

Gestiunea sesiunilor în PHP

Mihai Gabroveanu

Ce este un cookie?

- n Un **cookie** este pereche atribut-valoare trimisa de un server browserului web și apoi trimisă înapoi (nemodificată) de către navigator, de fiecare dată când accesează acel server
- n Trimiterea (setarea) unui cookie se face prin intermediul header-ului **Set-Cookie**: al unui răspuns HTTP trimis de catre server urmare a unei cereri
- n Un server poate trimite(seta) mai multe cookie-uri prin trimiterea de headere **Set-Cookie**: multiple.
- n Trimiterea înapoi a unui cookie se realizează prin intermediul headerului HTTP **Cookie**:

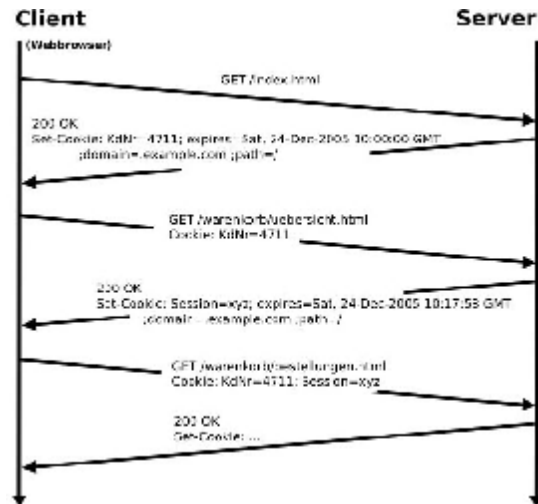
Setarea unui cookie: **Set-Cookie**

- n **Set-Cookie**: *name=value; [expires= date;] [path=path;] [domain= domain] [secure]*
- n unde
 - .. name= numele cookie-ului trimis browseului
 - .. value= valoarea asignată cookie-ului
 - .. date= data expirării cookie-ului
 - .. path= restrictionează trimiterea cookie-ului numai pentru cererile ce solicită o resursă ce incepe cu calea specificată
 - .. domain= restrictionează trimiterea cookie-ului pentru un anumit domeniu
 - .. secure = restrictionează trimiterea cookie-ului numai pe https

Trimiterea înapoi a cookie-urilor: **Cookie**

- n Browser-ul compară URL-ul cererii ce urmeaza sa o faca si domeniul care a trimis cookie-ul.
- n Daca calea si domeniul coincid atunci in cererea HTTP trimisă de browser se trimite si valoarea cookie-ului printr-un header de forma:
Cookie: name1=value1 ; name2=value2

Exemplu



Gestiunea sesiunilor în PHP

5

Ce este o sesiune?

- Protocolul HTTP este un protocol state-less. Fiecare cerere/raspuns sunt individuale si nu depind de celelalte.
- Un server web nu poate determina daca o cerere este legata de una anterioara, de la acelasi client.
- Implementarea unui mecanism prin care putem identifica ca o cerere face parte dintr-o succesiune de cereri de la acelasi client este esentiala in dezvoltarea aplicatiilor web
 - Exemplu: realizarea unor pagini ce necesita autentificare
- Sesiunea** este un de mecanism ce permite retinerea informatiilor despre starea clientului.
 - Implementarea sesiunilor se realizeaza cel mai adesea prin intermediul cookie-urilor

Gestiunea sesiunilor în PHP

7

Durata de viata a unui cookie

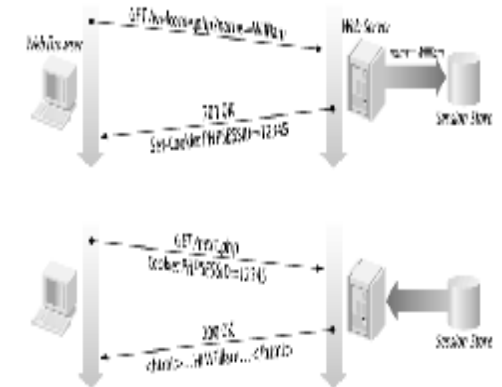
- Cookie-uri de sesiune:** acestea sunt cookie-uri retinute in memoria browserului.
 - La inchiderea browserului acestea sunt sterse automat
 - Nu pot fi utilizate pentru urmarirea pe termen lung
 - Sunt sigure, deoarece niciun program extren nu are acces la acestea
- Cookie-uri persistente:** acestea sunt cookie-uri sunt stocate intr-un fisier al browserului
 - Sunt disponibile pe termen lung
 - Potential nesigure, deoarece fisierul in care sunt stocare poate fi accesibil din alte programe

Gestiunea sesiunilor în PHP

6

Cum functioneaza sesiunile?

- Clientul prin intermediul browserului face o cerere initiala catre server
- Serverul identifica adresa IP a clientului, browserul, genereza un identificator de sesiune unic pe baza acestora, memoreaza la nivelul serverul o serie de informatii legate de client si trimite inapoi id-ul sesiuni sub forma unui cookie.
- La urmatoarele cereri clientul va trimite acel session id.
- Serverul va utiliza session id-ul pentru a obtine datele stocate la nivelul serverului asociate sesiunii.



Gestiunea sesiunilor în PHP

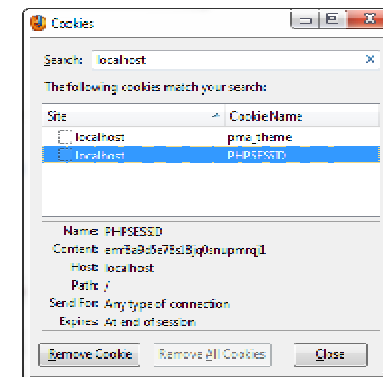
8

Crearea unei sesiuni in PHP

- n Pentru a crea o sesiune in PHP utilizam functia:
`session_start();`
- n Apelul acestei functii trebuie sa se realizeze la inceputul scriutului inainte de generarea oricarui continut HTML
- n Daca serverul nu a mai " vazut " userul respectiv anterior, atunci este creata o noua sesiune
- n Altfel, datele asociate sesiuni curente sunt incarcate in tabloul asociativ `$_SESSION[]`
- n Tabloul `$_SESSION[]` permite stocarea de date ce pot fi obtinute in paginile urmatoare solicitate in cadrul sesiunii create

Unde sunt memorate datele dintr-o sesiune?

- n La nivelul clientului session ID-ul este memorat un cookie cu numele PHPSESSID
- n la nivelul serverului, datele asociate sesiunii se memoreaza intr-un fisier stocat intr-un director temporar (eg:
`c:\xampp\tmp\sess_emf8a...`)
 - Pentru aplicatii datele pot fi memorate in cadrul unei baze de date SQL



Accesarea datelor dintr-o sesiune

- n Memorarea unei valori intr-o sesiune:
`$_SESSION["id_valoare"] = valoare;`
- n Obtinerea unei valori dintr-o sesiune
`$id_variabila = $_SESSION["id_valoare"];`
- n Testarea existentei unei date intr-o sesiune

```
if (isset($_SESSION["id_valoare"])){  
    ...  
}
```

Timeout-ul unei sesiuni

- n Deoarece protocolul HTTP este state-less, este dificil pentru server sa determine cand un user si-a terminat sesiunea
- n In mod ideal, la terminarea sesiuni de lucru userul ar trebui sa faca logout explicit; totusi multi nu fac acest lucru
- n Clientul sterge cookie-urile prin inchiderea browserului
- n Serverul sterge automat sesiunile vechi dupa o anumita perioada de inactivitate
 - Sesiunile vechi consuma resurse si pot genera riscuri de securitate
 - Putem ajusta intervalul de valabilitate printr-o setare globala a PHP-ului (vezi php.ini) sau apeland functia `session_cache_expire` function
 - Putem distruge explicit o sesiune apeland functia `session_destroy`



Utilizarea sesiunilor pentru autentificare

- n **Forma de Logare:** utilizatorului ii se cere sa introduca detaliile de autentificare intr-un formular, care apoi este transmis scriptului de autentificare
- n **Verificarea datelor:** scriptul verifica informatiile de autentificare
- n **Setarea starii:** In cazul in care informatiile sunt valide, in script se inregistreza una sau mai multe variabile de sesiune care sa fie apoi folosita ca referinta pentru starea autentificata
- n **Verificarea starii:** scripturile ce necesita autentificare pentru a fi rulate testeza valoarea variabilelor setate la pasul anterior, si in cazul in care acestea au valorile corespunzatoare executa actiunile protejate
- n **Logout:** la log-ul sunt sterse stearea (cu unset) variabilele de sesiune ce marcau starea de "autentificat".



Exemplu

- n Exemplu practic de implementare a unui Guestbook.