Autentificare primară cu .htaccess şi .htpasswd

Autentificarea primară (Basic Authentication) este implementată chiar la nivelul serverului web (în cazul nostru Apache httpd). Spre deosebire de alte forme de autentificare, aceasta este cea mai simplu de pus în practică inclusiv de către un dezvoltator începător.

Cum funcţionează? În momentul în care clientul (vizitatorul) solicită o anumită pagină protejată, serverul trimite un răspuns cu codul 401 (Authentication needed) la care browser-ul reacţionează solicitându-i vizitatorului un nume de utilizator şi o parolă pe care mai apoi le trimite (într-o formă codificată) înapoi la server. În cazul în care datele de autentificare sunt corecte serverul permite accesarea respectivei resurse (pagini). În caz contrar va trimite din nou respectivul cod de răspuns (401).

Cum se pune în practică? Prima etapă este stabilirea resursei care se doreşte protejată şi a utilizatorilor care vor avea acces la ea. Serverul va fi informat despre aceste amănunte printr-un fişier cu numele .htaccess care trebuie creat (sau modificat, în sensul adăugării a încă 3-4 linii, în caz că există deja) în interiorul directorului respectiv.

Exemplu:

```
AuthName “Admin”
AuthType Basic
AuthUserFile /home/www/.htpasswd
Require User dancld
```

- ***AuthName*** **specifică un şir de caractere (stabilit de programator) care va fi afişat utilizatorului în momentul în care acesta încearcă să acceseze resursa protejată.**

- ***AuthType*** **trebuie să fie “Basic” pentru autentificarea primară (subiectul acestui text). Pentru alte tipuri de autentificări puteţi consulta documentaţia specificată la sfârşitul textului.**

- ***AuthUserFile*** **specifică întotdeauna calea (absolută!) către fişierul unde sunt stocate parolele (numele poate fi ales de utilizator, se recomandă folosirea numelui standard .htpasswd pentru a preveni posibilitatea accesării acestui fişier din browser de către vizitatori). Calea absolută (raportată la rădăcina fizică a serverului, nu la rădăcina host-ului virtual) se poate afla numai prin consultarea variabilei de mediu *$DOCUMENT_ROOT* (se creează un fişier php care conţine doar linia “<?=$DOCUMENT_ROOT?>” şi se accesează în browser). Din experienţa mea: serverul ţine seama numai de fişierul cu parole aflat în rădăcina host-ului virtual şi le ignoră pe cele aflate în vreun subdirector.**

- ***Require User*** **defineşte o listă de utilizatori cărora li se permite să acceseze resursa protejată. Fiecăruia dintre ei trebuie să îi corespundă o linie din fişierul de parole.**

Următoarea etapă este construirea unui fişier de parole. Un exemplu de astfel de fişier, în concordanţă cu exemplul de mai sus, este:

```
dancld:lGdugFynarS4.
```

După cum se observă, în acest fişier fiecare linie este de forma "<nume utilizator>:<parolă criptată>". Criptarea parolelor este realizată cu algoritmul DES Standard, folosind o „cheie" de criptare pe două caractere (acelaşi algoritm care este folosit şi pentru codificarea parolelor în Linux/Unix).

Crearea şi întreţinerea unui astfel de fişier se face în sistemul de operare Linux cu ajutorul programului `htpasswd` (instalat o dată cu serverul web Apache httpd). O soluţie rapidă pentru generarea de parole criptate este apelarea funcţiei *crypt()* într-un script PHP: un program de o linie, şi anume <?=crypt("parola_mea","cc")?> va afişa exact şirul de caractere de adăugat după ":" în fişierul de parole. Evident, în loc de "parola_mea" veţi pune parola dorită de dvs., iar în loc de "cc" cheia de 2 caractere folosită pentru codificare (vezi documentaţia aferentă).