##
#Title: Default CSRF $_POST Token Protection###
#Author: pH4nToM################################
#Translate: cobra90nj cobra90nj@gmail.com######
##

Che cos' è ?

Bene, prima di entrare nei dettagli su come proteggersi
dagli attacchi di tipo CSRF, permettetemi di spiegare
prima di cosa si tratta esattamente.
Cross Site Request Forgery (noto anche con il nome di
XSRF, CSRF o Cross Site Reference Forgery) ha il compito
di fare eseguire del codice alla vittima dal suo browser.
Il suo compito è di solito legato all' URL
(http://site/stocks?buy=100&stock=ebay) che consentono
di eseguire azioni specifiche quando devono essere eseguite.
Se un utente è connesso ad un sito ed un attacante inganna
il browser della vittima, effettua una richiesta di
questo tipo, (esempio di URL di prima) l' azione allora viene
compiuta, eseguita e registrata come utente.
Tipicamente un attaccante inserisce un codice maligno
HTML o Java Script in un sito web o per email, che
chiede alla vittima di eseguire quell' URL, la vittima
ignara lo esegue.
Come avrete capito questi tipi di attacchi sono difficili
da individuare, potenzialmente lasciando un utente
a girare sul sito dove l' attaccante ha piazzato il link
maligno, prima o poi l' utente dovrà pure cliccarci sopra.

Ora l' elaborazione di un modulo è anche possibile
attraverso il CSRF, ecco un esempio dove un utente
malintenzionato può creare un campo di input, come
specificato nella pagina web da attaccare:

<input type="text" name="guidare"/>

l' attaccante successivamente può anche creare un
auto invio utilizzando del codice JavaScript,
lasciando l' utente ignro di ciò che è accaduto.

Tokens
In questo esempio spiego come creare un token, e
come si forma una sterilizzazione prima dell'
invio. Questo è il mio sistema, e molto facile
da modificare:

Fase-1
Inserire questa codifica dopo ogni

<form method="POST">

in tutte le funzioni "echo()", assumendo
una forma importante:

Codice da inserire:

```
<input Type="hidden" name="token" value="".$_SESSION['token']."" />
```

Fase-2
Inserire questo codice al tuo file di configurazione:

```
if (isset($_USER['id'])) { // la nostra funzione per controllare se un utente è connesso o meno
   if (empty($_SESSION['token']) || !isset($_SESSION['token'])) { // se non c' è nessun token per
impostare
      $_SESSION['token'] = strrev(md5($_USER['password'])); // impostare un token con codifica
di stringa in md5 (password)
   }
   if (CSRF_PROTECTED != false) { // se non è definita CSRF_PROTECTED restituisce falso
      if ($_POST) { // se ci sono dati post
         if ($_POST['token'] != $_SESSION['token']) { // se l' input token non è uguale al token di
sessione
            header("Location: /index.php"); // redirect in index
            die(); // fermare tutti i dati che stanno arrivando
         }
      }
   }
}
```

Fase-3
Ad ogni pagina che è consentito accedere ad un utente
e si vuole disattivare questa protezione, inserire questo
codice all' iniziono dlle pagine:

```
define("CSRF_PROTECTED", false);
```

Conclusione
Cosi termina il tutorial, spero di essere stato chiaro.