

Integrační příručka ASiC Factory .NET, v1.3

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

Copyright

Všetky práva vyhradené

Tento dokument je vlastníctvom spoločnosti DITEC, a. s. Žiadna jeho časť sa nesmie akýmkoľvek spôsobom (elektronickým, mechanickým) poskytnúť tretej strane, rozmnožovať, kopírovať, vrátane spätného prevodu do elektronickej podoby, bez písomného povolenia spracovávateľa.

Popisné charakteristiky dokumentu

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Podnázov	ASiC Factory .NET, v1.3	
Ref. číslo	GOV_ZEP.253	Verzia 8

Vypracoval	Mikuš Michal	Podpis	Dátum 19. 2. 2025
Preveril	Priezvisko Meno	Podpis	Dátum
Schválil	Priezvisko Meno	Podpis	Dátum

Formulár	Dokument		
Ref. číslo	Fo 11	Dátum poslednej aktualizácie	Dátum 03.01.2013

Akceptované dňa : <Dátum akceptácie>

Za <Objednávateľ>:

Za <Dodávateľ>.::

<Meno zodpovednej osoby>

<Meno zodpovednej osoby>

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

Záznamy o zmenách

Autor	Popis zmien	Dátum	Verzia
M.Mikuš	Zpracovaná 6.verzia špecifikácie; vzniká v1.1 produktu	19.5.	1
R. Vittek	Aktualizácia systémových požiadaviek	18.8.	2
M.Mikuš	Pridaná kontrola v metóde CheckContainer.	26.10.2016	3
M.Mikuš	Len pre CAdES: Metóda GetInfo vracia MimeType aj keď nie je určený v ASiCManifest. Určuje sa podľa koncovky súboru a konfiguračného súboru mimetypes.xml (přibudol konfiguračný súbor mimetypes.xml); vzniká verzia 1.2 produktu	12.5.2017	1
M.Mikuš	Pridanie zoznamu nepodpísaných súborov do výstupu metódy GetInfo.	12.9.2018	2
M.Mikuš	Zpracovaná špecifikácia verzie 9. Presnejšia detekcia typu kontajnera pre XAdES – Simple/Extended. Tu nie je spätná kompatibilita. Presunuté „kritické kontroly“ do metódy Initialize. Verzia produktu na 1.3.	6.3.2019	3
M.Mikuš	Zpracovaná verzia špecifikácie 10. Doplnenie a upresnenie funkcionality pre Java a zosúladenie postupu pre GetInfo.	12.11.2020	4
M.Mikuš	Aktualizácia systémových požiadaviek	30.8.2023	7
M.Mikuš	Doplnenie chybového hlásenia. Aktualizácia systémových požiadaviek.	28.1.2025	7

Pripomienkovanie a kontrola

Autor	Stanovisko	Dátum	Verzia

Rozdeľovník

	Priezvisko Meno	Firma, Funkcia
Originál		
Kópia		
Kópia		
Kópia		

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

Obsah

1.	Úvod	6
2.	Použité zdroje	7
3.	Systémové požiadavky	9
4.	Architektúra	10
4.1.	Postavenie v rámci nadradenej aplikácie	10
4.2.	Vnútna architektúra.....	10
5.	Špecifikácia API.....	11
5.1.	.NET API	11
5.1.1.	Triedy použité ako výstupné hodnoty metód	13
5.2.	Popis funkcií	14
5.2.1.	Metódy spoločné pre XAdES aj CAdES	14
5.2.1.1.	konštruktor.....	14
5.2.1.2.	metóda initialize.....	14
5.2.1.3.	metóda initialize.....	14
5.2.1.4.	metóda initialize.....	15
5.2.1.5.	metóda initialize.....	15
5.2.1.6.	metóda initialize.....	15
5.2.1.7.	IsInitialized.....	15
5.2.1.8.	metóda GetType.....	16
5.2.1.9.	metóda ErrorMessage.....	17
5.2.1.10.	metóda CheckContainer	17
5.2.1.11.	metóda JoinContainers	19
5.2.1.12.	metóda JoinContainers	21
5.2.1.13.	metóda GetInfo	21
5.2.1.14.	metóda GetInfo	24
5.2.1.15.	metóda GetVersion	24
5.2.2.	Metódy pre CAdES.....	24
5.2.2.1.	metóda GetCadesSignature	24
5.2.2.2.	metóda CreateCadesSignatureAsicS.....	25
5.2.2.3.	metóda CreateCadesSignatureAsicE.....	27
5.2.2.4.	metóda CreateTimeStampAsicS	30
5.2.2.5.	metóda PrepareTimeStampAsicE	31
5.2.2.6.	metóda FinalizeTimeStampAsicE.....	33
5.2.2.7.	metóda ReplaceCMS	34
5.2.2.8.	trieda DataObject	35

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

5.2.2.8.1.	konštruktor	35
6.	Scenáre použitia	36
6.1.1.	Získavanie informácií z kontajnera	36
6.1.2.	Spájanie kontajnerov	36
6.1.3.	Získanie CAdES podpisu alebo časovej pečiatky z ASiC kontajnera	36
6.1.4.	Vytváranie ASiC kontajnera s CAdES podpisom	36
6.1.5.	Vytváranie ASiC-s kontajnera s časovou pečiatkou	37
6.1.6.	Vytváranie ASiC-e kontajnera s časovou pečiatkou	37
6.1.7.	Doplnenie ASiC-e kontajnera o nový CAdES podpis	37
6.1.8.	Doplnenie ASiC-e kontajnera o novú časovú pečiatku	37
6.1.9.	Nahradenie CAdES podpisu alebo časovej pečiatky v ASiC kontajnery	38
7.	Návratové kódy	39

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

1. Úvod

Tento dokument je určený pre používateľov softvérového komponentu ASiC Factory .NET a vývojárov aplikácií pre vytváranie a spracovanie zaručených elektronických podpisov formátu XAdES_ZEP.

Komponent ASiC Factory .NET je určený na

- vytváranie kontajnera ASiC v súlade s [1] a [2] pre digitálne podpisy typu CAdES [3],
- spracovanie kontajnera a získavanie informácií o jeho štruktúre a o obsiahnutých podpisoch (CAdES aj XAdES),
- spájanie dvoch kontajnerov do jedného (CAdES aj XAdES).

Komponent je určený na integráciu do komplexnejších systémov ako pomocná knižnica a neposkytuje užívateľské rozhranie, takže jej popis obsahuje iba zoznam funkcií a ich vstupno-výstupné charakteristiky (popísané v časti 5).

Systémové požiadavky sú zhrnuté v časti 3 a príklady použitia komponentu sú uvedené v časti 6.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

2. Použité zdroje

- [1] ETSI TS 102 918 v1.3.1 (2013-06). Associated Signature Containers (ASiC). Electronic Signatures and Infrastructures (ESI). http://www.etsi.org/deliver/etsi_ts/102900_102999/102918/01.03.01_60/ts_102918v010301p.pdf
- [2] ETSI TS 103 174 v2.1.1 (2013-06). ASiC Baseline Profile. Electronic Signatures and Infrastructures (ESI). Technical Specification. http://www.etsi.org/deliver/etsi_ts/103100_103199/103174/02.01.01_60/ts_103174v020101p.pdf
- [3] ETSI TS 101 733 V2.2.1 (2013-04). Electronic Signatures and Infrastructures (ESI);. CMS Advanced Electronic Signatures (CAAdES). http://www.etsi.org/deliver/etsi_ts/101700_101799/101733/02.02.01_60/ts_101733v020201p.pdf
- [4] ETSI TS 103 171 V2.1.1 (2012-03). Electronic Signatures and Infrastructures (ESI);. XAdES Baseline Profile http://www.etsi.org/deliver/etsi_ts/103100_103199/103171/02.01.01_60/ts_103171v020101p.pdf.
- [5] ETSI TS 101 733 V2.2.1 (2013-04). Electronic Signatures and Infrastructures (ESI);. CMS Advanced Electronic Signatures (CAAdES). http://www.etsi.org/deliver/etsi_ts/101700_101799/101733/02.02.01_60/ts_101733v020201p.pdf
- [6] ETSI TS 103 173 V2.2.1 (2013-04). Electronic Signatures and Infrastructures (ESI);. CAAdES Baseline Profile. Technical Specification. http://www.etsi.org/deliver/etsi_ts/103100_103199/103173/02.02.01_60/ts_103173v020201p.pdf
- [7] IETF, RFC 5652, Cryptographic Message Syntax (CMS), <http://tools.ietf.org/html/rfc5652>.
- [8] IETF RFC 3161: "Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)", <https://www.ietf.org/rfc/rfc3161.txt>
- [9] PKWARE ".ZIP Application Note". <http://www.pkware.com/support/zip-application-note>
- [10] Odporúčania NIST ohľadom hašovacích funkcií. <http://csrc.nist.gov/groups/ST/hash/policy.html>
- [11] Špecifikácia ASiC Factory, DITEC, a.s., 2015

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná příručka	
Referencia	GOV_ZEP.253	Verzia 8

- [12] OASIS: "Open Document Format for Office Applications (OpenDocument) Version 1.2; Part 3: Packages" 29 September 2011. OASIS Standard.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

3. Systémové požiadavky

Použitie knižnice ASiC Factory .NET vyžaduje:

- OS – MS Windows 7 SP1, Windows 8.x, Windows 10, Windows 11,
- platforma – .Net framework, verzia 4.6.2 alebo vyššia,

Ak je aplikácia ASiC Factory .NET spúšťaná z web portálu pomocou aplikácie D.Launcher v1.x, tak požiadavky na web prehliadač sú:

- MS Internet Explorer, v7.0 alebo vyššia (IE 7/8/9 len 32 bit, IE 10/11 32 aj 64 bit), Mozilla Firefox, v45 alebo vyššia, Google Chrome v51 alebo vyššia (prípadne Chromium), Opera v38 alebo vyššia, MS Edge v25 alebo vyššia.

Ak je aplikácia ASiC Factory .NET spúšťaná z web portálu pomocou aplikácie D.Launcher v2.x a rozšírenia D.Bridge, tak požiadavky na web prehliadač sú nasledovné:

- MS Internet Explorer 11 (len 32bit verzia), Mozilla Firefox 78, 89, 91, 101, Google Chrome 91, 100, 101, Chromium 91, 100, 101, Opera 76, 78, Microsoft Edge 91, 96, 97; vo webovom prehliadači nainštalované a povolené rozšírenie D.Bridge 2, pre MS Internet Explorer sa vyžaduje vypnutý chránený režim.

ASiC Factory .NET x64 je možné spúšťať výlučne pomocou aplikácie D.Launcher. Systémové požiadavky pre aplikáciu D.Launcher sú špecifikované v rámci používateľskej príručky pre aplikáciu D.Launcher.

Pre aplikáciu ASiC Factory .NET nie sú potrebné vyššie hardwarové požiadavky, ako vyžaduje samotný operačný systém, prípadne platforma .Net framework 4.6.2 alebo vyššia.

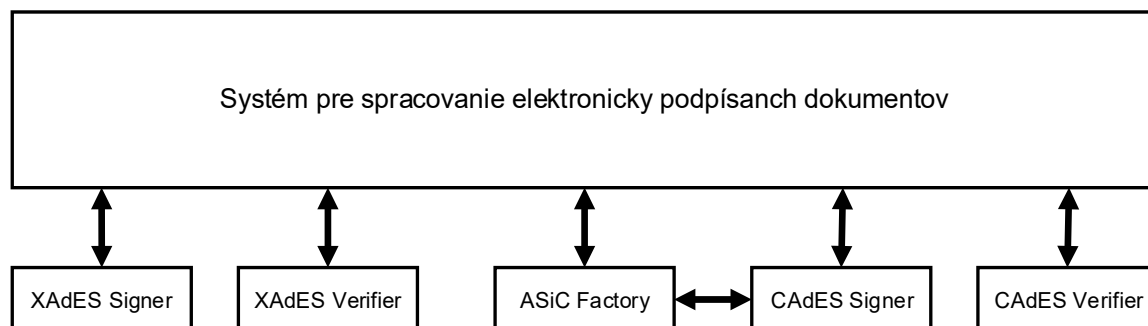
Podrobný popis požiadaviek na prevádzku aplikácie je totožný s požiadavkami na prevádzku, ktoré sú špecifikované v rámci dokumentácie produktu D.Signer/XAdES .NET.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

4. Architektúra

4.1. Postavenie v rámci nadradenej aplikácie

Tento komponent poskytuje volajúcej aplikácii rozhranie na vytváranie a spracovanie dátových kontajnerov ASiC definovaných v špecifikácii [1] a [2]. Volajúca aplikácia ku tomu poskytuje všetky potrebné údaje iba v prípade XAdES podpisov. Požiadavky na manipuláciu s CAdES podpismi vyžadujú, aby komponent ASiC Factory využíval služby komponentu CAdES Signer.



Obr. 1: Komponent ASiC Factory v rámci systému na spracovanie elektronicky podpísaných dokumentov.

4.2. Vnútrotná architektúra

Funkčné požiadavky na komponent ASiC Factory vymedzujú iba jednoduché úlohy, ako napr. extrakcia informácií z existujúceho ASiC-u alebo spájanie dvoch vstupných ASiC-ov. Vzhľadom na fakt, že tieto úlohy nevyžadujú dátovú štruktúru a iba jednoducho spracovávajú vstupné dáta, bude vytvorená iba jedna trieda, ktorá neobsahuje špeciálne dátové štruktúry.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

5. Špecifikácia API

Na základe technológie volajúcej aplikácie je možný prístup k metódam buď priamo referencovaním(.NET trieda Ditec.Zep.AsicFactory.AsicFactory), alebo pomocou alternatívnych rozhraní (COM, NPAPI) popísaných v separátnom dokumente.

Knižnica neposkytuje GUI a teda oznamovanie výstupov z knižnice je plne na strane volajúcej aplikácie.

5.1. .NET API

Hlavná trieda:

Ditec.Zep.AsicFactory.AsicFactory

Konštruktor:

```
public AsicFactory();
```

Vlastnosti:

```
string ErrorMessage{ get; }
```

Metódy:

```
void Initialize(string appIdentification);
```

```
void Initialize(
    string appIdentification,
    string asicB64
```

```
);
```

```
void Initialize(
    string appIdentification,
    byte[] asic
```

```
);
```

```
void Initialize(
    string appIdentification,
    string asicB64,
    string fileName,
```

```
);
```

```
void Initialize(
    string appIdentification,
    byte[] asic,
    string fileName,
```

```
);
```

```
bool IsInitialized();
```

```
int CheckContainer();
```

```
int JoinContainers(
    byte[] asicA,
    byte[] asicB,
    out byte[] joinedAsic
```

```
);
```

```
int JoinContainers(
    string asicAB64,
    string asicBB64,
    out string joinedAsicB64
```

```
);
```

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

```

AsicInfo GetInfo();
string GetInfo(int type);
AsicType GetType();
int GetCadesSignature(
    string signaturePathName,
    bool returnData,
    out Signature signature
);
int CreateCadesSignatureAsicS(
    string digestAlgName,
    string signaturePolicyIdentifier,
    List<byte[]> certificates,
    List<byte[]> certPathRestrictions,
    string username,
    string password,
    DataObject dataObject,
    out byte[] asic,
    string[] hsmUrls = null
);
int CreateCadesSignatureAsicE(
    string digestAlgName,
    string signaturePolicyIdentifier,
    List<byte[]> certificates,
    List<byte[]> certPathRestrictions,
    string username,
    string password,
    List<DataObject> dataObjects,
    out byte[] asic,
    string[] hsmUrls = null
);
int CreateTimeStampAsicS(
    byte[] timeStampToken,
    DataObject dataObject,
    out byte[] asic
);
int PrepareTimeStampAsicE(
    string digestAlgName,
    List<DataObject> dataObjects,
    out byte[] asicManifest,
    out string manifestGUID
);
int FinalizeTimeStampAsicE(
    byte[] timeStampToken,
    string manifestGUID,
    out byte[] asic
);
int ReplaceCMS(
    string signaturePathName,
    byte[] cms,
    out byte[] asic

```

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

```
);
string GetVersion();
```

5.1.1. Triedy použité ako výstupné hodnoty metód

```
public class AsicInfo
{
    AsicType Type
    string FileName
    int SignatureFileCount
    int SignatureInfoListCount
    List<SignatureInfo> SignatureInfoList
    int UnsignedObjectInfoListCount
    List<UnsignedObjectInfo> UnsignedObjectInfoList
}
public enum AsicType
{
    Unknown = 0,
    ASiC_S_CMS = 1,
    ASiC_E_CMS = 2,
    ASiC_S_XAdES = 3,
    ASiC_E_XAdES = 4
}
public class SignatureInfo
{
    string SignatureId
    string SignatureVersion
    List<ProductInfo> ProductInfos
    string SignaturePathName
    int SignedObjectInfoListCount
    List<SignedObjectInfo> SignedObjectInfoList
    string X509CertificateDataBase64
    DateTime? Time
}
public class ProductInfo
{
    string ProductName
    string ProductVersion
}
public class SignedObjectInfo
{
    string ObjectIdentifier
    string ObjectPathName
    string Description
    string MimeType
    string DataB64
    string Encoding
}
public class Signature
{
    AsicType Type
    byte[] CmsStructure
    byte[] AsicManifest
    List<DataObject> DataObjects
}
public class DataObject
```

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

```

{
    public DataObject(string contentType, byte[] fileData)
    public DataObject(string contentType, string fileName, byte[] fileData)

    string ContentType
    string FileName
    byte[] FileData
}
public class UnsignedObjectInfo
{
    string ObjectPathName
    string MimeType
    string DataB64
}

```

5.2. Popis funkcií

Nasleduje stručný popis metód triedy Ditec.Zep.AsicFactory.AsicFactory. Metódy sú rozdelené do dvoch skupín. Prvá skupina metód poskytuje základnú funkcionálnosť pre podpisy XAdES aj CAdES, druhá skupina pokrýva špeciálne požiadavky na manipuláciu s podpismi typu CAdES.

5.2.1. Metódy spoločné pre XAdES aj CAdES

Inicializácia komponentu

Pri využití funkcionality komponentu ASiC Factory musí klientska aplikácia vytvoriť inštanciu triedy AsiCFactory, ktorej následne pomocou metódy `initialize()` odovzdá identifikátor klientskej aplikácie a nepovinne aj ASiC kontajner.

Následne musí klientska aplikácia zavolať metódu `isInitialized`, aby zistila, či bola štruktúra ASiC kontajnera syntakticky správna a inštancia je korektne inicializovaná. Ak to tak nie je, klientska aplikácia nesmie volať iné metódy komponentu ASiC Factory na tejto inštancii, okrem `getErrorMessage`, prípadne opätovnej `initialize`.

5.2.1.1. konštruktor

vstupné parametre:

žiadne

Vytvorenie inštancie triedy.

5.2.1.2. metóda initialize

vstupné parametre:

- textový reťazec `appIdentification` - identifikátor klientskej aplikácie pre použitie v auditných záznamoch,

Metóda vytvorí prázdnu inštanciu triedy. Slúži iba na spájanie kontajnerov.

5.2.1.3. metóda initialize

vstupné parametre:

- textový reťazec `appIdentification` - identifikátor klientskej aplikácie pre použitie v auditných záznamoch,

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- binárny reťazec `asic` – vstupná obálka vo formáte ASiC,

Spoločný popis funkčnosti je v poslednej metóde `initialize`.

5.2.1.4. metóda `initialize`

vstupné parametre:

- textový reťazec `appIdentification` - identifikátor klientskej aplikácie pre použitie v auditných záznamoch,
- textový reťazec `asicB64` – vstupná obálka vo formáte ASiC,

Spoločný popis funkčnosti je v poslednej metóde `initialize`.

5.2.1.5. metóda `initialize`

vstupné parametre:

- textový reťazec `appIdentification` - identifikátor klientskej aplikácie pre použitie v auditných záznamoch,
- binárny reťazec `asic` – vstupná obálka vo formáte ASiC,
- string `fileName` – názov kontajnera.

Spoločný popis funkčnosti je v poslednej metóde `initialize`.

5.2.1.6. metóda `initialize`

vstupné parametre:

- textový reťazec `appIdentification` - identifikátor klientskej aplikácie pre použitie v auditných záznamoch,
- textový reťazec `asicB64` – vstupná obálka vo formáte ASiC,
- string `fileName` – názov kontajnera.

Ak je na vstupe kontajner, tak sa zistí typ kontajnera volaním `getType()`. V závislosti od typu kontajnera sa vykonajú nasledovné kritické kontroly:

- A) XAdES-S – žiadne kontroly,
- B) XAdES-E – žiadne kontroly,
- C) CAdES-S – skontroluje sa, či v root-adresári nie je viac ako jeden objekt. Ak áno, tak inicializácia skončí s chybou,
- D) CAdES-E – skontroluje sa prítomnosť súboru `ASiCManifest.xml`.

5.2.1.7. `IsInitialized`

vstupné parametre:

žiadne

návratová hodnota:

- bool

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

popis:

Metóda vráti true, ak je inštancia korektne inicializovaná, teda ak počas konštruktora a následnej inicializácie nebola zistená žiadna chyba v štruktúre kontajnera. Inak vráti false a aplikácia nesmie volať žiadne iné metódy (okrem `getErrorMessage` a opätovnej inicializácie).

5.2.1.8. metóda GetType

Metóda zistí typ kontajnera a typ podpisov, ktoré sa vyskytujú vo vstupnom kontajneri.

vstupné parametre:

žiadne

výstupné parametre:

žiadne

návratová hodnota:

- `AsicType` type – jedna z hodnôt { `Unknown` (ak je výstupom celé číslo, tak 0), `ASiC-s CMS` (1), `ASiC-e CMS` (2), `ASiC-s XAdES` (3), `ASiC-e XAdES` (4)}.

popis:

Metóda nazrie do adresára META-INF a skontroluje prítomnosť súborov „`*signatures*.xml`“ (t.j. XML typ podpisov), „`timestamp.tst`“, alebo „`signature.p7s`“ (teda CMS typ podpisov). Tiež skontroluje prítomnosť súboru „`ASiCManifest*.xml`“ (teda typ kontajnera ASiC-e).

Ak sa v kontajneri nachádzajú oba typy (XML aj CMS), alebo žiaden, alebo nastane prípad, že kontajner obsahuje aj „`*timestamp*.tst`“ aj „`*signature*.p7s`“ a neobsahuje „`ASiCManifest*.xml`“ tak výstupom je hodnota „`Unknown`“ (resp. 0 pre rozhrania, kde je výstupom iba celé číslo).

Ak je v kontajneri len súbor „`timestamp.tst`“, alebo „`signature.p7s`“ a zároveň kontajner neobsahuje súbor „`ASiCManifest*.xml`“, tak vráti „`ASiC-s CMS`“ (resp. 1). Ak kontajner obsahuje súbor „`ASiCManifest*.xml`“, tak vráti „`ASiC-e CMS`“ (resp. 2).

Ak sú v kontajneri len súbory „`*signatures*.xml`“, tak typ kontajnera sa identifikuje prioritne podľa:

1) obsahu súboru

- A) ak sa v kontajneri nachádza viacero súborov s podpismi, alebo viacero dátových súborov, tak sa nastaví ASiC-e XAdES,
- B) ak sa nachádza len jeden súbor s podpisom a jeden dátový objekt a v podpise je element `ds:Reference` použitý bez atribútu `Uri` (implicitná referencia), tak sa nastaví ASiC-S XAdES,
- C) ak nenastala žiadna z predchádzajúcich, tak sa pokračuje bez určenia typu,

2) súboru mimetype,

- A) obsahuje string „`application/vnd.etsi.asic-s+zip`“, tak sa nastaví ASiC-s XAdES

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

B) obsahuje string „application/vnd.etsi.asic-e+zip“, tak sa nastaví ASiC-e XAdES

C) mimetype sa v kontajneri nenachádza, tak sa pokračuje bez určenia typu,

3) koncovky súboru (.scs, .asics, .sce, .asice) tak sa nastaví príslušný typ ASiC-s alebo ASiC-e XAdES.

Ak tieto hodnoty nesedia, tak typ sa určí v bode s vyššou prioritou a do internej premennej ErrorMessage sa nastaví príslušná chyba „Nekorektný obsah súboru mimetype.“, alebo „Nekorektná koncovka v názve kontajnera.“. Ak by nastal prípad, že typ sa určí v bode 1 a je chybný aj mimetype aj koncovka, tak sa vráti len chyba s vyššou prioritou.

Ak sa uvedenými metódami nepodarilo presne určiť typ kontajnera (koncovka nebola daná a mimetype sa nenachádza), tak sa vráti hodnota „ASiC-e XAdES“ a do ErrorMessage sa nastaví chyba „Nie je špecifikovaný mimetype kontajnera.“.

5.2.1.9. metóda ErrorMessage

Metóda vráti posledné chybové hlásenie, ktoré nastalo pri volaní niektorej z ostatných metód.

vstupné parametre:

žiadne

výstupné parametre:

žiadne

návratová hodnota:

- reťazec znakov – popis poslednej chyby.

popis:

Metóda vráti na výstup obsah internej premennej ErrorMessage.

5.2.1.10. metóda CheckContainer

Metóda skontroluje korektnosť vstupného kontajnera podľa špecifikácie, pričom nekontroluje typ kontajnera (.sce, .scs) ani obsah mimetype súboru. Predpokladá sa, že vstupom je buď ASiC-s, alebo ASiC-e.

vstupné parametre:

žiadne

výstupné parametre:

žiadne

návratová hodnota:

- chybový kód

popis:

Ak bola inštancia triedy AsiCFactory vytvorená bez vstupného parametra `asic`, vráti chybový kód: „Nepovolený typ operácie.“

Metóda vykoná tieto kroky:

1) určí typ podpisov (XAdES/CMS) a pre CMS aj typ kontajnera (S/E) volaním metódy `getType()`,

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

2) v prípade XAdES:

- A) metóda ďalej prehľadá súbory s podpismi (*signatures*.xml v adresári META-INF), validuje ich vzhľadom na XAdESSignatures schému a skontroluje prítomnosť referencovaných dátových súborov.
- B) metóda skontroluje prítomnosť súboru „manifest.xml“ v adresári META-INF, ak ho nenájde, vráti chybu,
- C) V prípade, ak „manifest.xml“ v kontajneri je, tak sa validuje voči príslušnej XML schéme a skontroluje, či referencuje všetky súbory mimo adresára META-INF a či nereferencuje sám seba, t.j. „/META-INF/manifest.xml“. V súbore manifest.xml skontroluje prítomnosť položky <file-entry> s atribútom full-path rovným „/“. Taktiež sa skontrolujú duplicity v položkách <file-entry> daného manifest.xml podľa hodnoty full-path.
- D) Do internej premennej sa zapíše počet nájdených XML podpisov (elementov ds:Signature prítomných v root-ovom elemente každého *signatures*.xml súboru – teda vnorené podpisy nespracováva!²).

3) v prípade CMS štruktúry metóda ďalej prehľadá súbory ASiCManifest*.xml, validuje vzhľadom na ASiCManifest.xsd a zároveň skontroluje prítomnosť odkazovaného súboru s podpisom a aj referencovaných podpisovaných súborov. Skontroluje sa, či názvy súborov s podpisom vyhovujú predpisu „*signature*.p7s“ alebo „*timestamp*.tst“.

Ak kontajner neobsahuje ASiCManifest*.xml súbory, tak sa skontroluje prítomnosť práve jedného z „timestamp.tst“ alebo „signature.p7s“ a prítomnosť jediného dátového objektu v koreňovom adresári (okrem prípustného súboru mimetype).

Do internej premennej sa zapíše počet nájdených CMS štruktúr, ktoré sú referencované z nejakého ASiCManifest súboru.

4) v prípade typu Unknown sa vráti chyba „Kontajner neobsahuje len špecifikovaný typ podpisu XAdES/CMS.“

Možné chybové hlásenia:

- 1) OK.
- 2) ASiC-S neobsahuje práve jeden dátový súbor.
- 3) ASiC-S obsahuje viacero podpisových súborov.
- 4) Kontajner neobsahuje len špecifikovaný typ podpisu XAdES/CMS.
- 5) Súbor (jeho meno) nie je validný vzhľadom na XAdESSignatures schému.
- 6) Súbor (meno) nie je validný vzhľadom na ASiCManifest schému.
- 7) (Meno referencujúceho súboru): V kontajneri chýba súbor (meno referencovaného súboru).

² V prípade, ak by sme povolili spracovávanie vnorených ds:Signature elementov by sme museli analyzovať, či nie sú obsiahnuté v podpísaných dátach – ktoré môžu obsahovať ľubovoľný text a teda nemusia byť v súlade s XSD schémou.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- 8) (Meno referencujúceho súboru): Chybná referencia na podpísaný objekt, porušená integrita údajov. (meno referencovaného súboru).
- 9) ASiC kontajner neobsahuje manifest.xml.
- 10) manifest.xml nie je validný vzhľadom na schému.
- 11) manifest.xml nereferencuje všetky súbory mimo adresára META-INF.
- 12) manifest.xml referencuje sám seba.
- 13) manifest.xml obsahuje duplicity v položkách <file-entry>.
- 14) Nepovolený typ operácie.
- 15) Inicializácia bola neúspešná.
- 16) manifest.xml neobsahuje korektný element <file-entry> pre koreňový element kontajnera.
- 17) Nekorektný obsah súboru mimetype.
- 18) Nekorektná koncovka v názve kontajnera.
- 19) Manifest (meno manifest-súboru) odkazuje na signature/timestamp s chybným názvom: (meno referencovaného súboru).

5.2.1.11. metóda JoinContainers

vstupné parametre:

- binárne reťazce `asicA`, `asicB` – vstupné obálky vo formáte ASiC,

výstupné parametre:

- binárny reťazec `joinedAsic` – výstupná obálka vo formáte ASiC-e

návratová hodnota:

- chybový kód

popis:

Ak bola inštancia triedy `AsiCFactory` vytvorená so vstupným parametrom `asic`, vráti chybový kód: "Nepovolený typ operácie."

Metóda vykoná tieto kroky:

- 1) skontroluje korektnosť oboch vstupných kontajnerov volaním metódy `checkContainer()` a pokračuje len v prípade, ak sú oba „OK“,
- 2) zistí typy podpisov v oboch kontajneroch volaním metódy `getType()` a pokračuje len v prípade, ak je návratová hodnota:
 - "ASiC-e CMS" (2).
 - "ASiC XAdES" (3)
- 3) v prípade CAdES podpisov:

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- I) Skontroluje či názvy všetkých súborov "META-INF/ASiCManifest*.xml", "META-INF/*signature*.p7s", "META-INF/*timestamp*.tst" sú jedinečné v rámci oboch kontajnerov. Tie súbory manifestov, podpisov a časových pečiatok, ktoré majú rovnaký názov, musia byť identické. Ak nie, vráti chybový kód.
 - II) Skontroluje či názvy všetkých súborov dátových objektov v root adresári sú jedinečné. Tie súbory dátových objektov, ktoré majú rovnaký názov, musia byť identické. Ak nie, vráti chybový kód.
 - III) Vytvorí nový ASiC kontajner, v root adresári vytvorí mimetype súbor s hodnotou "application/vnd.etsi.asic-e+zip". Vytvorí adresár "META-INF", do ktorého skopíruje všetky súbory "META-INF/ASiCManifest*.xml", "META-INF/*signature*.p7s", "META-INF/*timestamp*.tst" z oboch vstupných kontajnerov a do root adresára skopíruje všetky súbory dátových objektov (identické súbory s identickým názvom sa skopírujú iba raz).
- 4) v prípade XAdES podpisov sa najprv skontroluje manifest.xml, či nie je podpísaný niektorým z podpisov. Ak je, tak metóda skončí s chybou „Kontajner obsahuje podpísaný manifest.xml.“. Následne skontroluje, či niektorý z kontajnerov neobsahuje element ds:Reference bez atribútu URI (prípád ASiC-S). Ak áno, tak skončí s chybou „Nezlučiteľné typy ASiC kontajnerov.“. Potom:
- I) zmení (ak existuje) obsah mimetype súboru v asicA na hodnotu „application/vnd.etsi.asic-e+zip“,
 - II) skopíruje prvý kontajner asicA do výstupného joinedAsic, zistí informácie o obsahu prvého kontajnera pomocou getInfo(asicA) a ak sa tam nenachádzal súbor „/META-INF/manifest.xml“, tak ho vytvorí aj s platnými dátami (podľa špecifikácie [12]), pričom hodnotu mimetype zistí z príslušného podpisu z elementu DataObjectFormat. Druhý kontajner asicB spracováva postupne po každom „*signatures*.xml“ súbore:
 - a) najprv zistí, či sa daný „*signatures*.xml“ už v kontajneri asicA nachádza. Ak áno, tak ho preskočí a pokračuje ďalším. Ak sa tam nenachádzal, tak skontroluje, či ho možno skopírovať,
 - b) potom zistí, či daný súbor „*signatures*.xml“ treba premenovať, ak áno, tak ho premenuje,
 - c) nakoniec skopíruje (premenovaný) „*signatures*.xml“ a aj príslušné dátové súbory. Ak v asicB existoval „manifest.xml“, tak do výstupného „manifest.xml“ skopíruje príslušné položky <file-entry>, pričom nevytvára duplicity (t.j. kopíruje len také položky, ku ktorým aj kopíruje dátový súbor). Ak neexistoval, tak ich vytvorí podľa špecifikácie [12], pričom hodnotu mimetype zistí z príslušného podpisu z elementu DataObjectFormat.

Súbor „*signatures*.xml“ možno skopírovať, ak:

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- i) neobsahuje také podpisy, ktorých atribút Id je zhodný s Id niektorého podpisu vo výstupnom kontajneri,
- ii) neobsahuje také dátové súbory, ktoré sa s rovnakým menom, rovnakou cestou a rôznym obsahom už vyskytujú vo výstupnom kontajneri.

Ak súbor nemožno skopírovať, metóda vráti chybové hlásenie a skončí. Súbor „*signatures*.xml“ treba premenovať, ak sa vo výstupnom kontajneri už nachádza súbor s rovnakým názvom. V takom prípade sa rozparsuje suffix mena „*signatures*“ a určí sa taký suffix, ktorý sa tam ešte nenachádza.

- 5) Ak nenastala žiadna chyba, metóda vráti na výstupe obálku vo formáte ASiC-e a návratovú hodnotu „OK“.

Možné chybové hlásenia:

- 1) Kontajnery obsahujú rôzne podpisy s rovnakým Id.
- 2) Kontajnery obsahujú rôzne dátové súbory s rovnakým názvom a cestou.
- 3) Kontajnery obsahujú rôzne manifesty, podpisy alebo časové pečiatky s rovnakým názvom a cestou.
- 4) Nezlúčiteľné typy ASiC kontajnerov.
- 5) Kontajner obsahuje podpísaný manifest.xml.
- 6) Nepovolený typ operácie.
- 7) Inicializácia bola neúspešná.

5.2.1.12. metóda JoinContainers

vstupné parametre:

- textové reťazce `asicAB64`, `asicBB64` – vstupné obálky vo formáte string base64,

výstupné parametre:

- textový reťazec `joinedAsicB64` – výstupná obálka ASiC-e vo formáte string base64,

návratová hodnota:

- chybový kód

popis:

Pretážená metóda určená primárne pre jednoduchšie rozhrania NPAPI, COM a JavaAppletu.

5.2.1.13. metóda GetInfo

vstupné parametre:

žiadne

výstupné parametre:

žiadne

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

návratová hodnota:

- štruktúra `AsicInfo` – obsahujúca všetky informácie o vstupnom kontajneri

popis:

Metóda naplní štruktúru `ASiCInfo` a vráti ju na výstup. V prípade, ak niektoré položky štruktúry nemožno vyplniť (v podpise sa nenachádzajú, alebo daný atribút nemajú), tak sa vyplnia prázdny reťazcom.

Metóda najprv zaradí všetky dátové objekty (t.j. všetko v root-adresári okrem adresára META-INF a súboru mimetype) do zoznamu nepodpísaných. Potom prechádza postupne všetky podpisy v META-INF (pre ASiC-e CMS prechádza `ASiCManifest` súbory) a dátové súbory referencované z podpisu presúva do príslušnej štruktúry v `SignatureInfo` pre daný podpis. Ak sa referencovaný dátový súbor nenájde, tak sa vráti chyba. Ak sa nenájde podpis, tak sa pokračuje ďalším.

Štruktúra `ASiCInfo` obsahuje:

- `AsicType type` – typ podpisov (rovnako ako výstup `GetType()`),
- `String FileName` – názov kontajnera,
- `int SignatureFileCount` – počet súborov s podpismi,
- `int SignatureInfoListCount` – počet podpisov (v jednom súbore môže byť viac podpisov),
- `List<SignatureInfo> SignatureInfoList` – zoznam podpisov,
- `int UnsignedObjectInfoListCount` – počet nepodpísaných súborov v kontajneri,
- `List<UnsignedObjectInfo> UnsignedObjectInfoList` – zoznam nepodpísaných súborov v kontajneri.

Štruktúra `SignatureInfo` obsahuje:

- `string SignatureId` – Id atribút konkrétneho podpisu (vyplní sa len v prípade XAdES podpisov),
- `string SignatureVersion` – identifikátor verzie podpisu z elementu `xzep:SignatureVersion` v štruktúre podpisu,
- `List<ProductInfo> ProductInfos` – identifikátory produktov, ktoré boli použité pre vytvorenie podpisu, a ich verzie z elementu `xzep:ProductInfos` v štruktúre podpisu,
- `string SignaturePathName` – cesta a názov súboru podpisu alebo časovej pečiatky v rámci ASiC kontajnera
- `int SignedObjectInfoListCount` – počet podpísaných objektov,
- `List<SignedObjectInfo> SignedObjectInfoList` – zoznam podpísaných súborov/objektov (v prípade CMS ASiC-e sa tento zoznam získa z príslušného `ASiCManifest.xml`, v prípade CMS ASiC-s sa k podpisu alebo časovej pečiatke vzťahuje vždy len jeden dátový objekt),

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- `string X509CertificateDataBase64` – podpisový certifikát v base64,
- `date Time` – čas, v ktorom bola vydaná časová pečiatka (vyplní sa len v prípade timestamp tokenu).

Štruktúra `ProductInfo` obsahuje:

- `string ProductName` – meno produktu,
- `string ProductVersion` – verzia produktu.

Štruktúra `SignedObjectInfo` obsahuje, v závislosti od typu podpisu:

- `string ObjectPathName` – cesta a názov súboru v rámci daného ASiC,
- `string ObjectIdentifier` – identifikátor objektu (iba pre XAdES, nepovinné),
- `string Description` – popis objektu (iba pre XAdES, nepovinné),
- `string MimeType` – mimetype objektu,
- `string DataB64` – dáta objektu v base64 kódovaní,
- `string Encoding` – kódovanie dát v objekte (iba pre XAdES, nepovinné).

Pre určenie `MimeType` dátového objektu sa pre formát CAdES používa nepovinný element v `ASiCManifest.xml` (len pre CAdES-E). Ak by tento spôsob zlyhal (napr. elementy sa v štruktúre nenachádzajú, alebo sa jedná o obálku typu CAdES-S), tak sa `MimeType` objektu skúsi určiť podľa koncovky referencovaného súboru. Ak sa dá určiť koncovka súboru, tak sa podľa konfiguračného súboru `mimetypes.xml` určí zodpovedajúci `MimeType`. Ak sa koncovka súboru nedá určiť, alebo sa v `mimetypes.xml` nenachádza, tak bude výstupom položky `MimeType` null.

Pre XAdES sa `MimeType` určuje na základe povinného elementu `DataObjectFormat`. Ak sa tento nenachádza, tak výstupom je null.

Pozn. Pre XML údaje sa v tejto metóde vráti vždy len referencovaný objekt. Nie sú vykonané dodatočné Transformácie, ktoré môžu jeho obsah ďalej zmeniť. Presné podpísané údaje (t.j. po vykonaní všetkých transformácií) vracia komponent `DVerifierSvrXadesBP` aplikácie `D.Verifier-Svr/Xades`.

Štruktúra `UnsignedObjectInfo` obsahuje zoznam súborov, ktoré nie sú podpísané (v prípade XML to znamená, že do nich nesmeruje žiadna referencia) a informácie o nich:

- `string ObjectPathName` – cesta a názov súboru v rámci daného ASiC,
- `string MimeType` – metóda skúsi určiť typ dátového objektu na základe prípony a súboru `mimetypes.xml`,
- `string DataB64` – dáta objektu v base64 kódovaní.

Zoznam `UnsignedObjectInfoList` sa získa tak, že sa inicializuje zoznamom všetkých súborov v kontajneri mimo adresára META-INF a vyškrtnú sa všetky súbory, do ktorých smeruje nejaká referencia z nejakého podpisu.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

5.2.1.14. metóda GetInfo

vstupné parametre:

- celé číslo `type` – určuje požadovaný typ výstupu

výstupné parametre:

žiadne

návratová hodnota:

- reťazec znakov `AsicInfo` – obsahujúca všetky informácie o vstupnom kontajneri vo formáte XML alebo JSON

popis:

Preťažená metóda naplní štruktúru `ASiCInfo` a vráti ju v požadovanom výstupnom formáte XML (ak `type` bol rovný nule), alebo JSON (ak `type` bol rovný jednej).

5.2.1.15. metóda GetVersion

vstupné parametre:

žiadne

výstupné parametre:

žiadne

návratová hodnota:

- reťazec znakov – obsahujúci informácie o verzii komponentu `ASiC Factory`.

popis:

Výstupný reťazec bude vo formáte JSON a bude obsahovať tieto informácie:

- `name` – celý názov komponentu,
- `version` – verzia komponentu.

5.2.2. Metódy pre CAdES

5.2.2.1. metóda GetCadesSignature

vstupné parametre:

- string `SignaturePathName` – cesta a názov súboru podpisu alebo timestamp tokenu, ktorý sa má vrátiť (nepovinný parameter – v prípade absencie sa automaticky vracia prvá štruktúra z obálky),
- bool `returnData` – indikuje, či sa má vrátiť aj zoznam podpísaných súborov (v prípade `ASiC-e` podpísaný manifest a referencované dátové objekty, v prípade `ASiC-s` jeden podpísaný dátový objekt),

výstupné parametre:

- štruktúra `Signature` – obsahujúca štruktúru podpisu a podpísané údaje,

návratová hodnota:

- chybový kód

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

popis:

Ak bola inštancia triedy `AsicFactory` vytvorená bez vstupného parametra `asic`, vráti chybový kód: "Nepovolený typ operácie."

V prípade ak kontajner obsahuje podpisy XAdES, metóda vráti na výstup prázdnu štruktúru a skončí.

Ak je vstupný kontajner korektný a číslo požadovaného podpisu alebo timestamp tokenu `number` je menšie alebo rovné počtu podpisov, resp. timestamp tokenov), tak naplní štruktúru `Signature` a vráti ju na výstup. V opačnom prípade sa vráti prázdna štruktúra a chybové hlásenie bude zapísané v internej štruktúre.

Štruktúra `Signature` obsahuje:

- `string type` – typ podpisov (CMS),
- binárny reťazec `CMSstructure` – podpis/pečiatka vo formáte CMS, DER kódovaný,
- binárny reťazec `asicManifest` – podpísaný, resp. opečiatkovaný manifest v prípade ASiC-e kontajnera,
- `List<DataObject> dataObjects` -- v prípade ASiC-S je to len jeden podpísaný dátový objekt, v prípade ASiC-e jeden alebo viacero dátových objektov referencovaných v podpísanom manifeste
Trieda `DataObject` je definovaná v časti 5.2.2.8 nižšie.

Možné chybové hlásenia:

- 1) ASiC-E neobsahuje ASiCManifest pre daný podpis.
- 2) Podpis s danou cestou neexistuje.
- 3) Nepovolený typ operácie.
- 4) Inicializácia bola neúspešná.

5.2.2.2. metóda `CreateCadesSignatureAsicS`

Volaním metódy `Sign` komponentu `D.Signer-SVR/CAdES` vytvorí CAdES podpis pre vstupný dátový objekt. Následne vytvorí adresárovú štruktúru kontajnera ASiC-s podľa špecifikácie ASiC [1] a vráti ju na výstup.

vstupné parametre:

- textový reťazec `digestAlgName` - algoritmus digitálneho odtlačku,
- textový reťazec `signaturePolicyIdentifier` – OID podpisovej politiky, nepovinné
- množina binárnych reťazcov `certificates` – kolekcia certifikátov v DER kódovaní. Prvý je vždy podpisový certifikát. Ostatné certifikáty sú nepovinné (`SignedData.Certificates`),

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- množina binárnych reťazcov `certPathRestrictions` - kolekcia certifikátov v DER kódovaní. Prvý je vždy podpisový certifikát (`SignedData.SignerInfo.signedAttrs.id-aa-signingCertificate`). Ak klientska aplikácia nechce použiť reštrikcie pri overení podpisu, tak je NULL,
- textový reťazec `username` a textový reťazec `password` - autentifikačné údaje klientskej aplikácie, ktoré sa overujú voči konfiguračnému súboru podpisovača,
- `DataObject dataObject` – objekt na podpísanie, trieda `DataObject` je definovaná v časti 5.2.2.8 nižšie,
- množina textových reťazcov `hsmUrls` – parametre na dynamickú inicializáciu HSM zariadenia.

výstupné parametre:

- binárny reťazec `asic` – výstupná obálka vo formáte ASiC-s

návratová hodnota:

- chybový kód

popis:

Ak bola inštancia triedy `AsiCFactory` vytvorená so vstupným parametrom `asic`, vráti chybový kód: "Nepovolený typ operácie."

Metóda vytvorí inštanciu modulu `D.Signer-SVR/CAdES`, následne zavolá metódu `Sign`, v rámci ktorej špecifikuje vstupné parametre pre podpis:

- dátový objekt na podpísanie,
- autentifikačné dáta,
- algoritmus digitálneho odtlačku,
- nepovinne podpisovú politiku,
- podpisový certifikát a nepovinne všetky certifikáty v certifikačnej ceste podpisového certifikátu,
- nepovinne kolekciu certifikátov, ktoré musia byť použité pri vystavaní a overení certifikačnej cesty podpisového certifikátu,
- príznak pre vytvorenie externého (detached) podpisu - ASiC kontajner môže obsahovať iba externé (detached) podpisy, preto príznak bude vždy nastavený na true
- úroveň validácie dátových objektov – používa sa iba pre S/MIME obálku, teda hodnota sa nastaví vždy na 0 (bez validácie)

Ak volanie metódy `Sign` skončilo s chybou, metóda vráti príslušný chybový kód.

Volaním metódy `GetCAdESSignature()` získa CMS štruktúru vytvoreného CAdES podpisu.

V koreňovom adresári výstupného ASiC-u sa vytvorí príslušný dátový súbor s názvom „fileName“ a následne v koreňovom adresári vytvorí adresár META-INF, do ktorého sa vloží súbor vytvoreného podpisu s názvom „signature.p7s“.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

V koreňovom adresári kontajnera sa vytvorí súbor s názvom „mimetype“, ktorého obsahom bude hodnota „application/vnd.etsi.asic-s+zip“.

Možné chybové hlásenia:

- 1) Nepovolený typ operácie.
- 2) Inicializácia bola neúspešná.

5.2.2.3. metóda **CreateCadesSignatureAsicE**

Ak pri inicializácii komponentu nebol načítaný ASiC kontajner, metóda vytvorí adresárovú štruktúru kontajnera ASiC-e podľa špecifikácie ASiC [1]. Pripraví ASiC Manifest, do ktorého sú pridané referencie podpisu a vstupných dátových objektov na podpísanie. Referencie dátových objektov získa volaním metódy `CalculateHash` komponentu `D.Signer-SVR/CAdES`. Následne volaním metódy `Sign` komponentu `D.Signer-SVR/CAdES` vytvorí CAdES podpis pre ASiC Manifest, ktorý vloží do kontajnera a vytvorenú štruktúru ASiC-e vráti na výstupe.

Ak pri inicializácii komponentu bol načítaný ASiC-e kontajner (obsahujúci iba CAdES, resp. CMS podpisy), metóda do existujúcej štruktúry kontajnera ASiC-e pridá nový CAdES podpis, k nemu prislúchajúci manifest a dátové objekty referencované v manifeste. Najprv sa do kontajnera ASiC-e pridajú nové dátové objekty. Ak má niektorý zo vstupných dátových objektov na podpísanie rovnaký názov ako existujúci dátový objekt v rozširovanom kontajneri a súbory sú identické, nebude do kontajnera pridaný a v manifeste nového podpisu bude referencovaný pôvodný dátový objekt. Ak nie sú na vstupe uvedené žiadne dátové objekty na podpísanie, budú v manifeste nového podpisu referencované všetky súbory dátových objektov v rozširovanom ASiC kontajneri. Následne sa pripraví ASiC Manifest, do ktorého sú pridané referencie podpisu a vstupných dátových objektov na podpísanie. Referencie dátových objektov sa získajú volaním metódy `CalculateHash` komponentu `D.Signer-SVR/CAdES`. Následne volaním metódy `Sign` komponentu `D.Signer-SVR/CAdES` vytvorí CAdES podpis pre ASiCManifest, ktorý vloží do kontajnera a rozšírenú štruktúru ASiC-e vráti na výstupe.

vstupné parametre:

- textový reťazec `digestAlgName` - algoritmus digitálneho odtlačku,
- textový reťazec `signaturePolicyIdentifier` – OID podpisovej politiky, nepovinné,
- množina binárnych reťazcov `certificates` – kolekcia certifikátov v DER kódovaní. Prvý je vždy podpisový certifikát. Ostatné certifikáty sú nepovinné (`SignedData.Certificates`),
- množina binárnych reťazcov `certPathRestrictions` - kolekcia certifikátov v DER kódovaní. Prvý je vždy podpisový certifikát (`SignedData.SignerInfo.signedAttrs.id-aa-signingCertificate`). Ak klientska aplikácia nechce použiť reštrikcie pri overení podpisu, tak je NULL,

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- textový reťazec `username` a textový reťazec `password` - autentifikačné údaje klientskej aplikácie, ktoré sa overujú voči konfiguračnému súboru podpisovača,
- dátové objekty `dataObject` na podpísanie – pri vytváraní nového ASiC kontajnera povinný aspoň jeden, pri rozširovaní ASiC kontajnera o nový CAdES podpis sú dátové objekty nepovinné (ak nie sú uvedené, budú v manifeste nového podpisu referencované všetky súbory dátových objektov v rozširovanom ASiC kontajneri),
- množina textových reťazcov `hsmUrls` – parametre na dynamickú inicializáciu HSM zariadenia.

výstupné parametre:

- binárny reťazec `asic` – nová alebo rozšírená obálka vo formáte ASiC-e

návratová hodnota:

- chybový kód

popis:

- 1) Ak pri inicializácii komponentu nebol načítaný ASiC kontajner, metóda vytvorí adresárovú štruktúru kontajnera ASiC-e podľa špecifikácie ASiC [1]. V koreňovom adresári ASiC kontajnera sa vytvorí súbor s názvom „mimetype“, ktorého obsahom bude hodnota „application/vnd.etsi.asic-s+zip“ a adresár „META-INF“.
- 2) Ak pri inicializácii komponentu bol načítaný ASiC kontajner, volaním metódy `getType()` zistí typ kontajnera. Ak je návratová hodnota iná ako ASiC-e CMS, vráti chybový kód „Nesprávny typ kontajnera.“
- 3) Do ASiC kontajnera metóda pridá nové dátové objekty (v koreňovom adresári ASiC-u sa vytvorí príslušný dátový súbor s názvom „fileName“). Ak na vstupe nie je uvedený názov súboru dátového objektu, pre názov súboru v ASiC obálke sa vygeneruje GUID.
 - Ak sa vytvára nový ASiC kontajner, sú automaticky pridané všetky vstupné dátové objekty.
 - Ak nie sú na vstupe uvedené žiadne dátové objekty na podpísanie
 - ⇒ a zároveň pri inicializácii komponentu bol načítaný ASiC kontajner, budú v manifeste nového podpisu referencované všetky súbory dátových objektov v rozširovanom ASiC kontajneri.
 - ⇒ a zároveň pri inicializácii komponentu nebol načítaný ASiC kontajner, metóda vráti chybový kód: „Pri vytváraní nového ASiC kontajnera je povinné zadanie aspoň jedného dátového objektu.“
 - Ak má niektorý zo vstupných dátových objektov na podpísanie rovnaký názov ako existujúci dátový objekt v rozširovanom kontajneri

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

⇒ a zároveň sú súbory identické, nebude do kontajnera pridaný a v manifeste nového podpisu bude referencovaný pôvodný dátový objekt.

⇒ a zároveň súbory nie sú identické, metóda vráti chybový kód: "Duplicitný názov dátového objektu na podpísanie: [fileName]".

4) Vygeneruje sa názov súboru podpisu „signature_GUID.p7s“ (jedinečný identifikátor GUID v názve súboru podpisu umožňuje neskoršie spájanie viacerých ASiC-e kontajnerov do jedného).

5) Volaním metódy `CalculateHash(fileName, digestMethod, outDigestValue)` komponentu D.Signer-SVR/CAdES vyráta hodnotu digitálneho odtlačku pre jednotlivé dátové objekty zo vstupného parametra `dataObject` (alebo dátové objekty z rozširovaného ASiC kontajnera, pravidlá na zaradenie referencií dátových objektov do manifestu sú popísané vyššie).

6) Pripraví sa ASiCManifest, do ktorého sú pridané referencie podpisu a vstupných dátových objektov na podpísanie. XML schéma podľa špecifikácie ASiC [1]:

- `AsiCManifest`: root element. Obsahuje jeden element `SigReference` a jeden alebo viacero elementov `DataObjectReference`,
- `SigReference`: tento element obsahuje URI ukazujúce na CAdES podpis alebo časovú pečiatku vzťahujúce sa na metadáta v manifeste a ich príslušný `MimeType`,
- `DataObjectReference`: tento element obsahuje atribúty `URI`, `MimeType` a nepovinne `Rootfile`, referencujúce v tomto poradí dátový objekt, MIME typ a boolean `Rootfile` atribút; obsahuje elementy `DigestMethod` a `DigestValue`, ktoré obsahujú algoritmus digitálneho odtlačku a hodnotu odtlačku pre príslušný dátový objekt,
- `Extension`: tento element sa zatiaľ nepoužíva.

Do elementu `SigReference` sa naplní URI ukazujúce na CAdES podpis (MIME typ sa naplní hodnotou „application/x-pkcs7-signature“), pre jednotlivé dátové objekty zo vstupného parametra `dataObject` (alebo dátové objekty z rozširovaného ASiC kontajnera, pravidlá na zaradenie referencií dátových objektov do manifestu sú popísané vyššie) vytvorí element `DataObjectReference` a naplní atribúty `URI` a `MimeType` (atribút `Rootfile` sa nenaplní), do elementov `DigestMethod` a `DigestValue` sa naplnia parametre metódy `CalculateHash`.

7) Vytvorený manifest (názov súboru „AsiCManifest_GUID.xml“ – rovnaký GUID ako pre podpis) sa uloží do adresára „META-INF“.

8) Metóda vytvorí inštanciu modulu D.Signer-SVR/CAdES, následne zavolá metódu *Sign*, v rámci ktorej špecifikuje vstupné parametre pre podpis:

- ASiC Manifest na podpísanie,
- autentifikačné dáta,

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- algoritmus digitálneho odtlačku,
- nepovinne podpisovú politiku,
- podpisový certifikát a nepovinne všetky certifikáty v certifikačnej ceste podpisového certifikátu,
- nepovinne kolekciu certifikátov, ktoré musia byť použité pri vystavaní a overení certifikačnej cesty podpisového certifikátu,
- príznak pre vytvorenie externého (detached) podpisu - ASiC kontajner môže obsahovať iba externé (detached) podpisy, preto príznak bude vždy nastavený na true,
- úroveň validácie dátových objektov – používa sa iba pre S/MIME obálku, teda hodnota sa nastaví vždy na 0 (bez validácie).

9) Ak volanie metódy `Sign` skončilo s chybou, metóda vráti príslušný chybový kód.

10) Volaním metódy `GetCAdESSignature()` komponentu `D.Signer-SVR/CAdES` získa CMS štruktúru vytvoreného CAdES podpisu.

11) Súbor vytvoreného podpisu s názvom „signature_GUID.p7s“ sa vloží do ASiC kontajnera do adresára META-INF.

12) Metóda vráti na výstupe nový ASiC-e kontajner alebo pôvodný ASiC-e kontajner rozšírený o nový CAdES podpis a k nemu sa vzťahujúci manifest a dátové objekty.

5.2.2.4. metóda `CreateTimeStampAsicS`

Vytvorí adresárovú štruktúru kontajnera ASiC-s podľa špecifikácie ASiC [1], vloží do nej časovú pečiatku aj s dátovým objektom, ku ktorému sa časová pečiatka vzťahuje a vráti ju na výstup.

vstupné parametre:

- binárny reťazec `timeStampToken` – časová pečiatka vo formáte CMS, DER kódovaná,
- `DataObject dataObject`, ku ktorému sa vzťahuje časová pečiatka (`DataObject` je definovaný v časti 5.2.2.8 nižšie),

výstupné parametre:

- binárny reťazec `asic` – výstupná obálka vo formáte ASiC-s

návratová hodnota:

- chybový kód

popis:

Ak bola inštancia triedy `AsiCFactory` vytvorená so vstupným parametrom `asic`, vráti chybový kód: “Nepovolený typ operácie.”

Metóda vytvorí adresárovú štruktúru kontajnera ASiC-s podľa špecifikácie ASiC [1]. V koreňovom adresári ASiC kontajnera sa vytvorí súbor s názvom „mimetype“, ktorého obsahom bude hodnota „application/vnd.etsi.asic-s+zip“ a adresár „META-INF“. Do koreňového adresára ASiC kontajnera sa vloží dátový súbor s názvom

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

„fileName“ a do adresára „META-INF“ sa vloží súbor časovej pečiatky s názvom „timestamp.tst“.

Metóda vráti na výstupe nový ASiC-s kontajner.

Možné chybové hlásenia:

- 1) Nepovolený typ operácie.
- 2) Inicializácia bola neúspešná.

5.2.2.5. metóda **PrepareTimeStampAsicE**

Ak pri inicializácii komponentu nebol načítaný ASiC kontajner, metóda vytvorí adresárovú štruktúru kontajnera ASiC-e podľa špecifikácie ASiC [1]. Pripraví ASiC Manifest, do ktorého sú pridané referencie časovej pečiatky a vstupných dátových objektov. Referencie dátových objektov získa volaním metódy `CalculateHash` komponentu `D.Signer-SVR/CAAdES`.

Ak pri inicializácii komponentu bol načítaný ASiC-e kontajner (obsahujúci iba CAAdES, resp. CMS podpisy), metóda do existujúcej štruktúry kontajnera ASiC-e pridá nový manifest pre časovú pečiatku, v ktorom sú referencované vstupné dátové objekty. Najprv sa do kontajnera ASiC-e pridajú nové dátové objekty. Ak má niektorý zo vstupných dátových objektov rovnaký názov ako existujúci dátový objekt v rozširovanom kontajneri a súbory sú identické, nebude do kontajnera pridaný a v manifeste novej časovej pečiatky bude referencovaný pôvodný dátový objekt. Ak nie sú na vstupe uvedené žiadne dátové objekty, budú v manifeste novej časovej pečiatky referencované všetky súbory dátových objektov v rozširovanom ASiC kontajneri. Následne sa pripraví ASiC Manifest, do ktorého sú pridané referencie časovej pečiatky a vstupných dátových objektov. Referencie dátových objektov sa získajú volaním metódy `CalculateHash` komponentu `D.Signer-SVR/CAAdES`.

Klientskej aplikácii vráti metóda na výstupe ASiC Manifest, pre ktorý klientska aplikácia vyžiada časovú pečiatku od časovej autority. Následne klientska aplikácia volaním metódy `finalizeTimeStampAsicE(timestampToken, out asic)` získa ASiC-e kontajner s časovou pečiatkou.

vstupné parametre:

- textový reťazec `digestAlgName` – algoritmus digitálneho odtlačku pre výpočet referencií dátových objektov v ASiC manifeste,
- `List<DataObject> dataObjects` — objekty na podpísanie, trieda `DataObject` je definovaná v časti 5.2.2.8 nižšie.

výstupné parametre:

- binárny reťazec `asicManifest` – pripravený ASiC manifest, pre ktorý má byť vyžiadaná časová pečiatka
- textový reťazec `manifestGUID` – GUID súboru manifestu časovej pečiatky

návratová hodnota:

- chybový kód

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

popis:

- 1) Ak pri inicializácii komponentu nebol načítaný ASiC kontajner, metóda vytvorí adresárovú štruktúru kontajnera ASiC-e podľa špecifikácie ASiC [1]. V koreňovom adresári ASiC kontajnera sa vytvorí súbor s názvom „mimetype“, ktorého obsahom bude hodnota „application/vnd.etsi.asic-s+zip“ a adresár „META-INF“.
- 2) Ak pri inicializácii komponentu bol načítaný ASiC kontajner, volaním metódy `getType()` zistí typ kontajnera. Ak je návratová hodnota iná ako ASiC-e CMS, vráti chybový kód „Nekorektný typ kontajnera.“
- 3) Do ASiC kontajnera metóda pridá nové dátové objekty (v koreňovom adresári ASiC-u sa vytvorí príslušný dátový súbor s názvom „fileName“). Ak na vstupe nie je uvedený názov súboru dátového objektu, pre názov súboru v ASiC obálke sa vygeneruje GUID.
 - Ak sa vytvára nový ASiC kontajner, sú automaticky pridané všetky vstupné dátové objekty.
 - Ak nie sú na vstupe uvedené žiadne dátové objekty na podpísanie
 - ⇒ a zároveň pri inicializácii komponentu bol načítaný ASiC kontajner, budú v manifeste nového podpisu referencované všetky súbory dátových objektov v rozširovanom ASiC kontajneri.
 - ⇒ a zároveň pri inicializácii komponentu nebol načítaný ASiC kontajner, metóda vráti chybový kód: „Pri vytváraní nového ASiC kontajnera je povinné zadanie aspoň jedného dátového objektu.“
 - Ak má niektorý zo vstupných dátových objektov na podpísanie rovnaký názov ako existujúci dátový objekt v rozširovanom kontajneri
 - ⇒ a zároveň sú súbory identické, nebude do kontajnera pridaný a v manifeste nového podpisu bude referencovaný pôvodný dátový objekt.
 - ⇒ a zároveň súbory nie sú identické, metóda vráti chybový kód: „Duplicitný názov dátového objektu: [fileName]“.
- 4) Vygeneruje sa názov súboru časovej pečiatky „timestamp_GUID.tst“ (jedinečný identifikátor GUID v názve súboru podpisu umožňuje neskoršie spájanie viacerých ASiC-e kontajnerov do jedného).
- 5) Volaním metódy `CalculateHash(fileName, digestMethod, outDigestValue)` komponentu D.Signer-SVR/CAdES vyrába hodnotu digitálneho odtlačku pre jednotlivé dátové objekty zo vstupného parametra `dataObject` (alebo dátové objekty z rozširovaného ASiC kontajnera, pravidlá na zaradenie referencií dátových objektov do manifestu sú popísané vyššie).
- 6) Pripraví sa ASiCManifest, do ktorého sú pridané referencie časovej pečiatky a vstupných dátových objektov na podpísanie. XML schéma podľa špecifikácie ASiC [1]:

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- **AsiCManifest:** root element. Obsahuje jeden element **SigReference** a jeden alebo viacero elementov **DataObjectReference**,
- **SigReference:** tento element obsahuje URI ukazujúce na CAdES podpis alebo časovú pečiatku vzťahujúce sa na metadáta v manifeste a ich príslušný **MimeType**,
- **DataObjectReference:** tento element obsahuje atribúty **URI**, **MimeType** a nepovinne **Rootfile**, referencujúce v tomto poradí dátový objekt, MIME typ a boolean **Rootfile** atribút; obsahuje elementy **DigestMethod** a **DigestValue**, ktoré obsahujú algoritmus digitálneho odtlačku a hodnotu odtlačku pre príslušný dátový objekt,
- **Extension:** tento element sa zatiaľ nepoužíva.

Do elementu **SigReference** sa naplní URI ukazujúce na časovú pečiatku (TS zatiaľ nie je vyžadovaný, ale už je vygenerovaný názov súboru TS) a MIME typ („application/x-pkcs7-signature“), pre jednotlivé dátové objekty zo vstupného parametra **dataObject** (alebo dátové objekty z rozširovaného ASiC kontajnera, pravidlá na zaradenie referencií dátových objektov do manifestu sú popísané vyššie) vytvorí element **DataObjectReference** a naplní atribúty **URI** a **MimeType** (atribút **Rootfile** sa nenaplní), do elementov **DigestMethod** a **DigestValue** sa naplnia parametre metódy **CalculateHash**.

- 7) Vytvorený manifest (názov súboru „AsiCManifest_GUID.xml“ – rovnaký GUID ako pre časovú pečiatku) sa uloží do adresára „META-INF“.
- 8) Metóda vráti na výstupe ASiC Manifest, pre ktorý klientska aplikácia vyžiada časovú pečiatku od časovej autority. Následne klientska aplikácia volaním metódy **finalizeTimeStampAsicE(timestampToken, out asic)** získa ASiC-e kontajner s časovou pečiatkou.

Možné chybové hlásenia:

- 1) Nekorektný typ kontajnera.

5.2.2.6. metóda **FinalizeTimeStampAsicE**

Metóda doplní časovú pečiatku manifestu do „pripraveného“ ASiC-e kontajnera (pripravený volaním metódy **prepareTimeStampAsicE**) a na výstupe vráti ASiC-e kontajner.

vstupné parametre:

- binárny reťazec **timestampToken** – časová pečiatka manifestu vo formáte CMS, DER kódovaná,
- textový reťazec **manifestGUID** – GUID súboru manifestu časovej pečiatky

výstupné parametre:

- binárny reťazec **asic** – výstupná obálka vo formáte ASiC-s

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

návratová hodnota:

- chybový kód

popis:

Vykoná sa kontrola, či je inštancia triedy `AsiCFactory` pripravená na doplnenie časovej pečiatky. Ak nie, metóda vráti chybový kód: "ASiC kontajner neobsahuje ASiCManifest pre daný manifestGUID."

Metóda do adresára "META-INF" vloží súbor časovej pečiatky s názvom "timestamp_manifestGUID.tst".

Metóda vráti na výstupe nový ASiC-e kontajner.

Možné chybové hlásenia:

- 1) ASiC kontajner neobsahuje ASiCManifest pre daný manifestGUID.
- 2) Inicializácia bola neúspešná.

5.2.2.7. metóda ReplaceCMS

Klientska aplikácia volaním metódy `getInfo()` získa informácie o obsahu ASiC kontajnera a následne volaním metódy `getCadesSignature(number, returnData)` pre vybrané poradové číslo CMS štruktúru CAdES podpisu alebo časovej pečiatky. Následne môže klientska aplikácia rozšíriť CMS štruktúru o ďalšie `SignerInfo` (volaním metódy `AddSign` komponentu `D.Signer-SVR/CAdES`) alebo doplniť CAdES podpis na vyššiu formu (volaním rôznych metód komponentu `D.Verifier-SVR/CAdES`). Následne klientska aplikácia môže volaním metódy `replaceCMS(number)` nahradiť vybranú CMS štruktúru v ASiC kontajneri.

vstupné parametre:

- `string SignaturePathName` – cesta a názov súboru CMS štruktúry podpisu alebo časovej pečiatky v ASiC kontajneri
- binárny reťazec `CMS` – CAdES podpis alebo časová pečiatka vo formáte CMS

výstupné parametre:

- binárny reťazec `asic` – výstupná obálka vo formáte ASiC

návratová hodnota:

- chybový kód

popis:

Ak bola inštancia triedy `AsiCFactory` vytvorená bez vstupného parametra `asic`, vráti chybový kód: "Nepovolený typ operácie."

Volaním metódy `getType()` zistí typ kontajnera. Ak je návratová hodnota iná ako ASiC-s CMS alebo ASiC-e CMS, vráti chybový kód "Nesprávny typ kontajnera."

Ak sa v kontajneri nenachádza súbor CMS štruktúry `SignaturePathName`, metóda vráti chybový kód: „Podpis s danou cestou neexistuje.“. Inak bude súbor nahradený hodnotou zo vstupného parametra `CMS`.

Metóda vráti na výstupe nový ASiC kontajner s nahradenou CMS štruktúrou CAdES podpisu alebo časovej pečiatky.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

Možné chybové hlásenia:

- 1) Nepovolený typ operácie.
- 2) Inicializácia bola neúspešná.
- 3) Podpis s danou cestou neexistuje.

5.2.2.8. trieda DataObject

Táto trieda sa používa na manipuláciu s podpísanými objektami pri volaní CAdES-ových metód. Pre XAdES podpisy sa nevyužíva.

5.2.2.8.1. konštruktor

Vstupné parametre:

- textový reťazec `contentType` – MIME typ dátového objektu na podpísanie,
- textový reťazec `fileName` – názov súboru (vrátane prípony) dátového objektu na podpísanie; atribút je *nepovinný*,
- binárny reťazec `fileData` – dátový objekt na podpísanie

Popis:

Ak nie je uvedený nepovinný parameter `fileName`, pre názov súboru v ASiC obálke sa vygeneruje GUID.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

6. Scenáre použitia

V tejto časti sú rozobraté hlavné prípady použitia komponentu ASiC Factory.

6.1.1. Získavanie informácií z kontajnera

V prípade, ak volajúca aplikácia potrebuje získať len základné informácie o kontajneri, musí dodržať nasledovný postup:

- 4) zavolať konštruktor, kde zadá požadovaný kontajner ako vstupný parameter,
- 5) overiť pomocou `IsInitialized()`, či vstupný kontajner bol korektný. Popis prípadnej chyby je možné získať volaním metódy `GetErrorMessage()`,
- 6) v prípade potreby zavolať aj metódu `GetInfo()`, ktorá vráti informácie o obsahu kontajnera (podrobný popis výstupnej štruktúry je v 5.2.1.13).

6.1.2. Spájanie kontajnerov

Postup v prípade, ak volajúca aplikácia chce spojiť dva kontajnery do jedného, je nasledovný:

- 1) zavolať konštruktor triedy bez vstupného parametra `asic`,
- 2) zavolať metódu `JoinContainers()`, ktorá dostane na vstupe oba ASiC kontajnery. Metóda skontroluje korektnosť vstupných dát a výsledný kontajner vráti vo výstupnom parametri. Návrátová hodnota metódy je chybový kód, popis chyby je možné získať volaním `GetErrorMessage()`.

6.1.3. Získanie CAdES podpisu alebo časovej pečiatky z ASiC kontajnera

Postup v prípade, ak volajúca aplikácia chce získať CAdES podpis alebo časovú pečiatku z ASiC kontajnera, je nasledovný:

- 1) zavolať konštruktor, kde zadá požadovaný kontajner ako vstupný parameter,
- 2) overiť pomocou `IsInitialized()`, či vstupný kontajner bol korektný. Popis prípadnej chyby je možné získať volaním metódy `GetErrorMessage()`,
- 3) v prípade potreby zavolať metódu `GetInfo()`, ktorá vráti informácie o obsahu kontajnera,
- 4) pomocou metódy `GetCadesSignature` získa CMS štruktúru CAdES podpisu alebo časovej pečiatky (nepovinne aj manifest a podpísané dátové objekty),

6.1.4. Vytváranie ASiC kontajnera s CAdES podpisom

V prípade, ak volajúca aplikácia potrebuje vytvoriť nový ASiC kontajner s CAdES podpisom, je postup nasledovný:

- 1) zavolať konštruktor triedy bez vstupného parametra `asic`,
- 2) vytvoriť dátový objekt (resp. kolekciu objektov) volaním konšuktora triedy `dataObject()`,

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- 3) pomocou metód `CreateCadesSignatureAsicS` alebo `CreateCadesSignatureAsicE` vytvorí a získa na výstupe ASiC-s alebo ASiC-e kontajner s CAdES podpisom.

6.1.5. Vytváranie ASiC-s kontajnera s časovou pečiatkou

V prípade, ak volajúca aplikácia potrebuje vytvoriť nový ASiC-s kontajner s časovou pečiatkou, je postup nasledovný:

- 1) zavolať konštruktor triedy bez vstupného parametra `asic`,
- 2) pomocou metódy `CreateTimeStampAsicS` vytvorí a získa na výstupe ASiC-s kontajner s časovou pečiatkou.

6.1.6. Vytváranie ASiC-e kontajnera s časovou pečiatkou

V prípade, ak volajúca aplikácia potrebuje vytvoriť nový ASiC-e kontajner s časovou pečiatkou, je postup nasledovný:

- 1) zavolať konštruktor triedy bez vstupného parametra `asic`,
- 2) pomocou metódy `PrepareTimeStampAsicE` sa pripraví štruktúra ASiC-e kontajner, metóda vráti na výstupe manifest,
- 3) klientska aplikácia vyžiada pre manifest časovú pečiatku od časovej autority,
- 4) pomocou metódy `FinalizeTimeStampAsicE` doplní časovú pečiatku do kontajnera a na výstupe získa ASiC-e kontajner s časovou pečiatkou.

6.1.7. Doplnenie ASiC-e kontajnera o nový CAdES podpis

V prípade, ak volajúca aplikácia potrebuje doplniť CAdES podpis do ASiC-e kontajnera, je postup nasledovný:

- 1) zavolať konštruktor, kde zadá požadovaný kontajner ako vstupný parameter,
- 2) overiť pomocou `IsInitialized()`, či vstupný kontajner bol korektný. Popis prípadnej chyby je možné získať volaním metódy `GetErrorMessage()`,
- 3) v prípade potreby zavolať metódu `GetInfo()`, ktorá vráti informácie o obsahu kontajnera,
- 4) pomocou metódy `CreateCadesSignatureAsicE` vytvorí a získa na výstupe ASiC-e kontajner doplnený o nový CAdES podpis.

6.1.8. Doplnenie ASiC-e kontajnera o novú časovú pečiatku

V prípade, ak volajúca aplikácia potrebuje doplniť časovú pečiatku do ASiC-e kontajnera, je postup nasledovný:

- 1) zavolať konštruktor, kde zadá požadovaný kontajner ako vstupný parameter,

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

- 2) overiť pomocou `IsInitialized()`, či vstupný kontajner bol korektný. Popis prípadnej chyby je možné získať volaním metódy `GetErrorMessage()`,
- 3) v prípade potreby zavolať metódu `GetInfo()`, ktorá vráti informácie o obsahu kontajnera,
- 4) pomocou metódy `PrepareTimeStampAsicE` sa pripraví štruktúra ASiC-e kontajner, metóda vráti na výstupe manifest,
- 5) klientska aplikácia vyžiada pre manifest časovú pečiatku od časovej authority,
- 6) pomocou metódy `FinalizeTimeStampAsicE` doplní časovú pečiatku do kontajnera a na výstupe získa ASiC-e kontajner s časovou pečiatkou.

6.1.9. Nahradenie CAdES podpisu alebo časovej pečiatky v ASiC kontajneri

V prípade, ak volajúca aplikácia potrebuje nahradiť CMS štruktúru podpisu alebo časovej pečiatky v ASiC-s alebo ASiC-e kontajneri, je postup nasledovný:

- 1) zavolať konštruktor, kde zadá požadovaný kontajner ako vstupný parameter,
- 2) overiť pomocou `IsInitialized()`, či vstupný kontajner bol korektný. Popis prípadnej chyby je možné získať volaním metódy `GetErrorMessage()`,
- 3) v prípade potreby zavolať metódu `GetInfo()`, ktorá vráti informácie o obsahu kontajnera,
- 4) pomocou metódy `GetCadesSignature` získa CMS štruktúru CAdES podpisu alebo časovej pečiatky (nepovinne aj manifest a podpísané dátové objekty),
- 5) klientska aplikácia môže rozšíriť CMS štruktúru o ďalšie `SignerInfo` (volaním metódy `AddSign` komponentu `D.Signer-SVR/CAdES`) alebo doplniť CAdES podpis na vyššiu formu (volaním rôznych metód komponentu `D.Verifier-SVR/CAdES`),
- 6) pomocou metódy `ReplaceCMS` nahradí CMS štruktúru CAdES podpisu alebo časovej pečiatky v kontajneri a na výstupe získa upravený ASiC kontajner.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

7. Návratové kódy

Komponent ASiC Factory môže vracaa chybové hlásenia z komponentu D.Signer-SVR/CAdES, alebo nasledujúce chybové hlásenia.

Kód	Popis
0	OK.
2	ASiC-S neobsahuje práve jeden dátový súbor.
3	ASiC-S obsahuje viacero podpisových súborov.
5	Kontajner neobsahuje len špecifikovaný typ podpisu XAdES/CMS.
6	Súbor (jeho meno) nie je validný vzhľadom na XAdESSignatures schému.
7	Súbor (meno) nie je validný vzhľadom na ASiCManifest schému.
8	(Meno referencujúceho súboru): V kontajneri chýba súbor (meno referencovaného súboru).
9	(Meno referencujúceho súboru): Chybná referencia na podpísaný objekt, porušená integrita údajov. (meno referencovaného súboru).
10	Kontajnery obsahujú rôzne podpisy s rovnakým Id.
11	Kontajnery obsahujú rôzne dátové súbory s rovnakým názvom a cestou.
12	Kontajnery obsahujú rôzne manifesty, podpisy alebo časové pečiatky s rovnakým názvom a cestou.
13	Nezlúčiteľné typy ASiC kontajnerov.
14	Kontajner obsahuje podpísaný manifest.xml.
15	ASiC-E neobsahuje ASiCManifest pre daný podpis.
16	ASiC kontajner neobsahuje ASiCManifest pre daný manifestGUID.
17	ASiC kontajner neobsahuje manifest.xml
18	manifest.xml nie je validný vzhľadom na schému.
19	manifest.xml nereferencuje všetky súbory mimo adresára META-INF.
20	manifest.xml referencuje sám seba.
21	Podpis s danou cestou neexistuje.
22	Nekorektný typ kontajnera.
23	manifest.xml neobsahuje korektný element <file-entry> pre koreňový element kontajnera.
24	Nekorektný obsah súboru mimetype.
25	Nekorektná koncovka v názve kontajnera.

Projekt	GOV_ZEP	A3019_002
Dokument	Integračná príručka	
Referencia	GOV_ZEP.253	Verzia 8

26	Manifest (meno manifest-súboru) odkazuje na signature/timestamp s chybným názvom: (meno referencovaného súboru).
27	manifest.xml obsahuje duplicity v položkách <file-entry>.
101	Neočakávaná chyba.
102	Inicializácia bola neúspešná.
103	Nepovolený typ operácie.