Datasikkerhed i PHP

En kort gennemgang af hvad der sker, når PHP modtager data, hvorfor stripslashes og addslashes er nødvendige og hvordan man bruger dem bedst.

Egentlig datavalidering bliver ikke behandlet i denne artikel.

Skrevet den **12. Feb 2009** af **sandbox** I kategorien **Programmering / PHP** | ⭐⭐☆☆☆

Når du modtager data, sker følgende:

- PHP får oplysninger om felternes indhold og cookie-indhold fra webserveren.
- PHP kører måske addslashes() på brugerdata. [Se note 1]
- PHP putter data i variabler, der kan bruges af dit script.
** $_GET får data fra adressen (og formularer med method='get')
** $_POST får data fra formularer med method='post'
** $_COOKIE får data fra cookies.
** $_REQUEST får data fra de tre ovennævnte. [Se note 2]
** Måske bliver data også puttet i variable som f.eks. $navn [Se note 2 og 3]
- Dit script starter.

Forestil dig følgende login-situation:
  * Formularen havde to felter: navn og password
  * Der er IKKE kørt addslashes(), eller du HAR kørt stripslashes()
  * SELECT brugerid FROM tabel WHERE navn='$navn' and password='$password'
  * $navn="admin" og password="glup' OR '' = '" giver følgende:
    SELECT brugerid FROM tabel WHERE navn='admin' AND password='glup' OR '' = ''

Hvis du har en bruger, der hedder admin, er personen nu logget ind.

\ betyder bare: Næste tegn skal ikke opfattes normalt, men bare indsættes som det er. Den bruges også i PHP, hvis du vil bruge en " inde i en streng, hvor der er " uden om. Hvis du havde slashes i variablerne, ville MySQL have ignoreret \' så den ville have forsøgt at finde en bruger med passwordet glup' OR '' = ', da din query så havde set således ud:

SELECT brugerid FROM tabel WHERE navn='admin' AND password='glup\' OR \'\' = \''

Når du henter data ud fra tabellen igen, kan det også være, at der bliver kørt en automatisk addslashes på dem[Se note 4]. Den skal du kun fjerne med stripslashes, hvis du ikke vil putte indholdet i database igen. Jeg vil foreslå at du kun fjerner dem, lige når du udskriver data - dog før du eventuelt manipulerer output med striptags(), nl2br() eller lignende.

Her er to funktioner, som du kan bruge i stedet for addslashes() og stripslashes():

```
function my_addslashes($streng){
  return get_magic_quotes_gpc()?$streng:addslashes($streng);
}

function my_stripslashes($streng){
```

```
  return get_magic_quotes_runtime()?stripslashes($streng):$streng;
}
```

Fordelen ved at bruge disse funktioner istedet for bare at kalde addslashes eller stripslashes normalt er, at så virker dit script stadig hvis du skifter til et andet hotel med andre indstillinger, eller hvis dit hotel pludselig laver om på indstillingerne.

Noter:

1)
Dette kan ændres i php.ini. I dit script, kan du bruge get_magic_quotes_gpc() for at finde ud af om det er slået til, så du kan lave scriptet, så det virker både på servere med og uden automatisk addslashes().

2)
Hvis det skulle ske, at variable fra flere steder har samme navn, er det enten variables_order (i nyere udgave) eller gpc_order (i ældre udgaver), der bestemmer hvilke der kommer igennem.

3)
Hvis register_globals er slået til. Dette anbefales ikke, og den har som standard været slået fra i nyere PHP-versioner.

4) Tilsvarende til note 1, findes der funktionen get_magic_quotes_runtime() som kan fortælle dig om dette er aktiveret.

Hvis du har forslag til rettelser i artiklen, må du meget gerne skrive til mig. Jeg er brugeren ole på serveren sandbox.adsl.dk

Kommentar af erikjacobsen d. 25. Feb 2004 | 1

Eller brug en pakke fra PEAR:
http://pear.php.net/package/HTML_QuickForm

Kommentar af bromer d. 04. Mar 2004 | 2

Kommentar af hermandsen d. 12. Mar 2004 | 3

Ganske fin lille artikel, desværre ikke vildt dybtegående, men stadig god!! :)

Kommentar af xyborx d. 16. Apr 2004 | 4

Lille simpel artikel, der dog har en meget vigtig pointe. Til mysql queries er det måske også værd at kigge på funktionen mysql_escape_string()

Kommentar af stevenizzle d. 20. Oct 2005 | 5

fin artikel :D