Dokuz Eylül Üniversitesi Bilgi İşlem Dairesi

Sunum konusu:

Web Güvenliği

Web Güvenliği

Bugün Türkiye'de önemli kurumların bilgi güvenliği ile ilgili zayıflıkları bulunduğunu görüyoruz. Bilginin güvensiz bir şekilde internet ortamında yayınlanmasının çok ağır sonuçları olabilir. Özlük bilgilerinin, banka hesaplarının, öğrenci notlarının, ticari sırların, e-postaların güvenliği önemli ve oldukça hassas bir konudur.

Zayıflık nedenlerini sınıflandıralım

1. Kullanıcı taraflı zaafiyetler
2. Ağ yönetimi ile ilgili zaafiyetler
3. Uygulama ile ilgili zaafiyetler
4. Sistem Yönetimi ile ilgili zaafiyetler

Genel olarak yapılan hataları bu şekilde sınıflandırabiliriz. Bu sınıflandırmaya ekleme ve çıkarmalar yapılabilir. Daha çok uygulama ve sistem yönetimi ile ilgili hatalar üzerinde duracağız. Sıraladığımız tüm hatalar birbiri ile içiçedir. Zaman kazanma ve bütünlük sağlama açısından uygulama ve sistem yönetimi ile ilgili zayıflıkları birarada ele alacağız.

Kullanıcı taraflı zaafiyetler

Kullanıcı tarafından tercih edilen işletim sisteminin doğru şekilde yönetilememesi sonucunda zararlı yazılımlarla şifre çalınması, oturum bilgilerinin çalınması (trojan, keylogger, spy gibi...).
Belirlenen zayıf şifreler (12345, admin, nimda, 11111, 19831983 gibi).
Aynı şifrenin heryerde kullanılması.
Phishing (Sahte formlar)..

Bugün oneway function yöntemi ile şifrelenmiş geri döndürülemeyeceği zannedilen hash'ler "rainbow table" olarak isimlendirilen tablo yapısı üzerinde çalıştırılan çapraz sorgularla kırılabilmektedir. Bence bu noktada yapılan en kritik hata, zayıf parola seçimleridir.

Ağ Yönetimi ile ilgili zaafiyetler

Ağ trafiğinin dinlenmesi, paketlerin analiz edilmesi...
Arp spoofing, Man-in-the-middle yöntemi gibi..

Bu yöntemlerle http, pop3 gibi protokoller ile gönderilen paketler yakalanabilir...

Uygulama ve sistem yönetimi ile ilgili zaafiyetler

Web uygulamalarından bahsedildiğinde farklı kategoride, kullanılan çok sayıda programlama ve veritabanı alternatifi akla gelmektedir. Perl, Python, Java, PHP, ASP, PostgreSQL, MSSQL, MySQL, Oracle gibi.

Genel olarak benzer mantık hatalarının yapıldığını görüyoruz. Örneklerde PHP programlama dilinden ve MySQL ilişkisel veritabanı yazılımından faydalanılmıştır. Web güvenliği gibi geniş bir konuyu sıraladığımız teknolojilerle sınırlandırmak elbette mümkün değildir.

XSS Cross Site Scripting nedir?

Alınan girdilerin süzgeçten geçirilmeden kullanılması sonucu, saldırgan zararlı içeriklerle programın çalışma şeklini istenmeyen etkiler doğuracak şekilde değiştirebilir.
Örnek senaryo: e-posta web arabirimi gelen e-posta'daki bilgileri süzgeçten getirmeden ekrana basmaktadır.

Directory Traversal nedir?

Kısaca izin verilmeyen dizinlere erişim sağlanması olarak özetlenebilir. Örneğin; apache - php konfigurasyonu yaparken virtual hostlarınızda openbasedir tanımlaması yapmazsanız istenmeyen dizinlerdeki dosyalar için file inclusion sorunu yaşayabilirsiniz. İstenmeyen dizinlerdeki dosyalarınız okunabilir, dizinlerinize dosya yazılabilir.

show_source, include, require, system, shell_exec, passthru, exec gibi fonksiyonlar ile directory traversal zayıflıklarından faydalanabilirsiniz.

File Inclusion LFI - RFI

Local file inclusion ve remote file inclusion olarak iki taraflı düşünülmelidir.

```
<?
include($_GET[“page”]);
?>
```

GET /?page=/etc/passwd
gibi uzaktaki veya yerel bir dosyanın uygulamaya dahil edilmesi söz konusu olabilir.

Arbitrary file upload

Dosya yüklemelerinde yapılan uzantı kontrol hataları akla gelmelidir. Sunucuya çalıştırılabilir, yorumlanabilir dosya yüklenmesi sonucu, sunucunuza sağlanabilecek erişimler ile sisteminiz ele geçirilebilir. CGI, PHP, ASP, PY gibi yorumlanan, çalıştırılabilir dosyaların yüklenmesi engellenmelidir.

Command Injection nedir?

Php programlama dili ile system, shell_exec, passthru, exec gibi fonksiyonlar kullanılarak sistem komutları çalıştırılabilir. Yine asp 'ler için fso (File System Object) ve diğer programlama dillerinde de benzer fonksiyon, methot veya sınıflar mevcuttur.

SQL Injection nedir?

Veritabanı destekli uygulamalarda girişlerin süzgeçten geçirilmemesi; sonrasında, sql komutları kullanılarak tabloların okunması, yazılması, değiştirilmesi... dosyaların okunup, dosyaların yüklenmesi gibi sonuçlar doğurabilecek zayıflıklar akla gelmelidir.

Örnek kod 1 union sorgusu
Örnek kod 2 load_file fonksiyonu
Örnek kod 3 authentication bypass

Kernel intrusion

İşletim sisteminin çekirdeği kullanılarak yapılan saldırılar akla gelmelidir. Linux işletim sistemi için sistemi root'layabileceğiniz çok sayıda local root exploit i mevcuttur.

Kernel intrusion

KERNEL 2.2.*
2.2.x Exploit : ptrace.
KERNEL 2.4.*
2.4.17 Exploits : ptrace, uselib24, ong_bak, mremap, newlocal, brk, brk2.
2.4.18 Exploits : ptrace, uselib24, ong_bak, mremap, newlocal, brk, brk2
2.4.19 Exploits : ptrace, uselib24, ong_bak, mremap, newlocal, brk, brk2
2.4.20 Exploits : ptrace, uselib24, ong_bak, mremap, module_loader, elflbl, brk, brk2
2.4.21 Exploits : ptrace, uselib24, ong_bak, mremap, brk, brk2, w00t
2.4.22 Exploits : ptrace, uselib24, ong_bak, mremap, hatorihanzo, brk, brk2, w00t
2.4.23 Exploit : ptrace, uselib24, ong_bak, mremap
2.4.24 Exploit : ptrace, uselib24, ong_bak, mremap
2.4.25 Exploits : ptrace, uselib24, ong_bak, mremap
2.4.26 Exploits : ptrace, uselib24, ong_bak, mremap
2.4.27 Exploits : ptrace, uselib24, ong_bak, mremap
KERNEL 2.6.*
2.6.2 Exploit : mremap_ptea
2.6.3 Exploit : Krad
2.6.4 Exploit : Krad
2.6.5 Exploit : Krad
2.6.6 Exploit : Krad
2.6.7 Exploit : Krad
2.6.8 Exploit : Krad

Önlemler

*Linux sistem yönetimi ile ilgili tercihler
*Linux partition tablosu ile ilgili yapılabilecek düzenlemeler ve nedenleri
*MySQL kullanıcıları ile ilgili seçimler
*php.ini dosyasının düzenlenmesi
*Apache virtualhost tanımlamaları

Linux sistem yönetimi ile ilgili tercihler

Sistem yöneticisi kesinlikle linux kernel'i için güvenlik yamaları yapmalıdır, dosya dizinler ile ilgili izinler gözden geçirilmelidir.

Linux partition tablosu ile ilgili yapılabilecek düzenlemeler ve nedenleri

Bildiğiniz gibi tmp dizini izinleri herkes tarafından okunabilir ve yazılabilir şekilde düzenlenmiştir. /tmp 'i ayrı bir partition olarak tanımlamak fstab dosyasında no-exec, no-suid olarak bağlamak mantıklı olacaktır. Aksi durumda saldırganın dosya atabileceği bir dizin olarak düşünürülürse dosyaların çalıştırılması kesinlikle sorun yaratacaktır.

MySQL kullanıcıları ile ilgili seçimler

Global yetkileri olan mysql kullanıcısı ile kesinlikle web uygulamları kullanmamanızı öneriyorum. load_file fonksiyonu genel kullanıcılar tarafından kullanılan bir fonksiyondur. Dolayısıyla eğer genel kullanıcı ile web uygulaması geliştiriyorsanız bunun sonuçlarına katlanmak zorunda kalabilirsiniz.

php.ini dosyasının düzenlenmesi

php.ini dosyasında allow_url_fopen off yapmalısınız. Disable_functions bölümüne aşağıda fonksiyonları eklemelisiniz:
disable_functions = show_source, system, shell_exec, passthru, exec, phpinfo, po pen, proc_open

Apache virtualhost tanımlamaları

Örnek virtual host tanımlaması:

```
<VirtualHost 212.174.115.20:80>
        ServerName   blackdaemons.com:80
        ServerAlias  www.blackdaemons.com
        ServerAdmin  "okan@izmirx.com"
        DocumentRoot /home/okan/domains2/blackdaemons.com/httpdocs
        Alias        /webmail /var/www/data/webmail
        DirectoryIndex index.php index.html
        Options -Indexes
        php_admin_value open_basedir "/home/okan/domains2/blackdaemons.com/httpd
ocs:/var/tmp:/usr/local/www/data/webmail"
        CBandLimit 4000M
        CBandExceededURL http://www.izmirx.com/hata/bandwidth_exceeded.html
        CBandScoreboard /home/okan/domains2/blackdaemons.com/httpdocs/scoreboard
        CBandPeriod 4W
<Location /cband-status>
SetHandler cband-status
</Location>
<Location /cband-status-me>
SetHandler cband-status-me
</Location>
<Directory /home/okan/domains2/blackdaemons.com/httpdocs>
  order allow,deny
  allow from all
</Directory>
</VirtualHost>
```

Sorularınız

Hazırlayan: Ali Okan YÜKSEL
E-posta: okan@deu.edu.tr

Dokuz Eylül Üniveritesi
Bilgi İşlem Dairesi
Şubat 2007, İZMİR