Web Teknolojileri ve Güvenlik Konuları - Özet

XAMPP nedir?

XAMPP, Apache Friends tarafından dağıtılan ücretsiz ve kolay kurulabilen bir web sunucu paketidir. İçinde Apache HTTP Server, MariaDB (veya MySQL), PHP ve Perl gibi bileşenler bulunur. Geliştiriciler yerel makinede PHP ve veritabanı tabanlı uygulamalar geliştirmek, test etmek ve öğrenmek için XAMPP kullanırlar. Canlı üretim ortamı yerine geliştirme ve test amaçlı tasarlanmıştır; bu yüzden güvenlik ayarları varsayılan olarak geliştirmeye uygun biçimde gevşetilmiş olabilir.

LDAP nedir?

LDAP (Lightweight Directory Access Protocol), ağ üzerindeki dizin (directory) bilgilerine erişmek ve yönetmek için kullanılan bir protokoldür. Kullanıcı hesapları, kimlik doğrulama bilgileri, e-posta adresleri ve kaynaklar (sunucular, yazıcılar vb.) gibi hiyerarşik olarak düzenlenmiş bilgileri saklamak ve sorgulamak için idealdir. OpenLDAP gibi uygulamalar LDAP sunucusu olarak yaygın şekilde kullanılır.

Cookie (Çerez) nedir, neden önemlidir ve nasıl oluşturulur?

Çerezler, bir web sunucusunun kullanıcı tarayıcısına gönderdiği, küçük veri parçalarıdır. Tarayıcı çerezleri kaydeder ve aynı siteye yapılan sonraki isteklerde tekrar gönderir. Çerezler oturum yönetimi (login durumu), tercihleri hatırlama, kişiselleştirme ve izleme gibi amaçlarla kullanılır. Güvenlik ve gizlilik açısından önemlidir: oturum çerezlerinin ele geçirilmesi kimlik hırsızlığına yol açabilir. Basit bir örnek (HTTP header ile): Set-Cookie: sessionId=abc123; HttpOnly; Secure; SameSite=Lax. JavaScript ile bir çerez oluşturma örneği: document.cookie = "kullanici=Elif; path=/; max-age=3600; Secure; SameSite=Lax".

Drupal nedir?

Drupal, içerik yönetim sistemi (CMS) kategorisinde açık kaynaklı bir platformdur. İçerik editörleri, tasarımcılar ve geliştiriciler için modüler ve esnek yapı sağlar. Geniş eklenti (module) ekosistemi ve güçlü kullanıcı/roller yetkilendirme sistemi ile kurumsal ve topluluk siteleri için tercih edilir.

Cross-Site Scripting (XSS) nedir ve nasıl önlenir?

XSS, bir web uygulamasına kötü amaçlı script (genelde JavaScript) enjekte edilmesi ile gerçekleşir ve bu script mağdurun tarayıcısında çalışır. XSS türleri: Stored (kalıcı), Reflected (yansıtmalı) ve DOM-based (tarayıcı taraflı). Etkileri arasında oturum çalma, sayfa içeriği değiştirme ve kimlik sahtekarlığı bulunur. Önleme yaklaşımları: kullanıcı girdilerini uygun şekilde doğrulamak ve çıkışta (output) HTML encode/escape yapmak, Content Security Policy (CSP) kullanmak ve çerezleri HttpOnly/Secure olarak ayarlamak.

CAPTCHA nedir?

CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart), çevrimiçi kullanıcıların insan mı yoksa bot mu olduğunu ayırt etmek için kullanılan testlerdir. Görsel karakter tanıma, nesne seçme, davranış tabanlı analiz veya sesli alternatifler gibi çeşitli türleri vardır. Amaç spam, otomatik kayıt ve scraping gibi kötü amaçlı otomasyonları engellemektir.

Web sayfası kodlarının şifrelenmesi / korunması nasıl yapılır?

Tam anlamıyla 'müşteri tarafı' (client-side) kaynak kodunu gizlemek mümkün değildir; tarayıcı çalıştırabilmek için HTML/CSS/JS açıkça gönderilmelidir. Ancak kodu anlamayı zorlaştırmak ve bazı içerikleri korumak için yöntemler vardır:
- JavaScript obfuscation (karmaşıklaştırma) ve minification: değişken isimlerini küçültme, kod akışını zorlaştırma.
- Sunucu tarafı (server-side) mantığı ve hassas verileri sunucuda tutmak; istemciye yalnızca gerekli verileri göndermek.
- PHP gibi sunucu tarafı dilleri için ionCube gibi bytecode encoder/obfuscator kullanmak; bu yöntemle kaynak kodun düz metin hâli sunucuda daşınmaz.
Not: Kod koruma çözümleri kesin güvenlik sağlamaz; yetenekli bir saldırgan kodu tersine mühendislik ile analiz edebilir. Güvenlik katmanlı olmalıdır.

ionCube nedir?

ionCube, PHP kodunu bytecode'a çevirip şifreleyerek dağıtmak ve lisanslama özellikleri sağlamak üzere kullanılan ticari bir araçtır. Encoded (şifrelenmiş) PHP dosyalarını çalıştırmak için sunucuda ionCube Loader eklentisi gerekir. Bu sayede PHP kaynak kodunun düz metin hâli sunucu dışına çıkmış olsa bile okunması zorlaşır.

XAMPP - Hızlı Başlangıç ve phpinfo Örneği

Adımlar:

1. XAMPP kontrol panelinden Apache ve MySQL/MariaDB servislerini başlatın.
2. Proje dosyalarınızı htdocs klasörüne koyun (örn: C:/xampp/htdocs/myproject).
3. Tarayıcıda http://localhost/myproject/phpinfo.php adresini açarak PHP konfigürasyonunuzu görüntüleyin.

phpinfo.php örneği:

```
<?php
phpinfo();
?>
```

LDAP - Örnek ldapsearch ve PHP ile Bind

Komut satırı ile LDAP sorgulama örneği (OpenLDAP sunucusuna bağlanma):

ldapsearch örneği:

```
ldapsearch -x -H ldap://ldap.example.com -D "cn=admin,dc=example,dc=com" -W -b
"dc=example,dc=com" "(uid=ali)"
```

PHP ile LDAP bind ve basit sorgu örneği:

PHP LDAP örneği:

```
<?php
$ldapconn = ldap_connect("ldap.example.com") or die("Could not connect to LDAP
server.");
```

```
ldap_set_option($ldapconn, LDAP_OPT_PROTOCOL_VERSION, 3);
if($ldapbind = ldap_bind($ldapconn, "cn=admin,dc=example,dc=com",
"admin_password")) {
    $result = ldap_search($ldapconn, "dc=example,dc=com", "(uid=ali)") or
die("Error in search");
    $entries = ldap_get_entries($ldapconn, $result);
    print_r($entries);
} else {
    echo "LDAP bind failed.";
}
?>
```

Çerezler (Cookies) - Örnekler ve Öznitelikler Tablosu

HTTP header ile Set-Cookie örneği ve JavaScript ile çerez oluşturma örneği:

HTTP header (Set-Cookie) örneği:

```
Set-Cookie: sessionId=abc123; Path=/; HttpOnly; Secure; SameSite=Strict; Max-
Age=3600
```

JavaScript ile çerez oluşturma:

```
document.cookie = "kullanici=Elif; path=/; max-age=3600; Secure; SameSite=Lax";
```

HttpOnly	*JavaScript tarafından erişimi engeller*	*Oturum çerezleri için önerilir*
Secure	*Sadece HTTPS üzerinden gönderilir*	*Zorunlu olarak kullanın*
SameSite	*CSRF riskini azaltır (Strict/ Lax/None)*	*Varsayılan SameSite=Lax iyi bir başlangıç*
Path/Domain	*Çerezin kullanılabileceği yol ve domain*	*En dar kapsam kullanın*
Max-Age/Expires	*Çerezin ömrünü belirler*	*Oturum çerezleri için kısa ömür*

Drupal - Basit Modül İskeleti Örneği

Drupal 8/9/10 için basit bir modul .module dosyası ve info.yml örneği:

mymodule.info.yml örneği:

```
name: 'My Module'
type: module
core_version_requirement: ^8 || ^9 || ^10
package: Custom
```

mymodule.module örneği:

```
<?php
use Drupal\Core\Routing\RouteMatchInterface;

function mymodule_help($route_name, RouteMatchInterface $route_match) {
  switch ($route_name) {
```

```
    case 'help.page.mymodule':
      return t('Yardım metni...');
  }
}
?>
```

Cross-Site Scripting (XSS) - Kırılabilir Örnek ve Düzeltme

Vulnerable PHP örneği (kullanıcı girdisini filtresiz yazma):

Vulnerable örnek:

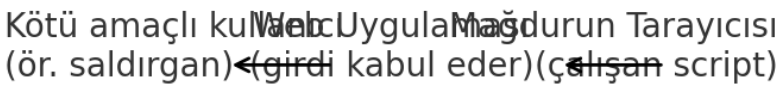
```
<?php
echo "Hoşgeldin, " . $_GET['name'];
?>

// Eğer name parametresine <script>alert('xss')</script> gönderilirse, script
çalışır.
```

Güvenli çıktı (HTML encode) örneği:

Fix (PHP):

```
<?php
echo 'Hoşgeldin, ' . htmlspecialchars($_GET['name'], ENT_QUOTES, 'UTF-8');
?>
```



CAPTCHA (reCAPTCHA) - Sunucu Tarafı Doğrulama Örneği

reCAPTCHA doğrulama (PHP örneği):

```
<?php
$recaptcha_secret = 'YOUR_SECRET_KEY';
$response = $_POST['g-recaptcha-response'];
$verify = file_get_contents("https://www.google.com/recaptcha/api/siteverify?
secret={$recaptcha_secret}&response={$response}");
$captcha_success = json_decode($verify);
if ($captcha_success->success) {
```

```
  // işlem devam eder
} else {
  // bot olma ihtimali
}
?>
```

Kod Koruma - ionCube ve JS Obfuscation Örnekleri

ionCube: PHP kodunu encode eder; sunucuda ionCube Loader gerektirir. Aşağıda JS için basit 'obfuscation' (elle yapılmış küçük örnek) gösterilmiştir.

Basit JS obfuscation örneği (manüel):

```
function greet(name) {
  var a = 'Hel' + 'lo, ';
  var b = name.split('').reverse().reverse().join('');
  return a + b;
}
// Bu basit taktikler minimal gizleme sağlar; gerçek obfuscator daha karmaşık dönüşümler yapar.
```

Özet Tablolar: XSS Önlemleri

Input Validation	*Girdi tiplerini ve uzunluğunu sınırlar*	*Tüm kullanıcı girdilerinde*
Output Encoding	*HTML/JS/URL encode ile çıktıyı güvenli hale getirir*	*Dinamik içerik çıktısı*
Content Security Policy (CSP)	*Tarayıcının hangi kaynakları yükleyebileceğini sınırlar*	*Ekstra katman, inline scriptleri engellemek için*
HttpOnly Cookies	*JS ile çerez erişimini engeller*	*Oturum çerezleri için*