Ders 4

PHP Sessions

İçindekiler

- PHP betiklerinin işleyişine tepeden bir bakış
- SESSIONS nedir? Neden böyle bir şeye ihtiyaç vardır?
- Basit kod örnekleri
- SESSIONS'un çalışma mekanizması
- SESSIONS ne zaman kullanılır? Ne zaman kullanılmaz?
- SESSIONS ve güvenlik
- Sessionlarla alakalı fonksiyonlardan bazıları

PHP betiklerinin işleyişine tepeden bakış

- PHP kullanıcıdan gelen istek üzerine çalışmaya başlayan, isteği cevapladıktan sonra sonlanan bir yapıya sahiptir.
- PHP programları, masaüstü uygulamaları gibi sürekli açık kalan programlar değildir.
- Örneğin siz bir programı (mesela Winamp) açtığınızda, kapat düğmesine basılana (veya çökene kadar) açık kalacaktır.
- Yazdığınız PHP programı ise, istediklerinizi yapacak, bir çıktı oluşturacak ve sonlanacaktır.

İki çalışma arası veri tutma ihtiyacı

- Word'ü açın. İçine satırlarca yazı yazın. Sonra kaydetmeden kapatın.
- Word'ü tekrar açın.
- Yazılar nereye gitti?
- Bu sorunu çözmek için yazdıklarımızı bir yere kaydetmemiz gerekir.
- Her kullanıcının kendi belgesini, ayrı dosyalar olarak kaydetmesi gerekir.
- Böylece veriler, programın açılıp kapanması arasında kaybolmaz.

Sessions

- PHP Sessions bir önceki slaytta anlatılan metodu PHP için uygulayan bir sistemdir.
- Her ziyaretçiye bir kimlik numarası verilir. Ziyaretçi bizim sitemizde gezdiği sürece her sayfa isteğiyle beraber bu kimlik numarasını da gönderir.
- Böylece PHP ziyaretçiyi tanır ve bu ziyaretçi hakkında bilgi tutabilir.

SESSIONS Çalışma Mantığı

- 1. Kullanıcı siteyi ilk defa ziyaret eder.
- 2. Sayfa yaratılırken kullanıcıya bir kimlik numarası gönderilir.
- 3. Kullanıcı ikinci bir sayfa açmak ister. Bu istek esnasında kendi kimlik numarasını da gönderir.
- 4. PHP kullanıcıyı tanır, ve kimlik numarasını kullanarak onun hakkındaki bilgilere ulaşır.

Basit bir sessions örneği

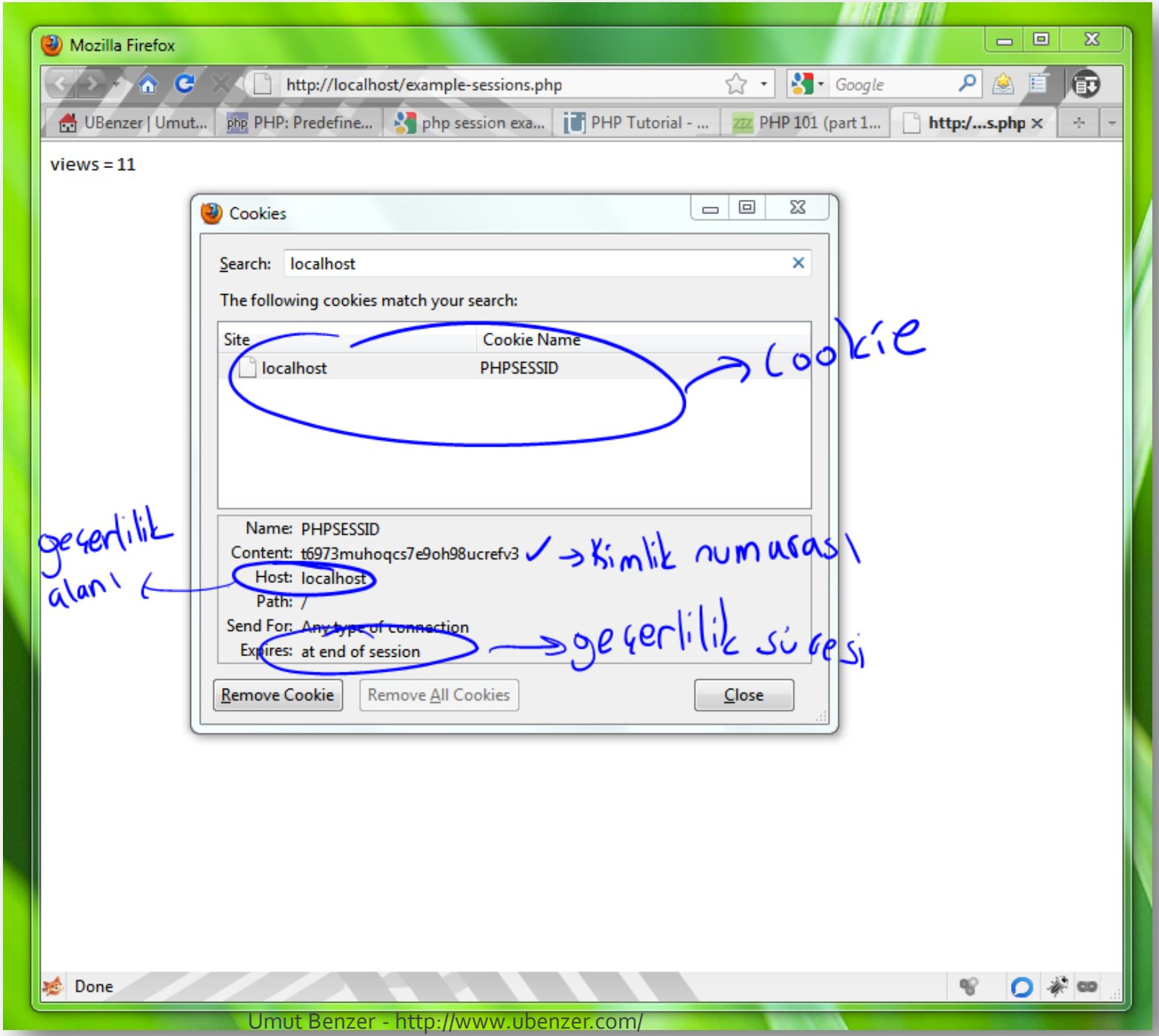
```
<?php
session_start();
if(isset($_SESSION['views'])) {
        $_SESSION['views'] = $_SESSION['views']+ 1;
} else {
        $_SESSION['views'] = 1;
}
echo "views = ". $_SESSION['views'];
?>
```

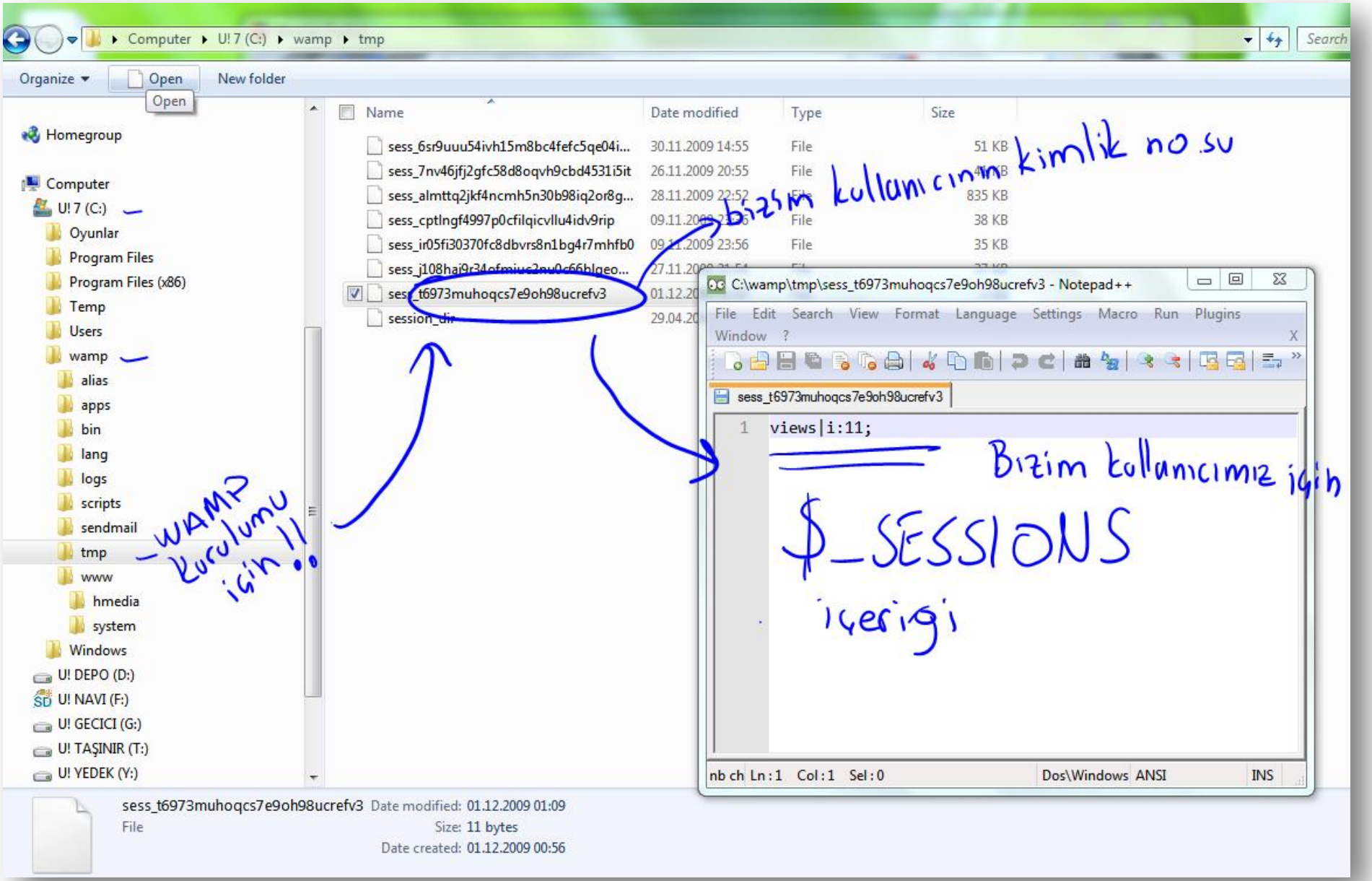
Sessionsların Mekanizması

- Sessions'un ne olduğunu ve uygulamasının nasıl olduğunu gördük. Şimdi de mekanizmasını göreceğiz.
- $_SESSIONS süper değişkeninde saklanan tüm veriler her session id için ayrı bir dosyada olmak üzere, geçici bir klasörde depolanır.
- Bu klasör PHP kurulumları arasında farklılık gösterebilir. WAMP ile gelen kurulumda dosyalar C:\wamp\tmp içerisinde saklanmaktadır.

Sessionsların Mekanizması

- Sessionlar uzun süreli veri saklamak için değildir. (bir veritabanı gibi kullanamazsınız)
- Session bilgileri şu şekilde «expire» olur:
  - Kullanıcı tarayıcıyı kapatabilir. Kullanıcı tarafında kimlik numarası Cookie'ler ile tutulur. Tarayıcı kapanınca bu cookieler silinir. Sunucu tarafında bilgi durmaya devam eder, ancak kimse erişmez. Belirli bir süre geçince PHP'nin «garbage collector»u zamanı geçmiş Session'ları siler.


Mozilla Firefox
http://localhost/example-sessions.php
Google
UBenzer | Umut...
PHP: Predefine...
php session exa...
PHP Tutorial - ...
PHP 101 (part 1...
http:/...s.php
views = 11
Cookies
Search: localhost
The following cookies match your search:
Site
Cookie Name
localhost
PHPSESSID
Cookie
Name: PHPSESSID
Content: t6973muhoqcs7e9oh98ucrefv3
Kimlik numarası
Host: localhost
Path: /
Send For: Any type of connection
Expires: at end of session
geçerlilik süresi
geçerlilik alanı
Remove Cookie
Remove All Cookies
Close
Done


Computer ▸ U! 7 (C:) ▸ wamp ▸ tmp
Search
Organize
Open
New folder
Open
Homegroup
Computer
U! 7 (C:)
Oyunlar
Program Files
Program Files (x86)
Temp
Users
wamp
alias
apps
bin
lang
logs
scripts
sendmail
tmp
www
hmedia
system
Windows
U! DEPO (D:)
U! NAVI (F:)
U! GECICI (G:)
U! TAŞINIR (T:)
U! YEDEK (Y:)
Name
Date modified
Type
Size
sess_6sr9uuu54ivh15m8bc4fefc5qe04i... 30.11.2009 14:55 File 51 KB
sess_7nv46jfj2gfc58d8oqvh9cbd4531i5it 26.11.2009 20:55 File
sess_almttq2jkf4ncmh5n30b98iq2or8g... 28.11.2009 22:52 File 835 KB
sess_cptlngf4997p0cfilqicvllu4idv9rip 09.11.2009 File 38 KB
sess_ir05fi30370fc8dbvrs8n1bg4r7mhfb0 09.11.2009 23:56 File 35 KB
sess_t6973muhoqcs7e9oh98ucrefv3 01.12.20
session_dir 29.04.20
C:\wamp\tmp\sess_t6973muhoqcs7e9oh98ucrefv3 - Notepad++
File Edit Search View Format Language Settings Macro Run Plugins
Window ?
sess_t6973muhoqcs7e9oh98ucrefv3
1 views|i:11;
nb ch Ln:1 Col:1 Sel:0
Dos\Windows ANSI INS
sess_t6973muhoqcs7e9oh98ucrefv3 Date modified: 01.12.2009 01:09
File
Size: 11 bytes
Date created: 01.12.2009 00:56
bizim kullanicinin kimlik no.su
WAMP Kurulumu için !!
Bizim kullanicimiz için
$_SESSIONS
icerigi

Sessions ne zaman kullanılır, ne zaman kullanılmaz?

- Kullanıcının «hatırlanması» gerektiği durumlarda sessions kullanılır.
- Ancak $_SESSIONS içerisinde tutulan değişkenlerin her istekte okunup, değişikliklerin doğrudan diske yazıldığını ve diske yazılır hale getirilmek için «serialize» edilip okunmak için «deserialzie» edildiği görülmektedir. Bu da çok aşırı bilginin $_SESSIONS ile saklanması durumunda ciddi performans kayıplarına neden olur.
- Bu yüzden iki sayfa arası çok aşırı bilgi tutacaksak $_SESSIONS yerine veritabanı kullanmak tercih edilmelidir.

Sessions ne zaman kullanılır, ne zaman kullanılmaz?

- Kullanıcının «hatırlanması» gerektiği durumlarda sessions kullanılır.
- Ancak $_SESSIONS yapısı gereği tarayıcı kapana kadar çalışır. Bu yüzden «beni hatırla» benzeri uzun vadeli «hatırlama» gerektiğinde sessionlara yardımcı cookieler kullanmak gerekecektir.
- Ayrıntılar laboratuvarda anlatılacaktır.

Sessions & Güvenlik

- Sessionlar bir kimlik numarası esasına göre çalıştıklarından tek başlarına ciddi birer güvenlik açığı teşkil ederler.
- İnternet programlamasında «kullanıcıya giden her veri çalınabilirmiş» gibi düşünüp biraz paranoyakça davranmak gerekmektedir.
- Hiçbir önlem alınmamışsa, kullanıcı session kimliğini tutan cookieyi çaldırdığında hırsız sistemde kullanıcıymış gibi gezebilecektir.

Sessions & Güvenlik

- Session Fixation
  - URL injection
  - Phishing POST form
  - XSS
  - vs vs...
- Session Hijacking
  - IP restriction?
  - User Agent restriction?
- Other

Session Fonksiyonları

- **session_start ()**
  - **Bir session başlatır.**
  - **Bir «session başlatmak» ile anlatılmak istenen şudur:**
  - **Kullanıcı için bir kimlik numarası belirlenir.**
  - **Bu kimlik numarası kullanıcıya gönderilmek üzere cookielere eklenir.**
  - **Sessionların sunucu üzerinde tutalacağı klasörde session bilgilerini tutacak bir dosya oluşturularak $_SESSIONS süper değişkeni ilklenir.**

Session Fonksiyonları

- string **session_id(**[$degisken]**)**
  - **Eğer varsa güncel sessiondaki kullanıcının kimlik numarasını verir. Bunun için bir session başlatılmış olmalıdır.**
  - **Ayrıca isteğe bağlı olarak $degisken değişkeni aracılığı ile kendi istediğimiz id'yi atamamız mümkündür.**
  - **Session id değerini kendimiz atıyorsak, bunu session başlamadan önce yapmalıyız.**

Session Fonksiyonları

- string **session_name(**[$degisken]**)**
  - **Session kimlik numarasının tutulacağı değişkenin adını belirlemeye/öğrenmeye yarar.**
  - **Değişiklikler session_start()'tan önce yapılmalıdır.**

Session Fonksiyonları

- session_regenerate_id ()
  - Güncel sessiondaki bilgileri saklayarak, sessionun kimlik numarasını değiştirir.
  - Kullanıcıya yeni kimlik numarası gönderilir.

Session Fonksiyonları

- string **session_save_path** ([string $path])
  - **Session bilgilerinin sunucuda nereye kaydedileceğinin öğrenilmesini ve değiştirilmesini sağlar.**
  - **Kayıt klasörü değişiklikleri session_start()'tan önce yapılmalıdır.**
  - **Bu fonksiyon paylaşılan hosting ortamlarında session bilgilerini güvenli bir klasöre taşımak için kullanılabilir.**
  - **Session save path ile belirtilen klasör için PHP'nin dosya yazma/okuma izni mutlaka olmalıdır.**

Session Fonksiyonları

- **session_set_cookie_params** (

  int $lifetime,

  string $path,

  string $domain,

  bool $secure = false,

  bool $httponly = false)

  - **Sessionun kimlik numarasının gönderileceği cookienin özelliklerinin değiştirilmesini sağlar.**

Session Fonksiyonları

- lifetime:
  - Cookie kullanıcının bilgisayarında ne kadar kalacak?
  - Varsayılan: «0». Kullanıcı tarayıcıyı kapatana kadar.
  - **Hangi sorunlar çıkabilir?**
- path:
  - Cookie domaindeki hangi klasörlere gönderilecek?
  - Varsayılan: «/» Hepsine.

Session Fonksiyonları

- domain:
  - Cookie hangi domainlerde geçerli olacak?
- secure:
  - Cookie sadece güvenilir bağlantılarda mı gönderilecek?
- httponly:
  - Cookie sadece HTTP protokolü ile kurulan bağlantılarda mı gönderilecek?

Burada anlatılan özellikler sadece session cookieleri için değil, tüm cookieler için geçerli özniteliklerdir.

Session Fonksiyonları

- **session_unset** ()
  - Güncel session ile ilgili tüm bilgileri bellekten atar. $_SESSION süper değişkeninin içini boşaltır ve sessiona ait bilgilerin tutulduğu dosyayı boşaltır.
  - Bu fonksiyon kullanıcıya gönderilmiş session ID'sini değiştirmez. Sadece içeriğini siler.

Yararlanılan Kaynaklar

- http://www.php.net/manual/en/session.security.php
- http://www.tizag.com/phpT/phpsessions.php
- http://devzone.zend.com/article/646
- http://php.net/manual/en/function.ini-set.php
- http://tr.php.net/manual/en/session.configuration.php
- http://www.sitepoint.com/blogs/2004/03/03/notes-on-php-session-security/
- http://phpsec.org/projects/guide/4.html
- http://www.php.net/manual/en/intro.session.php
- http://www.php.net/manual/en/session.examples.php
- http://www.acrossecurity.com/papers/session_fixation.pdf
- http://www.php.net/manual/en/ref.session.php
- http://www.php.net/manual/en/function.session-id.php
- http://www.php.net/manual/en/function.session-name.php
- http://www.php.net/manual/en/function.session-regenerate-id.php
- http://www.php.net/manual/en/function.session-save-path.php
- http://www.php.net/manual/en/function.session-set-cookie-params.php
- http://www.php.net/manual/en/function.session-start.php
- http://www.php.net/manual/en/function.session-unset.php