BLM306 YAZILIM LAB III.

Proje 3

PROJE TESLİM TARİHİ: 03.05.2017

Proje: ZAYIF (VULNERABLE) WEB UYGULAMALARI GELİŞTİRİLMESİ

- **Proje istenilen dilde geliştirilebilir.**
- **Proje kaynak kodları (sourcecode) CD içerisinde raporla beraber teslim edilecektir. Ayrıca her grup E-Destek üzerinden de projenin kaynak kodları ve raporlarını yükleyecektir. Yükleme tarihi saati ve formatı hakkında ayrıntılı bilgi daha sonra duyurulacaktır.**
- **Proje grupları 2 kişilik oluşturulacaktır. Grup oluşturma linki aşağıda verilmiştir.**
- **Proje ile alakalı sorularınız için** meltem.kurt@kocaeli.edu.trve burcu.kir@kocaeli.edu.tr**adreslerine ulaşabilirsiniz.**

Giriş

Kasıtlı olarak bırakılmış güvenlik zafiyetlerini (vulnerability) içeren zayıf (vulnerable) web uygulamaları, uygulamalı siber güvenlik eğitimlerinde önemli bir rol oynamaktadır. Öğrenciler, yasal olmayan bir şekilde gerçek sistemlere saldırmadan bu tip uygulamalar üzerinde öğrendiklerini tecrübe ederek kendilerini geliştirebilmektedirler.

Farklı programlama dillerinde (PHP, ASP, C# .NET, Node.js, PERL, Java, Ruby on Rails, Python, C++, ColdFusion vb.) yazılmış vulnerable web uygulamaları bulunmaktadır. Fakat, Türkçe yazılmış ve iyi bir şekilde dokümantasyonu yapılmış vulnerable web uygulamaları yok denecek kadar azdır.

Bu projenin amacı ülkemizdeki siber güvenlik derslerinde kullanılabilecek bir düzeyde vulnerable web uygulamaları geliştirmektir. Bunun için sadece üzerinde zafiyetler bulunan bir web uygulaması geliştirmek yeterli değildir, bu uygulamadaki zafiyetlerin neden kaynaklandığının ve bu zafiyetlerin nasıl sömürüleceğinin (exploitation) de açık ve anlaşılır bir şekilde dokümantasyonun yapılması gereklidir.

Proje Tanımı ve Yapmanız Gerekenler

Proje Gereksinimleri

1. Web uygulaması şu zafiyetleri içermelidir:
    a. SQL Enjeksiyonu (SQL Injection)
    b. Siteler Arası Betik Çalıştırma (XSS - Cross Site Scripting)
    c. Komut Enjeksiyonu (CommandInjection)

2. Web uygulaması istenilen bir programlama dili kullanılarak geliştirilebilir.
3. Web uygulaması en az bir tarayıcı üzerinden (Chrome, Firefox, Safari veya Internet Explorer) sorunsuz bir şekilde kullanılabilmelidir.
4. Her bir zafiyet için web uygulamasında ayrı bir sayfa geliştirilmelidir. Bir sayfada birden fazla zafiyet bulunmamalıdır.
5. Zafiyetin bulunduğu her bir sayfada bulunan bir "yardım" bağlantısına tıklandığında pop-up şeklinde açılan bir sayfada ilgili zafiyetin sömürmek için örnek bir URL, bir paragraflık açıklaması, kaynak koddaki hangi kısımdan dolayı ve neden bu zafiyetin ortaya çıktığı, kaynak koddaki hangi değişiklikle ve nasıl bu zafiyetin engellenebileceği yazılmalıdır.

Proje Sunumu

1. **Sunum sırasında web uygulaması herhangi bir nedenden dolayı çalışmazsa veya uygulamaya erişilemezse proje başarısız sayılacaktır.** Proje ekibi sunum öncesi gerekli testleri yapıp çalışan bir vulnerable web uygulamasıyla sunuma gelmelidir. Proje için gerekli tüm bağımlılıklar öğrencinin bilgisayarında yüklü olmalıdır. Sunum sırasında E-Destek üzerinden projenizi indirip çalıştıracağınız için internet erişiminiz mutlaka olmalıdır.

Proje Değerlendirme Yönergesi Açıklamalar

1. **PROJE RAPORU:** Her proje sonunda teslim edilmesi gereken projenizi her yönüyle açıklayacağınız bir dokümandır.
   - **1.1. Problem Tanımı:** Proje kapsamında sizden çözüm bulmanız beklenen problem ile ilgili açıklama yapılması gerekmektedir. Burada amaç projenin ne kadar anlaşıldığını test etmektir.
   - **1.2. Yapılan Araştırmalar:** Proje geliştirilmesi aşamasında karşılaşılan sorunlara nasıl çözümler bulunduğu ve bu konularda yapılan araştırmalar açıklanmalıdır.
   - **1.3. Tasarım**
     - **1.3.1. Akış şeması:**Proje içerisinde yer alan algoritma ve işlemleri şekilsel olarak ifade edecek şema oluşturmanız beklenmektedir.
     - **1.3.2. Yazılım mimarisi:** Projenin kodlanması aşamasında kullanılacak kod yapısı ve geliştirme aşamalarını gösteren bir yapı hazırlanması beklenmektedir.
     - **1.3.3. Veri tabanı diyagramı:** Projeye ait ER diyagramının oluşturulması beklenmektedir.
   - **1.4. Genel Yapı:** Projenizi genel yapısı bakımından her yönüyle özetlemeniz gerekmektedir.
   - **1.5. Referanslar:** Proje geliştirilirken ve araştırma aşamasında faydalanılan kaynaklar rapor dokümanının en altında listelenmeli ve dokuman içerisinde de ilgili yerlerde indekslenmelidir.

     Referans formatı aşağıda verilen örneklere uygun olmalıdır.

     Kitap, çok yazarlı
     Larson, G. W., Ellis, D. C.,&Rivers, P. C. (1984). Essentials of chemicaldependencycounseling. New York: Columbia UniversityPress.

     Report from a privateorganization (author&publishersame)

     NationalLeagueforNursing. (1990). Self-studyreportforcommunityhealthorganizations (Pub. No. 21-2329). New York: Author.

     Unpublishedmaster'sthesis

     Paulosky, K. A. (1997). Knowledge andattitudes of painandactivities of nurseadministrators. Unpublishedmaster'sthesis, Northern Michigan University, Marquette, Michigan.

     Article in a journal (continuouspaginationthroughoutvolume)

     Burke, R. J., Shearer, D., &Deszca, E. (1984). Correlates of burnoutphasesamongpoliceofficers. GroupandOrganizationalStudies, 9, 451-466.

     Article in a Popular Magazine

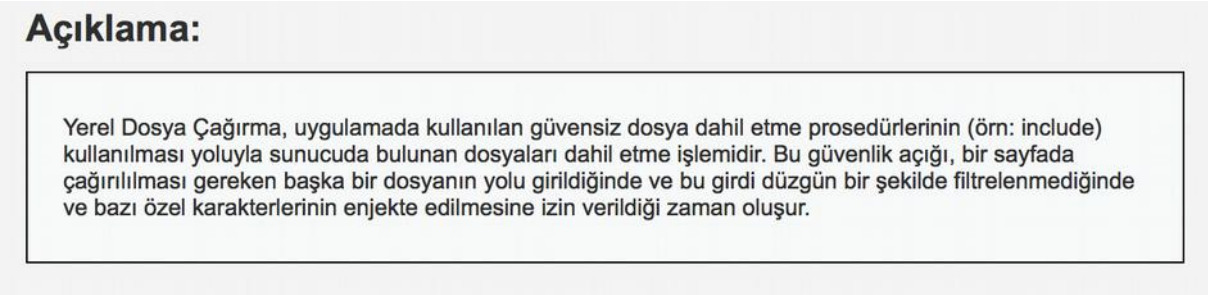
Caloyianis, N. (1998, September). Greenlandsharks. NationalGeographic, 194, 60- 71.
Web Site
http://en.wikipedia.org/wiki/Neural_network (Access date: 07.10.2013)

2. GELİŞTİRİLEN UYGULAMADA BULUNAN HER BİR ZAFİYET İÇİN PUANLAMA ADIMLARI

(NOT: SQL ENJEKSİYONU, XSS VE KOMUT ENJEKSİYONU DIŞINDA BİR ZAFİYET İÇİN PUAN VERİLMEYECEKTİR):

2.1. (2 puan) Zafiyetin bir paragraflık açıklaması.

Örnek:



2.2. (5 puan)Bu zafiyetin geliştirilen uygulamanın kaynak kodundaki hangi güvensiz kodlama pratiğinden kaynaklandığının detaylı açıklaması (bu zafiyetin uygulamada neden var olduğunun ilgili kod kısmıyla açıklanması).

Örnek:

```php
<?php
$file = $_GET['file']; // Burada çağırılmak istenen dosyayı bir GET isteğiyle
URL'den 'file' parametresiyle alıyoruz.
if(isset($file)) // Eğer 'file' parametresi geldiyse
{
    include("pages/$file"); // çağırılmak istenen dosyayı getir
}
else // file paremetresi gelmediyse
{
    include("index.php"); // index.php sayfasını çağır
}
?>
```

Yukarıda görüldüğü gibi 'file' parametresi URL üzerinden alındıktan sonra herhangi bir filtreleme veya kontrol işleminden geçirilmeden 'include' fonksiyonu içerisinde kullanılmaktadır. Bir saldırgan, 'file' parametresine sayfa.php gibi beklenen bir girdi yerine ../../../../etc/passwd gibi bir girdi girerek sunucudaki önemli dosyalara erişebilir. Kullanıcı tarafından gelen girdi sunucu tarafında herhangi bir kontrol ve filtreleme işlemine tabi tutulmadan 'include' fonksiyonu içerisinde kullanıldığı için bu açık ortaya çıkmıştır.

2.3. (13 puan)Bu zafiyeti sömüren bir GET isteği için URL (yani bu zafiyetin sömürüsü için verilen örnek URL tarayıcının adres çubuğuna girildiğinde

zafiyetin var olduğu ve sömürüldüğü açık bir şekilde görülmelidir). Örnek sömürü (exploitation) isteği verilmezse veya verilen örnek sömürü kodu çalışmazsa zafiyetin uygulamada var olduğu kanıtlanamamış kabul edilecektir ve bu kısımdan puan alamayacaksınız.

Örnek:

2.4. (5 puan)Kodda hangi değişiklik (iyileştirme) yapılarak bu zafiyetin ortaya çıkmasının engellenebileceği ve bu değişikliğin neden zafiyeti engellediğinin detaylı açıklaması.

Örnek:

```php
<?php
$file = $_GET['file']; // Burada çağırılmak istenen dosyayı bir GET isteğiyle
URL'den 'file' parametresiyle alıyoruz.
if(isset($file)) // Eğer 'file' parametresi geldiyse
{
 $ok = array('news.php', 'contact.php', 'team.php'); // burada çağırılmasına
 izin verilen sayfalar yer alıyor (whitelist)
 $include = in_array($_GET['page'], $ok) ? $_GET['page'] : 'index.php'; // Eğer
 'file' paremetresinde gelen değer whitelist içerisindeyse 'include' değişkenine
 ata, değilse 'index.php' değerini ata
 include("pages/$include"); // çağırılmak istenen dosyayı getir
}
else // file paremetresi gelmediyse
{
    include("index.php"); // index.php sayfasını çağır
}
?>
```

Yukarıda görüldüğü gibi 'file' parametresi URL üzerinden alındıktan sonra kullanıcı tarafından gelen girdi 'ok' isminde bir dizide bulunan değerlerle karşılaştırılıyor. Eğer girdi bu değerlerden birisiyle eşleşmediyse kullanıcı tarafından gelen girdi dikkate alınmıyor ve varsayılan değer olan 'index.php' sayfası çağırılıyor. Burada beyaz liste (whitelist) yöntemi ile kullanıcıdan gelen girdi kontrol edilmiş ve beyaz listede yer alan beklenen girdiler dışında gelebilecek girdiler göz ardı edilmiştir. Yani, bir saldırgan ../../../etc/passwd gibi girdiler de dahil olmak üzere 'news.php', 'contact.php', 'team.php' dışında herhangi bir girdi gönderse bile uygulama bunu 'include' fonksiyonuna sokmamaktadır.

2.5. (15 puan) Uygulamanın kurulum dokümantasyonu. Bu dokümantasyonun adımları takip edilerek uygulama sorunsuz bir şekilde kurulabilmelidir. Kurulum dokümantasyonu açık ve anlaşılır olmalıdır.

Kurulum dokümanında aşağıdaki bilgiler **kesinlikle** yer almalıdır:

- Kurulumun yapılabileceği işletim sistemi
- Web sunucu yazılımının kurulumu (Apache, nginx vs.)
- Veritabanı yazılımının kurulumu (MySQL vs.)
- Veritabanının kurulumu (bunun için bir SQL dosyası sağlanmalıdır)
- Uygulamanın kurulumu

2.6. (10 puan)Rapor

TOPLAM PUANLAMA: Her bir zafiyet toplam 25 puan olmak üzere 3 farklı zafiyet için de aynı adımları (2.1, 2.2, 2.3, 2.4) gerçekleyeceksiniz. 3 zafiyet için toplam puan 25*3=75 puandır. Rapor ve dökümantasyon toplam 25 puan olup proje tamamı 100 puan üzerinden değerlendirilecektir. Dökümantasyona ve rapora gerekli özen gösterilmelidir. Aksi halde bu kısımdan verilecek puan beklentinizi karşılayamayacaktır.

İNTİHAL: İNTERNETTEN ALINAN KOD PARÇACIKLARI MUTLAKA KOD İÇERİSİNDE BELİRTİLECEK VE AÇIKLAMA SATIRI İLE KAYNAK GÖSTERİLECEKTİR. AKSİ DURUMDA KOPYA OLARAK DEĞERLENDİRİLECEKTİR. KOPYA ÇEKTİĞİ YA DA KOPYA VERDİĞİ TESPİT EDİLEN ÖĞRENCİLER SUNUMA ALINMAYACAKTIR.

PROJELER İKİ KİŞİLİK GRUPLAR HALİNDE YAPILACAKTIR!

Proje gruplarının oluşturulacağı link:

https://docs.google.com/spreadsheets/d/1dQPAqq1wirYVl4AM1ii5X2IN5UTn1dj5qMbq3A6Z0kE/edit?usp=sharing