Sikre dit login system bedre.

Denne lille guide vil hjælpe dig med at sikre dit php login system bedre.

Skrevet den **12. Feb 2009** af **rasmus-madsen** I kategorien **Programmering / PHP** | ★☆☆☆☆

Hejza der ude.
Jeg er rimelig ny til php, men har søgt en del om sikkerhed. Denne guide vil vise den mest sikre bedste måde jeg er kommet frem til.

Det første man skal forhindre, er den mest kendte måde. Faktisk også den nemmeste.

```
"user=''or''='"
```

Hvis man indsætter det i formen som brugernavn og det samme i password, vil man kunne komme ind.
Eks:
Har man denne form kode:

```
<form method="post" action="login.php">
    <input type="Text" value="" name="user">
    <input type="password" value="" name="pass">
    <input type="submit" value="Login">
  </p>
</form>
[div]
Isætter man "user=''or''='" i i brugernavn og "pass=''or''='" i password feltet.
Dette kan man forhindre ved at bruge denne kode:
[div]
$user = $_POST["user"];
$user = mysql_escape_string($user);
```

Så virker den ikke mere.

Derefter skal du md5-kryptere dit password.
Man at kryptere en streng på denne måde:

```
$password = md5($streng);
```

Det kan godt være det bliver sværere og gætte passwordet nu, men nogen hackere gætter det ikke. De kontrollere hvad der sendes og modtages til din computer. Derfor skal der gøres to ting for at sikre de ikke kan det. Vi vil i stedet for at md5-kryptere passwordet når dataen er sendt til siden, gøre dette inden dataen sendes.:

Form siden

```
<form method="post" action="login.php" onSubmit="this.user.value = hex_md5(this.pass.value);
return true;">
    <input type="Text" value="" name="user">
    <input type="password" value="" name="pass">
    <input type="submit" value="Login">
```

```
  </p>
</form>
```

Filen md5.js kan hentes på www.rasmus-madsen.dk/md5.js

Hackeren kan stadig væk finde ud af hvad der sendes og modtages. Derefter vil han "lave" egen login boks på sin egen computer. Eks

```
<form method="post" action="www.domine.dk\login.php"
   <input type="Text" value="" name="user">
   <input type="password" value="" name="pass">
   <input type="submit" value="Login">
  </p>
</form>
```

Så undgår hackeren hermed at kryptere det password som er krypteret i forvejen. Derfor skal vi nu kontrollere hvor dataen kommer fra. Data skal komme fra vores egen login-form (serveren):

```
if ($test == "http://www.domine.dk/login_form.php") {
//login
}else{
echo "Du ender data fra et andet sted end tilladt";
}
```

Hvis Hackere nu finder det password som sendes fra din computer, vil han ikke kunne bruge det til noget. Hvis han taster det ind i vores login-form hvor vil passwordet blive krypteret to gange, og derfor ikke være rigtigt.

Håber det hjalp nogen !

Skrevet af Rasmus Madsen.

Kommentar af limemedia d. 25. Jan 2004 | 1

Artiklen giver et indblik i nogle af de problem stillinger der er mht login formularer og muligheder for at "snuppe bagvejen" - tager lidt for givet at et kodeord altid bliver kompromitteret ved aflæsning over nettet.

Kommentar af minau d. 29. Apr 2004 | 2

Kommentar af radion d. 13. Feb 2006 | 3

Hvor definerer du hvad $test indeholder?

if ($test == "http://www.domine.dk/login_form.php") {

Kommentar af htm d. 23. Jul 2004 | 4

Kommentar af jamola d. 18. Jan 2005 | 5

Kommentar af mxs d. 29. Feb 2004 | 6

:) Husk nu at den rigtige hacker også spiller skuespil eller laver en seroøs forum eller andet hvor andre melder sig som bruger med de password og brugernavn som de også bruger andre steder på nettet. Med det kan en hacker NEMT komme ind!! Ellers sys jeg det er en god artikel som giver en bruger et indblik i hvor og hvad man skal kigge efter!!

Kommentar af googolplex d. 28. Jan 2004 | 7

Meget overfladisk, men et par enkelte fif er da ok.

Kommentar af net-base.dk d. 26. Feb 2004 | 8

Syntes ikke denne vejledning er særlig i dybte gående. kunne godt have været bedere

Kommentar af squashguy d. 25. Jan 2004 | 9

Fint nok at påpege det med sql-injektion, og at lave password til en hash inden form submittes. Det hjælper dog ikke at kontrollere HTTP_REFERER, da denne sagtens kan forfalskes. Så sniffer hackeren passwordet, kommer han nok ind alligevel..

Kommentar af stevenizzle d. 20. Oct 2005 | 10

sleet ikke mine 5 point værd