

Forum:[Hack Forums](#)

Thread: [\[help\] Need a guide on how to make a secure admin section](#)

Webman.io - June 7, 2012, 2:05 pm

Need a guide on how to make a secure admin section with login and logout function.

And then also with some admin secure pages which only the admin can see when logged in.

Anyone know such?

Novalty - June 7, 2012, 2:16 pm

I could code you something like this if you wanted. Shouldn't be too hard!

Webman.io - June 7, 2012, 2:34 pm

(Jun 7, 2012 02:16 PM)novalty Wrote: [\[CITE\]](#) I could code you something like this if you wanted. Shouldn't be too hard!

But I am not pretty good at this yet.

I have worked with SQL and PHP before, but I have no clue on making login.

As I don't know what it would require.

Novalty - June 7, 2012, 2:54 pm

Here is a very basic shell for the code you want. Fill it up with as much stuff as you want but this is the very basics of a MySQL login system.

I have not protected it against any kinds of attacks but if you search around for a minute or two you can find very good resources for protecting your scripts.

I would also recommend recoding my example into OOP, and using PDO for the database work.

login.php

PHP Code

```
<?php
```

```
$errors = array();
```

```
include 'functions.php';
```

```
if(isset($_POST['submit'])) {  
    if(userExists($_POST['user'])) {  
        if(authenticate($_POST['user'], $_POST['pass'])) {  
            login();  
            header('Location: private.php');  
            die();  
        } else {  
            $errors[] = 'You have entered an invalid username or password.';  
        }  
    } else {  
        $errors[] = 'That username does not exist.';
```

```

    }
}

?>
<h1>Login</h1>
<ul>
<?php
foreach($errors as $error) {
    echo "<li>{$error}</li>";
}
?>
</ul>
<form action="" method="post">
    <input type="text" name="user" />
    <input type="password" name="pass" />
    <input type="submit" name="submit" value="Login" />
</form>

```

functions.php

PHP Code

```
<?php
```

```

function authenticate($user, $pass) {
    $salt = getSalt($user);
    $pass = hashPass($pass, $salt);
    $query = mysql_query("SELECT COUNT(id) AS rows FROM users WHERE username = '{
$username}' AND password = '{$pass}'");
    $row = mysql_fetch_assoc($query);
    return ($row['rows'] > 0);
}

```

```

function userExists($user) {
    $query = mysql_query("SELECT COUNT(id) AS rows FROM users WHERE username = '{
$username}'");
    $row = mysql_fetch_assoc($query);
    return ($row['rows'] > 0);
}

```

```

function getSalt($user) {
    $query = mysql_query("SELECT salt FROM users WHERE username = '{$username}'");
    $row = mysql_fetch_assoc($query);
    return $row['salt'];
}

```

```

function hashPass($pass, $salt) {
    return hash('sha512', $pass . $salt);
}

```

```

function login() {
    $_SESSION['loggedin'] = true;
}

```

```

function logout() {

```

```

$_SESSION['loggedin'] = false;
}

function loggedIn() {
    return (!empty($_SESSION['loggedin']) && $_SESSION['loggedin'] === true);
}

```

?>
private.php
PHP Code
<?php

```

if(!loggedIn()) {
    header('Location: login.php');
    die();
}

```

?>

private stuff

database structure
Code name: users

id - username - password

Webman.io - June 7, 2012, 10:36 pm

(Jun 7, 2012 02:54 PM)novalty Wrote: [CITE](#) Here is a very basic shell for the code you want. Fill it up with as much stuff as you want but this is the very basics of a MySQL login system.

I have not protected it against any kinds of attacks but if you search around for a minute or two you can find very good resources for protecting your scripts.

I would also recommend recoding my example into OOP, and using PDO for the database work.

login.php
PHP Code
<?php

```

$errors = array();

```

```

include 'functions.php';

```

```

if(isset($_POST['submit'])) {
    if(userExists($_POST['user'])) {
        if(authenticate($_POST['user'], $_POST['pass'])) {
            login();
            header('Location: private.php');
            die();
        } else {
            $errors[] = 'You have entered an invalid username or password.';
        }
    } else {
        $errors[] = 'That username does not exist.';
    }
}

```

```

    }
}

?>
<h1>Login</h1>
<ul>
<?php
foreach($errors as $error) {
    echo "<li>{$error}</li>";
}
?>
</ul>
<form action="" method="post">
    <input type="text" name="user" />
    <input type="password" name="pass" />
    <input type="submit" name="submit" value="Login" />
</form>

```

functions.php

PHP Code

```
<?php
```

```

function authenticate($user, $pass) {
    $salt = getSalt($user);
    $pass = hashPass($pass, $salt);
    $query = mysql_query("SELECT COUNT(id) AS rows FROM users WHERE username = '{
$username}' AND password = '{$pass}'");
    $row = mysql_fetch_assoc($query);
    return ($row['rows'] > 0);
}

```

```

function userExists($user) {
    $query = mysql_query("SELECT COUNT(id) AS rows FROM users WHERE username = '{
$username}'");
    $row = mysql_fetch_assoc($query);
    return ($row['rows'] > 0);
}

```

```

function getSalt($user) {
    $query = mysql_query("SELECT salt FROM users WHERE username = '{$username}'");
    $row = mysql_fetch_assoc($query);
    return $row['salt'];
}

```

```

function hashPass($pass, $salt) {
    return hash('sha512', $pass . $salt);
}

```

```

function login() {
    $_SESSION['loggedin'] = true;
}

```

```

function logout() {

```

```
$_SESSION['loggedin'] = false;
}

function loggedIn() {
    return (!empty($_SESSION['loggedin']) && $_SESSION['loggedin'] === true);
}
```

```
?>
private.php
PHP Code
<?php
```

```
if(!loggedIn()) {
    header('Location: login.php');
    die();
}
```

```
?>
```

private stuff

database structure
Code name: users

id - username - password
thanks a lot for this. But it looks like it also needs a salt field in the database?
But what should I put in there to make it work?

Novalty - June 8, 2012, 2:25 am

Oh haha! My bad :P

To get yourself as a working user, run a password and a salt through the hashPass() function.

Pass - your desired password
Salt - Random string of characters. Use something like the following: sfr620aw2

Echo out the returned value from that function. Copy the value/hash that you see on your screen and put it into your database under 'password'.
Then copy the salt you used and put that into the 'salt' column in the database

Dankki - June 8, 2012, 3:35 am

You should `mysql_real_escape_string` the values before trying to check them from the db.

Novalty - June 8, 2012, 5:05 am

(Jun 8, 2012 03:35 AM)Dankki Wrote: [\[CITE\]](#) You should `mysql_real_escape_string` the values before trying to check them from the db.

I had stated I did not provide any protection

novalty Wrote:I have not protected it against any kinds of attacks but if you search around for a minute or two you can find very good resources for protecting your scripts.

Bassline - June 8, 2012, 5:56 am

Perhaps:

```

Code <?php
include("lib.php");
$player = check_user($secret_key, $db);
if($player->rank != "Administrador")
{
header("Location: personagem.php");
exit;
}
include("templates/private_header.php");
head("Admin Panel","adminpanel.png");
startbox();
if(isset($_GET['page']))
{
include("admin/" . $_GET['page'] . ".php");
}
else
{
?>
<A href="admin.php?page=code1">Kill Players</a><br>
<A href="admin.php?page=code2">Heal Players</a><br>

```

```

<?php
}
endbox();
include("templates/footer.php");
?>

```

i want to know if im prone to hacking. the code1.php and code2.php uses a custom query library that is included in lib.php so there is no way to execute them directly without falling in to an error.

Also in My template i have:

```

if($player->rank == "Administrador")
{
echo "<a href='admin.php'>Admin Panel</a>";
}

```

Taken from:

<http://stackoverflow.com/questions/6650032/is-this-a-secure-php-administration-page>

systemcarnage - June 8, 2012, 6:38 am

you could encrypt the data that is being transferred from client to server or other way around..