İSTANBUL KÜLTÜR ÜNİVERSİTESİ

BİLGİ SİSTEMLERİ ve TEKNOLOJİLERİ DAİRE BAŞKANLIĞI

GÜVENLİ GELİŞTİRME POLİTİKASI (GGP)

HAZIRLAYANLAR	KONTROL EDEN	ONAYLAYAN
Ufuk Dikme	İsmail Koç	
Bilgi Güvenliği Yönetim Müdürü	Bilgi Sistemleri ve Teknolojileri Daire Başkanı	Rektörlük Temsilcisi

Doküman Kod	IKU-BSTDB-GGP-001	**Revizyon Tarihi**	26.02.2022
Yayın Tarihi	13.10.2020	**Revizyon No**	GGP -001-2.0

İÇİNDEKİLER

1. AMAÇ

Bu politika, T.C. İstanbul Kültür Üniversitesi bünyesinde geliştirilmiş ve geliştirilmekte olan yazılımların güvenliğinin sağlanması için, Bilgi Sistemleri ve Teknolojileri Daire Başkanlığında çalışan yazılım geliştiriciler, veri tabanı ve sistem yöneticilerinin uyması gereken kural ve kontrollerin tanımlanması amacıyla hazırlanmıştır.

2. KAPSAM

Bu politika, T.C. İstanbul Kültür Üniversitesi Bilgi Sistemleri ve Teknolojileri Daire Başkanlığı bünyesinde geliştirilen yazılımların bakımı, geliştirilmesi ve desteğini veren çalışanların uyması gereken kuralları kapsamaktadır.

3. DAYANAK

- ISO/IEC 27001 Bilgi Güvenliği Yönetim Sisteminin Ek.A.14.2.1 maddesi.
- 15.03.2018 tarihli ve 19924119-719-E.21240 sayılı "2016-2019 Ulusal Siber Güvenlik Eylem Planı" konulu YÖK yazısında, üniversitelerin ISO27001 Bilgi Güvenliği Yönetim Sertifikası alması ve iş süreçlerini bu şekilde yapılandırması gerektiği ifade edilmiştir.

4. TANIMLAR VE KISALTMALAR

KISALTMALAR	TANIMLAR
BSTDB	Bilgi Sistemleri ve Teknolojileri Daire Başkanlığı
İKÜ	T.C. İstanbul Kültür Üniversitesi

5. İLGİLİ DOKÜMANLAR

No	İLGİLİ ARAÇLAR
1	IKU BSTDB Bilgi Güvenliği Disiplin Politikası
2	IKU BSTDB Bilgi Güvenliği İhlal Olayları Prosedürü
3	IKU BSTDB Şifre Yönetimi Politikası

6. GÜVENLİ GELİŞTİRME POLİTİKASININ HEDEFLERİ VE PRENSİPLERİ

6.1. Uygulamaların kayıt altına aldığı veya kullandığı her türlü bilginin yetkisiz erişime kapalı olması gerekmektedir.

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

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

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

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

6.6. Uygulamalar, geliştirme ortamından canlı ortama 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.

6.7. 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.

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

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

6.10. Uygulamaların yayın yaptığı içerik, dizinler, sunucuların arayüzleri vb. hiçbir kaynak, kontrol dışı erişilebilir olmamalıdır.

6.11. Uygulamaların üzerinde koştukları sunucular, servis verdikleri dizinlerin içeriklerini listelememelidir.

6.12. Arama motorları tarafından görüntülenmemesi istenen dizinler varsa, bunlar için robots.txt ile önlem alınmalıdır. Yalnız, sayfa içerisinde köprülenmeyen bağlantıların / dizinlerin (örneğin yönetim sayfası) güvenlik sorunu oluşturmaması adına robots.txt dosyasına eklenmemesi gerekmektedir.

6.13. Gerekmedikçe POST/GET dışındaki HTTP metotlarına izin verilmemelidir.

6.14. 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.

6.15. Kritik işlemlerde CSRF saldırılarına karşı "token" gibi güvenlik önlemleri alınmalıdır.

6.16. GET ve POST isteklerindeki HTTP parametreleri değiştirilerek üçüncü şahısların bilgilerine yetkisiz olarak erişilmemelidir.

6.17. Uygulamayı çalıştıran sistem kullanıcısının, hizmet verilen dizin dışındaki yetkileri kaldırılmalıdır.

6.18. Veri tabanı kullanıcısının sadece uygulamanın kullandığı veri tabanı kaynaklarına erişim hakkı olmalıdır.

6.19. Veri tabanı kullanıcısının veri tabanına sadece uygulama sunucu IP adresinden bağlantı hakkı olmalıdır.

6.20. Sunucu üzerinde bulunan ve web tabanlı istatistik sağlayan uygulamalara erişim herkese açık olmamalıdır.

6.21. Kısıtlı erişim gerektiren bütün URL'lere, fonksiyonlara, obje referanslarına, servislere, uygulama verilerine, kullanıcı bilgilerine, güvenlik yapılandırma dosyalarına erişim denetlenmelidir.

6.22. Yetki hakkının artık gerekmediği durumlarda (örneğin şirketi terk etme, projede rol değiştirme gibi) en kısa sürede ilgili haklar iptal edilmelidir.

6.23. Yönetim paneli gibi kritik dizinlerin isimleri kolay tahmin edilebilir olmamalıdır (admin, yönetici, administrator, yönetim, panel vb.).

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

6.25. Ön tanımlı kullanıcı hesapları sistemden, veri tabanından ve uygulamadan kaldırılmalıdır.

6.26. Umumi olmayan bütün kaynaklara ve sayfalara erişim için sunucu tarafında kimlik doğrulaması yapılmalıdır.

6.27. 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.

6.28. Bütün başarılı ve başarısız login işlemleri ve kaynaklara erişim denemeleri kayıt altına alınmalıdır.

6.29. DDoS saldırısı barındıracak veya şifre deneme-yanılma gibi kaba kuvvet saldırılarına açık tüm formlara anti-otomasyon güvenlik kontrolleri uygulanmalıdır.

6.30. SOAP, Restful, XML-RPC gibi teknolojilerle geliştirilmiş web servislerine erişimlerde kimlik doğrulama kontrolü uygulanmalıdır.

6.31. Hassas bilgiler içeren web sayfalarının tarayıcılarda belleğe alınmaması için autocomplete, cachecontrol, pragma gibi gerekli HTTP/HTML başlıkları kullanılmalıdır.

6.32. 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.

6.33. Oturum bilgisi zaman aşımına uğrayacak şekilde yapılandırılmalıdır.

6.34. 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.

6.35. Oturum bilgisini içeren çerezlerin (COOKIE) domain ve yol (path) bilgileri ilgili site için en uygun şekilde sınırlandırılmalıdır.

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

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

6.38. Uygulama domain isimlerine ait hassas bilgilerin Google/Bing gibi arama motorları tarafından indekslenmediği kontrol edilmelidir.

6.39. Güvenliğin tahsis edilmesinde ileri kriptografiden yararlanılmak zorundadır.

6.40. Güvenli web trafiği için (SSL) güçlü şifreleme algoritmaları kullanılmalıdır, güvensiz algoritmalar inaktif hale getirilmelidir.

6.41. SSL sunucusunun "renegotiation" özelliği kapatılarak sunucu servis dışı bırakma ve MITM saldırılarına karşı korunaklı hale getirilmelidir.

6.42. Zayıf parolaların kullanımına izin verilmemelidir.

6.43. Parola hash değerleri oluşturulurken tuz (salt) verisi kullanılmalıdır.

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

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

6.46. Kullanıcıdan gelen tüm girdiler sunucu tarafında pozitif veri kontrolünden geçmelidir. Girdiler veri kontrolünden geçmeden önce "canonicalization" işlemine tabi tutulmalıdırlar.

6.47. Karşıdan dosya yükleme işlemlerinde yüklenilen dosya üzerinde isim, boyut, tip ve içerik kontrolü yapılmalıdır.

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

6.49. Uygulama hizmete girmeden önce sızma testleri yapılmalıdır.

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

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

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

6.53. 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.

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

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

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

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

6.58. Mümkün olduğu sürece, program kaynak kütüphanesi işletimdeki sistemler içinde tutulmamalıdır.

6.59. Program kaynak kodu ve program kaynak kütüphanesi oluşturulmuş prosedürler ile yönetilmelidir.

6.60. Destek personeli, program kaynak kütüphanesine sınırsız erişim yetkisine sahip olmamalıdır.

6.61. Yazılımcılar, program kaynak kütüphanesinin ve ilişkili öğelerin güncellenmesi ve program kaynaklarının yayınlanmasını sadece uygun yetki alındıktan sonra yapmalıdır.

6.62. Program listeleri güvenli bir ortamda saklanmalıdır.

6.63. Program kaynak kütüphanelerine yapılan tüm erişimlerin denetim günlüğü tutulmalıdır.

6.64. Program kaynak kütüphanelerinin sürdürülmesi ve kopyalanması sıkı değişim kontrol prosedürlerine tabi olmalıdır.

6.65. Geliştirme ve işletim yazılımı, farklı etki alanlarında veya dizinlerde, farklı sistem ve bilgisayar işlemcilerinde çalışmalıdır.

6.66. İşletimdeki sistemlerin ve uygulamalarının değişiklikleri işletimdeki sistemlere uygulanmadan önce test ya da hazırlama ortamında test edilmelidir.

6.67. Istisnai durumlar dışında, testler işletimdeki sistemler üzerinde yapılmamalıdır.

6.68. Derleyiciler, editörler ve diğer geliştirme araçları veya sistem programları istenilmediği zaman işletimdeki sistemden erişilebilir olmamalıdır.

6.69. Kullanıcılar, işletim ve test sistemi için farklı kullanıcı profilleri kullanmalıdır ve menüler hata riskini azaltmak için uygun kimlik doğrulama mesajlarını göstermelidir.

6.70. Hassas veriler, test sistemi için eşdeğer kontroller sağlanmadan test sistemi ortamlarına kopyalanmamalıdır.

7. GÜVENLİ GELİŞTİRME POLİTİKASININ YAPTIRIMLARI

Bu politikanın ihlali durumunda, Bilgi Güvenliği Komisyonu ve ilgili yöneticinin onaylarıyla "IKU BSTDB Bilgi Güvenliği Disiplin Politikası" ve "IKU BSTDB Bilgi Güvenliği İhlal Olayları Prosedürü" belgelerinde belirtilen hususlar ve ilgili maddeleri esas alınarak işlem yapılır.

8. REVİZYON BİLGİSİ

Revizyon Numarası	Yürürlük Tarihi	Revizyon Tanımı	Hazırlayanlar

GGP-001-2.0		6.6 Uygulamalar, geliştirme ortamından ~~prodüksiyon ortamına~~ (canlı ortama) 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.	Ufuk Dikme