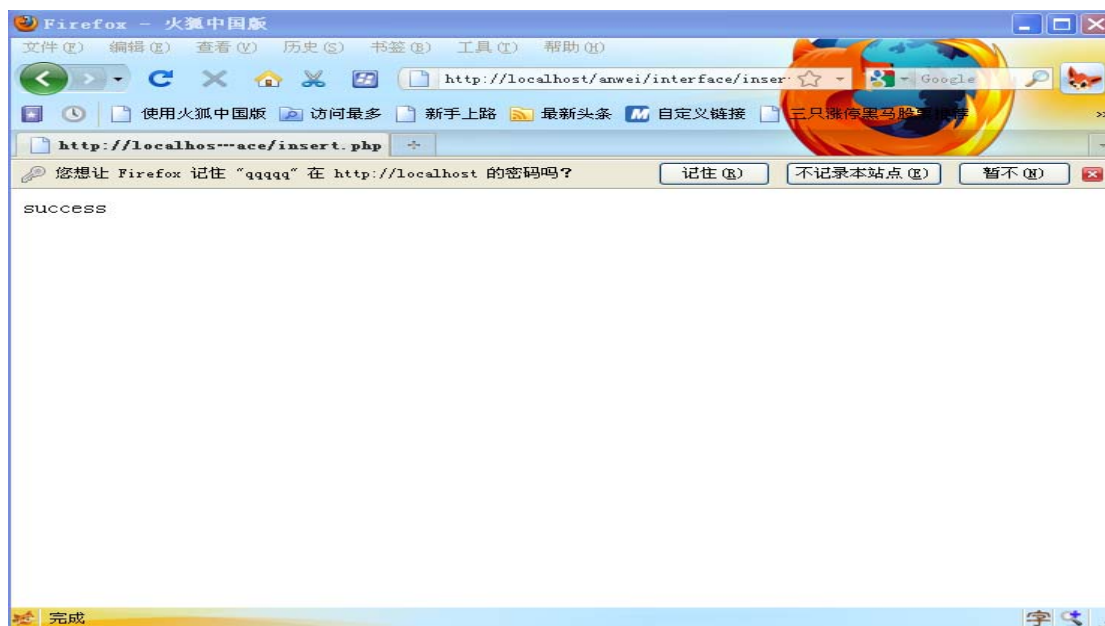


Figure 4.49 Interface of Insert Correctly

4.5 The Background Operation of Creating New User

Some commands to create a new user are similar to log-in operations. The background operation of creating a new user also uses the command to connect MySQL and MD5 introduced in 4.2. So I will show how to insert information to MySQL database here. Firstly, show all the codes based on PHP in the background operation of creating a new user.

```
<?php

$con = mysql_connect("localhost","root","19870824");

if (!$con)

    {

        die('Could not connect: ' . mysql_error());

    }

mysql_select_db("anwei", $con);

$str="$_POST[PASSWORD]";

$str1= md5($str);

$sql="INSERT INTO anan (NAME, PASSWORD)

VALUES

('$_POST[NAME]','$str1)";

if (!mysql_query($sql,$con))

    {
```

```
die('Error: ' . mysql_error());  
  
}  
  
echo "success";  
  
mysql_close($con)  
  
?>
```

'\$sql="INSERT INTO anan (NAME, PASSWORD)' is to insert the information to MySQL databased. Think about the secure transactions, the password will be encoded by MD5() before stored in MySQL database.

For example, we input a new name Peter and create a new secret password.



Figure 4.50 Input Values In INSERT Web Page

After we click 'INSERT', you can the data stored in MySQL database will be like this:

| | |
|-------|----------------------------------|
| Peter | 5ebe2294ecd0e0f08eab7690d2a6ee69 |
|-------|----------------------------------|

Figure 4.51 Peter' Password Stored In MySQL Database

Creating a new user is completed and you can use your own name and password to log in the site.

5 CONCLUSION

5.1 Challenge of My Thesis Research and Solution

The aim of my study was to implement the authentication with PHP and MySQL. The more practical aim of the thesis was to set up more secure database avoiding attacks from hackers. There is a challenge for my application. Because of the wide use of MD5, some softwares try to break MD5. As a result, the use of MD5 is not safe. Luckily, I finally resolved the problem.

I have introduced the algorithm of MD5. Now I will show the process of hashing data. MD5 has the initial vector IV, this is the key to resolve this problem. The algorithm of MD5 is stored by A, B, C, D four buffer registers while one register is 32 bits. The initial values are:

a = 0x67452301

b = 0x EFCDAB89

c = 0x 98BADCFE

d = 0x 10325476

We can see four sexadecimal values regarded as an initial seed. From the algorithm of MD5, we know the main processes are xor or mode operation, to process 512 bits in every group, to march 4*16 operation. So we can just change the initial values to solve the problem. A little change can build another algorithm of MD5. For example, we can alter the initial value (a=0x6745230) to 0x67452300. Then you can create your own MD5 through changing the initial.

5.2 The Future of MD5

Actually, any algorithm has its leak, although MD5 is widely used in the world. In the future, the MD5 will be replaced, who will the next one?

For a long time, password bounded always hammer at new encryption algorithm' research. Besides the adoptive encryption algorithm is not MD5 on the highly confidential area. Various governments and big companies all investigate independent technical encryption algorithm. The excellent representative are SHA-1 and SHA-224 and so on. After the MD5 decoding report was published, NIST expressed that they will spread a safer encryption algorithm, SHA-224、SHA-256、SHA- 384 and SHA-512. Compared to MD5 128 bits encryption, those algorithms seem to improve a lot on encryption digit and safety performance. Although MD5 must be replaced, it will still be used for a long time because of its open source and free features. Besides there is no real effective rapid breaking method.

References:

- [1] <URL http://baike.baidu.com/view/99.htm?fr=ala0_1_1>
- [2] PHP and MySQL Web Development Third Edition, by Luke Welling and Laura Thomson
- [3] <URL <http://zhidao.baidu.com/question/34780415.html>>
- [4] <URL http://baike.baidu.com/view/24816.htm?fr=ala0_1_1>
- [5]<URL
[http://image.baidu.com/i?ct=503316480&z=0&tn=baiduimagedetail&word=mysql
&in=23251&cl=2&cm=1&sc=0&lm=-1&pn=1&rn=1&di=1282818555&ln=2000&fr
=&ic=0&s=0&se=1](http://image.baidu.com/i?ct=503316480&z=0&tn=baiduimagedetail&word=mysql&in=23251&cl=2&cm=1&sc=0&lm=-1&pn=1&rn=1&di=1282818555&ln=2000&fr=&ic=0&s=0&se=1)>
- [6] <URL http://baike.baidu.com/view/28283.htm?fr=ala0_1>
- [7] <URL <http://linux.chinaitlab.com/special/apache/Index.html>>
- [8] <URL http://en.wikipedia.org/wiki/GNU_Privacy_Guard>
- [9] <URL <http://www.gnupg.org/>>
- [10] <URL <http://en.wikipedia.org/wiki/MD5>>

Appendix A-1

3.php

```
<html>
```

```
<body>
```

```
<form action="XIAO.php" method="post">
```

```
NAME: <input type="text" name="NAME" />
```

```
PASSWORD: <input type="password" name="PASSWORD" />
```

```
<input type="submit" name="login" value="login" />
```

```
</form>
```

```
<form action="ANAN.php" method="post">
```

```
<input type="submit" name="submit" value="new user">
```

```
</body>
```

```
</html>
```

XIAO.php

```
<?php
```

```
$con = mysql_connect("localhost","root","19870824");
```

```
if (!$con)
```

```
{  
  
    die('Could not connect: ' . mysql_error());  
  
}  
  
mysql_select_db("anwei", $con);  
  
$str=$_POST[PASSWORD];  
  
$str1= md5($str);  
  
$result=mysql_query("SELECT * FROM anan  
  
WHERE NAME='$_POST[NAME]' AND PASSWORD='$str1' ");  
  
if($row = mysql_fetch_array($result))  
  
    {  
  
        echo "welcome";  
  
        echo "<br />";  
  
    }  
  
else{  
  
    echo "wrong";}
```

ANAN.php

<html>

<body>


```
<form action="insert.php" method="post">  
  
NAME: <input type="text" name="NAME" />  
  
PASSWORD: <input type="password" name="PASSWORD" />  
  
<input type="submit" name="INSERT" value="INSERT" />  
  
</form>  
  
</body>  
  
</html>
```

Insert.php

```
<?php  
  
$con = mysql_connect("localhost","root","19870824");  
  
if (!$con)  
  
    {  
  
        die('Could not connect: ' . mysql_error());  
  
    }  
  
mysql_select_db("anwei", $con);  
  
$str=$_POST[PASSWORD];  
  
$str1= md5($str);  
  
$sql="INSERT INTO anan (NAME, PASSWORD)
```

VALUES

```
($_POST[NAME]','$str1');
```

```
if (!mysql_query($sql,$con))
```

```
{
```

```
    die('Error: ' . mysql_error());
```

```
}
```

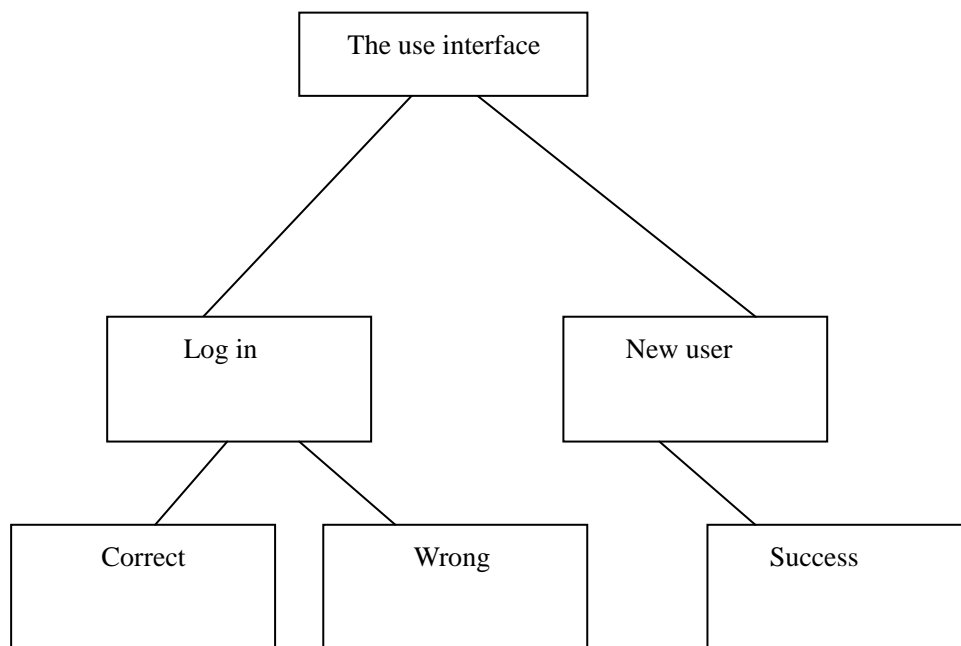
```
echo "success";
```

```
mysql_close($con)
```

```
?>
```

Appendix A-2

Flow Chart of My Application:



Appendix A-3

LIST OF FIGURES

Figure 2.1 The logo of PHP

Figure 2.2 The logo of MySQL

Figure 2.3 Interface of MySQL Query Browser

Figure 2.4 The logo of Apache

Figure 2.5 Interface of Apache

Figure 3.1 The process of PGP Encryption

Figure 3.2 Interface of GPG

Figure 3.3 The background operation of GPG

Figure 4.1 Interface of Installation Wizard 1

Figure 4.2 Interface of Installation Wizard 2

Figure 4.3 Interface of Installation Wizard 3

Figure 4.4 Interface of Installation Wizard 4

Figure 4.5 Interface of Installation Wizard 5

Figure 4.6 Interface of Installation Wizard 6

Figure 4.7 Interface of Installation Wizard 7

Figure 4.8 Interface of Installation Wizard 8

Figure 4.9 Shortcutkey of Apache

Figure 4.10 Test Apache web server

Figure 4.11 The file of httpd

Figure 4.12 Interface of uncompress

Figure 4.13 Configure the uncompress path

Figure 4.14 The file of php.ini 1

Figure 4.15 The file of php.ini 2

Figure 4.16 The file of php.ini 3

Figure 4.17 The file of php.ini 4

Figure 4.18 The file of php.ini 5

- Figure 4.19 The file of apache httpd
- Figure 4.20 Interface of MySQL setup wizard 1
- Figure 4.21 Interface of MySQL setup wizard 2
- Figure 4.22 Interface of MySQL setup wizard 3
- Figure 4.23 Interface of MySQL setup wizard 4
- Figure 4.24 Interface of MySQL setup wizard 5
- Figure 4.25 Interface of MySQL setup wizard 6
- Figure 4.26 Interface of MySQL setup wizard 7
- Figure 4.27 Interface of MySQL setup wizard 8
- Figure 4.28 Interface of MySQL setup wizard 9
- Figure 4.29 Interface of configuration MySQL 1
- Figure 4.30 Interface of configuration MySQL 2
- Figure 4.31 Interface of configuration MySQL 3
- Figure 4.32 Interface of configuration MySQL 4
- Figure 4.33 Interface of configuration MySQL 5
- Figure 4.34 Interface of configuration MySQL 6
- Figure 4.35 Interface of configuration MySQL 7
- Figure 4.36 Interface of configuration MySQL 8
- Figure 4.37 Interface of configuration MySQL 9
- Figure 4.38 Interface of configuration MySQL 10
- Figure 4.39 Interface of configuration MySQL 11
- Figure 4.40 Interface of configuration MySQL 12
- Figure 4.41 Input values
- Figure 4.42 Correct log-in
- Figure 4.43 Incorrect log-in
- Figure 4.44 Four possible functions F
- Figure 4.45 One MD5 operation.
- Figure 4.46 The data stored in MySQL database
- Figure 4.47 Interface of log-in web page
- Figure 4.48 Interface of INSERT web page

Figure 4.49 Interface of insert correctly

Figure 4.50 Input values in INSERT web page

Figure 4.51 Peter' password stored in MySQL database