

TASNİF DIŐI



**TÜBİTAK BİLGEM
KAMU SERTİFİKASYON MERKEZİ**

EYP UYUM DEĞERLENDİRME ÖN HAZIRLIK REHBERİ

Doküman Kodu

REH.05.02

Revizyon No

03

Revizyon Tarihi

01.06.2018

TASNİF DIŐI

REVİZYON GEÇMİŐI		
Revizyon No	Revizyon Nedeni	Revizyon Tarihi
00	İlk Çıkıő.	05.06.2017
01	Arayüz ile ilgili kontroller eklenmiőtir.	15.10.2017
02	Güvenilir algoritmalar için beyaz liste kavramı eklendi. İmzanın sunucuda yükseltilmesiyle ilgili madde eklendi. Doküman kodu güncellendi.	08.12.2017
03	Doküman biçimi yeni Őablona aktarılmıőtir. Doküman kodu güncellenmiőtir. Dokümanın eski revizyonları Kamu SM doküman yönetim sisteminde "REH-001-019" kodu ile yer almaktadır.	01.06.2018

İÇİNDEKİLER

1	<i>Amaç ve Kapsam</i>	3
2	<i>Kısaltmalar</i>	4
3	<i>İmza Oluřturma Testleri Ön İsterleri</i>	5
4	<i>İmza Doğrulama Testleri Ön İsterleri</i>	9
5	<i>Uyum Deđerlendirme Test Süreci Hakkında Bilgilendirmeler</i>	11
6	<i>Ek-A İmza Arřivleme Rehberi</i>	12

1 Amaç ve Kapsam

Bu doküman, e-imza uyum deđerlendirme çalıŐması öncesi kuruma gönderilen test paketinin içeriđi hakkında bilgi vermek ve kurum tarafından ön hazırlık olarak yerine getirilmesi gereken Őartları belirtmek için oluşturulmuŐtur. Uyum deđerlendirme çalıŐması format kontrolü, imza oluŐturma, dođrulama ve arŐivleme testleri olmak üzere dört bölümden oluŐmaktadır. Testlerin ayrıntılı biçimde yapılabilmesi için Kamu SM tarafından hazırlanmış olan Kamu SM Test Suit çalıŐması kullanılmaktadır.

Kamu kurumlarının Elektronik Belge Yönetim Sistemlerinin (EBYS), Elektronik YazıŐma Paketi (EYP) oluŐturması/dođrulaması durumunda EYP e-imza uyum deđerlendirme çalıŐması yapılacaktır. Bu çalıŐma, Elektronik YazıŐma Paketi'ndeki elektronik imzanın uluslararası standartlara uygunluđunun kontrolünden ibarettir. İmza uygulamasının oluŐturduđu imzalı dosya tiplerinin CAdES ES X-Long veya ES-A olması gerekmektedir.

EYP e-imza uyum deđerlendirme çalıŐması kapsamında EYP'nin temel bileŐenlerinin kontrolü yapılmaktadır. Kalkınma Bakanlığı'nca yayımlanan "e-YazıŐma Teknik Rehberi"nde yer alan kontrol listesi kapsam dıŐındadır.

Bu doküman;

CWA 14170: Security Requirements for Signature Creation Applications (İmza OluŐturma Uygulamaları için Güvenlik Gereksinimleri),

CWA 14171: Procedures for Electronic Signature Verification (Elektronik İmza Dođrulama için Prosedürler) ,

ETSI TS 101 733: Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES),

T.C. Kalkınma Bakanlığı e-YazıŐma Teknik Rehberi standartları referans alınarak hazırlanmıştır.

2 Kısaltmalar

CADES:	CMS Advanced Electronic Signature - CMS Gelişmiş Elektronik İmza
CMS:	Cryptographic Message Syntax- Kriptografik Mesaj Sözdizimi
CWA:	CEN (Comité Européen De Normalisation) Workshop Agreement-CEN Çalıştay Kararları
DED:	Dahili Elektronik Doküman
EBYS:	Elektronik Belge Yönetim Sistemleri
ES-A:	Archival Electronic Signature - Arşiv Elektronik İmza
ES X-Long:	EXtended Long Electronic Signature - Genişletilmiş Uzun Elektronik İmza
ETSI:	European Telecommunications Standards Institute-Avrupa Telekomünikasyon Standartları Enstitüsü
EYP:	Elektronik Yazışma Paketi
Kamu SM:	Kamu Sertifikasyon Merkezi
XML:	Extensible Markup Language - Genişletilebilir İşaretleme Dili
Zaman Damgası:	E-imza mevzuatında tanımlanan Zaman Damgası

3 İmza OluŐturma Testleri Öm İsterleri

1. TÜBİTAK API kullanan uygulamalar API'nin son sürümünü ve son sürümü barındıran paketten çıkan politika dosyasını kullanmalıdır. API'nin son versiyonuna <https://yazilim.kamusm.gov.tr/> adresinden ulaşabilirsiniz. Eski sürüm API kullanan uygulamalar değerdendirilmeyecektir.
2. Testlerde Kamu SM Test Suit'in kullanılabilmesi için size gönderilen, RootCerts.rar klasöründe bulunan test köklerinin uygulamanın güvenli kökler dizinine eklenmesi gerekmektedir. TÜBİTAK API kullananlar ilgili politika dosyası içerisinde aŐađıdaki düzenlemeyi yaparak bu hazırlığı tamamlayabilirler.

```
<trustedcertificate>
  <class
    name="tr.gov.tubitak.uekae.esya.api.certificate.validation.
    find.certificate.
    trusted.TrustedCertificateFinderFromFileSystem">
    <param name="dizin" value="D:\Trusted\" />
  </class>
  <class
    name="tr.gov.tubitak.uekae.esya.api.certificate.validation.
    find.certificate.trusted.TrustedCertificateFinderFromECertSt
    ore">
    <param name="securitylevel" value="legal" />
  </class>
</trustedcertificate>
```

Politika dosyasında ayrıca aŐađıdaki ayarların yapılması gerekmektedir.

```
Bu kısım yorum satırına çekilmelidir.

<!--class name =
"tr.gov.tubitak.uekae.esya.api.certificate.validation.find.crl.CRLFinderFromECertStore"/>

<class name =
"tr.gov.tubitak.uekae.esya.api.certificate.validation.find.crl.CRLFinderFromECertStore">
<param name = "getactivecrl" value="true"/>
</class -->
```

3. Doğrudan uzun dönemli imza oluşturulduđu takdirde zaman damgası alırken sistemsel bazı aksaklıklar oluşabileceğinden imza oluŐturma işlemi gerçekteşemeyebilir. Kullanıcı tarafında basit elektronik imza oluşturulduktan sonra sunucuda uzun dönemli imzaya çevrilerek bu aksaklıkların önüne geçilmesi tavsiye edilmektedir.
4. İmza oluŐturma testleri, sadece imzanın formatının yukarıda belirtilen uluslararası standartlara uygunluğunun kontrolünü değil sertifika doğrulama kontrollerini de içermektedir. Bu nedenle imza uygulaması bu kontrolleri yerine getirmelidir.
5. Sertifika doğrulama, yanlış PIN girme, bloke olmuş kartla imzalama ve benzeri hatalı durumlarla ilgili kullanıcı bilgilendirmelerinde standart hata kabul edilmeyecektir. Kullanıcı bilgilendirmeleri hatayı net ifade edecek şekilde olmalıdır.

6. İmza oluŐturma esnasında, birden fazla sertifikanın bulunduĐu kartlarda, kullanıcıya imza oluŐturma iŐlemi için karttaki sertifikaları seĐme hakkı verilmeli ve **sadece nitelikli sertifikalar gösterilmelidir**. Sertifika seĐim ekranı akıllı kartın çıkarılması, yeni kart takılması vb. durumlarda yenilenmelidir. (Bu iŐlem, yenileme butonu, kart okuyucunun sürekli olarak dinlenmesi gibi yöntemler ile yapılabilir.)
7. Uygulamanın, sertifika içeriĐini kullanıcı istediĐi takdirde gösterecek Őekilde geliŐtirilmesi gerekmektedir. Uygulama sertifikayı, iŐletim sisteminin sertifika görüntüleyicisi vasıtasıyla da gösterebilir.
8. İmza oluŐturma iŐleminin yapıldıĐı ekrana, "**Bu imza 5070 Sayılı Elektronik İmza Kanunu'na göre güvenli elektronik imzadır.**" ibaresi eklenmelidir. Kullanıcıya imzalamadan vazgeĐme seĐeneĐi sunulmalıdır.
9. Kullanıcıya imza oluŐturma esnasında, imzaladıĐı içeriĐi deĐiŐtirilemez bir Őekilde görüntüleme imkânı verilmelidir.
10. İmzalanmasına izin verilen belge türleri, Kalkınma Bakanlığı tarafından yayımlanan Birlikte Çalışabilirlik Esasları Rehberi'nin güncel sürümünün "2.1. Dosya Sunumu ve DeĐiŐimi" bölümünde tanımlanmaktadır. Metin Tabanlı Belgeler için PDF/A, Sayısal Grafik ve Diyagramlar için GIF, Sayısal FotoĐraflar için JPEG belge türlerine izin verilmektedir. Bunlar dıŐındaki belge türlerinin imzalanması uygulama tarafından kabul edilmemelidir.
11. Uygulama tarafından imzalanmasına izin verilen belge türü olarak PDF/A-1 ve PDF/A-2 belgeleri doĐrudan kabul edilebilirken; PDF/A-3 belgeleri PDF/A uyumlu olmayan ek içerebildiĐinden uygulamanın aŐaĐıdaki yöntemlerden birini tercih etmesi gerekmektedir:
 - a. Belge formatı PDF/A-3 olduĐu için imzalanmasına izin vermemelidir.
 - b. İmzalanacak belgenin PDF/A dıŐında ek içerdini tespit etmeli ve imzalanmasına izin vermemelidir.
12. Uygulamada imzacının PIN bilgisinin saklanması son derece tehlikelidir. PIN bilgisinin ve son kullanıcı bilgisayarındaki hakların saldırganlar tarafından ele geĐirilmesi halinde kiŐi adına imza oluŐturulabilir. Bu sebeple PIN bilgisinin saklanması tavsiye edilmez.
13. PIN bilgisi saklandıĐı takdirde imza oluŐturma uygulamasının, karta "log in" olduktan belirli bir süre sonra (maksimum 30 dakika) "log out" olması gerekmektedir. Uygulamanın bir kez "log in" olup, sınırsız imzalamaya izin vermesi kabul edilmemektedir. Güvenlik aĐısından önerilen metod, imzalama iŐlemi bittikten sonra hemen "log out" olunmasıdır.
14. Sertifika seĐimi yapılmadan PIN giriŐine izin verilmemelidir. İmza oluŐturma uygulamasındaki PIN girme alanının, PIN girilmeye baŐladıktan bir süre sonra imzalama iŐlemi bitirilmediĐi takdirde temizlenmesi gerekmektedir. PIN alanını temizlemek için bekleme süresi maksimum 30 saniyedir.
15. Uygulama arayüzünde imzalı ve imzasız belgeler ayırt edilebilir olmalıdır. Belgenin niteliĐine göre ilgili seĐenekler gösterilmelidir.

16. Kullanıcı, imzalama işleminden sonra imzanın oluşup oluşmadığına dair anlaşılır bir şekilde bilgilendirilmelidir.
17. Uygulama, imza oluşturma aşamasında zaman damgası alırken, zaman damgasının geçerlilik kontrolünü yapmalıdır. TÜBİTAK API kullananlar bu özelliđi aktive etmek için imza oluşturma kodundaki parametrelere aşağıda belirtilen eklemeyi yapmalıdır:

CADES	<code>params.put (EParameters.<i>P_VALIDATE_TIMESTAMP_WHILE_SIGNING</i>, true);</code>
--------------	--

Testte kullanılacak zaman damgası sunucularının erişim bilgileri aşağıda verilmiştir.

Kullanıcı Adı: 1		Şifre: 12345678 (Tüm hesaplar için kullanıcı adı ve şifre aynıdır.)	
TSA1: http://zdsA1.test2.kamusm.gov.tr	TSB: http://zdsB.test2.kamusm.gov.tr		
TSA2: http://zdsA2.test2.kamusm.gov.tr	TSC1: http://zdsC1.test2.kamusm.gov.tr		
TSA3: http://zdsA3.test2.kamusm.gov.tr	TSC2: http://zdsC2.test2.kamusm.gov.tr		
TSA4: http://zdsA4.test2.kamusm.gov.tr	TSC3: http://zdsC3.test2.kamusm.gov.tr		
TSA5: http://zdsA5.test2.kamusm.gov.tr			

Sunucu erişim bilgilerinin kod içeriğinden ayarlanması ve uyum değerlendirme testlerinden sonra deđişmesi durumunda uygulama özet deđeri bozulacaktır. Bu nedenle zaman damgası sunucu erişim bilgilerinin koddan bağımsız bir şekilde konfigüre edilmesi tavsiye edilmektedir.

18. EYP’de imzaya dahil olan ve olmayan bileşenler ile ilgili kullanıcı bilgilendirilmelidir.
19. BelgelMza.xml bileşeni e-Yazışma Teknik Rehberi’ne göre zorunlu bir bileşen deđildir. Paket içerisinde yer aldığı durumda da imzaya dahil edilemediğinden deđiştirilmeye açıktır. EYP imza dođrulama uygulamasının imzacı bilgilerini imza içerisinden deđil, bu bileşenden alarak göstermesi durumunda hata oluşabileceğinden, BelgelMza.xml bileşeninin kullanılması tavsiye edilmemektedir.
20. NES içeriğinde maddi limit bilgisi olduđu takdirde sertifika dođrulama yapılırken aşağıdaki yöntemlerden biri izlenmelidir:

- Sertifikada bulunan maddi limit ile belgenin maddi içeriğinin karşılaştırılmadığı durumda, kullanıcıya imzacı sertifikasında maddi limit bilgisi olduđu uyarısı verilmeli ve imzanın oluşturulup oluşturulmayacağı EBYS uygulaması politikalarına göre belirlenmelidir.
- Sertifikada bulunan maddi limit ile belgenin maddi içeriğinin karşılaştırıldığı durumda:
 - Sertifika maddi limiti belge maddi deđeri için yeterli ise imza oluşturulmalıdır.
 - Sertifika maddi limiti belge maddi deđeri için yeterli deđil ise imza oluşturulmamalıdır.

21. Uygulama toplu imzalamayı destekliorsa aŐağıdaki maddeler saĐlanmalıdır:

- a. Toplu imzalama sürecinde kullanıcıya özgü toplu imzalama listesi oluşturulabilmelidir. Toplu imzalama listesi imzalanacak belgelerden veya belge süreçlerinden oluşacak şekilde tasarlanmalıdır. Belge süreci, aynı belgeye ait ek, nüsha vb. bileŐenlerin tamamıdır.
- b. Kullanıcı imzalanacak belgeyi detaylı görüntüleyebilmeli ve toplu imzalama listesine eklemek isterse, detaylı görüntüleme ekranından ekleyebilmelidir.
- c. Toplu imzalama listesine eklenen belgeler kullanıcı hatasına mahal verecek şekilde toplu olarak seçilememelidir.
- d. Toplu imzalama listesi görüntüleme ekranında listedeki belgelerin adı, konu başlığı ve listeye eklenme tarihi görülmelidir.
- e. Uygulamada boŐta bekleme süresi (maksimum 30 dakika) belirlenmelidir. Bu süre boyunca işlem yapılmadığında toplu imzalama listesi inaktif olmalıdır.
- f. Uygulamada güvenli bekleme süresi maksimum 8 saat olarak belirlenmelidir. Bu süre sonunda toplu imzalama listesi sıfırlanmalıdır.

22. Uygulama arŐivleme modulüne sahip olmalıdır. ArŐivleme modülü Ek-A'da verilen "İmza ArŐivleme Rehberi"nde belirtilen talimatlara uygun olarak geliştirilmelidir.

4 İmza Doğrulama Testleri Ön İsterleri

1. Talep edilen imza türüne göre göndermiş olduğumuz imzalı dosya paketini EBYS uygulamasına dâhil etmeniz ve doğrulama testleri için hazır hale getirmeniz gerekmektedir.
2. Uygulama arayüzünde imzalı ve imzasız belgeler ayırt edilebilir olmalıdır. Belgenin niteliğine göre ilgili seçenekler gösterilmelidir.
3. Uygulama, doğrulanacak imzalı belgenin seçilmesine ve imzalanan içeriğin gösterilmesine imkân vermelidir.
4. İmza doğrulama ekranında imzacının isim bilgisi, imza zamanı ve imzanın genel doğrulama sonucu açıkça belirtilmelidir. Kullanıcı istediği takdirde imzacının sertifika bilgilerinin ve doğrulama sonuçlarının tamamını ayırt edilebilir şekilde görüntüleyebilmelidir. Uygulama sertifikayı işletim sisteminin sertifika görüntüleyicisi vasıtasıyla da gösterebilir.
5. Seri/Paralel imzacılar, doğrulama ekranında hiyerarşik bir düzende ağaç yapısına benzer şekilde gösterilmeli ve imzalar doğrulanırken kullanıcıyı net bir şekilde bilgilendirecek genel doğrulama sonucu ve imzacıların ayrı ayrı doğrulama sonuçları yer almalıdır.
6. İmza doğrulama sonuçları kullanıcıya anlaşılır şekilde gösterilmelidir. Kullanıcı bilgilendirmelerinde standart hata dönülmemeli, bilgilendirmeler hatayı net ifade edecek şekilde olmalıdır.
7. İmzaya dahil olan imza özelliklerinden "ContentHints" imza özelliği ile imzalanan dosya türü karşılaştırılmalıdır. Eşleşmediği durumlarda imza doğrulanmamalıdır ve kullanıcı bilgilendirilmelidir.
8. TÜBİTAK API kullananlar, imza doğrulama kodundaki parametrelere aşağıda belirtilen eklemeleri yapmalıdır:

CADES (Native API)	<pre>params.put(EParameters.P_FORCE_STRICT_REFERENCE_USE, true);</pre>
CADES (Common API)	<p style="color: red;">esya-signature-config.xml'de params içerisine aşağıdaki parametre eklenmelidir.</p> <pre><!--loosening below 2 settings will cause warnings instead of validation failure--> <!--referenced validation data must be used for cert validation is set true--> <force-strict-reference-use>true</force-strict-reference-use></pre>

9. Uygulamada geçersiz imza özet algoritması kontrolü yapılmalıdır. Bu sebeple uygulamanın kabul edilen özet algoritmaları listesi olmalıdır ve bu listede BTK'nın 13 Temmuz 2017 tarihli Resmi Gazete'de yayımlanan "ELEKTRONİK İMZA İLE İLGİLİ SÜREÇLERE VE TEKNİK KRİTERLERE İLİŐKİN TEBLİĐDE DEĐİŐİKLİK YAPILMASINA DAİR TEBLİĐ" in 1. maddesinde belirtilen özetleme algoritmaları bulunmalıdır. "Uygulama, doğrulama aşamasında imza algoritması içerisinde izin verilen algoritmalar dışında bir algoritma tespit ettiğinde, eđer imza arşivlenmemişse, "Geçersiz Özet Algoritması, İmza En Kısa Sürede Arşivlenmelidir" uyarısı dönmelidir. İmzanın statüsü geçerliyse, özet algoritması uyarısı sebebiyle geçersiz hale getirilmemelidir.
10. Üstveri, Üst Yazı, BelgeHedef ve DED ekleri bileşenlerinin özetlerinin PaketOzeti bileşeni içerisinde doğru bir şekilde yer aldığı kontrol edilmelidir.
11. PaketOzeti bileşeninin imzalandığı kontrol edilmelidir.
12. Dağıtım listesinde bulunan kuruma özgü ekler kontrol edilmelidir. Dağıtım listesinde konulmamış ek listesinde yer alan eklerin kontrolü yapılmamalıdır.
13. İmzacı bilgilerinin imza içerisinde alınması gereklidir.
14. İmzacı sertifikası içeriğinde maddi limit bilgisi varsa aşağıdaki yöntemlerden biri izlenerek imza kontrol edilmelidir:
 - a. Sertifikada bulunan maddi limit ile belgenin maddi içeriğinin karşılaştırılmadığı durumda, kullanıcıya imzacı sertifikasında maddi limit bilgisi olduğu uyarısı verilmeli ve imzanın doğrulanıp doğrulanmayacağı EBYS uygulaması politikalarına göre belirlenmelidir.
 - b. Sertifikada bulunan maddi limit ile belgenin maddi içeriğinin karşılaştırıldığı durumda:
 - i. Sertifika maddi limiti belge maddi değeri için yeterli ise imza doğrulanmalıdır.
 - ii. Sertifika maddi limiti belge maddi değeri için yeterli değil ise imza doğrulanmamalıdır.

5 Uyum Deđerlendirme Test Süreci Hakkında Bilgilendirmeler

Deđerlendirmeye alınan kuruluşlar tarafından, yukarıda belirtilen maddelerin uyum deđerlendirme çalışması öncesi tamamlanması gerekmektedir. Bu çalışmalar sonrasında uyum deđerlendirme süreci başlayacaktır. Bu maddeler uyum deđerlendirme sürecinin temel maddeleridir, deđerlendirme sürecinde bu maddeler dışında farklı kontroller de yapılacaktır.

Ön Hazırlık çalışması tamamlandıktan sonra kuruma randevu tarihi verilir. Randevu tarihi uyum deđerlendirme hizmetinin resmi olarak başladığı tarihtir. Test süreci ilk randevu tarihinden itibaren en fazla 50 iş günü içerisinde tamamlanmış olmalıdır, aksi takdirde sürecin bedeli kurum tarafından ödenerek yeniden başlatılması gerekmektedir.

EYP Uyum Deđerlendirme testleri dört ana bölümden oluşmaktadır. İlk bölüm olan Format Testi'nde, uygulama kullanılarak oluşturulmuş imzalı dosyaların ilgili standartlarda belirtilen imza formatlarına uygunluğu kontrol edilir. Format Testi'nden sonraki bölüm olan İmza Oluşturma Testi'nde, sertifika ve zaman damgası doğrulama ile ilgili kontroller yapılarak güvenli elektronik imza oluşturma yazılımının uygunluğu test edilir. İmza Doğrulama Testi'nde imza doğrulama testleri yapılarak güvenli elektronik imza doğrulama yazılımının uygunluğu test edilir. Son bölüm olan İmza Arşivleme Testi'nde ise arşivleme uygulamasının deđerlendirilmesi yapılmıştır.

Yukarıda ana bölümleri belirtilen Uyum Deđerlendirme test maddelerinin uygulamada sağlanmadığı tespit edilirse, kuruma eksikliklerini tamamlaması, uygulamadaki hataları düzeltmesi için zaman tanınır. Kurum hataları düzelttiğini belirttikten sonra tekrar randevu tarihi verilerek test sürecine devam edilir. Test yapılan uygulama, EYP Uyum Deđerlendirme Raporu'nda belirtilen zorunlu olarak sağlanması gereken maddelerin tamamını sağlayana kadar, kuruma hataları iletme, randevu verme ve test sürecine devam etme işlemleri tekrarlanır. EYP Uyum Deđerlendirme Raporu'nda belirtilen tüm zorunlu maddeler sağlandığında, Uyum Deđerlendirme testleri tamamlanmış olur.

Uyum Deđerlendirme testleri tamamlandıktan sonra uygulamanın arayüz, politika, imza oluşturma ve imza doğrulama kısımlarına ait SHA-256 özet deđerlerinin 1 (bir) iş günü içinde, Uyum Deđerlendirme Taahhütname Formu'nda belirtilmesi ve bu formun ıslak imzalı ve kaşeli halinin tarafımıza iletilmesi gerekmektedir. Aksi takdirde süreç yenilenecektir. Kurumun Uyum Deđerlendirme Taahhütname Formu'nda belirtmiş olduğu özet deđerleri, Uyum Deđerlendirme Raporu'na yazılır ve raporun onaylanmasının ardından www.kamusm.gov.tr internet sitesinden ilan edilir.

6 Ek-A İmza Arşivleme Rehberi

Arşiv imza, e-imzalı belgelerin sertifika makamına ait kök/alt kök, OCSP ve zaman damgası sertifikalarının geçerlilik süresinden daha uzun bir süre saklanması gerektiđi durumlarda kullanılması gereken imza tipidir.

Arşivleme, sertifika makamına ait sertifikaların geçerlilik süresinin sonuna yaklaşıması, sertifikaların iptal olması veya kullanılan algoritmaların geçerliliđini yitirmesi durumlarında yapılır. Arşivlemenin yukarıdaki durumlar oluşmadan önce yapılmasında da bir sakınca yoktur. Arşivleme, hali hazırda arşiv tipindeki imzalı dosyaların içindeki son arşiv zaman damgasının geçerliliđi tehlikeye girdiđi takdirde, ESHS tarafından yeni bir hiyerarşiden yayınlanmış zaman damgası ayarları girilerek tekrarlanmalıdır. Uygulama tarafında ise ilgili altyapı sağlanmış olmalıdır.

Aşağıdaki bölümde arşivleme ihtiyacı gerektirecek senaryolar belirlenmiştir.

Arşivleme Senaryoları:

1. Sertifika ile ilgili senaryolar
 - a. OCSP sertifikasının iptal olma ve süresinin dolma durumu
 - b. İmza ZD sertifikasının iptal olma ve süresinin dolma durumu
 - c. "İmza ve referans zaman damgası" ya da "referans zaman damgası" sertifikasının (eđer imzada varsa) iptal olma ve süresinin dolma durumu
 - d. Arşiv ZD sertifikasının iptal olma ve süresinin dolma durumu
 - e. Alt kök sertifikasının iptal olma ve süresinin dolma durumu
 - f. Kök sertifikasının süresinin dolma durumu
2. Güvenilir kök deposu deđişiklikleri ile ilgili senaryolar
 - a. Kök Sertifikasının kara listeye girme durumu
3. Algoritma geçersizlikleri ile ilgili senaryolar
 - a. İmza zaman damgası algoritmasının geçersiz olma durumu
 - b. "İmza ve referans zaman damgası" ya da "referans zaman damgası" (eđer imzada varsa) algoritmasının geçersiz olma durumu
 - c. Arşiv zaman damgası algoritmasının geçersiz olma durumu
 - d. İmza algoritmalarının geçersiz olma durumu

Senaryoları sağlamak adına kurulması istenen işleyiş aşağıda anlatılmakta ve sözde (pseudo) kodu verilmektedir.

1. Sistemin arka planında çalışacak uygulama kullanıcıdan bağımsız olarak toplu işlem (batch process) yapmalıdır.
2. İçerisine güvenilirliğini yitirmiş kök sertifikaların özet deđerlerinin eklenebileceđi kara liste (blacklist) yapısı kurulmalıdır.
3. Güvenli algoritmaların eklenebileceđi beyaz liste (whitelist) yapısı kurulmalıdır.

CADES API'de arşivleme işleminin aşağıda yazan sözde koda göre gerçekleşmesi tavsiye edilmektedir.

```
//Arşiv kontrolünden geçecek imzalar toplanır ve tek tek kontrolden geçirilir.
List<Signature> signatureFileList = getAllSignatures();

foreach(Signature s in signatureFileList){
    //Öncelikle imza doğrulaması yapılır, imza doğrulanmadığı takdirde arşivlenmez ve hata loglanır.
    //imza doğrulama işleminde algoritmalar için whitelist ve sertifikalar için blacklist kontrolü
    yapılmaz.
    if (verifySignature(s)) {

        //İlk seviyedeki bütün paralel imzacılar alınır.
        List<Signer> parallelSigList = s.getParallelSignerList();

        //Belirlenen paralel imzacıların imzaları tek tek kontrol edilir.
        foreach (Signer parallelSig in parallelSigList) {

            //Paralel imzanın tipi alınır.
            SignatureType parallelSigType = parallelSig.getType();

            //Kontrol edilen paralel imzacının bütün alt imzacıları DFS(Depth First Search)
            //veya BFS(Breadth First Search) ile alınır.
            //Paralel imzacının kendisi listeye dahil değildir.
            List<Signer> subSignerList = getAllSubSigners(parallelSig);

            //Listede tipi XLONG olmayan imza varsa önce paralel imzacı tipi Arşivse XLONG tipine
            çevrilir.
            //Sonra paralel imzacıya ait alt imzalardan XLONG tipinde olmayan imzalar XLONG tipine
            çevrilir.
            foreach (Signer subSigner in subSignerList) {
                if (subSigner.getType() != SignatureType.ESXLong) {

                    //Paralel imza tipi Arşivse XLONG tipine çevrilir.
                    if (parallelSigType == SignatureType.ESA) {
                        downgradeToXLong(parallelSig);
                        parallelSigType = SignatureType.ESXLong;
                        log("Arşiv imza XLONG tipine çevrildi.");
                    }

                    //Alt imza XLONG tipine çevrilir.
                    upgradeToXLong(subSigner);
                    log("Alt imza XLONG tipine çevrildi.");
                }
            }

            //Paralel imza arşiv tipinde değilse arşivlenir.
            if (parallelSigType != SignatureType.ESA) {
                archive(parallelSig);
                log("Paralel imza arşiv tipinde olmadığı için arşivlendi.");
                continue;
            }

            //Son arşiv zaman damgasının algoritmaları alınır.
            List<String> lastArchiveTSAlgorithmList = getLastArchiveTSAlgorithms(parallelSig);

            //Geçerli algoritma listesi alınır. Eğer yakın zamanda geçersiz olacak algoritma varsa
            //bu listeden çıkarılmalıdır.
            //Son arşiv zaman damgasındaki algoritmalar geçerli algoritma listesiyle karşılaştırılır.
            boolean isAlgorithmInWhiteList = isAlgorithmInWhiteList(lastArchiveTSAlgorithmList);

            //Son arşiv zaman damgasında geçersiz algoritma varsa imza arşivlenir.
            if (!isAlgorithmInWhiteList) {
                archive(parallelSig);
                log("Son arşiv zaman damgasında geçersiz algoritma bulunması sebebiyle imza yeniden
                arşivlendi.");
                continue;
            }

            //Son arşiv zaman damgası sertifikası alınır.

```

```

Certificate lastArchiveTSCertificate = getLastArchiveTSCertificate(parallelSig);

//Son arşiv zaman damgası sertifika süresinin dolmasına 2 aydan az kaldıysa yeni arşiv
zaman damgası
//ayarları yapılır ve imza arşivlenir.
Date certificateExpirationDate = getCertificateExpirationDate(lastArchiveTSCertificate);
if (certificateExpirationDate < Date.now + 2 months) {
    newArchiveTsSettings();
    archive(parallelSig);
    log("Sertifika süresinin dolmasına 2 aydan az kaldığı için imza yeniden
arşivlendi.");
    continue;
}
//Son arşiv zaman damgası sertifikası doğrulanamazsa yeni arşiv zaman damgası ayarları
yapılır ve
//imza arşivlenir.
if (!verifyCertificate(lastArchiveTSCertificate)) {
    newArchiveTsSettings();
    archive(parallelSig);
    log("Sertifika doğrulanamadığı için imza yeniden arşivlendi.");
    continue;
}

//Son arşiv zaman damgası kök sertifikası alınır.
Certificate lastArchiveTSRootCertificate =
getLastArchiveTSRootCertificate(lastArchiveTSCertificate);
//Son arşiv zaman damgası kök sertifikasının kara listede olup olmadığına bakılır. İptal
olan
//kök sertifikalar öncelikle bu listeye eklenmiş olmalıdır.
boolean isInRootCertificateBlackList =
isInRootCertificateBlackList(lastArchiveTSRootCertificate);

//Son arşiv zaman damgası kök sertifikası kara listedeyse yeni arşiv zaman damgası
ayarları yapılır
//ve imza arşivlenir.
if (isInRootCertificateBlackList) {
    newArchiveTsSettings();
    archive(parallelSig);
    log("Son arşiv zaman damgası kök sertifikası kara listede olduğu için imza yeniden
arşivlendi.");
    continue;
}

log("İmzanın arşivlenmesine gerek yok.");
}
} else
log("İmza doğrulanamadığı için arşivlenmedi.");
}

```

Arşivleme testlerinin tarafımızca yapılabilmesi için toplu işlemi (batch process) tetikleyip logların alınabileceği ve sonrasında arşivlenen dosyaların indirilebileceği geçici basit bir arayüz yapılmalıdır. Arayüz, kara listeye eklenen kök sertifikalarını ve beyaz listeye eklenen özet algoritmalarını görmeye imkan vermelidir.