YAZILIM GÜVENLİĞİ POLİTİKASI

1. AMAÇ:

Bu doküman Erzincan Üniversitesinde geliştirilmiş ve geliştirilmekte olan tüm uygulamalarında yazılımın her türlü güvenliğinin sağlanması amacıyla tüm yazılım geliştiriciler, veri tabanı ve sistem yöneticileri tarafından uyulması gereken kuralları ve kontrol edilmesi gereken adımları tanımlamak üzere hazırlanmıştır.

2. KAPSAM:

Bu politika, Erzincan Üniversitesi BİDB tarafından geliştirilen uygulamaların (örneğin web sayfaları, masaüstü uygulamaları, web otomasyonları v.b.) geliştirilmesi, bakım ve desteğinin sağlanması gibi faaliyetleri yürüten tüm akademik ve idari personelin uyması gereken kuralları kapsamaktadır.

3. SORUMLULAR:

Üniversitemizin farklı birimlerinde yürütülen internet uygulamaları, bilgi sistemleri alanına giren her konuda akademik ve idari birimlerin gereksinim duyduğu her türlü alt yapı, donanım ve yazılım hizmetleri Bilgi İşlem Daire Başkanlığı tarafından yürütülmektedir.

4. UYGULAMALAR:

Erzincan Üniversitesinde kullanıma açılan yazılım ürünlerinin (uygulamaların) aşağıda sıralanan kategoriler altında yer alan kontrol adımlarının doğrulanmasının sağlanması gerekmektedir. Aksi takdirde uygulamaların güvenliğine dair somut göstergelere sahip olunamayacaktır. Bu kategoriler 9 başlık altında toplanmıştır:

- Veri Koruması
- Hata Yönetimi ve Kayıt Tutma
- Erişim Kontrolü
- Kimlik Sınama
- Oturum Yönetimi
- İletişim Güvenliği
- Girdi Denetimi
- Çıktı Kodlama

4.1. Veri Koruması

Uygulamaların kayıt altına aldığı veya kullandığı her türlü bilginin yetkisiz erişime kapalı olması gerekmektedir. Bu amaç doğrultusunda aşağıdaki adımlar kontrol edilmelidir.

4.1.1. Web, uygulama ve veri tabanı sunucularının sistem bileşenleri hakkındaki kritik bilgiler (sunucu adı ve sürümü, kullanılan program sürümü v.b.) gizlenmelidir.

4.1.2. Uygulama çatısı, veri tabanı, uygulama sunucusu ve web sunucusu gibi kullanılan yazılımların güvenlik yamaları en üst seviyede olmalıdır.

4.1.3. ASP.NET, PHP, STRUTS gibi kullanılan uygulama çatılarının güvenlik özellikleri aktif hale getirilmelidir.

4.1.4. Kullanılan parolalar ve parolamı unuttum kontrol soru ve cevapları gibi diğer hassas veriler açık metin olarak saklanmamalıdır.

4.1.5. Uygulamalar, geliştirme ortamından yapım ortamına aktarılırken gereksiz olan dosyalar (örneğin test kodlar, demo programlar, yedek dosyalar) silinmeli, şayet gerek yoksa kaynak kod aktarılmamalı ve de aktarılacak olan kaynak kodlardaki yorum satırları silinmelidir. Aktarım esnasında dosyalarda istenmeyen bir değişikliğin olmaması garanti edilmelidir.

4.2. Hata Yönetimi ve Kayıt Tutma

Uygulamalar hata aldığında veya beklenmedik bir durum ile karşılaşıldığında, çalışma zamanında üretilen hata mesajlarında teknik detayların ve hatalara ait günlük kayıtlarının hiçbir şekilde son kullanıcıya açılmaması gerekmektedir.

4.2.1. Uygulamada oluşan hatalar ve uygulama sunucusu ön tanımlı hata mesajları kullanıcıya detaylı olarak gösterilmemelidir.

4.2.2. Uygulama üzerinden yapılan kritik işlemler hem uygulama seviyesinde hem de sunucu seviyesinde kayıt altına alınmalıdır.

4.3. Erişim Kontrolü

Uygulamaların yayın yaptığı içerik, dizinler, sunucuların ara yüzleri vb. hiçbir kaynak, kontrol dışı erişilebilir olmamalıdır.

4.3.1. Uygulamaların üzerinde koştukları sunucular, servis verdikleri dizinlerin içeriklerini listelememelidir.
4.3.2. Gerekmedikçe POST/GET dışındaki HTTP metotlarına izin verilmemelidir.
4.3.3. Ana sistem için gereksiz olan dosyalara (örneğin yedekleme, arşiv, test, geliştirme için kullanılan dosyalar) erişim engellenmeli ve sistemdeki gereksiz uygulamalar (örneğin ön tanımlı sunucu sayfaları, demo uygulamalar) kaldırılmalılar.
4.3.4. Flash uygulamalarında crossdomain.xml ve SilverLight uygulamalarında clientaccesspolicy.xml yapılandırma dosyalarında uygulanan politikaların güvenli olup olmadığı kontrol edilmelidir.
4.3.5. Kritik işlemlerde CSRF saldırılarına karşı "token" veya "CAPTCHA" gibi güvenlik önlemleri alınmalıdır.
4.3.6. GET ve POST isteklerindeki HTTP parametreleri değiştirilerek üçüncü şahısların bilgilerine yetkisiz olarak erişilmemelidir.
4.3.7. Uygulamayı çalıştıran sistem kullanıcısının, hizmet verilen dizin dışındaki yetkileri kaldırılmalıdır.
4.3.8. Veri tabanı kullanıcısının sadece uygulamanın kullandığı veri tabanı kaynaklarına erişim hakkı olmalıdır.
4.3.9. Veri tabanı kullanıcısının veri tabanına sadece uygulama sunucu IP adresinden bağlantı hakkı olmalıdır.
4.3.10. Sunucu üzerinde bulunan ve web tabanlı istatistik sağlayan uygulamalara erişim herkese açık olmamalıdır.
4.3.11. Kısıtlı erişim gerektiren bütün URL'lere, fonksiyonlara, obje referanslarına, servislere, uygulama verilerine, kullanıcı bilgilerine, güvenlik yapılandırma dosyalarına erişim denetlenmelidir.
4.3.12. Yetki hakkının artık gerekmediği durumlarda (örneğin şirketi terk etme, projede rol değiştirme gibi) en kısa sürede ilgili haklar iptal edilmelidir.
4.3.13. Yönetim paneli gibi kritik dizinlerin isimleri kolay tahmin edilebilir olmamalıdır (admin, yönetici, administrator, yönetim, panel v.b.).

4.4. Kimlik Sınama

Erişime açılan her kaynak kimlik denetimine tabi tutulma yöntemini de kullanmak zorundadır.

4.4.1. Ön tanımlı kullanıcı hesapları sistemden, veri tabanından ve uygulamadan kaldırılmalıdır.
4.4.2. Umumi olmayan bütün kaynaklara ve sayfalara erişim için sunucu tarafında kimlik doğrulaması yapılmalıdır.
4.4.3. Kullanıcı adı ve parola ile kimlik doğrulamasının yapıldığı kontroller tek tip hata mesajı vermek suretiyle kullanıcı adları listeleme saldırılarına engel olmalıdırlar. Örnek bir hata mesajı "Girdiğiniz kullanıcı adı ve/veya parola yanlıştır." seklinde olabilir.
4.4.4. Bütün başarılı ve başarısız login işlemleri ve kaynaklara erişim denemeleri kayıt altına alınmalıdır.
4.4.5. DoS saldırısı barındıracak veya şifre deneme-yanılma gibi kaba kuvvet saldırılarına açık tüm formlara CAPTCHA veya farklı anti-otomasyon güvenlik kontrolleri uygulanmalıdır.
4.4.6. SOAP, Restful, XML-RPC gibi teknolojilerle geliştirilmiş web servislerine erişimlerde kimlik doğrulama kontrolü uygulanmalıdır.

4.5. Oturum Yönetimi

Kimlik doğrulama sonrası açılan ve bir erişim izninin süresini tanımlayan oturumun yönetilmesinde aşağıdaki kontrol adımlarına uyulacaktır.

4.5.1. Hassas bilgiler içeren web sayfalarının tarayıcılarda belleğe alınmaması için autocomplete, cachecontrol, pragma gibi gerekli HTTP/HTML başlıkları kullanılmalıdır.
4.5.2. Oturum yönetimi için kullanılan ve uygulamayı kullanan bütün kullanıcılar için tekil olması gereken değerlerin (session id, token v.b.) güçlü bir rastgele veri üretecinden temin edildiği ve tahmin edilemez derecede karmaşık olduğu kontrol edilmelidir.
4.5.3. Oturum bilgisi zaman aşımına uğrayacak şekilde yapılandırılmalıdır.
4.5.4. Uygulamalarda başarılı kimlik doğrulama ve tekrarlayan kimlik doğrulama (re-authentication) neticesinde her zaman yeni bir oturum bilgisi oluşturulmalıdır. Çıkış işleminden sonra da var olan oturum bilgisi geçersizleştirilmelidir.
4.5.5. Oturum bilgisini içeren çerezlerin (COOKIE) domain ve yol (path) bilgileri ilgili site için en uygun şekilde sınırlandırılmalıdır.

4.5.6. Kullanılan çerez değerleri için httponly parametresi tanımlı olmalıdır. Buna ek olarak, HTTPS protokolü kullanılan bağlantılarda kullanılan çerez değerleri için secure parametresi tanımlı olmalıdır.

4.5.7. Başarılı login işlemleri sonrası kullanıcı HTTP 302 ile dâhili sayfalara yönlendirilmelidir.

4.5.8. Başarılı kimlik doğrulaması sonucu erişilen uygulamalarda sistemden tekrar çıkmak (logout) için gerekli linkler sağlanmalıdır.

4.5.9. Uygulama domain isimlerine ait hassas bilgilerin google/bing gibi arama motorları tarafından indekslenmediği kontrol edilmelidir.

4.6. Kriptoloji

Güvenliğin tahsis edilmesinde ileri kriptografiden yararlanılmak zorundadır. Aşağıdaki metotlar ve kullanım talimatları yerine getirilmelidir.

4.6.1. Kullanıcılara (zarf, sözlü, e-posta yoluyla) dağıtılan başlangıç parolalar, kullanıcılar uygulamaya ilk giriş yaptıklarında değiştirilmeye zorlanmalıdır.

4.7. İletişim Güvenliği

4.7.1. Uygulama ile son kullanıcı arasında aktarılan kullanıcı adı, parola, kredi kartı no, adres gibi hassas veriler HTTPS protokolü üzerinden aktarılmalıdır.

4.8. Girdi Denetimi

Uygulamalarda kullanıcılara arayüz üzerinden girişi yaptırılan veya sistemin yapısı gereği parametre geçilebilen her türlü bilgi girişi aşağıdaki listede yer alan kontrollerden geçirilmelidir.

4.8.1. Güvensiz kaynaklardan veri alarak aritmetik işlem yapan uygulamalar, gerekli tam sayı üst sınır ve alt sınır kontrollerini gerçekleştirmelidirler. Karşıdan dosya yükleme işlemlerinde yüklenilen dosya üzerinde isim, boyut, tip ve içerik kontrolü yapılmalıdır.

4.8.2. Kullanıcı parametrelerini kullanarak farklı sitelere yönlendirme yapan uygulamalarda ilgili parametrelere pozitif girdi denetimi uygulanmalı ve bu sayede olta saldırılarına engel olunmalıdır.

4.8.3. Kullanıcıdan gelen CR/LF karakterleri uygulama tarafında oldukları gibi HTTP cevap başlıklarında kullanılmamalıdır.

4.8.4. Uygulama hizmete girmeden önce sızma testleri yapılmalıdır.

4.8.5. Genelde uygulamaların arama özelliğini kötüye kullanarak veri tabanı üzerinde çok detaylı arama yaptırarak işlemciyi meşgul eden SQL genel arama karakter (%,* v.b.) saldırılarına karşı arama süresini kısıtlamak suretiyle önlem alınmalıdır.

4.9. Çıktı Kodlama

Uygulamaların ürettiği her türlü yanıt nesnesi (HTML veya Düz Metin) aşağıdaki listede yer alan kontrollerden geçirilmelidir.

4.9.1. Kullanıcıdan gelen veriler işletim sistemi komut satırına girmeden kontrol edilmeli ve düzgünleştirme işleminden (escape) geçirilmelidir.

4.9.2. SQL enjeksiyonuna karşı prepared statement/parameterized query/bind variables/pozitif veri kontrolü yöntemlerinden biri veya birkaçı kullanılmalıdır.

4.9.3. XSS saldırılarına karşı bütün kullanıcı girdileri dışarı aktarılmadan önce sunucu tarafında özel karakter kodlama (output encoding) işleminden geçirilmelidir. Güvenlik seviyesini artırmak için bu işlem kullanıcı girdilerinin tip, uzunluk, içerik denetlemesi yapılarak desteklenebilir.

4.9.4. Kullanıcıdan gelen ve dosya erişim işlemlerinde kullanılan girdiler normalizasyon işlemine tabi tutulmalıdır.

4.9.5. Kullanıcıdan veri alarak LDAP'a bağlanan uygulamalar, gerekli girdi kontrollerini gerçekleştirmeli ve bu girdileri LDAP düzgünleştirme işleminden (escape) geçirmelidir.

4.9.6. Güvensiz kaynaklardan veri alarak XPath sorguları yapan uygulamalar, gerekli girdi kontrollerini gerçekleştirmeli ve bu girdileri XPath düzgünleştirme işleminden (escape) geçirmelidir.

4.9.7. Uygulamalar, uygun olan her sayfada çerçeve engelleyici önlemleri (frame busting) almalıdırlar.

4.9.8. Web servisleri için kullanılan çatıların klasik XML saldırılarına (örneğin cok büyük XML verileri, çok sık tekrarlanan XML tag'leri) ve parametre manipülasyonlarına karşı korunaklı olmaları sağlanmalıdır.