

**TVORÍME VEDOMOSTNÚ SPOLOČNOSŤ**

**x070E - Detailná špecifikácia rozhrania - volanie služieb**

Projekt je spolufinancovaný Európskou úniou  
 Európsky fond regionálneho rozvoja

d

dd

Európsky fon

**Elektronické služby zdravotníctva**

kód ITMS projektu: **2110120009**

www.informatizacia.sk  
 www.opis.gov.sk

**Základné informácie o dokumente**

|  |  |
| --- | --- |
| **Názov dokumentu** | eSO1\_SP\_X070E\_Detailna\_specifikacia\_rozhrania\_Volanie\_sluzieb |
| **Aktuálna verzia** | 3.0.12 |
| **Vlastník dokumentu** | NCZI |

**História dokumentu**

|  |  |  |  |
| --- | --- | --- | --- |
| **Verzia** | **Dátum** | **Autor** | **Popis verzie a zmien oproti predchádzajúcej verzii** |
| 2.7.2 | 14.09.2016 | Marián Novotňák | viď readme.txt |
| 2.7.3 | 5.10.2016 | Marián Novotňák | bez zmeny |
| 2.7.4 | 14.11.2016 | Marián Novotňák | bez zmeny |
| 2.7.5 | 29.11.2016 | Marián Novotňák | bez zmeny |
| 2.7.6 | 26.1.2017 | Marián Novotňák | bez zmeny |
| 2.7.7 | 6.2.2017 | VTur | Kap. 5.5.2 - vymenovanie podporovaných platforiem  Kap. 5.5.4.4 – nová kapitola, popis požiadavky na EventLogSource |
| 2.7.8 | 14.2.2017 | MNO. MRE | Kap.6 – doplnenie pripojenia prostredníctvom SSL |
| 2.7.81 | 10.3.2017 | vTUR | Kap. 5.5.4.4 – Pridanie vzorového kódu na vytvorenie požadovaného zdroja udalostí |
| 2.7.81 | 29.3.2017 | SEC | Kapitola 6.1.1 Zabezpečenie komunikácie medzi IS PZS a NZIS  - upresnenie TimeOut a cieľových adries |
| 2.8.0 | 3.4.2017 | vTUR | Kap. 5.5.1 – Aktualizované verejné rozhranie IehealthCryptoController pre verziu 4.3 a vyššie  Vložená nová kapitola 5.5.1.8 pre volanie „GetPatientPresenceConfirmation“  Kap. 5.5.4.3 Špecifikácia čítačiek eID kariet pre realizovanie potvrdenia na základe výzvy |
| 2.8.0 | 4.4.2017 | MNO | Aktualizácia kapitoly 5.5.8 Riadenie prístupu k údajom pacientov |
| 3.0 | 23.6.2017 | Martin Ret | Aktualizácaia v rámci iniciatívy ezdravie |
| 3.0.2 | 19.7.2017 | Svätoslav Straka | Viď. Readme.txt |
| 3.0.3 | 31.7.2017 | Vladimír Polák | Doplnený popis HTML textu (kapitola 5.4.6)  Doplnená nová chyba E999973 |
| 3.0.4 | 15.8.2017 | Ján Kuruc | Formálne úpravy |
| 3.0.5 | 27.9.2017 | Ján Kuruc | Doplnené koncové body kap 5.3  Konsolidované informácie o čitaškách čipových kariet ZPR a čip kariet ZPR kap 6.1.4 – detilný popis a drivre sa nachádzajú v dokumente eZdravie\_Integracny\_manual\_pre\_IS\_PZS\_x104\_vXY.docx |
| 3.0.6 | 27.12.2017 | Ján Mikolaj | Doplnené informácie vyhodnocovania citlivosti v Consent Managemwnte  Doplnené informácie k zadávaniu BOK alebo potvrdzovaniu výzvy pri potvrdení prítomnosti pacienta. |
| 3.0.7 | 31.1.2018 | Konz | Upresnenie popisu výnimek E999972; E999994; E999995 |
| 3.0.8 | 2.3.2018 | Konz | Doplnené usmernenie pre používanie HTML textov kap.5.4.6 |
| 3.0.9 | 23.3.2018 | Konz | Opravená informácia ohľadne prístupu všeobecného lekára k zdravotným záznamom pacienta podľa stupňa citlivosti v kapitole 6.1.8  Doplnenie kapitoly 9 HTML validátor |
| 3.0.10 | 2.10.2018 | Konz | Akt. Popisu rozhrania IEHealthCryptoController |
| 3.0.11 | 21.12.2018 |  | HTML Validator pre .NET 4.0  Doplnené príklady pre volanie služby potvrdenia prítomnosti a dovysvetlenie správnej implementácie.  Opravený text v kapitole 5.4.4.2.cez revíziu |
| 3.0.12 | 3.7.2019 | Pavol Sidó | Upravená kapitola Predpoklady a požiadavky klientského komponentu EhealthCryptoController – odstránenie duplicitných informácií  Úprava správnej implementácie volania na výzvu |
| 3.0.13 | 8.9.2020 | Integrácia ISPZS | Odstránená časť Predpokladané zmeny v špecifikácii |
| 3.0.14 | 2.11.2020 | Integrácia ISPZS | doplnené v kapitole 6.1.8 mapovanie stránok EZKO na paragrafové znenie zákona, ktoré špecifikuje, aká množina údajov reprezentuje datá stránka EZKO  doplnené v kapitole 5.5 resp. 5.5.1 ošetrenie SOAP Fault výnimky, ktorú môže vracať bezpečnostný komponent XG45 |
| 3.0.15 | 19.5.2021 | Konz | Doplnená kapitola 5.7 popisujúca dešifrovanie PDF dokumentu digitálneho covid certifikátu |
| 3.0.16 | 1.11.2021 | Integrácia ISPZS | Doplnenie textu pre chybové hlášky v kapitole 5.3.5.  Doplnenie textu v podkapitole 6.1.1.7. |
| 3.0.17 | 11.4.2022 | Konz | Upresnenie popisu výnimiek E999987 a E999988 – chyba archivačného modulu |

**Akceptačné riadenie**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Odberateľ** | | | **Dodávateľ** | | |
| **Dátum** | **Meno (Rola)** | **Podpis** | **Dátum** | **Meno (Rola)** | **Podpis** |
|  |  |  |  |  |  |

**Súvisiace produkty a referenčné materiály**

|  |  |
| --- | --- |
| **ID** | **Názov dokumentu a referencia** |
|  |  |

**Charakteristika dokumentu**

Cieľom dokumentu je popísať integráciu na servisné rozhrania systému NZIS.

Dokument popisuje aké správy musí byť integrovaný systém schopný odoslať a prijať a na akých koncových bodoch.

Dokument popisuje bezpečnostné mechanizmy ako autentizáciu, podpisovanie a šifrovanie, ktoré musia byť použité, pre úspešnú implementáciu rozhrania.

Dokument má byť určený:

Solution architektom - aby porozumeli servisným rozhraniam a vedeli ich začleniť do návrhu systémov

Vývojárom - aby vedeli implementovať rozhrania do klientského systému v zhode so špecifikáciou rozhrania

Testerom - aby vedeli vyhodnotiť, či implemenácia rozhrania klientského systému zodpovedá špecifikácii rozhrania

# Obsah

[Obsah 5](#_Toc72385224)

[1. Manažérske zhrnutie 7](#_Toc72385225)

[2. Úvod 7](#_Toc72385226)

[3. Relevantné štandardy 8](#_Toc72385227)

[4. Definície pojmov 8](#_Toc72385228)

[5. Volanie služieb NZIS 8](#_Toc72385229)

[5.1 Synchrónne volanie 9](#_Toc72385230)

[5.2 Asynchrónne volanie 9](#_Toc72385231)

[5.3 Koncové body komunikácie 10](#_Toc72385232)

[5.3.1 Definované rozhranie 11](#_Toc72385233)

[5.3.2 Popis rozhrania IeHealthService 11](#_Toc72385234)

[5.3.3 Popis rozhrania IeHealthSyncService 13](#_Toc72385235)

[5.3.4 Popis rozhrania IeHealthServiceReadResponseManager a IeHealthServiceReadResponseManagerTyped 15](#_Toc72385236)

[5.3.5 Chybové stavy 19](#_Toc72385237)

[5.4 Spoločné štruktúry pre správy 21](#_Toc72385238)

[5.4.1 eHtalkMessage 21](#_Toc72385239)

[5.4.2 EHRExtract 27](#_Toc72385240)

[5.4.3 XSD Služby 33](#_Toc72385241)

[5.4.4 Identifikácia záznamov 33](#_Toc72385242)

[5.4.5 Spoločné dátové typy 38](#_Toc72385243)

[5.4.6 HTML text 41](#_Toc72385244)

[5.4.7 Používanie nepovinných atribútov 42](#_Toc72385245)

[5.4.8 Synchronizácia času 42](#_Toc72385246)

[5.5 Reakcia na chyby v komunikácii 42](#_Toc72385247)

[5.5.1 Zachytenie SOAP Fault výnimky 43](#_Toc72385248)

[5.6 Tlač EZKO 44](#_Toc72385249)

[5.6.1 Ukážka spracovania odpovede zo služby VyhladajZaznamyEZKTlac 45](#_Toc72385250)

[5.7 Dešifrovanie PDF dokumentu digitálneho covid certifikátu 46](#_Toc72385251)

[6. Bezpečnostné mechanizmy 47](#_Toc72385252)

[6.1.1 Verejné rozhranie komponentu IEhealthCryptoController 47](#_Toc72385253)

[6.1.2 Špecifikácia technologickej platformy klientskeho komponentu EhealthCryptoController. 65](#_Toc72385254)

[6.1.3 Zadávanie PIN kódu pri použití elektronickych preukazov ZPr 66](#_Toc72385255)

[6.1.4 Predpoklady a požiadavky klientského komponentu EhealthCryptoController. 66](#_Toc72385256)

[6.1.5 Kontinuita poskytovania služieb 67](#_Toc72385257)

[6.1.6 Elektronický podpis 68](#_Toc72385258)

[6.1.7 Potvrdenie prítomnosti pacienta 69](#_Toc72385259)

[6.1.8 Riadenie prístupu k údajom pacientov – Consent Management 69](#_Toc72385260)

[7. Sieťová komunikácia 70](#_Toc72385261)

[7.1 Sieťová komunikácia na služby NZIS 70](#_Toc72385262)

[7.1.1 Zabezpečenie komunikácie medzi IS PZS a NZIS 70](#_Toc72385263)

[7.2 Inštalácia čítačky ePZP kariet 71](#_Toc72385264)

[7.3 Inštalácia ovládačov ePZP kariet 71](#_Toc72385265)

[7.4 Inštalácia dôveryhodného certifikátu certifikačnej autority NCZI 71](#_Toc72385266)

[7.5 Sumarizácia krokov potrebných k volaniu vystavenej služby 72](#_Toc72385267)

[7.5.1 Naplnenie informácii o odosielateľovi – SenderInfo. 72](#_Toc72385268)

[7.5.2 Príprava tela správy 72](#_Toc72385269)

[7.5.3 Získanie identity pacienta 72](#_Toc72385270)

[7.5.4 Príprava identifikačných údajov správy 73](#_Toc72385271)

[7.5.5 Volanie služby 73](#_Toc72385272)

[7.5.6 Získanie výsledku 73](#_Toc72385273)

[7.6 Popis príkladov 73](#_Toc72385274)

[8. HTML validátor 73](#_Toc72385275)

# Manažérske zhrnutie

Dokument je časťou špecifikácie, ktorá definuje štandardnú sadu správ a mechnanizmov pre ich výmenu ktorá umožní integráciu informačných systémov poskytovateľov zdravotnej starostlivosti (ambulancie, lekárne, laboratóriá) tak by boli pokryté procesy definované v rámci produktu *x005 - Model biznis procesov*.

Špecifikácia správ je vytvorená na základe informácií obsiahnutých v produktoch *003 – Model prípadov použitia* a *004 – Komponentový model.* Pre definíciu mechanizmov výmeny správ čerpá špecifikácia informácie z produktu *011 – Systémová architektúra*.

Kompletná špecifikácia produktu x070 pozostáva z následovných dokumentov:

1. Detailná špecifikácia rozhrania – úvodný dokument obsahujúci informácie o štrukturovaní a spôsobe čítania špecifikácie**.**
2. Volanie služieb - Dokument definujúci sadu mechanizmov výmeny správ- **Tento dokument.**
3. Prehľad služieb – Dokument popisuje sadu dostupných služieb, ich základné atribúty, popis.

# Úvod

**Kapitola 3**

Popisuje relevantné štandardy

**Kapitola 4**

Obsahuje definície pojmov

**Kapitola 5**

Detailne popisuje princípy volania služieb

**Kapitola 7**

Detailne popisuje kroky postupu volania služby IS PZS

**Kapitola 8**

Zoznam predpokladaných zmien v ďalších verziách produktu.

# Relevantné štandardy

|  |  |
| --- | --- |
| Štandard | Využitie |
| Http 1.1 | Volanie webových služieb |
| SOAP 1.2 | Protokol pre výmenu informácií v rámci webových služieb |
| XMLSignature | Špecifikácia pre elektronické podpisovanie časti XML dokumentu |
| XMLEncryption | Špecifikácia pre šifrovanie časti XML dokumentu |
| WS-Security | Rozšírenie SOAP, ktoré umožnuje aplikovať bezpečnostné vlastnoti na volanie webových služieb |
| WS-Trust | Špecifikácia na vydávanie, obnovovanie a overovanie bezpečnostných tokenov, ktorá umožňuje rozšírenie WS-Security |
| RFC 4122 | Špecifikácia pre Univerzálne jedinečný identifikátor |
| EN13606 | Vytváranie správ obsahujúcich zdravotné záznamy |
| Luhn\_algorithm | Algoritmus kontolného súčtu čísiel |

Tabuľka 1 - Relevantné štandardy

# Definície pojmov

|  |  |
| --- | --- |
| Pojem | Definícia |
| Používateľ služby | Program, ktorý volá webovú službu poskytovanú poskytovateľom. Používateľ vystupuje ako klient. |
| Poskytovateľ služby | Program, ktorý poskytuje webovú službu. Vystupuje ako server. |
| Musí | Indikuje, že tvrdenie v rámci špecifikácie je povinné. |
| Mal by | Indikuje doporučenie pre implementátora. |
| SOAP | Protokol pre výmenu štruktúrovaných dát založených na XML správach v distribuovanom počítačovom prostredí. |
| Externý identifikátor  (extID) | Identifikátor záznamu používaný na komunikáciu. |
| Identifikátor číselníkovej položky | Identifikátor položky číselníka používaný na komunikáciu. |
| Zdroj záznamu | Miesto, kde záznam vznikol. |
| JRUZ identifikator | Identifikátor položky číselníka získaná zo služieb JRUZ. |

Tabuľka 2 - Definície pojmov

# Volanie služieb NZIS

Volanie služieb NZIS je vždy iniciované zo strany externého IS. Pre jednotlivé služby NZIS je definovaný spôsob, akým môže prebiehať komunikácia. Pre služby poskytujúce jednoduchú funkcionalitu je možné zaručiť rýchlu odozvu s cieľom zabezpečiť komunikáciu synchrónnym spôsobom. Pre komplexnejšie služby je vzhľadom na optimalizáciu využívania systémových zdrojov je možný asynchrónny spôsob komunikácie. Integrujúci sa informačný systém musí používať spôsob komunikácie špecifikovaný pre jednotlivé služby.

## Synchrónne volanie

Pri synchónnom volaní vystupuje externý informačný systém v roli používateľa služby. Vytvorí zodpovedajúcu správu a odosiela ju na koncový bod NZIS pre synchrónnu komunikáciu. Zo systému NZIS dostáva na správu priamo odpoveď.

## Asynchrónne volanie

Pri asynchrónnom volaní pri iniciáci požiadavky na službu vystupuje informačný systém v roli používateľa služby.

Postup:

1. Informačný systém vytvorí zodpovedajúcu správu a odosiela ju na koncový bod NZIS pre asynchrónnu komunikáciu.
2. Zo systému NZIS dostáva potvrdenie prijatia správy.
3. Informačný systém po odoslaní požiadavky čaká najmenej 3 sekundy, následne oslovuje NZIS na koncovom bode „ResponseManager“, kde sa volaním služby získa odpoveď.
4. Ak bola požiadavka systémom NZIS spracovaná, odpoveď je dostupná.
5. V prípade, ak odpoveď nie je dostupná, musí informačný systém opakovať požiadavku po ďalších 3 sekundách.



## Koncové body komunikácie

Z pohľadu externých systémov budú pre ESO1 vystavené do internetu nasledujúce služby:

|  |  |  |  |
| --- | --- | --- | --- |
| Služba | Interface | Metóda | URI |
| ExternalService | IeHealthService | **Receive** | http |
| ExternalSyncService | IeHealthSyncService | **GetData** | http |
| ExternalRMService | IeHealthServiceReadResponseManager | **GetData** | http |
| ExternalRMService | IeHealthServiceReadResponseManagerTyped | **GetData** | http |

Tabuľka 3 - Vystavené služby

**UPOZORNENIE vystavené služby platné s novým certifikátom platným od 25.8.2017 su uvedené v tabuľke nižšie**

|  |  |  |  |
| --- | --- | --- | --- |
| Služba | Interface | Metóda | URI |
| ExternalService | IeHealthService | **Receive** | http |
| ExternalSyncService | IeHealthSyncService | **GetData** | http |
| ExternalRMService | IeHealthServiceReadResponseManager | **GetData** | http |
| ExternalRMService | IeHealthServiceReadResponseManagerTyped | **GetData** | http |

DNS meno servera: **services.infra.npz.sk** (services.preprod.npz.sk).pre certifikát platný do 30.9.2017

DNS meno servera: **services1.infra.npz.sk** (services2017.preprod.npz.sk).pre certifikát platný od 25.8.2017

DNS meno servera pre SSL komunikáciu : **ssl-zpr.npz.sk** (sslpzs.preprod.npz.sk)

**IAM ADFS služby pre konfiguráciu Cryptocontrolera (IamStsServiceBaseUri):**

|  |  |
| --- | --- |
| <https://sts-zpr.infra.npz.sk/adfs/services/>  (https://sts-zpr.preprod.npz.sk/adfs/services/) | Komunikácia cez SSL |
| <https://sts-zpr.infra.npz.sk/adfs/services/>  (https://sts-zpr.preprod.npz.sk/adfs/services/) | Komunikácia cez VPN |

Špecifikovanie finálnych URL k WSDL je predmetom integračného balíka. WSDL schémy sú uvedené v prílohe “Prilohy\Schemy\COMMON\WSDL”.

### Definované rozhranie

Funkcionalita systému NZIS je externým systémom dostupná prostredníctvom nasledovných služieb:

* WSDL ExternalService – rozhranie IeHealthService
* WSDL ExternalSyncService – rozhranie IeHealthSyncService
* WSDL ExternalRMService – rozhranie IeHealthServiceReadResponseManager
* WSDL ExternalRMServiceTyped – rozhranie IeHealthServiceReadResponseManagerTyped

### Popis rozhrania IeHealthService

**IeHealthService** je asynchrónne rozhranie smerovacej komponenty. Webová služba asynchrónneho rozhrania zabezpečuje spracovanie prichádzajúcich správ a ich následné smerovanie na ostatné služby.

Z pohľadu externého systému ako konzumenta služby poskytuje rozhranie **IeHealthService** službu **Service** prostredníctvom metódy **Receive** s jedným vstupným parametrom typu **eHTalkMessage** (popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage**).

Autentifikácia voči tejto službe bude totožná ako v prípade iných externých endpointov NZIS.

#### Metóda „Receive“

Zabezpečuje spracovanie asynchrónnej požiadavky na základe vyplnenej štruktúry **eHtalkMessage**.

##### Vstupné parametre

|  |  |  |
| --- | --- | --- |
| Parameter | Dátovy typ | Popis |
| message | eHtalkMessage | popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage** |

Tabuľka 4 - Vstupné parametre

##### Výstupná hodnota

Výstupom volania metódy *Receive* je objekt typu **eHtalkMessage** (popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage**).

##### Chybové stavy

V prípade chyby interného spracovania webová metóda odpovedá validnou eHtalkMessage štruktúrov, v ktorej je uvedený aj stav spracovania výsledku. Stav spracovania sa nachádza v návratovej štruktúre v časti Result -> Code. Zoznam všetkých chybových stavov je popísaný v tabuľke

| Code | Result | Popis chyby | Error/ warning |
| --- | --- | --- | --- |
| E999973 | Požadovaná služba nie je naďalej poskytovaná | Používate starú verziu služby Systému eZdravie. Požiadajte svojho dodávateľa o aktualizáciu informačného systému. Do dodania aktualizácie je potrebné všetky záznamy vytlačiť. | W |
| E999975 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | W |
| E999976 | Chyba pri volaní koncovej služby |  | E |
| E999977 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | E |
| E999983 | Chyba v podpise správy | Nevalidný podpis | W |
| E999984 | Chyba pri volaní autorizačného modulu | Neznama exception pri ziskavani autorizačného tokenu (napr. aut. sluzba nie je dostupná) | E |
| E999986 | Chyba pri volaní archivačného modulu | Nastal interný problém pri volaní služby | W |
| E999987 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | W |
| E999988 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | E |
| E999989 | Chyba RM | Nastal interný problém pri volaní služby | E |
| E999990 | Chyba RM | Sprava neexistuje | W |
| E999991 | Chyba v Security Token-e |  | W |
| E999992 | Chyba v Security Token-e | Neznama exception pri extrahovani SAML tokenu | E |
| E999973 | Chyba v Security Token-e | Časová platnosť SAML tokenu vypršala | E |
| E999993 | Požadovaná služba má odstávku | Požiadavku nie je možné vykonať z dôvodu prebiehajúcej údržby systému eZdravie. Pokračujte ďalej v práci s Vašim IS a je potrebné všetky záznamy vytlačiť. | W |
| E999994 | Neautorizovaný prístup k službe | Nepovolený prístup. Prosím skontrolujte si prihlasovacie údaje a údaje nahlásené do Národného registra zdravotníckych pracovníkov (tzv. pracovno-právne vzťahy). Ak sú údaje správne, kontaktujte NCZI a nahláste chybu. | W |
| E999995 | Chyba formátu správy | Nastal interný problém pri volaní služby | E |
| E999996 | Chyba formátu správy | Chybný formát CorrID | W |
| E999997 | Chyba formátu správy | Chybný formát MsgID | W |
| E999998 | Chyba formátu správy | Prázdny element ClassName | W |
| E999999 | Neočakávana chyba | Nastal interný problém pri volaní služby | E |

Tabuľka 10 – Chybové stavy spracovania požiadaviek.

### Popis rozhrania IeHealthSyncService

**IeHealthSyncService** je synchrónne rozhranie smerovacej komponenty. Webová služba synchrónneho rozhrania zabezpečuje spracovanie prichádzajúcich správ a ich následné smerovanie na ostatné služby.

Z pohľadu externého systému ako konzumenta služby poskytuje rozhranie **IeHealthSyncService** službu **SyncService** prostredníctvom metódy **GetData** s jedným vstupným parametrom typu **eHTalkMessage** (popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage**).

Autentifikácia voči tejto službe bude totožná ako v prípade iných externých endpointov NZIS.

#### Metóda „GetData“

Zabezpečuje spracovanie synchrónnej požiadavky na základe vyplnenej štruktúry **eHtalkMessage**.

##### Vstupné parametre

|  |  |  |
| --- | --- | --- |
| Parameter | Dátovy typ | Popis |
| message | eHtalkMessage | popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage** |

Tabuľka 5 - Vstupné parametre

##### Výstupná hodnota

Výstupom volania metódy *GetData* je objekt typu **eHtalkMessage** (popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage**).

##### Chybové stavy

V prípade chyby interného spracovania webová metóda odpovedá validnou eHtalkMessage štruktúrov, v ktorej je uvedený aj stav spracovania výsledku. Stav spracovania sa nachádza v návratovej štruktúre v časti Result -> Code. Zoznam všetkých chybových stavov je popísaný v tabuľke

| Code | Result | Popis chyby | Error/ warning |
| --- | --- | --- | --- |
| E999973 | Požadovaná služba nie je naďalej poskytovaná | Požadovaná služba nie je naďalej poskytovaná | W |
| E999975 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | W |
| E999976 | Chyba pri volaní koncovej služby |  | E |
| E999977 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | E |
| E999983 | Chyba v podpise správy | Nevalidný podpis | W |
| E999984 | Chyba pri volaní autorizačného modulu | Neznama exception pri ziskavani autorizačného tokenu (napr. aut. sluzba nie je dostupná) | E |
| E999986 | Chyba pri volaní archivačného modulu | Nastal interný problém pri volaní služby | W |
| E999987 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | W |
| E999988 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | E |
| E999989 | Chyba RM | Nastal interný problém pri volaní služby | E |
| E999990 | Chyba RM | Sprava neexistuje | W |
| E999991 | Chyba v Security Token-e |  | W |
| E999992 | Chyba v Security Token-e | Neznama exception pri extrahovani SAML tokenu | E |
| E999973 | Chyba v Security Token-e | Časová platnosť SAML tokenu vypršala | E |
| E999993 | Požadovaná služba má odstávku | Požadovaná služba má odstávku | W |
| E999994 | Neautorizovaný prístup k službe | Neautorizovaný prístup k službe | W |
| E999995 | Chyba formátu správy | Nastal interný problém pri volaní služby | E |
| E999996 | Chyba formátu správy | Chybný formát CorrID | W |
| E999997 | Chyba formátu správy | Chybný formát MsgID | W |
| E999998 | Chyba formátu správy | Prázdny element ClassName | W |
| E999999 | Neočakávana chyba | Nastal interný problém pri volaní služby | E |

Tabuľka 10 – Chybové stavy spracovania požiadaviek.

### Popis rozhrania IeHealthServiceReadResponseManager a IeHealthServiceReadResponseManagerTyped

Response Manager je komponent zabezpečujúci dočasné úložisko výstupných správ pre externé systémy, ktoré neboli doručené prijímateľovi z dôvodu jeho nedostupnosti príp. neexistencie externého endpointu. Pre externé systémy, ktoré nedisponujú možnosťou vystaviť koncový bod, kam by mohol NZIS doručiť odpoveď (napríklad informačné systémy za proxy serverom a pod.), existuje možnosť asynchrónnej komunikácie, kde je odpoveď pre externý systém po spracovaní uložená v NZIS a prevzatie odpovede je realizované ako samostatné volanie NZIS. Externý systém posiela požiadavku avšak neuvádza logickú adresu v elemente **ChannelInfoReply** v správe **EHTalkMessage** kam má byť odoslaná odpoveď.

Odpoveď bude uchovaná v komponente RM 60 minút a následne bude automaticky vymazaná.

Z pohľadu externého systému ako konzumenta služby bude komponent ResponseManager poskytovať službu **ReadMessageExternal** a **ReadMessageTypedExternal** prostredníctvom metódy **GetData** s jedným vstupným parametrom typu string. Služba je implementáciou rozhrania **IeHealthServiceReadResponseManager** pre službu ReadMessageExternal a **IeHealthServiceReadResponseManagerTyped** pre službu ReadMessageTypedExternal, ktoré obsahujú jednu metódu **GetData**. Na základe vstupného parametra **CorrelationID** bude vyhľadaná správa v dočasnom úložisku a v prípade existencie správy bude dodaná ako výsledok volania v nasledujúcom formáte:

* Pre službu ReadMessageExternal vo forme XElement. Interný formát správy v XElement-e je možné na strane klienta deserializovať do objektu **eHtalkMessage** (popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage**). Autentifikácia voči tejto službe bude totožná ako v prípade iných externých endpointov NZIS.
* Pre službu ReadMessageTypedExternal vo forme **eHtalkMessage**. Autentifikácia voči tejto službe bude totožná ako v prípade iných externých endpointov NZIS.

Z pohľadu autorizácie sa bude jednať o dodanie existujúcej správy na základe CorrelationID ako vstupného parametra pri volaní služby **ReadMessageExternal** a **ReadMessageTypedExternal**.

#### Metóda „GetData“ – služba ReadMessageExternal

Zabezpečuje volanie komponenty Response Managera na základe CorrelationID.

##### Vstupné parametre

|  |  |  |
| --- | --- | --- |
| Parameter | Dátovy typ | Popis |
| CorrelationID | string | Alfranumerický reťazec identifikujúci požadovanú správu, ktorá bola pomocou asynchronného volania poslaná prostredníctvom asynchronného rozhrania na NZIS. Formát a hodnota parametra má byť rovnaká ako tomu je pri odpovedi o z asynchronného rozhrania (pozri kapitolu 5.4.1.2 Header časť CorrelationID). |

Tabuľka 6 - Vstupné parametre

##### Výstupná hodnota

Výstupom volania metódy *GetData* je objekt typu **XElement**, reprezentujúci XML štruktúru objektu **eHtalkMessage** (popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage**).

##### Chybové stavy

Response Manager bude pri dotazovaný odpovedať štruktúrov XElement pričom element Result -> Code bude obsahovať nasledujúce možné stavy:

|  |  |  |
| --- | --- | --- |
| Code | Message | Popis |
| 0 |  | Korektné spracovanie požiadavky. Žiadna chyba nenastala ani počas spracovania asynchrónnej požiadavky a ani počas dotazovania RM. |
| E999990 |  | Požadovaná správa v RM neexistuje. Externý klient by sa môže opakovane dotazovať po uplynutí časového intervalu minimálne 30 sekúnd. |
| E999989 | Exception | Iná chyba RM. V časti Result -> Message a Result -> Details sa bude nachádzať detailneší popis chyby. |

Tabuľka 7 – Chybové stavy odpovede

V prípade, že Response Manager poskytne odpoveď pričom výsledom Result -> Code obsahuje iný stav ako je popísaný v tabuľke, jedná sa o chybu spracovania samotného volania biznis požiadavky. Zoznam všetkých chybových stavov sa nachádza v tabuľke

| Code | Result | Popis chyby | Error/ warning |
| --- | --- | --- | --- |
| E999973 | Požadovaná služba nie je naďalej poskytovaná | Požadovaná služba nie je naďalej poskytovaná | W |
| E999975 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | W |
| E999976 | Chyba pri volaní koncovej služby |  | E |
| E999977 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | E |
| E999983 | Chyba v podpise správy | Nevalidný podpis | W |
| E999984 | Chyba pri volaní autorizačného modulu | Neznama exception pri ziskavani autorizačného tokenu (napr. aut. sluzba nie je dostupná) | E |
| E999986 | Chyba pri volaní archivačného modulu | Nastal interný problém pri volaní služby | W |
| E999987 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | W |
| E999988 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | E |
| E999989 | Chyba RM | Nastal interný problém pri volaní služby | E |
| E999990 | Chyba RM | Sprava neexistuje | W |
| E999991 | Chyba v Security Token-e |  | W |
| E999992 | Chyba v Security Token-e | Neznama exception pri extrahovani SAML tokenu | E |
| E999973 | Chyba v Security Token-e | Časová platnosť SAML tokenu vypršala | E |
| E999993 | Požadovaná služba má odstávku | Požadovaná služba má odstávku | W |
| E999994 | Neautorizovaný prístup k službe | Neautorizovaný prístup k službe | W |
| E999995 | Chyba formátu správy | Nastal interný problém pri volaní služby | E |
| E999996 | Chyba formátu správy | Chybný formát CorrID | W |
| E999997 | Chyba formátu správy | Chybný formát MsgID | W |
| E999998 | Chyba formátu správy | Prázdny element ClassName | W |
| E999999 | Neočakávana chyba | Nastal interný problém pri volaní služby | E |

Tabuľka 10 – Chybové stavy spracovania požiadaviek.

#### Metóda „GetData“ – služba ReadMessageTypedExternal

Zabezpečuje volanie komponenty Response Managera na základe CorrelationID.

##### Vstupné parametre

|  |  |  |
| --- | --- | --- |
| Parameter | Dátovy typ | Popis |
| CorrelationID | string | Alfranumerický reťazec identifikujúci požadovanú správu, ktorá bola pomocou asynchronného volania poslaná prostredníctvom asynchronného rozhrania na NZIS. Formát a hodnota parametra má byť rovnaká ako tomu je pri odpovedi o z asynchronného rozhrania (pozri kapitolu 5.4.1.2 Header časť CorrelationID). |

Tabuľka 8 - Vstupné parametre

##### Výstupná hodnota

Výstupom volania metódy *GetData* je objekt typu **eHtalkMessage** (popis štruktúry sa nachádza v kapitole **5.4.1**. **eHtalkMessage**).

##### Chybové stavy

Response Manager bude pri dotazovaný odpovedať štruktúrov XElement pričom element Result -> Code bude obsahovať nasledujúce možné stavy:

|  |  |  |
| --- | --- | --- |
| Code | Message | Popis |
| 0 |  | Korektné spracovanie požiadavky. Žiadna chyba nenastala ani počas spracovania asynchrónnej požiadavky a ani počas dotazovania RM. |
| E999990 |  | Požadovaná správa v RM neexistuje. Externý klient by sa môže opakovane dotazovať po uplynutí časového intervalu minimálne 30 sekúnd. |
| E999989 | Exception | Iná chyba RM. V časti Result -> Message a Result -> Details sa bude nachádzať detailneší popis chyby. |

Tabuľka 9 – Chybové stavy odpoveďe

V prípade, že Response Manager poskytne odpoveď pričom výsledom Result -> Code obsahuje iný stav ako je popísaný v tabuľke, jedná sa o chybu spracovania samotného volania biznis požiadavky. Zoznam všetkých chybových stavov sa nachádza v tabuľke

| Code | Result | Popis chyby | Error/ warning |
| --- | --- | --- | --- |
| E999973 | Požadovaná služba nie je naďalej poskytovaná | Požadovaná služba nie je naďalej poskytovaná | W |
| E999975 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | W |
| E999976 | Chyba pri volaní koncovej služby |  | E |
| E999977 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | E |
| E999983 | Chyba v podpise správy | Nevalidný podpis | W |
| E999984 | Chyba pri volaní autorizačného modulu | Neznama exception pri ziskavani autorizačného tokenu (napr. aut. sluzba nie je dostupná) | E |
| E999986 | Chyba pri volaní archivačného modulu | Nastal interný problém pri volaní služby | W |
| E999987 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | W |
| E999988 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | E |
| E999989 | Chyba RM | Nastal interný problém pri volaní služby | E |
| E999990 | Chyba RM | Sprava neexistuje | W |
| E999991 | Chyba v Security Token-e |  | W |
| E999992 | Chyba v Security Token-e | Neznama exception pri extrahovani SAML tokenu | E |
| E999973 | Chyba v Security Token-e | Časová platnosť SAML tokenu vypršala | E |
| E999993 | Požadovaná služba má odstávku | Požadovaná služba má odstávku | W |
| E999994 | Neautorizovaný prístup k službe | Neautorizovaný prístup k službe | W |
| E999995 | Chyba formátu správy | Nastal interný problém pri volaní služby | E |
| E999996 | Chyba formátu správy | Chybný formát CorrID | W |
| E999997 | Chyba formátu správy | Chybný formát MsgID | W |
| E999998 | Chyba formátu správy | Prázdny element ClassName | W |
| E999999 | Neočakávana chyba | Nastal interný problém pri volaní služby | E |

Tabuľka 10 – Chybové stavy spracovania požiadaviek.

### Chybové stavy

Každá požiadavka na externé rozhranie webovú službu v riešení NZIS používa na nižšej úrovni HTTP protokol. Odpoveď na každú požiadavku sprevádza aj stav s akým bola daná požiadavka na úrovni HTTP spracovaná. Na nasledujúcej linke je zoznam stavov, ktorými štandardne odpovedá IIS 7.0 a 7.5 server: <http://support.microsoft.com/kb/943891>.

V prípade, že na úrovni HTTP je návratový kód 200, tzn. požiadavka klienta bola na úrovni http spracovaná úspešne. Úspešnosť/neúspešnosť spracovania na aplikačnej úrovni je klientovi zasielaná prostredníctvom odpovede v eHtalkMessage štruktúre v časti Result -> Code. V nasledujúcej tabuľke je popis spoločných stavov spracovania spolu s identifikáciou služby, ktorá daný stav v eHtalkMessage struktúre môže obsahovať. Okrem týchto chybových stavov systém vracia výnimky špecifické pre konkrétne služby. Ich popis je uvedený v rámci popisu služby (Dokument *Prehľad služieb*) a detailný zoznam je uvedený v prílohe *x070\_Exceptions\_list.xls*.

Popis chyby pri výnimkách je určený pre vývojárov a dizajnérov IS PZS. Nemá byť preposielaný priamo na koncového používateľa IS PZS.

| Code | Result | Popis chyby | Error/ warning |
| --- | --- | --- | --- |
| E999973 | Požadovaná služba nie je naďalej poskytovaná | Požadovaná služba nie je naďalej poskytovaná | W |
| E999975 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | W |
| E999976 | Chyba pri volaní koncovej služby |  | E |
| E999977 | Chyba pri volaní koncovej služby | Nastal interný problém pri volaní služby | E |
| E999983 | Chyba v podpise správy | Nevalidný podpis | W |
| E999984 | Chyba pri volaní autorizačného modulu | Neznama exception pri ziskavani autorizačného tokenu (napr. aut. sluzba nie je dostupná) | E |
| E999986 | Chyba pri volaní archivačného modulu | Nastal interný problém pri volaní služby | W |
| E999987 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | W |
| E999988 | Chyba pri volaní archivačného modulu | Digitálny podpis správy nebol overený (nesprávny elektronický podpis) alebo ESID pacienta nie je previazaný na SAML token v hlavičke eHtalk správy (pravdepodobne po reautentifikácii nedošlo k pregenerovaniu ESID s novým SAML tokenom). | E |
| E999989 | Chyba RM | Nastal interný problém pri volaní služby | E |
| E999990 | Chyba RM | Sprava neexistuje | W |
| E999991 | Chyba v Security Token-e |  | W |
| E999992 | Chyba v Security Token-e | Neznama exception pri extrahovani SAML tokenu | E |
| E999973 | Chyba v Security Token-e | Časová platnosť SAML tokenu vypršala | E |
| E999993 | Požadovaná služba má odstávku | Požadovaná služba má odstávku | W |
| E999994 | Neautorizovaný prístup k službe | Neautorizovaný prístup k službe | W |
| E999995 | Chyba formátu správy | Nastal interný problém pri volaní služby | E |
| E999996 | Chyba formátu správy | Chybný formát CorrID | W |
| E999997 | Chyba formátu správy | Chybný formát MsgID | W |
| E999998 | Chyba formátu správy | Prázdny element ClassName | W |
| E999999 | Neočakávana chyba | Nastal interný problém pri volaní služby | E |

Tabuľka 10 – Chybové stavy spracovania požiadaviek

## Spoločné štruktúry pre správy

Pre komunikáciu na všetkých službách je využívaná jednotná dátová štruktúra. Na najvyššej úrovni je správa definovaná štrukúrou **eHTalkMessage** (XSD schéma je v prílohe – eHTalkMessage.xsd).

**eHtalkMessage** je XML dokument. Štruktúra správy na úrovni logiky vychádza z navrhovaného štandardu (v rámci Komisie pre štandardizáciu informačných systémov verejnej správy) ***SkTalk***. **eHtalkMessage** nie je bez nutnosti dodatočnej transformácie s navrhovaným štandardom kompatibilné s ***SKTalk*** (iný namespace a iný názov koreňového elementu).

V nasledujúcich kapitolách je podrobne popísaná štruktúra **eHtalkMessage**, ako aj jednotlivé spôsoby výmeny záznamov v dátovej časti správy.

### eHtalkMessage



Obrázok 1 - eHtalkMessage

#### EnvelopeVersion

EnvelopeVersionje elementom obsahujúcim použitú verziu formátu pre komunikáciu. Pre verziu Ehtalk 2.0 bude obsahovať číslo „**2.0**“. Tento xml element je povinný.

Pre komunikáciu s NZIS musí element obsahovať konštantnú hodnotu reťazec „2.0“

Príklad:

Zápis bude vyzerať nasledovne:

<EnvelopeVersion>2.0</EnvelopeVersion>

#### Header

Táto časť je určená pre evidovanie typu správy, spôsobov doručenia správy a definície identifikácie odosielateľa správy. Tiež obsahuje možnosti pre identifikáciu postupov spracovania danej správy zo strany jednotlivých navzájom komunikujúcich systémov.

**MessageInfo**

Oblasť evidujúca základné atribúty potrebné pre určenie typu požadovaného spracovania, identifikáciu daného dokumentu a väzby na prípadné predchádzajúce dokumenty. Tiež obsahuje doručovacie adresy pre príjem tohto dokumentu a spiatočné odoslanie daných výsledkov spracovania dokumentu.

**Class**

Element Class je primárny identifikátor typu správy, podľa ktorého moduly EHEALTH a prijímateľ rozpoznávajú obsah dokumentu a na základe ktorého sa riadi spracovanie, overenie a smerovanie danej správy. Každý typ správy musí mať jedinečný identifikátor typu správy.

Element Class musí obsahovať reťazec zodpovedajúci volanej službe uvedený pri danej službe v položke „Názov služby“ uvedenej v dokumente X070 Detailná špecifikácia rozhrania – prehľad služieb.

Príklad:

<Class>DajPacientskySumar </Class>

**MessageID**

Identifikátor GUID jednoznačne identifikuje správu.

Element musí obsahovať náhodný identitikátor (GUID) vygenerovaným počas vytvárania správy v súlade s rfc4122. [[1]](#footnote-2)

Príklad:

Formát zápisu je 00000000-0000-0000-0000-000000000000.

Pre platformu Microsoft.net je možné guid vygenerovať následovným postupom:

Pri pretypovaní typu GUID na string je potrebné používať formátovanie „D“ podľa špecifikácie <http://msdn.microsoft.com/en-us/library/97af8hh4.aspx>.

Zápis:

var messageId = Guid. NewGuid.ToString(“D”);

**CorrelationID**

Identifikátor CorrelationID slúži na vzájomné previazanie všetkých správ týkajúcich sa jednej biznis služby. Jedná sa o náhodne vygenerované 128bitové číslo na strane NZIS. Napríklad v prípade asynchrónnej komunikácie bude externý systém spájať odoslanú požiadavku s prijatými odpoveďami na základe tohoto indentifikárora. Pri odosielaní požiadavky externým systémom na službu ESO1 budú identifikátory MessageID a CorrelationID zhodné a pri odpovedi z NZIS bude tento identifikátor predpísaný stranou NZIS.

**BusinessID**

BusinessID je identifikátor, ktorý identifikuje biznis správu, pričom tento identifikátor musí byť v tvare:

<xsd:restriction base="xsd:string">

<xsd:pattern value="^[0-9a-zA-Z-\_.]\*$" />

</xsd:restriction>

Jeho obsah/formát určuje poskytovateľ služby. V prípade požiadavky o opätovné zopakovanie biznis služby napr. z dôvodu chybného spracovania je potrebné uviesť tento parameter na strane klienta.

**ChannelInfoReply**

Obsahuje adresy, ktoré čakajú príjem výsledkov spracovania dokumentu.

Obsahuje logický názov endpoint-u, na ktorý má byť odpoveď po spracovaní odoslaná. V prípade, že tento element je prázdny bude odpoveď dočasne uložená v ESO1 a žiadateľ si ju môže vyzdvihnúť volaním služby ESO1.ResponseManager.svc.

Vzor vyplneného elementu:

<ChannelInfoReply>IS\_PZS\_Kramare</ChannelInfoReply>

Predtým, ako začne klient vypĺňať tento element je potrebné, aby si tento logický endpoint zaregistroval v ESO1. Registrácia logického endpoint-u nie je v aktuálnej verzii sieťovou infraštruktúrou podporovaná.

**SenderInfo**

Obsahuje konkrétne údaje potrebné pre identifikáciu odosielajúceho používateľa. Tento element je pre externé systémy povinný pozri kapitolu 8.1.1 Naplnenie informácii o odosielateľovi – SenderInfo.

**SecurityToken**

Pozri kapitolu 6.1.7 Potvrdenie prítomnosti pacienta.

**UserContext**

Údaje o identite, ktorá správu odoslala a v mene koho odoslala. Schéma „UserContext.xsd“ je súčasťou XSD príloh tohto produktu.

#### Body

Definovaná časť slúži na prenos informácii špecifických pre konkrétnu elektronickú službu a obsahuje dva elementy:

Obrázok 2 - Štruktúra element Body

Jedná sa o povinný element.

**Data**

**Pri volané služby musí externý systém vypĺňať tento element v súlade s definíciou volanej služby (Schémy sú uložené v prílohach dokumentu).**

Ich štruktúru predpisuje poskytovateľ služby vo forme XSD schémy.

V tele tohoto elementu je v závislosti od použitej služby štruktúra napr. **EHRExtract**, alebo štruktúra daná **XSD schémou** definovanou pre službu. V prípade štruktúry **EHRExtract** je presný obsah definovaný archetypom pre danú službu.

Záznamy sú vymieňané týmto spôsobom oboma smermi, pričom ich vzájomná kombinácia nie je možná. Avšak služba môže spracovávať požiadavku v jednom formáte a odpovedať v inom.

Zoznam služieb a ich typy dátových častí sú špecifikované v dokumente „Prehľad služieb“ (*eSO1\_SP\_x070B\_Detailna\_specifikacia\_rozhrania\_Prehlad\_sluzieb\_20120328\_1.0.docx*).

Obrázok 3 - Spôsob výmeny záznamov

V prípade štruktúry EHRExtract je presný obsah definovaný archetypom pre danú službu (dátových štruktúr). Pre definíciu archetypov slúži norma CEN/STN EN13606, ktorá špecifikuje jazyk pre popis – ADL 1.4.

Detailný popis EHRExtract a XSD služby je popísany v kapitolách [5.4.2- EHRExtract](#_EHRExtract_1) a [5.4.3 - XSD služby](#_XSD_Služby_1) tohto dokumentu.

**Result**

V prípade volania služby informačný systém nesmie vypĺňať element.

V prípade úspešného volania obsahuje vrátená správa:

<Code>0</Code>

alebo upozornenie

<Code>Wxxxxxx</Code>

V prípade neúspešného spracovanie obsahuje chybu (Exxxxxx) s uvedením jej kódu a popisu. Služby nebudú generovať SOAP exception a namiesto toho budú posielať chybu ako návratovú hodnotu v správe eHTalkMessage.

**Korektné spracovanie správy**

V prípade, že volaná služba ESO1 korektne prijme požiadavku z extérneho systému pri **asynchrónnom** spracovaní odošle o tom informáciu späť vo forme:

<eHtalkMessage>

...

<Body>

<Result>

<Code>0</Code>

</Result>

</Body>

</ eHtalkMessage>

V prípade, že volaná služba ESO1 korektne spracuje požiadavku externého systému pri **synchrónnom** spracovaní odošle odpoveď späť vo forme:

<eHtalkMessage>

...

<Body>

<Data>

...

</Data>

<Result>

<Code>0</Code>

</Result>

</Body>

</ eHtalkMessage>

Príklad informácie o upozornení napr.:

<eHtalkMessage>

...

<Body>

<Data>

...

</Data>

<Result>

<Code>W900001</Code>

<Message>Z dôvodu obmedzenia prístupu Vám neboli poskytnuté všetky zdravotné záznamy. V prípade potreby, požiadajte pacienta o prístup k údajom vložením eID do čítačky a zadania súhlasu pacienta (stlačením OK/ zadaním BOK).</Message>

</Result>

</Body>

</ eHtalkMessage>

Okrem elementov **Code** a **Message** môže systém ESO1 vrátiť aj detailnejší popis upozornenia v elemente **Details** (viac viď schéma *Warnings.xsd*). Môže sa jednať napr. o zoznam aplikovaných obmedzení napr.:

<eHtalkMessage>

...

<Body>

<Data>

...

</Data>

<Result>

<Code>W900001</Code>

<Message>Z dôvodu obmedzenia prístupu Vám neboli poskytnuté všetky zdravotné záznamy. V prípade potreby, požiadajte pacienta o prístup k údajom vložením eID do čítačky a zadania súhlasu pacienta (stlačením OK/ zadaním BOK).</Message>

<Details>

   <Warnings>

<Warning>Obmedzenie:</Warning>

<Warning>Poskytnuté sú len záznamy do citlivosti 3 a zároveň záznamy vzniknuté do 1.1.2019 alebo</Warning>

<Warning>Poskytnuté sú len vlastné záznamy </Warning>

        </Warnings>

</Details>

</Result>

</Body>

</eHtalkMessage>

**Chybové spracovanie správy**

V tomto prípade bude do elementu **Code** vložený alfanumerický kód chyby podľa zoznamu chýb pre volanú službu. Element Message bude nepovinný parameter a bude ho vypĺňať systém ESO1 a zasielať na externý systém len v prípade, že rozširuje pôvodný význam chyby.

Príklad informácie o chybe napr.:

<eHtalkMessage>

...

<Body>

<Data>

...

</Data>

<Result>

<Code>E1001</Code>

<Message>Objekt neexistuje</Message>

</Result>

</Body>

</ eHtalkMessage>

Okrem elementov **Code** a **Message** môže systém ESO1 vrátiť aj detailnejší popis chyby v elemente **Details**. Môže sa jednať napr. o zoznam chýb po validácii vstupu napr.:

<eHtalkMessage>

...

<Body>

<Data>

...

</Data>

<Result>

<Code>E9005</Code>

<Message>Chybný vstup</Message>

<Details>

    <Detail>

              <Attribute>Meno</Attribute>

<Error>Obsahuje nepovolene znaky.</Error>

        </Detail>

     <Detail>

<Attribute>Vek</Attribute>

<Error>Musi byt numericka hodnota.</Error>

           </Detail>

<Detail>

<Attribute>Email</Attribute>

<Error>Zly format.</Error>

        </Detail>

<Detail>

<Attribute>KodIndikacie</Attribute>

<Error>Neexistujuci kod.</Error>

         </Detail>

</Details>

</Result>

</Body>

</ eHtalkMessage>

Obrázok 4 - Štruktúra elemetnu Result

#### Príklad eHtalkMessage

V nasledujúcom texte je uvedený príklad štruktúry správy eHtalkMessage. Zelenou farbou sú vyznačené povinné parametre správy. Popis generovania MessageID a CorrelationID je špecifikovaný nižšie. Sivou farbou je vyznačená samotná správa pre koncového prijímateľa. Štruktúra tejto správy závisí od aplikačnej implementácie služby a externej definície štruktúry elementu Body.

<?xml version="1.0" encoding="utf-8"?>

<eHtalkMessage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://eHealth.gov.sk/eHtalkMessage">

<EnvelopeVersion>2.0</EnvelopeVersion>

<Header>

<MessageInfo>

<Class>classtest</Class>

<MessageID>9EE81CF2-7E87-4D85-BA23-16CD7C1A5760</MessageID>

<CorrelationID>9EE81CF2-7E87-4D85-BA23-16CD7C1A5760</CorrelationID>

<BusinessID>Nazov operacie</BusinessID>

<ChannelInfoReply />

</MessageInfo>

<SenderInfo />

</Header>

<Body>

<Data></Data>

<Result>

<Code>0</Code>

</Result>

</Body>

</eHtalkMessage>

### EHRExtract

Táto časť dokumentu popisuje technickú špecifikáciu na vytváranie dátovej časti v súlade so štandardom "CEN/STN EN13606 – Elektronická komunikácia zdravotných záznamov. "

Štandard definuje presnú a stabilnú informačnú architektúru pre výmenu časti, alebo celej dokumentácie elektronických zdravotných záznamov vzťahujúcich sa k jednému pacientovi (EHR). EHR je vnímané ako sada perzistentne uchovávaných záznamov vznikajúcich počas dlhého časového obdobia, potenciálne vznikajúcich u viacerých poskytovateľov alebo vo viacerých krajinách, ktoré sa vzťahujú k jednému prijímateľovi ZS.

CEN EN13606 referenčný model (RM) obsahuje všetky potrebné konštrukcie na podporu zodpovedajúcej komunikácie medzi poskytovateľmi ZS. Kontext informácie by mal vždy sprevádzať zdravotné záznamy.

Kontextové informácie okrem iného zahŕňajú autora zdravotného záznamu, poskytovateľa ZS a tak ďalej.

V súčasnom návrhu archetypov sa počíta s dátovým typom text, t.j. bez formátovania (iba v zmysle nový riadok, medzera).

#### Triedy CEN EN 13606

|  |  |
| --- | --- |
| Trieda CEN EN 13606 | Popis |
| EHR\_EXTRACT | Kontajner najvyššej úrovne časti alebo celého zdravotného záznamu jedného Prijímateľa ZS. |
| FOLDER | Slúži na kategorizovanie a organizáciu zdravotných záznamov. Umožnuje rozdeľovať záznamy do oblastí týkajúcích sa zdrav. starostlivosti. |
| COMPOSITION | Súbor informácií zväzujúcich sa k jednému autorovi(lekárovi), v dôsledku jednej udalosti alebo jednej návštevy. |
| SECTION | Obvykle odráža tok/sadu informácií v priebehu tvorby a zhromažďovania zdravotného záznamu. |
| ENTRY | Údaje vytvorený v dôsledku jednej akcie, jednoho pozorovania, jednej interpretácie alebo zámeru. |
| CLUSTER | Zaznamenáva viaczložkové dátové štruktúry ako sú časové rady, reprezentácia stĺpca tabuľky... atď |
| ELEMENT | Koncový uzol hierarchie zdravotného záznamu, ktorý obsahuje jedinú dátovú hodnotu. |

Tabuľka 11 - Triedy CEN EN 13606

EHR\_EXTRACT obsahuje zdravotné záznamy ako COMPOSITION, voliteľne organizovaných do hierarchie FOLDER.

COMPOSITION obsahuje ENTRY, voliteľne obsiahnutých v rámci hierarchie SECTION.

ENTRY obsahuje ELEMENT, voliteľne obsiahnutý v rámci hierarchie CLUSTER.

#### Povinné atribúty EHR Extractu

Jednotlivé úrovne EHR\_EXTRACT v súlade s normou CEN EN 13606 obsahujú presne definované atribúty, ktorých vyplnenie je povinné.

Na úrovni EHR\_EXTRACT ide o informácie:

* ehr\_system - Identifikácia systému PZS, v ktorom bol EHR Extract vytvorený. Max 50 znakov. Kód tvorcovi IS PZS vydáva prevádzkovateľ riešenia po prijatí žiadosti tvorcu IS PZS na overenie zhody.
* subject\_of\_care - Unikátny identifikátor príjmateľa zdravotnej starostlivosti, z ktorého bol EHR Extract vytvorený a je definovaný systémom PZS.
* time\_created - Dátum a čas, kedy bol vytvorený EHR Extract.
* rm\_id - Položka bude obsahovať reťazec "EN13606". Ide o identifikáciu a verziu referenčného modelu, podľa ktorého je EHR Extract vytvorený.
* EHR\_EXTRACT môže obsahovať buď štruktúry typ FOLDER alebo štruktúry typu COMPOSITION, ktoré sú umiestnené do uzlu all\_compositions.

Úroveň FOLDER  ide o informácie:

* name - Názov Folder sa uvádza ako obyčajný text (je umožnené ho uvádzať aj ako kódovanú hodnotu) a obsahuje názov kategorizácie zdravotných údajov. Preberá sa popis položky z archetypu. Napríklad "Pacientsky sumár " a pod.
* synthesised - V prípade, že sa jedná o syntetizovaný uzol (nie je definované archetypom) hodnota je "TRUE", a v prípade, že je uzol definovaný archetypom, tak hodnota je "FALSE".
* archetype\_id - Jedinečný identifikátor uzla archetypu, ktorému component zodpovedá (Referencia na ADL archetypu). Je uvedený názov archetypu a identifikátor uzla spojený znakom „/“ napr. CEN-EN13606-FOLDER.Pacientsky\_sumar.v1/at0000
* rc\_id - Globálny jednoznačný identifikátor, ktorým je uzol v hierarchii EHR identifikovaný v systéme, ktorý vykonal prvý zápis údajov. Identifikátor musí byť zachovaný a znovupoužitý kedykoľvek bude prvok použitý v inom EHR\_Extract-e (bližšia špecifikácia je uvedená v kapitole „Identifikácia zdravotných záznamov“).
* Folder môže obsahovať ďalšie štruktúry typu FOLDER ak to umožňuje archetyp alebo obsahuje štruktúry typu COMPOSITION, ktoré sú danej kategórie.

Na úrovni COMPOSITION ide o informácie:

* name - Názov Composition sa uvádza ako obyčajný text (je umožnené ho uvádzať aj ako kódovanú hodnotu) a obsahuje klinický koncept, ktorý zodpovedá zaslanému EHR a preberá popis položky concept z archetypu. Napríklad "Medikačný záznam", "Nežiadúca reakcie" a pod.
* rc\_id - Globálny jednoznačný identifikátor, ktorým je uzol v hierarchii EHR identifikovaný v systéme, ktorý vykonal prvý zápis údajov. Identifikátor musí byť zachovaný a znovupoužitý kedykoľvek bude prvok použitý v inom EHR\_Extract-e (bližšia špecifikácia je uvedená v kapitole „Identifikácia zdravotných záznamov“). Netýka sa syntetických elementov (viď Serializácia EHRExtract).
* archetype\_id - Jedinečný identifikátor uzla archetypu, ktorému component zodpovedá (Referencia na ADL archetypu). V prípade, že sa jedná o koreňový uzol, je uvedený názov napríklad „CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1“. V prípade iného uzla je uvedený názov archetypu a identifikátor uzla spojený znakom „/“ napr. CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1/at0029
* synthesised - V prípade, že sa jedná o syntetizovaný uzol (nie je definované archetypom) hodnota je "TRUE", a v prípade, že je uzol definovaný archetypom, tak hodnota je "FALSE".
* sensitivity – Položka udáva citlivosť údajov uvedených v kompozícii. Povolené hodnoty sú od „1“ po „5“, pričom najnižšia citlivosť je „1“ a najvyššia citlivosť je „5“. Údaje uvedené v danej kompozícii pri príslušnej citlivosti môžeme charakterizovať nasledovne:
  + "1" – Záznam je dostupný administratívnemu personálu na riadenie príjemcu zdravotnej starostlivosti pri zabezpečovaní zdravotníckych služieb.
  + "2" - Menej citlivé údaje, ktoré môžu byť dostupné širšiemu okruhu osôb, z ktorých nie všetci môžu poskytovať aktívnu starostlivosť pacientovi (napr. zamestnanci rádiológie).
  + "3" - Štandardná hodnota citlivosti údajov. Ide o údaje, ktoré by mali mať dostupná väčšina zdravotníckych pracovníkov, ktorí sa starajú o pacienta. Väčšina údajov EHR by mala mať takúto úroveň citlivosti.
  + "4" - Citlivé údaje prístupné iba pre dôvernú úzku skupinu zdravotníckych pracovníkov.
  + "5" - Veľmi citlivé údaje, ktoré môžu byť zdieľané jedným alebo dvomi ďalšími zdravotníckymi pracovníkmi alebo sú dostupné príjemcovi zdravotnej starostlivosti. Môže ísť napríklad o údaje o duševnom zdraví pacienta.
* composer.performer - Identifikácia zdravotníckeho pracovníka, ktorý záznam v pôvodnom systéme vytvoril. Položka je povinná ak má COMPOSITION položku synthesised = FALSE.
* composer.healthcare\_facillity – Identifikácia OU PZS kde bol záznam vytvorený. Položka je povinná ak má COMPOSITION položku synthesised = FALSE.
* composer.function – Identifikácia zdravotníckej odbornosti zdravotníckeho pracovníka, ktorý záznam v pôvodnom systéme vytvoril podľa číselníka Zdravotníckych odborností OID 1.3.158.00165387.100.10.34. Položka je povinná ak má COMPOSITION položku synthesised = FALSE.
* committal.ehr\_system - Identifikácia systému PZS, v ktorom bol pôvodný záznam uložený. Položka je povinná ak má COMPOSITION položku synthesised = FALSE.
* committal.time\_committed - Dátum a čas, kedy bol záznam uložený do ehr\_system a stal sa súčasťou EHR. Položka je povinná ak má COMPOSITION položku synthesised = FALSE.
* committal.committer - Používateľ zodpovedný za uloženie záznamu do EHR príjemcu zdravotnej starostlivosti. Je požadované, aby Composer a Committer bol vždy ten istý pracovník. Položka je povinná ak má COMPOSITION položku synthesised = FALSE.
* committal.version\_status – Status verzie danej kompozície z hľadiska medicínsko-právneho. Povolená hodnota je „VER01“, ak nie je v popise služby uvedené inak. Význam hodnoty je, že ide o verziu, ktorá bola odoslaná ako ukončená bez predpokladanej potreby ďalšej revízie. Položka je povinná ak má COMPOSITION položku synthesised = FALSE. V štruktúre [CS] je povinné len vyplnenie položky codeValue.

Na úrovni SECTION ide o informácie:

* name - Názov uvádzať ako obyčajný text (je umožnené ho uvádzať aj ako kódovanú hodnotu) a obsahuje klinický koncept, ktorý zodpovedá zaslanému EHR a preberá hodnotu popisu položky z archetypu. Napríklad "Diagnóza", "Alergén" a pod.
* rc\_id - Globálny jednoznačný identifikátor, ktorým je uzol v hierarchii EHR identifikovaný v systéme, ktorý vykonal prvý zápis údajov. Identifikátor musí byť zachovaný a znovupoužitý kedykoľvek bude prvok použitý v inom EHR\_Extract-e (bližšia špecifikácia je uvedená v kapitole „Identifikácia zdravotných záznamov“). Netýka sa syntetických elementov (viď Serializácia EHRExtract).
* archetype\_id - Jedinečný identifikátor uzla archetypu, ktorému component zodpovedá (Referencia na ADL archetypu). V prípade, že sa jedná o koreňový uzol, je uvedený názov napríklad „CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1“. V prípade iného uzla je uvedený názov archetypu a identifikátor uzla spojený znakom „/“ napr. CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1/at0029
* synthesised - V prípade že sa jedná o syntetizovaný uzol (nie je definované archetypom) hodnota je "TRUE", a v prípade, že je uzol definovaný archetypom, tak hodnota je "FALSE".

Na úrovni ENTRY ide o informácie:

* name - Názov uvádzať ako obyčajný text (je umožnené ho uvádzať aj ako kódovanú hodnotu) a obsahuje klinický koncept, ktorý zodpovedá zaslanému EHR a preberá hodnotu popisu položky z archetypu. Napríklad "Diagnóza", "Alergén" a pod.
* rc\_id - Globálny jednoznačný identifikátor, ktorým je uzol v hierarchii EHR identifikovaný v systéme, ktorý vykonal prvý zápis údajov. Identifikátor musí byť zachovaný a znovupoužitý kedykoľvek bude prvok použitý v inom EHR\_Extract-e (bližšia špecifikácia je uvedená v kapitole „Identifikácia zdravotných záznamov“).
* archetype\_id - Jedinečný identifikátor uzla archetypu, ktorému component zodpovedá (Referencia na ADL archetypu). V prípade, že sa jedná o koreňový uzol je uvedený názov napríklad „CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1“. V prípade iného uzla je uvedený názov archetypu a identifikátor uzla spojený znakom „/“ napr. CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1/at0029
* synthesised - V prípade, že sa jedná o syntetizovaný uzol (nie je definované archetypom) hodnota je "TRUE", a v prípade, že je uzol definovaný archetypom, tak hodnota je "FALSE".
* uncertainty\_expressed - Položka je povinná a definovaná iba pre Type = "ENTRY" a je vždy nastavená na "FALSE". Ak by bola nastavená na "TRUE", špecifikuje, že tento atribút obsahuje údaje, ktoré majú určitý stupeň neistoty a je potrebné byť opatrný pri automatizovanom spracovaní údajov.
* info\_provider.performer - Identifikácia zdravotníckeho pracovníka, ktorý záznam v pôvodnom systéme vytvoril. Položka je povinná ak má nadradená COMPOSITION položku synthesised = TRUE.
* info\_provider.healthcare\_facillity – Identifikácia OU PZS kde bol záznam vytvorený. Položka je povinná ak má nadradená COMPOSITION položku synthesised = TRUE.
* info\_provider.function – Identifikácia zdravotníckej odbornosti zdravotníckeho pracovníka, ktorý záznam v pôvodnom systéme vytvoril podľa číselníka Zdravotníckych odborností OID 1.3.158.00165387.100.10.34. Položka je povinná ak má nadradená COMPOSITION položku synthesised = TRUE.
* feeder\_audit.ehr\_system - Identifikácia systému PZS, v ktorom bol pôvodný záznam uložený. Položka je povinná ak má nadradená COMPOSITION položku synthesised = TRUE.
* feeder\_audit.time\_committed - Dátum a čas, kedy bol záznam uložený do ehr\_system a stal sa súčasťou EHR. Položka je povinná ak má nadradená COMPOSITION položku synthesised = TRUE.
* feeder\_audit.committer - Používateľ zodpovedný za uloženie záznamu do EHR príjemcu zdravotnej starostlivosti. Je požadované, aby Composer a Committer bol vždy ten istý pracovník. Položka je povinná ak má nadradená COMPOSITION položku synthesised = TRUE.
* feeder\_audit.version\_status – Status verzie danej kompozície s hľadiska medicínsko-právneho. Povolená hodnota je „VER01“ “, ak nie je v popise služby uvedené inak. Význam hodnoty je, že ide o verziu, ktorá bola odoslaná ako ukončená bez predpokladanej potreby ďalšej revízie. Položka je povinná ak má nadradená COMPOSITION položku synthesised = TRUE. V štruktúre [CS] je povinné len vyplnenie položky codeValue.

Na úrovni CLUSTER ide o informácie:

* name - Názov uvádzať ako obyčajný text (je umožnené ho uvádzať aj ako kódovanú hodnotu) a obsahuje klinický koncept, ktorý zodpovedá zaslanému EHR a preberá hodnotu popisu položky z archetypu. Napríklad "Diagnóza", "Alergén" a pod.
* archetype\_id - Jedinečný identifikátor uzla archetypu, ktorému component zodpovedá (Referencia na ADL archetypu). V prípade, že sa jedná o koreňový uzol, je uvedený názov napríklad „CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1“. V prípade iného uzla je uvedený názov archetypu a identifikátor uzla spojený znakom „/“ napr. CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1/at0029
* synthesised - V prípade, že sa jedná o syntetizovaný uzol (nie je definované archetypom) hodnota je "TRUE", v prípade, že je uzol definovaný archetypom, tak hodnota je "FALSE".
* structure\_type – Definuje typ štruktúry CLUSTER. Móže ísť buď o zoznam (LIST) alebo tabuľku (TABLE). V štruktúre [CS] je povinné len vyplnenie položky codeValue.

Na úrovni ELEMENT ide o informácie:

* name - Názov uvádzať ako obyčajný text (je umožnené ho uvádzať aj ako kódovanú hodnotu) a obsahuje klinický koncept, ktorý zodpovedá zaslanému EHR a preberá hodnotu popisu položky z archetypu. Napríklad "Diagnóza", "Alergén" a pod.
* archetype\_id - Jedinečný identifikátor uzla archetypu, ktorému component zodpovedá (Referencia na ADL archetypu). V prípade, že sa jedná o koreňový uzol je uvedený názov napríklad „CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1“. V prípade iného uzla je uvedený názov archetypu a identifikátor uzla spojený znakom „/“ napr. CEN-EN13606-ENTRY.Neziaduca\_reakcia.v1/at0029
* synthesised - V prípade, že sa jedná o syntetizovaný uzol (nie je definované archetypom) hodnota je "TRUE", a v prípade, že je uzol definovaný archetypom, tak hodnota je "FALSE".

Atribúty na uvedených úrovniach, ktoré nie sú uvedené v zozname sa na vstupe nesmú vypĺňať. Systém ich nespracováva a teda neposkytuje ani na výstupe služby.

#### Serializácia EHRExctract – prenos dát

Serializovaná správa EHRExtract je implementovaná prostredníctvom technológie XML. XML schéma je vizualizovaná na nasledujúcom obrázku a uvedená v prílohe „*EN13606-extract.xsd*“ a „*EN13606-RM.xsd*“. Schéma bola vytvorená v rámci EN 13606 Association [www.en13606.org](http://www.en13606.org). Serializácia je riadená podľa definície príslušného archetypu (\*.adl). Ak je potrené posielať údaje na nižšej úrovni ako FOLDER alebo COMPOSITION, v takom prípade je potrebné chýbajúce úrove synteticky doplniť.

Pre syntetické úrovne je potrebné nasledovné naplnenie identifikátorov:

*synthesised=”TRUE”*

*rc\_id.OID = “1.3.158.00165387.100.30.30.10”*

*rc\_id.extension = GUID*

Keďže sa syntetické elementy neperzistujú, nie je možné ukladať ich rc\_id pre ďalšie použitie, preto že pri ďalšom volaní (čítaní/zápise) sa vždy použije nové rc\_id. Rc\_id v tomto prípade slúži len na odlíšenie elementov v rámci jedného EHR\_EXTRACT. Mimo daného EHR\_EXTRACT stráca význam.



Obrázok 5 - EHRExtract UML model

### XSD Služby

XML schémou – XSD popisuje údaje, ktoré nie sú klinického charakteru t.j. nejedná sa o zdravotný záznam, ktorý bol vytvorený pre prijímateľa ZS lekárom u niektorého z poskytovateľov ZS.

#### Základné požiadavky

Špecifikácia je v súlade s nasledovným:

* Musí používať UTF-8 kódovanie (v súlade s ISVS-2010 § 13 bod c)
* Musí byť v súlade s definíciou XML Schema 1.0 (http://www.w3.org/TR/xmlschema-0/, http://www.w3.org/TR/xmlschema-1/, a <http://www.w3.org/TR/xmlschema-2/>.)
* Musí byť naformátovaná a validná v súlade s XML 1.0 (<http://www.w3.org/TR/REC-xml>)

Všetky elementy musia byť zaradené do menného priestoru.

### Identifikácia záznamov

V komunikácii medzi externým systémom a NZIS sú pri identifikácii záznamov používané **externé identifikátory (ExtID)** a **identifikátory číselníkových položiek**.

Identifikátor je pridelený pre každý objekt, ktorý vstupuje alebo vystupuje z NZIS.

Identifikátor je vždy definovaný dvomi zložkami:

* Typ záznamu - vyjadrený prideleným OID
* Kód záznamu - je jeden alfanumerický reťazec bez použitia špeciálnych znakov a diakritiky. Dĺžka je limitovaná tým, že musí byť manuálne natypovateľný, nadiktovateľný do telefónu a pod.

#### OID

Pre potreby identifikácie záznamov, ako hlavný rozlišovací mechanizmus je zvolený koncept OID. OID je definované na úrovni typov záznamov a to tak, aby bolo možné každý vstupný a výstupný objekt typovo rozlíšiť.

OID sú definované prostredníctvom statickej mapovacej tabuľky, ktorú publikuje prevádzkovateľ*.*

#### Identifikácia zdravotných záznamov

Na identifikáciu je použitý významový externý identifikátor generovaný na základe metodiky vychádzajúcej z týchto pravidiel. Identifikátor je generovaný u zdroja záznamu (IS PZS...). Na základe tohto identifikátora sa pristupuje a manipuluje so záznamom, t.j. pri aktualizácii, zrušení a podobne, sa využíva práve tento externý identifikátor.

Identifikátor nesmie obsahovať informáciu, z ktorej by sa priamo dal určiť PrZS. Navyše žiadne dva identifikátory nesmú umožňovať odvodenie príslušnosti zdravotných záznamov jednému PrZS, t.j. nesmú obsahovať časti (podreťazce), z ktorých by sa dalo určiť, že ide o toho istého PrZS.

ExtID je na strane NZIS validovaný a kontrolovaný na jednoznačnosť. Konzument služby musí implementovať opravné mechanizmy na základe výsledkov kontrol NZIS a musí byť schopný pregenerovania identifikátora v spornom prípade.

Implementácia externého identifikátora zdravotných záznamov

Externý identifikátor zdravotného záznamu je reprezentovaný tzv. descriptorom. Descriptor je definovaný dátovým typom II podľa CEN/TS 14796 a implementáciou II podľa EN 13606-1.



Obrázok 6 - Dátový typ II

**Externý identifikátor v Archetypoch**

Externý identifikátor záznamu sa priamo v definíciách štruktúr archetypov neuvádza, tento údaj je definovaný v referenčnom modely ako *RECORD\_COMPONENT.rc\_id*.

**Charakteristiky extID v implementácii NZIS**

Externý identifikátor eviduje a poskytuje NZIS ako aj zdroj záznamu. NZIS zodpovedá za to, že záznam s nejednoznačným identifikátorom nie je v NZIS spracovaný a identifikátor zostane po celú dobu platnosti záznamu nemenný t.j. nemá časovú platnosť.

Externý identifikátor je súčasťou záznamu.

Externé identifikátory zdravotných záznamov sú prenášané z externých systémov do NZIS a späť v nešifrovaniej podobe.

Sémantika štruktúry extID pre zdravotné záznamy:

* Typ záznamu - je vyjadrený prideleným OID ku každému typu zdravotného záznamu. Je definovaný na rozhraní príslušnej služby.
* Kód záznamu (extension) - je jeden numerický reťazec bez použitia špeciálnych znakov a diakritiky. Dĺžka je limitovaná. Skladá sa z nasledovných častí:
  + Identifikátor zdroja záznamu - je jednoznačný a musí byť pridelený
  + Identifikácia – je jednoznačný v rámci typu záznamu a identifikátora zdroja záznamu, generuje ho zdrojový systém

Spôsob vytvárania identifikátora zdravotného záznamu

Princípy vytvárania identifikátora záznamu sa aplikujú pre kód záznamu, t.j. extension.

**Formát kódu záznamu**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Z | Z | Z | Z | Z | V | V | V | V | V | R | R | N | N | N | N | N | N | N | N | C |

|  |  |  |  |
| --- | --- | --- | --- |
| identifikátor zdroja záznamu | Identifikátor inštalácie | Rok + identifikácia | Kontrolné číslo |
|  |  |  |  |

Kde Z, V, R, N a C sú číslice.

Kód záznamu je reprezentovaný na rozhraniach ako textový reťazec. V tom prípade je zľava rozšírený číslicami „0“ na požadovanú dĺžku osobitne pre časť Z, časť V a časť N.

Kontrolná číslica C:

* Je vypočítaná pomocou funkcie[[2]](#footnote-3) modulo 10 (je to teda samodetekujúci kód).
* Postup výpočtu:
  + Sčítajú sa čísla R a N na nepárnych pozíciách (pozície sú číslované zľava)
  + Čísla na párnych pozíciách sa vynásobia dvoma.V prípade ak je výsledkom násobenia číslo > 9 nahradí sa číslom, ktoré vznikne ako súčet jeho cifier. Tieto výsledné hodnoty sa pripočítajú ku výsledku z predchádzajúceho bodu.
  + Výsledok sa vynásobí 9 a zoberie sa posledná číslica čím získame kontrolnú číslicu. (alternatívne je možné zobrať z výsledku poslednú číslicu a odčítať od 10).

Príklad:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **R** | **R** | **N** | **N** | **N** | **N** | **N** | **N** | **N** | **N** | x |
| **Pozícia** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | x |
| **Identifikácia(N)** | 1 | 5 | 9 | 2 | 7 | 3 | 9 | 8 | 7 | 1 | x |
| **Párne pozície** |  | 10 |  | 4 |  | 6 |  | 16 |  | 2 | x |
| **Nepárne pozície** | **1** |  | **9** |  | **7** |  | **9** |  | **7** |  |  |
| **Súčty číslic** | 1 | 1 | 9 | 4 | 7 | 6 | 9 | 7 | 7 | 2 | =5**3** |

Výsledkm je súčet 53, a kontrolná číslica sa získa ako 53 \* 9 modulo 10:

53\*9 = 477

477 mod 10 = **7**

Tj. Celý identifikátor je 1592739871**7**

**Generovanie kódu záznamu**

Kód záznamu generuje výlučne IS PZS.

Identifikátor zdroja záznamu „Z“ je pridelený v rámci procesu overovania zhody IS s NZIS.

Číslo inštalácie „V“ prideľuje tvorca informačného systému

Identifikácia „N“, môže byť implementovaná napríklad ako sekvenčné číslo pričom je vytváraná pre každý typ záznamu samostatnou DB sekvenciou. Prvé dva znaky „RR“ reprezentujú posledné dve cifry z aktuálneho roku.

Príklad:

Identifikátor zdroja = 22

Inštalácia číslo 57

Rok 2015

Záznam číslo 92739871

Výsledný kód záznamu: 00022000571592739871**7**

Hlavným rozlišovacím prvkom je typ záznamu (OID).

#### Identifikácia subjektov

Subjekty používané resp. odkazované v NZIS sú evidované v JRÚZ. Jedná sa o nasledovné entity :

* Prijímateľ ZS
* Poskytovateľ ZS
* Odborný útvar poskytovateľa ZS
* Zdravotnícky pracovník
* Zdravotná poisťovňa

Na referencovanie subjektov sa používa externý identifikátor.

Implementácia externého identifikátora subjektov

Externý identifikátor subjektu (JRUZ identifikátor) je reprezentovaný tzv. descriptorom. Descriptor je definovaný dátovým typom II podľa CEN/TS 14796 a implementáciou II podľa EN 13606-1.



Obrázok 7 - Dátový typ II

**Charakteristiky v implementácii NZIS:**

Externý identifikátor subjektu eviduje a poskytuje JRÚZ. JRÚZ zodpovedá za to, že identifikátor je jednoznačný a zostane po celú dobu platnosti subjektu nemenný t.j. nemá časovú platnosť.

Externý identifikátor je použitý pri komunikácii s externým systémom PZS, laboratória či ZP.

Pre jeden subjekt je v rámci NZIS používaný pri komunikácii s externým systémom vždy len externý identifikátor (Napríklad pre prijímateľa ZS nebude pre zápis zdravotného záznamu používaný pre jednu službu rodné číslo a pre inú službu bezvýznamový identifikátor vygenerovaný v rámci systému...a pod.).

Externý identifikátor je súčasťou záznamu o entite v JRÚZ.

Externý identifikátor subjektu je prideľovaný jednoznačné minimálne na úrovni slovenského eHealth.

**Externý (JRUZ) identifikátor prijímateľa ZS sa pri komunikácii s NZIS používa výlučne v zašifrovanej forme** (viď kap 6.1.1.7 **Metóda „CreateEsid" a** 6.1.1.8 **Metóda „ExtractPatientIdentifierFromEsid**“)

Identifikátor subjektu je prideľovaný na celý život entity, t.j. pre danú entitu budú identifikátory platné po celú dobu existencie entity a identifikátor nemôže byť v budúcnosti použitý pre inú entitu.

Sémantika štruktúry extID pre subjekty:

* Typ subjektu - je vyjadrený prideleným OID.
* Kód záznamu (extension) - je jeden alfanumerický reťazec bez použitia špeciálnych znakov a diakritiky. Dĺžka je limitovaná. Zdrojom je JRUZ.

*Poznámka:* JRUZ poskytuje služby, ktoré na základe iných identifikátorov poskytnú externý identifikátor subjektu (napríklad: na základe RČ,…).

#### Identifikácia číselníkových položiek

V rámci komunikácie s NZIS WS službami sa využívajú číselníkové položky, ktoré sú centrálne evidované v JRUZ. JRUZ udržuje a poskytuje samotné údaje a históriu číselníkov, tiež poskytuje k nim synchronizačné služby.

Identifikátor pridelený položke v číselníku je nemenný a jednoznačný. Nemá časovú platnosť, aj keď položka v číselníku môže byť časovo ohraničená.

Jednému identifikátoru zodpovedá vždy práve jedna položka v číselníku.

Pri referencovaní sa na číselníkovú položku je nevyhnutné používať identifikátor položky v JRUZ a nie kód položky s časovou platnosťou. Položka číselníka totiž môže mať ukončenú platnosť a následne byť znovu založená s tým istým kódom.

Identifikátor položky v číselníku sa nepoužíva priamo v používateľskom rozhraní – nie je viditeľný.

Číselníkové položky sa využívajú ako v Archetypoch tak natívnych XSD štruktúrach. Navrhované použitie je principiálne rovnaké naprieč rôznym typom údajov.

Pre použitie v rámci riadiacich informácii komunikácie je použitá iná štruktúra, prispôsobená potrebnému použitiu. Napríklad: v rámci Security tokenu, Identity používateľa...atď. [Pre definíciu pozri eHtalk Message príslušnú kapitolu.](#_eHtalkMessage)

#### Implementácia identifikátora číselníkovej položky

Číselníkovú položku popisuje XSD element dátového typu „CV“ (ďalej len CV) v súlade s normou dátových typov CEN/TS 14796, ktorú využíva norma EN 13606.

Pravidlá využívania CV nie sú normou pevne dané, norma stanovuje len rámec. Z tohto dôvodu špecifikácia rozhraní definuje implementáciu CV podľa pravidiel popísaných nižšie.

Dátový typ CV je popísaný štruktúrou uvedenou na nasledujúcom obrázku.



Obrázok 8 - Dátový typ CV

Všeobecný význam jednotlivých atribútov:

* *nullFlavor* – vypĺňa sa vtedy, ak položka Číselníka nie je vyplnená. Tu sa definuje dôvod prečo nie je vyplnená
* *codingScheme* – respektíve jeho Object ID určuje o aký číselník sa jedná
  + ilustračný príklad[[3]](#footnote-4): pre Číselník Štát hodnota OID = „1.3.158.00165387.100.10.51“
* *codingSchemeName* – názov číselníka
  + ilustračný príklad: pre Číselník Štát hodnota = „Štát“
* *codingSchemeVersion* – verzia číselníka
  + ilustračný príklad: pre prvú verziu hodnota = „1“
* *codeValue* – identifikátor položky číselníka
  + ilustračný príklad: pre položku Slovenská republika hodnota = „703“
* *displayName* – názov položky číselníka
  + ilustračný príklad: pre položku Slovenská republika hodnota = „Slovenská republika“

**Charakteristiky CV v implementácii NZIS:**

V rozhraní sa nepoužíva atribút „*nullFlavor*“ – Inak povedané, ak je položka číselníka povinná, tak musí byť vyplnená hodnotou. Ak je nepovinná, tak sa vôbec neuvádza. NZIS nullFlavor neimplementuje, v prípade použitia správa neprejde validáciou.

V rozhraní sa nepoužíva atribút „*codingSchemeName*“

Ako povinné atribúty sú:

* „*codingScheme.oid*“
* „*codingSchemeVersion*“
* „*codeValue*“
* *„displayName“*

*Kontroly:*

*NZIS na vstupe vždy kontroluje štvoricu „codingScheme.oid“, „codingSchemeVersion“, „codeValue“, „displayName“ a neumožní spracovanie správy ak nie je použitá položka číselníka, ktorá sa v uvedenej verzii nachádza, alebo nie je dodržaný názov položky. V prechodnom období je možné použiť pre zápis záznamov aj verzie, ktoré už nie sú posledné platné.*

### Spoločné dátové typy

V rozhraní sa nepoužíva atribút „*nullFlavor*“, ktorý obsahujú všetky dátové typy – Inak povedané, ak je položka povinná, tak musí byť vyplnená. Ak je nepovinná, tak sa vôbec neuvádza. NZIS nullFlavor neimplementuje, v prípade použitia správa neprejde validáciou.



Obrázok 9 - Atribút nullFlavor v dátovom type

#### Dátový typ II a CV

Sú popísané v rámci kapitoly [5.4.4. Identifikácia záznamov](#_Identifikácia_záznamov).

V rozhraní sa nepoužívajú nasledovné atribúty:

* *nullFlavor*
* assigningAuthorityName
* validTime – nepoužíva sa

#### Dátový typ INT

V rozhraní sa používa ako celé číslo. Je vyplnený atribút „*value*“.



Obrázok 10 - Dátový typ INT

#### Dátový typ REAL

V rozhraní sa používa ako reálne číslo. Je vyplnený atribút „*value*“. Vo volaniach nebude akceptovaná hodnota reálneho čísla zapísaná v exponenciálnej forme.



Obrázok 11 - Dátový typ REAL

#### Dátový typ BL

V rozhraní sa používa ako boolean. Je vyplnený atribút „*value*“.



Obrázok 12 - Dátový typ BL

#### Dátový typ SIMPLE\_TEXT

V rozhraní sa používa ako text. Je vyplnený atribút „*originalText*“.

Atribúty „*language*“ a „*charset*“ sa nepoužívajú. Maximálny rozsah položiek typu SIMPLE\_TEXT je špecifikovaný v definícii archetypu. Pokiaľ veľkosť položky nie je v ADL šepcifikovaná, platí pravidlo, že maximálna akceptovateľná dĺžka textu je 4000 znakov.



Obrázok 13 - Dátový typ SIMPLE TEXT

#### Dátový typ TS

V rozhraní sa používa ako dátum a čas. Je vyplnený atribút „*time*“. Ak nie je uvedené časové pásmo, predpokladá sa použitie lokálneho časového pásma prevádzkovateľa systému.



Obrázok 14 - Dátový typ TS

#### Dátový typ DATE

V rozhraní sa používa ako dátum. Je vyplnený atribút „*date*“.



Obrázok 15 - Dátový typ DATE

#### Dátový typ PQ

V rozhraní sa používa ako fyzikálna veličina - nameraná hodnota s mernou jednotkou. Je vyplnený atribút „*value*“ a „units“. Hodnoty atribútu „*units*“ sú vypĺňané z číselníka merných jednotiek. Atribúty „*property*“ a „*precision*“ sa nepoužívajú. Vo volaniach nebude akceptovaná hodnota reálneho čísla zapísaná v exponenciálnej forme.



Obrázok 16 - Dátový typ PQ

### HTML text

V rámci EHRExtractu v atribútoch typu ELEMENT, je možné použiť formátovanie textu prostredníctvom HTML. Takéto formátovanie je povolené ak ELEMENT má v deklarácii constraintu uvedené "HTML". Takýto atribút má obmedzenie v počtoch znakov, prostredníctvom regulárneho výrazu, napríklad /[\s\S]{0,1000000}/, konkrétne hodnoty sú uvedené v príslušných ADL. Toto obmedzenie môže byť hlbšie olimitované systémovým parametrom pre veľkosť HTML.

HTML formát je v súlade so špecifikáciou

1. Spĺňa štandard XHTML vo verzii 1.0 - <http://www.w3.org/TR/xhtml1>
2. Neobsahuje nepovolené tagy, resp. obsahuje len povolené HTML tagy a atribúty uvedené v prílohe „x070E\_Html\_format.xlsx”

Príklad ADL deklarácie povolenia HTML:

ELEMENT[at0002] occurrences matches {0..1} matches {  -- Text HTML

  value existence matches {0..1} matches {

    SIMPLE\_TEXT[at0003] occurrences matches {1..1} matches {  -- SIMPLE\_TEXT

      originalText existence matches {0..1} matches {/[\s\S]{0,1000000}/; **"HTML"**}

    }

  }

}

#### Serializácia

HTML text musí byť do XML elementu originalText vložený ako reťazec znakov, nie je povolené požitie CDATA.

Príklad:

<originalText>&lt;!doctype html&gt;&lt;HTML&gt;

             &lt;HEAD&gt;

             &lt;/HEAD&gt;

             &lt;BODY&gt;

                    &lt;table&gt;

                            &lt;tr&gt;

                            &lt;td&gt;&lt;em&gt;Parameter&lt;/em&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;em&gt;Nameran&amp;aacute; hodotna&lt;/em&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;em&gt;Jednotka&lt;/em&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;em&gt;Ref. hodnoty&lt;/em&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;em&gt;Jednotka&lt;/em&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;em&gt;Pr&amp;iacute;pad&lt;/em&gt;&lt;/td&gt;

                            &lt;/tr&gt;

                            &lt;tr&gt;

                            &lt;td&gt;&lt;strong&gt;S Transfer&amp;iacute;n&lt;/strong&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;strong&gt;1&lt;/strong&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;strong&gt;g/l&lt;/strong&gt;&lt;/td&gt;

                            &lt;td&gt;2,0 - 3,6&lt;/td&gt;

                            &lt;td&gt;g/l&lt;/td&gt;

                            &lt;td&gt;&amp;nbsp;&lt;/td&gt;

                            &lt;/tr&gt;

                            &lt;tr&gt;

                            &lt;td&gt;&lt;strong&gt;S Ferit&amp;iacute;n&lt;/strong&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;strong&gt;128&lt;/strong&gt;&lt;/td&gt;

                            &lt;td&gt;&lt;strong&gt;&amp;micro;g/l&lt;/strong&gt;&lt;/td&gt;

                            &lt;td&gt;21,8 - 247,7&lt;/td&gt;

                            &lt;td&gt;&amp;micro;g/l&lt;/td&gt;

                            &lt;td&gt;muži&lt;/td&gt;

                            &lt;/tr&gt;

                            &lt;tr&gt;

                            &lt;td&gt;&amp;nbsp;&lt;/td&gt;

                            &lt;td&gt;&amp;nbsp;&lt;/td&gt;

                            &lt;td&gt;&amp;nbsp;&lt;/td&gt;

                            &lt;td&gt;4,6 - 204,0&lt;/td&gt;

                            &lt;td&gt;&amp;micro;g/l&lt;/td&gt;

                            &lt;td&gt;ženy&lt;/td&gt;

                            &lt;/tr&gt;

                     &lt;/table&gt;

             &lt;/BODY&gt;

&lt;/HTML&gt; </originalText>

### Používanie nepovinných atribútov

Kvôli minimalizácii dátových tokov je požadované aby nepovinné XML elementy neboli na vstupe správ uvádzané a to ani vo forme prázdnych elementov.

### Synchronizácia času

Čas NZIS je synchronizovaný celosvetovým družicovým systémom GPS.

## Reakcia na chyby v komunikácii

V prípade ak po volaní služby systém nedostane odpoveď, alebo obdrží chybovú správu, je potrebné podľa typu chyby ( kap. [5.3.5 - Chybové stavy](#_Chybové_stavy)) volanie opakovať. V prípade opakovanej chyby (3 opakovania) musí informačný systém prejsť do off-line režimu, kde sa dáta neodosielajú do NZIS.

Za chybu sa v tomto kontexte považuje jedno z nasledovného:

* Chyby HTTP protokolu (4xx a 5xx)
* Žiadna odpoveď v rámci http protokolu po dobu viac ako 60 sekúnd (timeout)
* Chyba v rámci odpovede s kódom E999989,E999993,E999995,E999999
* Chýbajúca odpoveď (E999990) v rámci response manager v prípade asynchrónneho volania po dobu 5 minút po odoslaní požiadavky
* SOAP Fault chybová sprava (viď kap. 5.5.1)

O opakované odoslanie sa systém môže pokúsiť najskôr o 5 minút. V prípade neúspechu volania je potrebné dobu čakania zvýšiť zakaždým na dvojnásobok, najviac však na 2 hodiny.

### Zachytenie SOAP Fault výnimky

Príklad SOAP Fault správy, ktorú vracia bezpečnostný komponent XG45:

<soapenv:Detail>

<XMLGWFault xmlns="">

<faultcode>**E999999**</faultcode>

<faultstring>**Integrity of message was broken.**</faultstring>

<faultactor>**XML gateway**</faultactor>

</XMLGWFault>

</soapenv:Detail>

</soapenv:Fault>

Príklad spôsobu programového zachytenia typovej SOAP Fault výnimky:

try

{

result = ((eHealthService.IeHealthSyncService)channel).GetData(message);

}

catch (FaultException<eHealthService.XMLGWFault> xmlGwFault)

{

Assert.Fail(xmlGwFault.Detail.faultstring);

}

catch (FaultException otherFault)

{

Assert.Fail(otherFault.Message);

}

Pričom platí, že contract služby obsahuje deklaráciu fault contract-u:

[System.CodeDom.Compiler.GeneratedCodeAttribute("System.ServiceModel", "4.0.0.0")]

[System.ServiceModel.ServiceContractAttribute(Namespace="[http://eHealth.gov.sk/eHtalk](http://ehealth.gov.sk/eHtalk)", ConfigurationName="IeHealthSyncService")]

public interface IeHealthSyncService

{

// CODEGEN: Generating message contract since element name value from namespace [http://eHealth.gov.sk/eHtalk](http://ehealth.gov.sk/eHtalk) is not marked nillable

[System.ServiceModel.OperationContractAttribute(ProtectionLevel=System.Net.Security.ProtectionLevel.Sign, Action="[http://eHealth.gov.sk/eHtalk/IeHealthSyncService/Receive](http://ehealth.gov.sk/eHtalk/IeHealthSyncService/Receive)", ReplyAction="[http://eHealth.gov.sk/eHtalk/IeHealthSyncService/ReceiveResponse](http://ehealth.gov.sk/eHtalk/IeHealthSyncService/ReceiveResponse)")]

[System.ServiceModel.FaultContractAttribute(typeof(XMLGWFault), Action="[http://eHealth.gov.sk/eHtalk/IeHealthSyncService/GetDataXMLGWFaultFault](http://ehealth.gov.sk/eHtalk/IeHealthSyncService/GetDataXMLGWFaultFault)", ProtectionLevel=System.Net.Security.ProtectionLevel.None, Name="XMLGWFault", Namespace="")]

GetDataResponse GetData(GetDataRequest request);

}

Ak sa vygeneruje z WSDL:

*svcutil \*.wsdl \*.xsd /language:C#*

## Tlač EZKO

NZIS poskytuje službu VyhladajZaznamyEZKTlac, ktorá umožňuje IS PZS získať údaje EZK pacienta pre účel ich tlače. Na výstupe služby sú XML údaje EZK, šifrované osobné údaje občana, metadáta pre dešifrovanie osobných údajov a XSLT dokument pre transformáciu údajov do HTML.

Celý proces tlače EZKO vyzerá nasledovne:

* Systém IS PZS vytvorí požiadavku VyhladajZaznamyEzkTlacRequest podľa špecifikácie.
* Systém IS PZS odošle požiadavku do ESB cez asynchrónne rozhranie IeHealthService.
* NZIS získa údaje PrZS pre požadované stránky EZK a získa osobné údaje PrZS šifrované na verejný kľúč používateľa IS PZS.
* Systém IS PZS získa asynchrónne odpoveď VyhladajZaznamyEzkTlacResponse, z ktorej extrahuje XML dokument umiestnený v elemente http://eHealth.gov.sk/EzkHealthRecordManagement/v1: VyhladajZaznamyEzkTlacResponse/DataPreTlac a uloží ho do premennej v pamäti.
* Pre každý nájdený XML element http://eHealth.gov.sk/EzkHealthRecordManagement/v1: VyhladajZaznamyEzkTlacResponse/MapaPreDesifrovanie/MapaPreDesifrovanie vykoná:
  + Vyhľadá v extrahovanom XML element pomocou Xpath výrazu špecifikovanom v elemente XpathZasifrovaneData (napr. „/DocumentData/PatientBasicInfoEncrypted“).
  + Dešifruje obsah tohto elementu volaním metódy **Metóda „DecryptDataForHealthProfessional”** (viď podkapitola 6.1.1.11) komponentu EhealthCryptoController.
  + Vyhľadá v extrahovanom XML element pomocou Xpath výrazu špecifikovanom v elemente XPathPreDesifrovaneData (napr. „/DocumentData/PatientBasicInfo“) a vloží údaje dešifrované v predchádzajúcom kroku do vnútra tohto elementu.
* Z odpovede VyhladajZaznamyEzkTlacResponse ďalej extrahuje XSLT dokument umiestnený v elemente http://eHealth.gov.sk/EzkHealthRecordManagement/v1: VyhladajZaznamyEzkTlacResponse/XmlTransformacia a uloží ho do premennej v pamäti.
* Systém IS PZS využije načítaný XSLT dokument pre transformáciu extrahovaného XML do HTML, ktoré je naformátované pre tlač a uloží tento HTML dokument na disk používateľa IS PZS.
* Používateľ IS PZS otvorí vygenerovaný HTML dokument v internetovom prehliadači a zvolí tlač dokumentu (Alternatívne môže HTML dokument v internetovom prehliadači otvoriť priamo systém IS PZS).
* V závislosti od konkrétneho internetového prehliadača je možné tlač prispôsobiť (upraviť hlavičku a pätu tlačených stránok)

### Ukážka spracovania odpovede zo služby VyhladajZaznamyEZKTlac

Odpoveď zo služby VyhladajZaznamyEZKTlac je typu VyhladajZaznamyEzkTlacResponse z XML menného priestoru <http://eHealth.gov.sk/EzkHealthRecordManagement/v1>.

Typ VyhladajZaznamyEzkTlacResponse obsahuje vlastnosti:

* DataPreTlac
  + Obsahuje XML dokument s vyžiadanými údajmi o EZKO občana zaserializovaný formou textového reťazca
* XmlTransformacia
  + Obsahuje XSLT dokument určený pre transformáciu XML dokumentu „DataPreTlac“ do HTML dokumentu vhodného pre tlač cez internetový prehliadač.
* MapaPreDesifrovanie
  + XML dokument obsahuje šifrované (osobné) údaje občana a v prípade ich vyžiadania aj kontaktné a ICE údaje. Pre ich dešifrovanie pomocou CryptoController komponentu je potrebné ich po kusoch (XML fragmentoch) vystrihnúť, dešifrovať a vložiť dešifrovaný XML fragment naspäť.
  + Vlastnosť obsahuje zoznam metadát určených pre vykonanie vyššie popísanej akcie dešifrovania.

Ukážka extrahovania XML dokumentu z odpovede služby VyhladajZaznamyEZKTlac a dešifrovania údajov podľa metadát v elemente MapaPreDesifrovanie v jazyku C#.

*Príklad predpokladá, že v premennej response, typu VyhladajZaznamyEzkTlacResponse (vygenerovaného z XSD) je uložená deserializovaná odpoveď z asynchrónnej služby VyhladajZaznamyEZKTlac.*

// Parse XML data string to System.Xml.Ling.XDocument type

XDocument doc = XDocument.Parse(**response.DataPreTlac**);

// Create instance of eHealth Cryptography Controller

IEhealthCryptoController ecc = new EhealthCryptoController();

// Decrypt each XML fragment specified in decryption map from response

foreach (var **mapa** in **response.MapaPreDesifrovanie**)

{

// Extract encrypted personal data

XElement encData = doc.XPathSelectElement(**mapa.XPathZasifrovaneData**);

// Decrypt personal data

if (encData != null)

{

// Decrypt data using instance of eHealth Cryptography Controller

byte[] encDataBytes = Convert.FromBase64String(encData.Value);

byte[] decDataBytes = ecc.DecryptDataForHealthProfessional(encDataBytes);

string decDataString = Encoding.UTF8.GetString(decDataBytes);

// Parse XML data string to System.Xml.Ling.XElement type

XElement decryptedXml = XElement.Parse(decDataString);

// Insert decrypted data XML fragment back into XML document

XElement decDataTarget = doc.XPathSelectElement(**mapa.XPathPreDesifrovaneData**);

decDataTarget.Add(decryptedXml);

}

}

Viac o dešifrovaní údajov z NZIS pomocou Crypto Controllera je popísané v kapitole 6.1.1.9 **Metóda „DecryptDataForHealthProfessional”**.

Po dešifrovaní údajov je možné výsledný XML dokument (uložený v príklade v premennej doc) pretransformovať na HTML dokument, ktorý je možné uložiť na pevný disk a následne otvoriť v internetovom prehliadači.

Ukážka následnej transformácie XML do HTML pomocou XSLT dokumentu v jazyku C#.using (XmlReader stylesheet = XmlReader.Create((TextReader)new StringReader(**response.XmlTransformacia**)))

{

XslCompiledTransform compiledTransform = new XslCompiledTransform();

compiledTransform.Load(stylesheet);

using (StringWriter stringWriter = new StringWriter())

{

using (XmlReader input = **doc**.CreateReader())

{

compiledTransform.Transform(input, (XsltArgumentList)null, (TextWriter)stringWriter);

string **html** = stringWriter.ToString();

}

}

}

## Dešifrovanie PDF dokumentu digitálneho covid certifikátu

NZIS poskytuje službu DajCovidCertifikat, ktorá vo svojom výstupe vráti PDF dokument požadovaného digitálneho covid certifikátu šifrovaného na certifikát zdravotníckeho pracovníka.

Pred použitím PDF dokumentu je potrebné ho dešifrovať pomocou ePZP karty s využitím CryptoControllera.

Viac o dešifrovaní údajov z NZIS pomocou Crypto Controllera je popísané v kapitole 6.1.1.9 **Metóda „DecryptDataForHealthProfessional”**.

Po dešifrovaní je možné binárne dáta, ktoré predstavujú obsah PDF dokumentu uložiť na pevný disk a následne otvoriť v prehliadači PDF dokumentov.

Ukážka dešifrovania PDF dokumentu a jeho uloženia na pevný disk (ilustračne je názov súboru odvodený na základe identifikátora certifikátu):

// Zavolaj sluzbu DajCovidCertifikat

DajCovidCertifikatResponse response = DajCovidCertifikat(request);

string base64EncryptedPdf = response.**Dokument**;

// Dekoduj binarne data a desifruj ich cez CryptoController

IEhealthCryptoController ecc = new EhealthCryptoController();

byte[] decryptedData = ecc.DecryptDataForHealthProfessional(Convert.FromBase64String(base64EncryptedPdf));

string pdfFileName = request.**IdCertifikatu**.Replace(":", "\_") + ".pdf";

// Uloz desifrovane binarne data PDF na disk

File.WriteAllBytes(Path.Combine(BaseDirectory, pdfFileName), decryptedData);

# Bezpečnostné mechanizmy

Pre zabezpečenie prístupu ku službám iba autentizovaným používateľom a zabezpečenie nepopierateľnosti autorstva sú do volania služieb implementované bezpečnostné mechanizmy. Bezpečnostné autentizačné mechanizmy využívajú pre svoju činnosť identifikačné predmety (čipové karty) zdravotníckeho pracovníka, ako aj prijímateľa zdravotnej starostlivosti, tieto karty sú vyžadované autentizačnými službami NZIS. Pre komunikáciu s identifikačnými predmetmi ako aj autentizačnými službami poskytne NZIS knižnicu - EhealthCryptoController, ktorý musí informačný systém poskytovateľa integrovať so svojimi klientskymi komponentami na jednotlivých pracovných staniciach.

### Verejné rozhranie komponentu IEhealthCryptoController

Verejné rozhranie „IEhealthCryptoController“ (pozri **Obrázok 17**), je definíciou množiny metód, ktoré klientsky komponent verejne poskytuje IS PZS za účelom realizácie požadovanej funkcionality bezpečnostných mechanizmov NZIS na strane informačných systémov IS PZS.

Toto rozhranie zároveň tvorí styčný bod medzi klientskym komponentom a IS PZS.



Obrázok 17 - Rozhranie IEhealthCryptoController (verzia 5.0 a vyššie).

V nasledujúcich podkapitolách sa nachádza detailnejší popis jednotlivých metód komponentu *EhealthCryptoController* ako aj príklad ich volania. Všetky príklady sú písané v programovacom jazyku C# pre Microsoft .NET Framework 4.0. Uvedené príklady nie sú plne kompilovateľnými časťami zdrojového kódu. Ide o fragmenty kódu z ktorých je vidieť praktický spôsob volania jednotlivých metód verejného rozhrania klientskeho komponentu *EhealthCryptoController*.

#### Metóda „GetHealthProfessionalIdentifier”

Vráti identifikátor zdravotníckeho pracovníka ktorého čipová karta je v čítacom zariadení na základe informácii obsiahnutých v autentifikačnom certifikáte daného zdravotníckeho pracovníka..

*Vstupné parametre*

Metóda nemá žiadne vstupné parametre.

*Návratová hodnota*

Výsledkom volania metódy *GetHealthProfessionalIdentifier* je reťazec znakov reprezentujúci JRÚZ identifikátor zdravotníckeho pracovníka, ktorého čipová karta je v čítacom zariadení.

*Výnimky*

|  |  |
| --- | --- |
| Výnimka | Popis |
| SmartCardNotPresentExecption | Nastane ak počas volania tejto metódy nie je v systéme prítomná žiadna čipová karta zdravotníckeho pracovníka s platným autentifikačným certifikátom. |
| CertificateNotFoundException | Výnimka nastane v prípade, keď sa namiesto čipových kariet použije ako úložisko certifikátov systémové úložisko pre aktuálne prihláseného používateľa (Microsoft, 2011), pričom sa v tomto úložisku nenachádza žiaden platný autentifikačný certifikát zdravotníckeho pracovníka. |
| TooManyCertificatesFoundException | Nastane v prípade, ak v nastavenom úložisku – čipovej karte alebo systémovom úložisku - zodpovedá požadovaným kritériám viac ako jeden certifikát.  Ak sa používajú čipové karty, táto výnimka znamená, že sú prítomné karty viacerých zdravotníckych pracovníkov súčasne. |

*Príklad*

....

using Iam.Client;

....

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

string hProIdentifier = null;

hProIdentifier = ecc.GetHealthProfessionalIdentifier();

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

Console.WriteLine("HealthProfessional identifier is: {0}", hProIdentifier);

#### Metóda „GetPatientData”

Metóda umožňuje získanie údajov (rodné číslo / JRÚZ identifikátor / Potvrdenie Prítomnosti) o prijímateľovi zdravotnej starostlivosti v jednom volaní. Výsledkom volania služby je XML dokument obsahujúci požadovanú množinu údajov (viac viď XSD schéma „*PatientDataResponse.xsd*“). Získanie potvrdenia prítomnosti prijímateľa (PPC) zdravotnej starostlivosti je cez centrálnou službou IAM a to v úrovni vyžadovat/nevyžadovať od prijímateľa zdravotnej starostlivosti zadanie BOK k jeho eID karte, a/alebo v úrovni Výzva/Potvrdenie.

**UPOZORNENIE**: Použtie tejto metódy si vyžaduje mať k systému pripojenú čítačku čipových kariet s podporou príslušnej funkcionality, pričom eID karta prijímateľa zdravotnej starostlivosti musí byť vložená v tejto čítačke (viac viď kap. 6.1.4.3 ).

*Vstupné parametre*

| Parameter | Dátový typ | Popis |
| --- | --- | --- |
| personData | Enum | Výber požadovných údajov pacienta:   * + BirthNumber – rodné číslo   + JruzIdentifier – JRUZ identifikátor pacienta   + PersonComputerNumber – PČO   + PersonMeaninglessIdentifier – bezvýznamový identifikátor |
| confirmationLevel | Enum | Určuje vybranú úroveň požadovaného PPC:   * + None – nepožaduje sa PPC   + PinRequired – požaduje sa PPC so zadaním PIN kódu k pacientovému elektronickému identifikačnému predmetu (eID)   + NoPinRequired – požaduje sa PPC bez zadania PIN kódu k eID   + ChallengeResponse – požaduje sa PPC s textovou výzvou |
| challenge | String | Obsahuje UTF-8 kódovaný text výzvy “Suhlasim s \npristupom k EZK” (bez diakritiky - presný text výzvy je overovaný na centrálnom serveri, ak nie je dodržaný je vrátené chybové hlásenie). Táto výzva je následne zobrazená na zobrazovacom zariadení čítačky čipových kariet v ktorej je vložená eID karta.  Ak má výzva byť zobrazená vo viacerých riadkoch, jednotlivé riadky v rámci výzvy sú oddelené znakom „\n“. |
|  |  |  |

*Návratová hodnota*

Výsledkom volania metódy *GetPatientData* je požadovaná množina údajov podľa schémy *PatientDataResponse.xsd* s obsahom reťazcov znakov reprezentujúcich JRÚZ identifikátor prijímateľa zdravotnej starostlivosti, RČ, PČO, bezvýznamový identifikátor resp. podpísaný XML dokument PPC prijímateľa zdrav. starostlivosti, ktorého elektronický identifikačný predmet je v čítacom zariadení.

*Výnimky*

|  |  |  |
| --- | --- | --- |
| Výnimka | Popis | |
| RequestedOperationFailedException | Nastane v prípade neočakávanej chyby operačného systému alebo sieťovej komunikácie.  Chyba, ktorá túto výnimku spôsobila je zahrnutá ako vlastnosť *InnerException* tejto výnimky. | |
| IncorectNumberOfEIDCardsDetected | | Nastane ak bolo odetegovaných prítomných viac ako jedna eID karta. |
| eIDCardNotFoundException | | Nenašla sa žiadna eID karta. |
| FeaturesNotSuportedByReaderException | | V systéme bola detegovaná eID karta ale čítačka, v ktorej je karta vložená nepodporuje funkcionalitu potrebnú na vykonanie potvrdenia výzvy (napr. čítačka nemá display a PinPad alebo nepodporuje príslušné inštrukcie podľa PC/SC špecifikácie). |
| IncorrectChallengeFormatException | | Nastane v prípade, ak text výzvy nezpodpovedá disponibilite danej čítačky (napr. obsahuje znakov na riadok ako je čítačka schopná zobraziť alebo obsahuje viac riadkov textu ako má čítačka).  Konkrétny detail je uvedený vo vlastnosti „**FormatErrorCode**“ danej výnimky, pričom hodnoty sú:   * **TooManyCharactersPerLine** * **TooManyLines**   Výnimka tiež obsahuje vlastnosti „**MaxNumberOfCharactersPerRow**“ a „**MaxNumberOfRows**“ ktoré obsahujú informáciu o maximálnom počte znakov na riadok pre danú čítačku ako aj o maximálnom počte riadkov. |
| TimeoutException | | Prijímateľ zdravotnej starostlivosti neodpovedal na príslušnú výzvu v stanovenom časovom limite.  Predvolená hodnota tohto časového limitu je 60 sekúnd. |

Okrem uvedených výnimiek, môžu pri volaní metódy *GetPatientData* nastať výnimky súvisiace s identifikáciou prítomného zdravotníckeho pracovníka. Tieto výnimky sú opísané v kapitole 6.1.1.5.

*Príklad*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

using Iam.Client;

...

string patientData = ecc.GetPatientData(

(Iam.Client.PatientData.BirthNumber

| Iam.Client.PatientData.JruzIdentifier

| Iam.Client.PatientData.PersonComputerNumber

| Iam.Client.PatientData.PersonMeaninglessIdentifier

) // Údaje o používateľovi

, Iam.Client.PresenceConfirmationLevel.NoPinRequired // Vrátane PPP bez PINu

, null // Text výzvy pre PPP je v tomto prípade NULL.

);

// Vypísanie získaného potvrdenia na obzaovku.

Console.WriteLine(patientData);

// Z výsledku extrahujeme napr. JRUZ identifikátor PrZS.

string przsId = Helper.ExtractFromXml(patientData, "/pd:patientDataResponse/pd:jruzIdentifier");

// A vypíšme identifikátor na obzaovku.

Console.WriteLine(przsId);

Extrahovanie PPT z odpovede GetPatientData

*Zdrojový kód predpokladá existenciu vygenerovanej triedy patientDataResponse zo schémy patientDataResponse.xsd pre účel deserializácie XML odpovede z GetPatientData metódy.*

var serializer = new XmlSerializer(typeof(patientDataResponse));

var reader = new StringReader(patientDataXmlText);

var pdResponse = (patientDataResponse)serializer.Deserialize(reader);

string ppTokenString = pdResponse.presenceConfirmation.Any[0].OuterXml;

// Konverzia PPC tokenu z textového reťazca do samostatneho XML dokumentu

XmlDocument pptTokenDoc = new XmlDocument();

pptTokenDoc.LoadXml(ppTokenString);

XmlElement **ppcTokenXml** = pptTokenDoc.DocumentElement;

Upozornenie:

Predpokladom pre úspešné overenie PPC tokenu v NZIS službách je kódovanie znaku nového riadku v texte výzvy v súlade s XML štandardom a teda \n (binárne 0a). V prípade použitia znaku \r\n (binárne 0d0a) skončí validácia podpisu PPC tokenu neúspešne. Uvedený príklad využíva XmlSerializer, ktorý vykoná transformáciu znaku nového riadku na \n automaticky.

Metóda Helper.ExtractFromXml() je nasledovná:

public static string ExtractFromXml(string xmlData, string xpath)

{

XmlDocument xmlDoc = new XmlDocument();

xmlDoc.LoadXml(xmlData);

var xmlnsman = new XmlNamespaceManager(xmlDoc.NameTable);

xmlnsman.AddNamespace("pd", "http://iam.ehealth.sk/2018-08/PatientDataResponse.xsd");

xmlnsman.AddNamespace("ppc", "http://iam.ehealth.sk/2017-03/PatientPresenceConfirmationResponse.xsd");

var node = xmlDoc.SelectSingleNode(xpath, xmlnsman);

if(node == null)

throw new Exception($"XPath \"{xpath}\" not found.");

string result = null;

result = node.InnerXml;

return result;

}

V premennej patientData bude po úspešnom volaní XML štruktúra obsahujúca požadované údaje.

*Príklad volania s potvrdením prítomnosti pacienta s BOK (PIN)*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

using Iam.Client;

...

string patientData = ecc.GetPatientData(

(PatientData.BirthNumber // RČ

| PatientData.JruzIdentifier // JRUZ ID

| PatientData.PersonComputerNumber // PČO

| PatientData.PersonMeaninglessIdentifier // BIFO

) // Údaje o používateľovi

, Iam.Client.PresenceConfirmationLevel.PinRequired // Vrátane PPP s PINu

, null // Text výzvy pre PPP je v tomto prípade NULL.

);

Zvyšná časť príkladu je zhodná s prvým príkladom volania GetPatientData.

*Príklad volania s potvrdením prítomnosti bez zadania BOK (PIN)*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

using Iam.Client;

...

string patientData = ecc.GetPatientData(

(PatientData.BirthNumber // RČ

| PatientData.JruzIdentifier // JRUZ ID

| PatientData.PersonComputerNumber // PČO

| PatientData.PersonMeaninglessIdentifier // BIFO

) // Údaje o používateľovi

, Iam.Client.PresenceConfirmationLevel.NoPinRequired // Vrátane PPP bez PINu

, null // Text výzvy pre PPP je v tomto prípade NULL.

);

Zvyšná časť príkladu je zhodná s prvým príkladom volania GetPatientData.

*Príklad volania s potvrdením prítomnosti pacienta výzvou*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

using Iam.Client;

...

string patientData = ecc.GetPatientData(

(PatientData.BirthNumber // RČ

| PatientData.JruzIdentifier // JRUZ ID

| PatientData.PersonComputerNumber // PČO

| PatientData.PersonMeaninglessIdentifier // BIFO

) // Údaje o používateľovi

, Iam.Client.PresenceConfirmationLevel.ChallengeResponse // Vrátane PPP s výzvou

, "Suhlasim s \npristupom k EZK" // Text výzvy pre volanie služby ZapisSuhlasOsobyPrePZS.

);

Zvyšná časť príkladu je zhodná s prvým príkladom volania GetPatientData.

*Príklad volania metódy GetPatientData bez potvrdzovania prítomnosti pacienta*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

using Iam.Client;

...

string patientData = ecc.GetPatientData(

(PatientData.BirthNumber // RČ

| PatientData.JruzIdentifier // JRUZ ID

| PatientData.PersonComputerNumber // PČO

| PatientData.PersonMeaninglessIdentifier // BIFO

) // Údaje o používateľovi

, Iam.Client.PresenceConfirmationLevel.None // Nepotrebujeme PPP, len údaje o pacientovi na základe eID

, null // Text výzvy pre PPP je v tomto prípade NULL.

);

Zvyšná časť príkladu je zhodná s prvým príkladom volania GetPatientData s výnimkou extrakcie PPC tokenu, keďže ten v tomto prípade nebude v patientData prítomný.

#### Metóda „SignByHealthProfessional”

Zabezpečuje realizáciu digitálneho podpisu zdravotníckeho pracovníka nad bližšie nešpecifikovanou štruktúrou dát.

*Vstupné parametre*

|  |  |  |
| --- | --- | --- |
| Parameter | Dátový typ | Popis |
| data | Byte[] | Dáta určené na digitálne podpísanie vo forme pola bajtov. |

*Návratová hodnota*

Výstupom volania metódy *SignByHealthProfessional* je pole bajtov, reprezentujúce detašovaný digitálny podpis dát v štruktúre PKCS #7.

*Výnimky*

|  |  |
| --- | --- |
| Výnimka | Popis |
| SmartCardNotPresentException | Nastane ak počas volania tejto metódy nie je v systéme prítomná žiadna čipová karta zdravotníckeho pracovníka s platným podpisovým certifikátom. |
| CertificateNotFoundException | Výnimka nastane v prípade, keď sa namiesto čipových kariet použije ako úložisko certifikátov systémové úložisko pre aktuálne prihláseného používateľa (Microsoft, 2011), pričom sa v tomto úložisku nenachádza žiaden platný podpisový certifikát zdravotníckeho pracovníka. |
| TooManyCertificatesFoundException | Nastane v prípade, ak v nastavenom úložisku – čipovej karte alebo systémovom úložisku - zodpovedá požadovaným kritériám viac ako jeden certifikát.  Ak sa používajú čipové karty, táto výnimka znamená, že sú prítomné karty viacerých zdravotníckych pracovníkov súčasne |

*Príklad*

Nasledovný príklad demonštruje digitálne podpísanie textu „Hello world“ zdravotníckym pracovníkom s využitím metódy *SignByHealthProfessional*.

....

using Iam.Client;

....

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

byte[] dataNaPodpis = Encoding.Default.GetBytes("Hello world.");

byte[] podpis = ecc.**SignByHealthProfessional**(dataNaPodpis);

string enkodovanyPodpisAkoRetazec = Convert.ToBase64String(podpis);

#### Metóda „SignXmlDocumentByHealthProfessional”

Metóda zabezpečuje realizáciu digitálneho podpisu XML dokumentov v zmysle štandardu XML-DSIG zdravotníckym pracovníkom.

*Vstupné parametre*

|  |  |  |
| --- | --- | --- |
| Parameter | Dátový typ | Popis |
| xml | string | Alfanumerický reťazec reprezentujúci XML dokument, ktorý sa má podpísať zdravotníckym pracovníkom. |

*Návratová hodnota*

Výstupom volania metódy *SignXmlDocumentByHealthProfessional* je alfanumerický reťazec znakov (*string*), reprezentujúci elektronicky podpísaný XML dokument v zmysle štandardu XML-DSIG..

*Výnimky*

|  |  |
| --- | --- |
| Výnimka | Popis |
| XmlException | Výnimka nastane v prípade syntaktickej chyby v XML dátach špecifikovaných v parametri *xml*. |
| SmartCardNotPresentException | Nastane ak počas volania tejto metódy nie je v systéme prítomná žiadna čipová karta zdravotníckeho pracovníka s platným podpisovým certifikátom. |
| CertificateNotFoundException | Výnimka nastane v prípade, keď sa namiesto čipových kariet použije ako úložisko certifikátov systémové úložisko pre aktuálne prihláseného používateľa (Microsoft, 2011), pričom sa v tomto úložisku nenachádza žiaden platný podpisový certifikát zdravotníckeho pracovníka. |
| TooManyCertificatesFoundException | Nastane v prípade, ak v nastavenom úložisku – čipovej karte alebo systémovom úložisku - zodpovedá požadovaným kritériám viac ako jeden certifikát.  Ak sa používajú čipové karty, táto výnimka znamená, že sú prítomné karty viacerých zdravotníckych pracovníkov súčasne |

*Príklad*

Nasledujúci zdrojový kód demonštruje využitie metódy „*SignXmlDocumentByHealthProfessional*“ na digitálne podpísanie vzorového XML dokumentu zdravotníckym pracovníkom.

....

using Iam.Client;

....

string xmlData = @"<?xml version=""1.0"" encoding=""ISO-8859-1""?>

<CATALOG>

<PLANT>

<COMMON>Bloodroot</COMMON>

<BOTANICAL>Sanguinaria canadensis</BOTANICAL>

<ZONE>4</ZONE>

<LIGHT>Mostly Shady</LIGHT>

<PRICE>$2.44</PRICE>

<AVAILABILITY>031599</AVAILABILITY>

</PLANT>

<PLANT>

<COMMON>Columbine</COMMON>

<BOTANICAL>Aquilegia canadensis</BOTANICAL>

<ZONE>3</ZONE>

<LIGHT>Mostly Shady</LIGHT>

<PRICE>$9.37</PRICE>

<AVAILABILITY>030699</AVAILABILITY>

</PLANT>

<PLANT>

<COMMON>Marsh Marigold</COMMON>

<BOTANICAL>Caltha palustris</BOTANICAL>

<ZONE>4</ZONE>

<LIGHT>Mostly Sunny</LIGHT>

<PRICE>$6.81</PRICE>

<AVAILABILITY>051799</AVAILABILITY>

</PLANT>

</CATALOG>

";

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

string podpisaneXml = ecc.**SignXmlDocumentByHealthProfessional**(xmlData);

Obsahom premennej *podpisaneXml* bude:

<?xml version="1.0" encoding="utf-8"?>

<CATALOG>

  <PLANT>

    <COMMON>Bloodroot</COMMON>

    <BOTANICAL>Sanguinaria canadensis</BOTANICAL>

    <ZONE>4</ZONE>

    <LIGHT>Mostly Shady</LIGHT>

    <PRICE>$2.44</PRICE>

    <AVAILABILITY>031599</AVAILABILITY>

  </PLANT>

  <PLANT>

    <COMMON>Columbine</COMMON>

    <BOTANICAL>Aquilegia canadensis</BOTANICAL>

    <ZONE>3</ZONE>

    <LIGHT>Mostly Shady</LIGHT>

    <PRICE>$9.37</PRICE>

    <AVAILABILITY>030699</AVAILABILITY>

  </PLANT>

  <PLANT>

    <COMMON>Marsh Marigold</COMMON>

    <BOTANICAL>Caltha palustris</BOTANICAL>

    <ZONE>4</ZONE>

    <LIGHT>Mostly Sunny</LIGHT>

    <PRICE>$6.81</PRICE>

    <AVAILABILITY>051799</AVAILABILITY>

  </PLANT>

  <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">

    <SignedInfo>

      <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />

      <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />

      <Reference URI="">

        <Transforms>

          <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />

        </Transforms>

        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />

        <DigestValue>kScO3W1gkm3YMDS8ABa++GXyjSI=</DigestValue>

      </Reference>

    </SignedInfo>

    <SignatureValue>EtyoSAzJsKKIDTsJDQQ5MZTiSKYgtFxefKYfr+rPoUY0aZhMPpulxkMUL0KgzPoEDCSWm06Ivs8o89aIMUtseG6Gin7vFvgdIbKXWYFn2Ap/CEKvMMklnCfZkuLKv6n+ys4K2vsj1Yd5TDkbNrK/A5E/67k+4sWn4PTDwE1c5HhghMC9YHa0JP5WoSPOodKtvXXJamHQ5Zplj0Z7rSOu8Y9jvu1uwtew/udc9c318EZk1dis2FkeM4F0CP3wPHESTmAyy9eppRxxEe0tmueCtPyaKhc2Ydg0JBJG02MHnui8SKEpKEDyF1lBEPwSLz3FjdXwYWfsKl8URASddvuxkQ==</SignatureValue>

    <KeyInfo>

      <X509Data>

        <X509Certificate></X509Certificate>

      </X509Data>

    </KeyInfo>

  </Signature>

</CATALOG>

#### Metóda „GetSamlTokenForHealthProfessional”

Metóda získa autentizačný SAML token potrebný pre autentizáciu zdravotníckeho pracovníka pri volaní jednotlivých vnútorných služieb NZIS.

*Vstupné parametre*

|  |  |  |
| --- | --- | --- |
| Parameter | Dátový typ | Popis |
| targetServiceIdentifier | string | Alfanumerický reťazec identifikujúci cieľovú službu NZIS, pre ktorú sa žiada vydanie autentizačného tokenu. |

*Návratová hodnota*

Výstupom volania metódy *GetSamlTokenForHealthProfessional* je Microsoft .NET Framework objekt typu *System.IdentityModel.Tokens.SecurityToken* reprezentujúci autentizačný SAML token zdravotníckeho pracovníka. Tento objekt sa použije pri volaní požadovanej služby NZIS. SAML token má platnosť 4hod.

*Výnimky*

|  |  |
| --- | --- |
| Výnimka | Popis |
| SmartCardNotPresentException | Nastane ak počas volania tejto metódy nie je v systéme prítomná žiadna čipová karta zdravotníckeho pracovníka s platným autentifikačným certifikátom. |
| CertificateNotFoundException | Výnimka nastane v prípade, keď sa namiesto čipových kariet použije ako úložisko certifikátov systémové úložisko pre aktuálne prihláseného používateľa (Microsoft, 2011), pričom sa v tomto úložisku nenachádza žiaden platný autentifikačný certifikát zdravotníckeho pracovníka. |
| TooManyCertificatesFoundException | Nastane v prípade, ak v nastavenom úložisku – čipovej karte alebo systémovom úložisku - zodpovedá požadovaným kritériám viac ako jeden certifikát.  Ak sa používajú čipové karty, táto výnimka znamená, že sú prítomné karty viacerých zdravotníckych pracovníkov súčasne. |
| UnknownServiceIdentifierException | Parameter *targetServiceIdentifier* obsahuje neznámy identifikátor a autentifikačný server nevie identifikovať cieľovú službu pre ktorú je potrebné vystaviť autentifikačný token. |
| UnableContactIdentityProviderException | Nie je možné kontaktovať autentifikačný server z dôvodu uplynutia časového limitu alebo z dôvodu nedostupnosti autentifikačnej služby. |
| TrustFailureException | Nie je možné vytvoriť zabezpečený komunikačný kanál s autentifikačnou službou, pretože klient nedôveruje príslušným certifikačným autoritám IAM. |
| SecureChannelFailureException | Výnimka nastane v prípade, ak počas procesu autentifikácie zlyhá zabezpečenie komunikácie. Toto môže nastať ak bola predčasne vytiahnutá čipová karta daného zdravotníckeho pracovníka z čítacieho zariadenia. |
| RequestedOperationFailedException | Nastane v prípade neočakávanej chyby operačného systému alebo sieťovej komunikácie.  Chyba, ktorá túto výnimku spôsobila je zahrnutá ako vlastnosť *InnerException* tejto výnimky. |

*Príklad*

Nesledujúci príklad demonštruje získanie autentizačného tokenu zdravotníckeho pracovníka pomocou metódy „GetSamlTokenForHealthProfessional” pre volanie testovacej služby s URI [*https://adfs2.ehealth-iam.local/WSTest/service.svc*](https://adfs2.ehealth-iam.local/WSTest/service.svc).

....

using Iam.Client;

using Iam.IdentityModel;

....

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

SecurityToken token = ecc.**GetSamlTokenForHealthProfessional**("https://adfs2.ehealth-iam.local/WSTest/service.svc");

CallService(tok);

....

Metóda CallService bude implementovať napr.:

var binding = new WS2007FederationHttpBinding(WSFederationHttpSecurityMode.Message);

binding.Security.Message.EstablishSecurityContext = false;

var factory = new ChannelFactory<eHealthService.IeHealthSyncService>(binding, new EndpointAddress(new Uri(ConfigurationManager.AppSettings["SyncExt"])));

factory.Credentials.SupportInteractive = false;

var channel = factory.CreateChannelWithIssuedToken<eHealthService.IeHealthSyncService>(token);

var result = channel.GetData(message);

//urob niečo s výsledkom volania;

Pričom je definované nasledovné

namespace eHealthService

{

    [ServiceContract(Namespace = "<http://eHealth.gov.sk/eHtalk>", ConfigurationName = "IeHealthSyncService", ProtectionLevel = ProtectionLevel.Sign)]

    public interface IeHealthSyncService

    {

        [XmlSerializerFormat]

        [TransactionFlow(TransactionFlowOption.Allowed)]

        [OperationContract(Action = "<http://eHealth.gov.sk/sync/eHtalk/IeHealthService/Receive>", ReplyAction = "<http://eHealth.gov.sk/sync/eHtalk/IeHealthService/ReceiveResponse>")]

        eHtalkMessage GetData(eHtalkMessage value);

    }

}

A v konfigurácii je mimo iné uvedené:

  <appSettings>

    <add key="SyncExt" value="<http://eh-bts01-tst/ESB.SyncServiceExt/SyncService.svc>" />

  </appSettings>

#### Metóda „EncryptForNzis”

Zašifruje bližšie nešpecifikovanú štruktúru dát pre definovanú množinu vnútorných služieb NZIS.

Metóda používa šifrovací algoritmus AES s dĺžkou kľúča 256 bitov v móde CBC (AlgorithmObjectID je „2.16.840.1.101.3.4.1.42“). Výstupné dáta sú následne zabalené do PKCS#7 enveloped message a vrátené ako pole bajtov.

*Vstupné parametre*

| Parameter | Dátový typ | Popis |
| --- | --- | --- |
| data | Byte[] | Dáta určené na zašifrovanie vo forme pola bajtov. |

*Návratová hodnota*

Výstupom volania metódy *EncryptForNzis* je pole bajtov, reprezentujúce zašifrované dáta v štruktúre PKCS #7.

*Výnimky*

|  |  |
| --- | --- |
| Výnimka | Popis |
| FileNotFoundException | Nie je možné nájsť lokálny súbor s certifikátmi služieb NZIS. |
| CertificateNotFoundException | Lokálny súbor s certifikátmi služieb NZIS neobsahuje žiadne platné šifrovacie certifikáty služieb NZIS. |
| RequestedOperationFailedException | Nastane v prípade neočakávanej chyby operačného systému.  Chyba, ktorá túto výnimku spôsobila je zahrnutá ako vlastnosť *InnerException* tejto výnimky. |

*Príklad*

Nesledujúci príklad demonštruje zašifrovanie textu „Hello world!” pre služby NZIS volaním metódy *EncryptForNzis*.

....

using Iam.Client;

....

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

string dataNaZasifrovanie = "Hello world!";

byte[] poleBajtovNaZasifrovanie = Encoding.UTF8.GetBytes(dataNaZasifrovanie);

EhealthCryptoController ecc = new EhealthCryptoController();

byte[] zasifrovanePoleBajtov = ecc.EncryptForNzis(poleBajtovNaZasifrovanie);

string zasifrovaneDataBase64 = Convert.ToBase64String(zasifrovanePoleBajtov);

#### Metóda „CreateEsid”

Metóda vytvorí ASN.1 štruktúru z nešifrovaného identifikátor prijímateľa zdravotnej starostlivosti (IDPrZS) a jednoznačnými identifikátormi „all\_compositions“ elementov vstupnej eHTalk správy a identifikátorom autentizačného tokenu. Následne túto štruktúru spolu s ďalšími metadátami digitálne podpíše s použitím privátneho kľúča podpisového certifikátu prítomného zdravotníckeho pracovníka do formy PKCS#7 štruktúry, ktorú následne celú zašifruje pre cieľové služby NZIS.

Po reautentifikácii a teda získaní nového autentizačného tokenu (SAML tokenu) je potrebné znovu vytvoriť ESID, nakoľko NZIS overuje väzbu ESIDu na príslušný SAML token. Z autentizačného tokenu zadaného ako vstupný parameter „Token“ metódy „CreateEsid“ sa prečíta atribút AssertionID a tento sa priloží s JRUZ identifikátorom PrZS do štruktúry, ktorá je následne spolu s Hash odtlačkom vypočítaným z eHtalk správy podpísaná a zašifrovaná vo forme ESID.

V prípade, že sa pri následnom volaní služby NZIS použije ESID, ktorý je previazaný s iným (napr. starším) autentizačným tokenom než s tokenom, ktorý sa využije pre vytvorenie autentifikovaného spojenia s ESB, dôjde na strane NZIS počas archivácie správy k zlyhaniu overenia väzby ESID na autentizačný token a NZIS vráti chybu E999987: Error in archivation.

Presný popis ASN.1 štruktúry ESID ako aj postup jeho vytvárania je definovaný v technickej špecifikácii ESID, ktorá je samostatným dokumentom.

*Vstupné parametre*

| Parameter | Dátový typ | Popis |
| --- | --- | --- |
| identifier | string | Reťazec znakov predstavujúci identifikátor prijímateľa zdravotnej starostlivosti. |
| Token | object | Autentizačný token zdravotníckeho pracovníka vrátený metódou „GetSamlTokenForHealthProfessional”. |
| ehtalkMessage | string | Serialozavaná eHTalk správa s ktorou má byť vytvorený ESID prepojený. |

*Návratová hodnota*

Výstupom volania metódy *CreateEsid* je reťazec znakov (string), reprezentujúci digitálne podpísaný a následne zašifrovaný identifikátor prijímateľa zdravotnej starostlivosti (ESID) transformovaný do Base64 formy.

*Výnimky*

| Výnimka | Popis |
| --- | --- |
| SmartCardNotPresentException | Nastane ak počas volania tejto metódy nie je v systéme prítomná žiadna čipová karta zdravotníckeho pracovníka s platným podpisovým certifikátom. |
| CertificateNotFoundException | Výnimka nastane v prípade, keď sa namiesto čipových kariet použije ako úložisko certifikátov systémové úložisko pre aktuálne prihláseného používateľa (Microsoft, 2011), pričom sa v tomto úložisku nenachádza žiaden platný podpisový certifikát zdravotníckeho pracovníka. |
| TooManyCertificatesFoundException | Nastane v prípade, ak v nastavenom úložisku – čipovej karte alebo systémovom úložisku - zodpovedá požadovaným kritériám viac ako jeden certifikát.  Ak sa používajú čipové karty, táto výnimka znamená, že môžu byť prítomné karty viacerých zdravotníckych pracovníkov súčasne |
| UnableToLoadEhtalkMessageXmlDataException | Nastane v prípade, ak ako vstupný parameter eHTalkMessage nie je platný XML dokument. |
| RcidNotFoundException | Výnimka nastane, ak sa pri výpočte kontrolného súčtu kompozícii vstupnej eHTalkMessage nepodarí identifikovať/nájsť „rc\_id“ danej kompozície.  Ak nastane táto výnimka, je potrebné skontrolovať, či eHTalkMessage má správnu štrukúru. |
| TooManyRcidException | Výnimka nastane, ak sa pri výpočte kontrolného súčtu kompozícii vstupnej eHTalkMessage nájde viacero „rc\_id“ pre danú kompozíciu a teda nie je možné jednoznačne určiť „rc\_id“ danej kompozície.  Ak nastane táto výnimka, je potrebné skontrolovať, či eHTalkMessage má správnu štrukúru. |
| RequestedOperationFailedException | Nastane v prípade ak zlyhá samotné vytváranie ESID. Detaily chyby, ktorá spôsobila túto výnimku sú obsiahnuté vo vlastnosti *InnerException*. |

*Príklad*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

// Získanie autentizačného tokenu ZPr.

SecurityToken zprAuthToken = ecc.GetSamlTokenForHealthProfessional("https://adfs2.ehealth-iam.local/WSTest/service.svc");

// Vytvorenie príslušnej eHTalk správy a jej serializovanie do premennej ehtalkMsg

string ehtalkMsg = "REPLACE\_THIS\_WITH\_SERUALIZED\_EHTALK\_MESSAGE";

string esid = ecc.**CreateEsid**("00000000018", zprAuthToken, ehtalkMsg);

#### Metóda „ExtractPatientIdentifierFromEsid”

Metóda dešifruje identifikátor prijímateľa zdravotnej starostlivosti z ESID vráteného v odpovedi na volanie NZIS služieb a vráti informáciu o overení digitálneho podpisu na získanom ESID, zoznam kontrolných súčtov elementov „all\_compositions“ pre ktoré bol daný ESID vytvorený a samotný dešifrovaný identifikátor prijímateľa zdravotnej starostlivosti.

*Vstupné parametre*

| Parameter | Dátový typ | Popis |
| --- | --- | --- |
| base64EncodedSsid | string | Reťazec znakov predstavujúci Base64 transformovaný ESID získaný v odpovedi na volanie NZIS služby. |

*Návratová hodnota*

Metóda má nasledovné výstupné parametre:

| Parameter | Dátový typ | Popis |
| --- | --- | --- |
| patientIdentifier | string | Dešifrovaný identifikátor prijímateľa zdravotnej starostlivosti. |
| compositionsHashList | CompositionHashCollection | Zoznam kontrolných súčtov elementov „all\_compositions“, pre ktoré bol daný ESID vytvorený. |
| signatureStatus | EsidSignatureStatus | Enumerácia hodnôt o výsledku overenia digitálneho podpisu na danom ESID. Možné hodnoty:   * „NotVerified“ – podpis nebol overovaný. * „OK“ – všetko je v poriadku (podpis aj podpisovateľ). * „UnableToVerifySigner“ – digitálny podpis je v poriadku ale X.509 certifikát podpisovateľa bol medzičasom exspirovaný, zrušený alebo ho nebolo možné overiť. * „InvalidSignature“ – digitálny podpis nie je platný. |

Nasledujúci obrázok detailnejšie zobrazuje štruktúru „CompositionHashCollection“.



Obr. 1 Štruktúra "CompositionHashCollection"

*Výnimky*

|  |  |
| --- | --- |
| Výnimka | Popis |
| SmartCardNotPresentException | Nastane ak počas volania tejto metódy nie je v systéme prítomná žiadna čipová karta zdravotníckeho pracovníka s platným šifrovacím certifikátom. |
| CertificateNotFoundException | Výnimka nastane v prípade, keď sa namiesto čipových kariet použije ako úložisko certifikátov systémové úložisko pre aktuálne prihláseného používateľa (Microsoft, 2011), pričom sa v tomto úložisku nenachádza žiaden platný šifrovací certifikát zdravotníckeho pracovníka. |
| TooManyCertificatesFoundException | Nastane v prípade, ak v nastavenom úložisku – čipovej karte alebo systémovom úložisku - zodpovedá požadovaným kritériám viac ako jeden certifikát.  Ak sa používajú čipové karty, táto výnimka znamená, že môžu byť prítomné karty viacerých zdravotníckych pracovníkov súčasne |
| RequestedOperationFailedException | Nastane v prípade ak zlyhá samotné overovanie a dešifrovanie ESID.  Detaily chyby, ktorá spôsobila túto výnimku sú obsiahnuté vo vlastnosti *InnerException*. |

*Príklad*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

using Iam.Client;

using eHealth.Security.Esid;

// ....

string esidFromNzisReponse;

// 1. Realizácia volania služby NZIS

// 2. Extrahovanie ESIDu z odpovede služby do premennej "esidFromNzisReponse"

// 3. Deklarovanie premenných pre návratové hodnoty metódy "ExtractPatientIdentifierFromEsid".

string przsID;

CompositionHashCollection compositionsHashList;

EsidSignatureStatus signatureStatus;

// Dešifrovanie ESID pomocou " ExtractPatientIdentifierFromEsid".

ecc.**ExtractPatientIdentifierFromEsid**(esidFromNzisReponse

, out przsID

, out compositionsHashList

, out signatureStatus

);

// Vypísanie dešifrovaného identifikátora na obrazovku

Console.WriteLine("Identifikátor prijímateľa zdravotnej starostlivosti je: {0}", przsID);

#### Metóda „DecryptDataForHealthProfessional”

Dešifruje údaje, zašifrované službami NZIS pre daného zdravotníckeho pracovníka.

*Vstupné parametre*

| Parameter | Dátový typ | Popis |
| --- | --- | --- |
| encryptedData | Byte[] | Údaje, vo forme PKCS#7 správy, určené na rozšifrovanie. |

*Návratová hodnota*

Výsledkom volania metódy *DecryptDataForHealthProfessional* sú rozšifrované údaje vo forme pola bajtov.

*Výnimky*

|  |  |
| --- | --- |
| Výnimka | Popis |
| SmartCardNotPresentException | Nastane ak počas volania tejto metódy nie je v systéme prítomná žiadna čipová karta zdravotníckeho pracovníka s platným šifrovacím certifikátom. |
| CertificateNotFoundException | Výnimka nastane v prípade, keď sa namiesto čipových kariet použije ako úložisko certifikátov systémové úložisko pre aktuálne prihláseného používateľa (Microsoft, 2011), pričom sa v tomto úložisku nenachádza žiaden platný šifrovací certifikát zdravotníckeho pracovníka. |
| TooManyCertificatesFoundException | Nastane v prípade, ak v nastavenom úložisku – čipovej karte alebo systémovom úložisku - zodpovedá požadovaným kritériám viac ako jeden certifikát.  Ak sa používajú čipové karty, táto výnimka znamená, že môžu byť prítomné karty viacerých zdravotníckych pracovníkov súčasne |
| RequestedOperationFailedException | Nastane v prípade ak zlyhá samotné dešifrovanie správy.  Detaily chyby, ktorá spôsobila túto výnimku sú obsiahnuté vo vlastnosti *InnerException*. |

*Príklad*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

string encDataString = String.Empty;

// 1. Do premennej encDataString načítajte Base64 kódovanú časť zašifrovaných dát z eHTalkMessage

byte[] encData = Convert.FromBase64String(encDataString);

byte[] decdata = ecc.DecryptDataForHealthProfessional(encData);

string strdata = UnicodeEncoding.Default.GetString(decdata);

Console.WriteLine("Rozšifrované údaje z NZIS:");

Console.WriteLine(strdata);

#### Metóda „IsTokenRelationshipToSmartCardValid”

Overí prítomnosť čipovej karty ZPr, ktorá obsahuje X.509 autentizačný certifikát na základe ktorého bol zdravotníckemu pracovníkovi vydaný daný autentizačný token.

*Vstupné parametre*

| Parameter | Dátový typ | Popis |
| --- | --- | --- |
| token | object | Autentizačný token zdravotníckeho pracovníka vrátený metódou „GetSamlTokenForHealthProfessional”. |

*Návratová hodnota*

Výsledkom volania metódy *IsTokenRelationshipToSmartCardValid* je pravdivostná hodnota (Boolean), ktorý má hodnotu „True“ ak je karta stále v čítacom zariadení a nebola medzičasom vytiahnutá, prípadne ak bol token vydaný na základe softvérového certifikátu. V iných prípadoch je vrátená hodnota „False“.

*Výnimky*

|  |  |
| --- | --- |
| Výnimka | Popis |
| ArgumentNullException | Nastane v prípade, ak hodnota vstupného parametra *token* je NULL hodnota. |
| ArgumentException | Nastane v prípade, ak bol ako hodnota parametru *token* zadaný autentizačný token, ktorý nebol získaný s použitím *GetSamlTokenForHealthProfessional*. |

*Príklad*

// Vytvorenie inštancie klientskeho komponentu IAM - EhealthCryptoController.

EhealthCryptoController ecc = new EhealthCryptoController();

// Definovanie premennej pre autentizačný token ZPr pre volanie testovacej služby.

SecurityToken zprAuthToken = null;

// Definovanie lokálneho počítadla neúspešných pokusov o volanie služby.

int attempts = 1;

// Definovanie návestia

GetTokenAndCallService:

// Získanie autentizačného tokenu ZPr pre testovaciu službu.

zprAuthToken = ecc.**GetSamlTokenForHealthProfessional**("https://adfs2.ehealth-iam.local/WSTest/service.svc");

// Realizácia volania NZIS služby:

// 1. Overenie prítomnosti karty

bool cardStillInReader = false;

cardStillInReader = ecc.**IsTokenRelationshipToSmartCardValid**(zprAuthToken);

if (cardStillInReader == true)

{

// Ak je karta stále v čítacom zariadení, pokračujem vo volaní služby

CallService(zprAuthToken);

}

else if (cardStillInReader == false && attempts < 2)

{

// Ak bola karta medzičasom vytiahnutá a počet neúspešných pokusov je menší ako dva,

// navýš počítadlo a pokús sa znova vyžiadať token a zavolať slubu.

attempts++;

goto GetTokenAndRetry;

}

else

{

// Inak vráť chybu.

throw new Exception("Smart card not present.");

}

### Špecifikácia technologickej platformy klientskeho komponentu EhealthCryptoController.

|  |  |
| --- | --- |
| Účel | Sprostredkovanie vybraných služieb NZIS pre informačné systémy poskytovateľov zdravotnej starostlivosti formou .Net knižnice pre desktopové aplikácie IS PZS. |
| Platforma komponentu | Microsoft .NET Framework 4.0 |
| Platforma operačného systému | Presný zoznam je umiestnený v dokumente TechnologickyChecklistPripojenieISPZS\_vXY.docx |
| Inštalačné súbory | Súčasť iintegračného balíčka |
| Ďalšie informácie | *EhealthCryptoController* je realizovaný formou Microsoft .NET „assembly“, ktoreho autenticita je zbezpečena digitalnym podpisom.  EhealthCryptoController využíva na komunikáciu s I&A subsystémom komunikačnú platformu *Windows Comunication Foundation* (WCF) s využitím bezpečnostných komunikačných štandardov WS-Trust |

Tabuľka 12 – Špecifikácia EhealthCryptoController

### Zadávanie PIN kódu pri použití elektronickych preukazov ZPr

Vyžaduje sa jeden PIN pre autentifikáciu, podpisovanie a dešifrovanie údajv v súvislosti so zdravotníckym pracovníkom. Tento PIN kód musí ZPr zadať vždy pri prvej autentifikácii, elektronickom podpise a dešifrovaní údajov, ktoré boli zašifrované priamo preňho. Celý proces zadávania a dočasného uchovávania PIN kódu pre jednotlivé operácie funguje plne v réžii Microsoft CryptoAPI[[4]](#footnote-5) - PIN po prvom zadaní, sa dočasne uchováva v chránenej pamäti operačného systému, do doby pokliaľ beží daná aplikácia, ktorá iniciovala jeho zadanie a/alebo daná čipová karta nebola vytiahnutá z čítačky.

### Predpoklady a požiadavky klientského komponentu EhealthCryptoController.

V nasledujúcich kapitolách sú zosumarizované nevyhnutné hardverové a softvérové predpoklady a požiadavky pre správne fungovanie klientského komponentu IAM s čipovými kartami zdravotníckych pracovníkov.

#### Čítačka čipových kariet zdravotníckych pracovníkov

Informácie o čítačkách čipových kariet zdravotníckych pracovníkov sú uvedené v dokumente eZdravie\_Integracny\_manual\_pre\_IS\_PZS\_x104\_vX.XX.docx.

#### Čipová karta zdravotníckeho pracovníka

Informácie o čipových kartách zdravotníckych pracovníkov sú uvedené v dokumente eZdravie\_Integracny\_manual\_pre\_IS\_PZS\_x104\_vX.XX.docx

#### Čítačka čipových kariet príjmateľov zdravotnej starostlivosti

Informácie o čítačkách čipových kariet prijímateľov zdravotnej starostlivosti sú uvedené v dokumente eZdravie\_Integracny\_manual\_pre\_IS\_PZS\_x104\_vX.XX.docx

#### eID middleware

Informácie o eID middleware (PPP aplikácia) sú uvedené v dokumente eZdravie\_Integracny\_manual\_pre\_IS\_PZS\_x104\_vX.XX.docx.

#### Ďalšie predpoklady

Komponent EhealthCryptoController zapisuje dôležité udalosti do systémového denníka udalostí (Windows EvengLog). Tieto udalosti sú zapisované pod zdrojom „*EhealthCryptoController*“.

Nakoľko pre vytvorenie takého zdroja udalostí (EventSource) v operačných systémoch Windows požaduje administrátorské oprávnenia, je potrebné aby IS PZS zabezpečil krektné vytvorenie uvedeného zdroja udalostí počas inštalácie svojho produktu.

Ak komponent EhealthCryptoController pri svojom prvom inicializovaní zistí absenciu vyššie uvedeného zdroja, pokúsi sa ho vytvoriť. Pokiaľ však proces, v rámci ktorého CryptoController je inicializovaný, nemá administtrátorské práva potrebné na jeho vytvorenie, inicializácia komponentu zlyhá s chybou.

#### Knižnice pre ECC

Komponent EhealthCryptoController tiež využíva knižnice „*BspConsulting.Security.SmartCards.dll*“ a „*BouncyCastle.Crypto.dll*“, ktoré tvoria neoddeliteľnú súčasť komponentu..

Príklad (PowerShell)

$eventsourcename = "ehealthcryptocontroller"

$logfileexists = [system.diagnostics.eventlog]::sourceexists($eventsourcename)

if (-not $logfileexists)

{

new-eventlog -logname "application" -source $eventsourcename

}

Príklad (C#)

using System.Diagnostics;

...

var sourceName = "EhealthCryptoController";

if (!EventLog.SourceExists(sourceName)

{

EventLog.CreateEventSource(sourceName);

}

### Kontinuita poskytovania služieb

Súčasťou programu implementácie eHealth je aj garancia poskytovanie eHealth služieb v prípade havárie, ktoré je zabezpečované prostredníctvom:

* Technických opatrení garantujúcich vysokú dostupnosť a obnoviteľnosť funkčnosti technických komponentov;
* spracovaním sústavy stratégií, procedúr a plánov, ktorých náplňou je zachovanie kontinuity poskytovania eHealth služieb a na ne nadväzujúcich zdravotníckych služieb.

Obrázok 18- Plán kontinutity

Technické opatrenie a plány obnovy (DRP) sú určené výhradne pre centrálne systémy eHealth. Aby bolo garantované vybavenie pacienta v elektronickej podobe, aj IS PZS musia mať aplikované princípy obnoviteľnosti funkčnosti po havárii (zálohovanie, redundancia, plány a pod.).

Alternatívne procedúry a procedúry návratu slúžia na popis aktivít, ktoré sú potrebné k tomu, aby:

* Bolo možné poskytnúť zdravotnú starostlivosť aj v prípade výpadku ktorejkoľvek zo služieb eHealth;
* Po opätovnom sprevádzkovaní služieb eHealth bolo možné skonsolidovať všetky údaje vytvorené v „off line“ režime.

K tomu boli prijaté nasledovné zásady:

* Súbežná prevádzka – využívanie elektronickej aj papierovej dokumentácie pri poskytovaní zdravotníckych služieb (pri nedostupnosti eHealth služieb bude využitá papierová dokumentácia);
* IS PZS musí byť v prípade neúspešného primárneho volania schopné opätovne vyvolať eHealth službu.

Problematika kontinuity poskytovania služieb je obsahovou náplňou dokumentov prislúchajúcich k produktu „068 - Náväzné dokumenty vrcholových bezpečnostných dokumentov". Dokument „068I - Stratégia kontinuity eHealth služieb“ obsahuje strategický prístup na zachovanie kontinuity každej z eHealth služieb (služby, ktoré majú rovnaké nároky na kontinuitu a budú sa na ne vzťahovať identické alternatívne procedúry a procedúry návratu, budú zgrupované do skupín). Dokument "068J - Technologická stratégia obnovy" obsahuje obnoviteľnosť funkčnosti NZIS v etape projektu eSO1 a tiež v konečnom štádiu riešenia.

### Elektronický podpis

Pri komunikácii informačných systémov s NZIS musí byť pre služby zápisu každá odoslaná správa, ktorá obsahuje zdravotný záznam (EHRExtrakt), opatrená elektronickým podpisom. Samotný podpis správy zabezpečí funkcia CreateEsid, ktorá je súčasťou rozhrania komponentu IEhealthCryptoController.

Vstupom pre túto službu je vždy kompletná správa, ktorá môže byť definovaná štruktúrou **all\_composition**. Výstupom je ESID štruktúra ktorá je s jednotlivými objektami all\_composition prepojená a obsahuje elektronický podpis ich kontrolných súčtov.

### Potvrdenie prítomnosti pacienta

Vybrané služby NZIS vyžadujú pre volanie doplnenie tokenu s potvrdením prítomnosti. Token je získavaný službou CryptoControllera „GetPatientData()“ – kap. 6.1.1.2. Ten je vložený do ehTalkMessage.Header.SenderInfo.SecurityToken.paccientPresenceConfirmationResponse. Platnosť tokenu je 20 minút od jeho získania.

Službou vyžadujúcou potvrdenie prítomnosti pacienta je služba „ZapisSuhlasOsobyPrePZS“. Ak pacient vyžaduje pre potvrdenie prítomnosti zadať svoj BOK je vytvorený token podľa kapitoly 6.1.1.2 **Metóda „GetPatientData()”** s parametrom **confirmationLevel** = **PinRequired**, ak pacient navyžaduje zadať BOK a stačí mu iba potvrdenie výzvy, je token vytvorený volaním s parametrom **confirmationLevel** = **ChallengeResponse** a text vyzvy nie je volitelny, musi byt presne "Suhlasim s \npristupom k EZK"  (s pouzitim oddelovaca \n, resp. chr(10) ).

Do 30 minút po úspešnom zavolaní služby „ZapisSuhlasOsobyPrePZS“,môže zdravotnícky pracovník získavať z NZIS zdravotné záznamy pacienta všetkých stupňov citlivosti (viď kapitola 5.4.2.2 atribút sensitivity).

### Riadenie prístupu k údajom pacientov – Consent Management

Prístup k údajom je riadený súhlasom pacienta na prístup k údajom a zoznamom doplňujúcich pravidiel pre prístup k údajom. Pre prístup k údajom v NZIS platí nasledovné:

1. Záznam je poskytnutý v prípade, že pacient poskytol súhlas na prístup k údajom (zapísaný zdravotníckym pracovníkom prostredníctvom služby ZapisSuhlasOsobyPrePZS alebo občanom prostredníctvom portálu).
2. Všeobecný lekár osoby, ktorý má vzťah definovaný v JRUZ, má prístup k všetkým záznamom osoby bez potreby súhlasu. Pri prístupe k lekárskym správam je s  obmedzením citlivosti maximálne na úroveň 3. Ak je potrebný prístup aj na správy s citlivosťou 5, je potrebné zapísať súhlas na prístup zdravotným údajom osoby s tokenom potvrdzujúcim prítomnosť pacienta (PatientPresenceConfirmation) podľa bodu 1.
3. Autorstvo záznamu neoprávňuje používateľa pre prístup k záznamu (okrem prípadu, ak je autor zároveň lekárom uvedeným v bodoch 1.,2., 3. alebo je služba určená na sprístupnenie záznamu pre autora prípadne žiadateľa – viď popis príslušnej služby)
4. Pre zápis záznamu do NZIS nie je potrebný súhlas pacienta
5. Prípadné výnimky a doplnenia týchto pravidiel sú uvedené v popise konkrétnej služby.
6. Consent Management vydáva súhlas k zdravotným záznamom aj s maximálnou povolenou citlivosťou. Ak je napríklad súhlas vydaný na citlivosť 3, je umožnené používateľovi prehliadať všetky záznamy s citlivosťou 1, 2 a 3. Ak je súhlas vydaný na citlivosť 5, je umožnené používateľovi prehliadať všetky záznamy. Pri vyhodnocovaní prístupu sa zohľadňuje súhlas s najvyššou citlivosťou.

Súhlasy je možné evidovať k nasledujúcim skupinám údajov:

|  |  |
| --- | --- |
| Kód | Skupina údajov (stránka EZKO) |
| OAU | Kontaktné a ICE údaje pacienta + Kontaktné údaje PS (Poskytovanie identifikačných údajov osoby podľa zákona 153/2013 § 5 (1) a)) |
| OUP | Osobný účet poistenca (Poskytovanie informácií z účtu poistenca podľa zákona 153/2013 § 5 (1) c)) |
| PHR | Vlastné záznamy osoby (Poskytovanie vlastných záznamov pacienta podľa zákona 153/2013 § 5 (1) d)) |
| LIEK | Medikačná história (Poskytovanie medikačných záznamov podľa zákona 153/2013 § 5 (1) b) 12.) |
| REC | Lekárske predpisy a lekárske poukazy (Poskytovanie preskripčných záznamov podľa zákona 153/2013 § 5 (1) b) 10.) |
| DISP | Záznam o výdaji lieku/zdr. Pomôcky (Poskytovanie dispenzačných záznamov podľa zákona 153/2013 § 5 (1) b) 11.) |
| PSU | Pacientsky sumár + Aktívna medikácia + Očkovania (Poskytovanie pacientskeho sumáru podľa zákona 153/2013 § 5 (1) b) 1. |
| DPS | Doplnkové zdravotné údaje PS (Poskytovanie doplnkových zdravotných údajov osoby podľa zákona 153/2013 § 5 (1) b) 15.) |
| LBZ | Laboratórne žiadanky (Poskytovanie žiadaniek na vyšetrenia spoločných vyšetrovacích a liečebných zložiek vrátane popisu vzorky podľa zákona 153/2013 § 5 (1) b) 3.) |
| LBV | Laboratórne výsledky (Poskytovanie záznamov o výsledku vyšetrenia spoločných vyšetrovacích a liečebných zložiek podľa zákona 153/2013 § 5 (1) b) 4.) |
| VVP | Vyšetrenia pacienta a lekárske správy (Poskytovanie záznamov z ambulantnej zdravotnej starostlivosti a ústavnej zdravotnej starostlivosti podľa zákona 153/2013 § 5 (1) b) 8. a 9.) |
| VLA | Odporúčanie (výmenný lístok) na ambulantnú ZS (Poskytovanie záznamov o odporúčaní lekára na špecializovanú ambulantnú zdravotnú starostlivosť podľa zákona 153/2013 § 5 (1) b) 6.) |
| VLU | Odporúčanie (výmenný lístok) na ústavnú ZS (Poskytovanie záznamov o odporúčaní lekára na prijatie do ústavnej zdravotnej starostlivosti podľa zákona 153/2013 § 5 (1) b) 7.) |

Pri službách, kde je vyžadovaný súhlas pacienta pre prístup k údajom je v prílohe „Prehľad služieb“ uvedený kód príslušnej skupiny údajov (stránky EZKO).

# Sieťová komunikácia

## Sieťová komunikácia na služby NZIS

### Zabezpečenie komunikácie medzi IS PZS a NZIS

Komunikácia medzi IS PZS a NZIS je realizovaná dvoma spôsobmi:

1. Pripojením do virtuálnej privátnej siete (VPN)
2. Pripojením cez internet pomocou SSL protokolu

#### VPN

Pre vytvorenie VPN spojenia je potrebné, aby pracovná stanica IS PZS mala prístup do Internetu pomocou SSL protokolu cez HTTPS / DTLS na portoch 443/tcp a 443/udp na cieľovú adresu <https://vpn.npz.sk> (<https://vpn.preprod.npz.sk>).

V rámci nadviazaného spojenia je potrebné zabezpečiť aby žiaden prvok prostredia PZS neblokoval nasledovné prístupy:

1. Services.infra.npz.sk – port TCP 80 certifikat platny do 30.9.2017 UPOZRONENIE PO TOMTO TERMINE je nevyhnutne pouzivat novy endpoint
2. services1.infra.npz.sk – port TCP 80 certifikat platny od 25.8.2017
3. sts-zpr.infra.npz.sk – port TCP 443
4. ocsp.infra.npz.sk – port TCP 80,443
5. pki.infra.npz.sk – port TCP 80,443

e) cms-zpr-aktiv.infra.npz.sk port, 443 (reset PIN)

Na pracovných staniciach používateľov IS PZS bude potrebné nainštalovať a nakonfigurovať špecializovaný VPN klient SW, následne bude šifrovanie komunikácie pre SW komponenty IS PZS plne transparentné.

VPN klient SW: **Cisco AnyConnect Secure Mobility Client,**

Získanie inštalačného programu: Je súčasťou integračného balíčka

Nastavenie pripojenia: Pri prvom úspešnom pripojeni VPN klienta na cieľovú adresu si VPN klient automaticky stiahne profil s potrebnými nastaveniami.

Autentifikácia: Pre prístup sa používa certifikát na čipovej karte používateľa. Vyžaduje sa overenie PIN-om.

Ukončenie spojenia: Ukončenie spojenia je vykonané buď po uplynutí timeoutu (po uplynutí doby nečinnosti tzn. 3600 sekúnd alebo po uplynutí maximálnej povolenej doby pripojenia v tomto prípade je to neobmedzené) alebo po odpojení od siete, vytiahnutí karty z čítačky alebo manuálnym odpojením od VPN.

Podporované OS: Windows 10, 8, 8.1, a 8.1 Update 1 x86 (32-bit) a x64 (64-bit)

#### Pripojenie cez internet a SSL

Pre vytvorenie spojenia prostredníctvom sieťe internet je potrebné aby pracovná stanica IS PZS mala prístup do internetu pomocou SSL protokolu na porte 443/TCP na cieľové adresy:

1. [ssl-zpr.npz.sk](https://services.infra.npz.sk)– port TCP
2. [sts-zpr.npz.sk](https://sts-zpr.infra.npz.sk)– port TCP 443
3. pki.npz.sk – port TCP 80
4. cms-zpr-aktiv.npz.sk port, 443 (reset PIN)

IS PZS musí zabezpečiť, že pri komunikácii voči službám NZIS je opakovane používané to isté SSL spojenie a nedochádza ku opätovnému nadväzovanieu SSL spojenia pri každom volaní služby. Timeout pre vyjednané SSL parametre spojenia na strane NPZ je 3600 sekúnd.

## Inštalácia čítačky ePZP kariet

Postup inštalácia a podporované čítačky ePZP kariet sú popísané v rámci dokumentu eZdravie\_Integracny\_manual\_pre\_IS\_PZS\_x104\_vXY.docx

## Inštalácia ovládačov ePZP kariet

MiddleWare pre použitie čipových kariet - Windows Mini Driver. Môže sa stiahnuť automaticky po vložení karty, alebo sa nainštaluje pri pripájaní IS PZS do NZIS. Drivre sú popísané v dokumente eZdravie\_Integracny\_manual\_pre\_IS\_PZS\_x104\_vXY.docx a priložené v rámci jeho príloh

## Inštalácia dôveryhodného certifikátu certifikačnej autority NCZI

Pre správne fungovanie pripojenia na služby NZIS je nutné zaradiť certifikáty certifikačných autorít NCZI do úložiska dôveryhodných koreňových a sprostredkovateľských certifikátov. Certifikát vydá prevádzkovateľ riešenia.

Príklad volania vystavenej služby informačným systémom PZS

Pre komunikáciu na všetkých službách je využívaná jednotná dátová štruktúra. Na najvyššej úrovni je správa definovaná štrukúrou eHTalkMessage (viď kapitola [5.4.1 – eHtalkMessage](#_eHtalkMessage)).V tejto kapitole uvedený popis krokov potrebných k tomu, aby táto štruktúra mohla byť pripravená a odoslaná na NZIS.

## Sumarizácia krokov potrebných k volaniu vystavenej služby

### Naplnenie informácii o odosielateľovi – *SenderInfo*.

Informácia sa skladá z častí *SecurityToken* a *UserContext*.

Časť *SecurityToken* je potrebné získať volaním služby *GetSamlTokenForHealthProfessional*, ktorú poskytuje *EhealthCryptoController* (viď kapitola 6.1.1.5 **Metóda „GetSamlTokenForHealthProfessional”** ). Podmienkou je prítomný bezpečnostný predmet (čipová karta) zdravotníckeho pracovníka. SecurityToken je špecifický pre službu, ktorá bude volaná.

Výsledkom je Microsoft .NET Framework objekt typu *System.IdentityModel.Tokens.SecurityToken.*

Platnosť tokenu je možné zistiť z poľa

XPATH: /trust:RequestSecurityTokenResponseCollection/trust:RequestSecurityTokenResponse/trust:Lifetime/wsu:Expires/text()

Pričom

wsu=”<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd>”

a

trust=“http://docs.oasis-open.org/ws-sx/ws-trust/200512“

Token musí byť používaný opakovane počas práce so službami NZIS a nový token vyžiadať až v momente keď pôvodný nie je možné použiť. Token nesmie byť použitý pokiaľ do konca jeho platnosti je menej ako 5 minút. Časť *UserContext* je potrebné naplniť identifikáciou odborného útvaru PZS a voliteľne identifikáciou špecializácie, v rámci ktorej zdravotnícky pracovník vystupuje.

### Príprava tela správy

Telo správy (časť *Body.Data*) je špecifické pre konkrétnu vystavenú službu (viď kapitola [5.4.1.3 – Body](#_Body)). Špecifikácie dátových štruktúr sú popísané v dokumente Prehľad služieb a jednotlivých schémach, ktoré tvoria prílohu produktu. Do dátových štruktúr, ktoré sú definované archetypom podľa CEN13606 je potrebné vygenerovať štruktúru do podoby EhrExtrakt (viď kapitola [5.4.2 – EHRExtract](#_EHRExtract_1)) aj použitím syntetických kontajnerov (viď kapitola [5.4.2.1 – Triedy CEN EN 13606](#_Triedy_CEN_EN)) a v nej je potrebné doplniť všetky povinné metadáta definované normou CEN13606 (viď kapitola [5.4.2.2 – Povinné atribúty EHR Extractu](#_Povinné_atribúty_EHR)).

Pri službách, ktoré vyžadujú nový identifikátor záznamu je pri jeho vytvorení postupovať podľa kapitoly [5.4.4.2 *–* Identifikácia zdravotných záznamov](#_Identifikácia_zdravotných_záznamov).

### Získanie identity pacienta

**Pri komunikácii s NZIS sa komunikuje výlučne prostredníctvom šifrovaného JRUZ identifikátora pacienta.** Identifikátor prijímateľa ZS, je zašifrovaný pre NZIS pomocou služby *CreateESID***,** ktorú poskytuje *EhealthCryptoController* (viď kapitola 6.1.1.7 **Metóda „CreateEsid”** ). Vstupom je identifikátor prijímateľa zdravotnej starostlivosti a pre zápisové služby EHR\_EXTRACTu aj telo správy (viď kap. 6.1.6 Elektronický podpis). Výstupom volania metódy CreateEsid je reťazec znakov (string), reprezentujúci digitálne podpísaný a následne zašifrovaný identifikátor prijímateľa zdravotnej starostlivosti (ESID) transformovaný do Base64 formy.

Samotné získanie JRUZ identifikátora, ktorý má byť zašifrovaný je možné volaním služby DajJRUZIdentifikátor (viď dokument Prehľad Služieb, kapitola 6 – Jednotná referenčná údajová základňa), ktorá umožňuje získať identifikátor napr. na základe rodného čísla pacienta alebo prostredníctvom volania služby komponentu IEhealthCryptoController – GetPatientData (viď kap. 6.1.1.2 **Metóda „GetPatientData”** ”). Takto získaný a zašifrovaný identifikátor prijímateľa ZS je potrebné použiť pri službách, ktoré na vstupe vyžadujú identitu pacienta.

### Príprava identifikačných údajov správy

Časť *Header.MessageInfo* špecifikuje, typ požadovaného spracovania, identifikáciu dokumentu a väzby na prípadné predchádzajúce dokumenty a adresu kde bude očakávaná odpoveď (viď kapitola [5.4.1.2 – Header](#_Header).

### Volanie služby

IS PZS vykoná volanie vystavenej služby na definovanom EndPointe (ESB) (viď kapitola [5.3 – Koncové body komunikácie](#_Koncové_body_komunikácie)). V prípade vybraných služieb, kde je pre prístup k dátam potrebný súhlas prijímateľa ZS je potrebné zapísať súhlas prostredníctvom služby ZapisSuhlasOsobyPrePZS. Viď kap. 6.1.8 Riadenie prístupu k údajom pacientov – Consent Management

### Získanie výsledku

Pri volaní synchrónnej služby IS PZS obdrží novú eHTalkMessage, ktorá obsahuje odpoveď z volanej vystavenej služby.

Pri volaní asynchrónnej služby ESB odpovie IS PZS s informáciou o úspešnom prijatí požiadavky a následne IS PZS dohľadáva odpoveď z volanej služby v komponente ResponseManager. Tento postup je popísaný v kapitole [5.2 – Asynchrónne volanie](#_Asynchrónne_volanie) a kapitole [5.4.1.2 - Header](#_Header), kde sú popísané vzájomne previazania požiadavky a odpovede.

V prípade ak po volaní služby systém nedostane odpoveď, alebo obdrží chybovú správu, je potrebné volanie opakovať – viď kapitola [5.6 - Reakcia na chyby v komunikácii](#_Reakcia_na_chyby).

## Popis príkladov

Všetky príklady sú dodávané vo formáte xml správ, ktoré majú byť odosielané a prijímané IS PZS. Sady príkladov za jednotlivé oblasti (EZKO, MPD, ALOK) sú vyhotovené z rôznych pohľadov.

Za oblasti EZKO, MPD, ako aj eAlokácie sú dodávané príklady pokrývajúce kombinácie vstupných a výstupných správ z hľadiska použitej štruktúry:

* Vstupná správa ADL štruktúra – výstupná správa XSD štruktúra;
* Vstupná správa ADL štruktúra – výstupná správa ADL štruktúra;
* Vstupná správa XSD štruktúra – výstupná správa XSD štruktúra.

Všetky uvedené príklady sú dostupné v prílohách produktu (časť príklady).

# HTML validátor

HTML Validátor je pomocná knižnica pre tvorcov ISPZS v rámci systému eZdravie. Účelom nástroja je mať možnosť lokálne skontrolovať HTML text pre potreby zaslania do systému NZIS. Týmto spôsobom je možné vykonať lokálnu validáciu, s rovnakým výsledkom ako je validácia NZIS systému.

Validátor nie je určený pre vytváranie kontrol pre koncového používateľa, jedná sa o middleware. Výstup validátora je čisto technický a jeho prípadná integrácia musí zohľadňovať kontext v akom je použitý. Štandardné použitie HTML validátora je pre vývojárov ISPZS počas tvorby integrácie.

Spustiteľný súbor sa nachádza pri ostatných binárných súboroch v x104\_Binary\_v.zip

**Namespace** - Ness.HtmlValidator

**Assembly** - Ness.HtmlValidator.dll

**Class** - public sealed class HtmlValidatorInternal

**Methods**

public static bool Validate(

string htmlString,

StringBuilder sb

)

**Vstupné parametre:**

htmlString – HTML na validáciu

sb – StringBuilder, slúžiaci na zber/zápis zistených výnimiek, musí byť vytvorený (nesmie byť NULL)

**Návratová hodnota:**

Návratová hodnota je typu bool vo význame:

* True – zadané html (parameter htmlString) je valídne, neboli zapísané žiadne chyby
* False – zadané html (parameter htmlString) nie je valídne, boli pridané validačné chyby (parameter sb)

**Výnimky:**

ArgumentException

* "Input html string is null or empty!"
  + ak bol ako htmlString parameter zadaný prázdny alebo null reťazec
* "StringBuilder is not created! (sb == null)"
  + ak StringBuilder zadaný ako druhý parameter nebol vytvorený

**Exception:**

Všeobecná výnimka, pre neštandardné / neočakávané chyby. Nemalo by sa stať, ale teoretická možnosť tu je.

**Závislosti:**

* .NET Framework 4.0 a vyššie
* Knižnice tretích strán
* HtmlAgilityPack vo verzii v1.6.0 (dostupný ako Nuget package)
* Príklad (C#)

**Kód:**

string inputInValid =

@"<!DOCTYPE html>

<html>

<head>

<meta charset=""utf-8""/>

<link rel=""stylesheet"" type=""text/css"" href="""">

</head>

<BODY>

<div>

<br />

<ul>

<li> Lorem <bdi dir=""rtl""> bdi: ipsun </bdi> Lorem ipsun </li>

<li> Lorem <bdo dir=""rtl""> bdo: ipsun </bdo> Lorem ipsun </li>

</ul>

<br />

</BODY>

</HTML>";

var sb = new StringBuilder();

if (!HtmlValidatorInternal.Validate(inputInValid, sb))

{

Console.WriteLine("HTML1 KO");

Console.WriteLine("");

Console.WriteLine(sb.ToString());

}

else

{

Console.WriteLine("HTML1 OK");

}

string inputValid =

@"<!DOCTYPE html>

<html>

<head>

<meta charset=""utf-8""/>

<link rel=""stylesheet"" type=""text/css"" href="""">

</head>

<BODY>

<div>

<br />

<ul>

<li> Lorem <bdi dir=""rtl""> bdi: ipsun </bdi> Lorem ipsun </li>

<li> Lorem <bdo dir=""rtl""> bdo: ipsun </bdo> Lorem ipsun </li>

</ul>

<br />

</div>

</BODY>

</HTML>";

sb.Clear();

if (!HtmlValidatorInternal.Validate(inputValid, sb))

{

Console.WriteLine("HTML2 KO");

Console.WriteLine("");

Console.WriteLine(sb.ToString());

}

else

{

Console.WriteLine("HTML2 OK");

}

**Výstup:**

**HTML1**: KO, End tag </div> was not found

**HTML2**: OK

1. http://tools.ietf.org/html/rfc4122 [↑](#footnote-ref-2)
2. <http://en.wikipedia.org/wiki/Luhn_algorithm> [↑](#footnote-ref-3)
3. Všetky príklady sú len ilustratívne v zmysle rámca použitia CV definovaného normou CEN/TS 14796. Tento príklad v žiadnom prípade neilustruje použitie v  NZIS. [↑](#footnote-ref-4)
4. <http://technet.microsoft.com/fr-fr/library/ff404300(v=ws.10)> [↑](#footnote-ref-5)