ČESKÁ SPRÁVA SOCIÁLNÍHO ZABEZPEČENÍ

Křížová 25, 225 08 Praha 5

Podávací a dotazovací protokol pro e-podání ČSSZ

Systém pro zpracování e-podání ČSSZ

Verze 1.4~~1. 23~~4, datum aktualizace ~~31~~16. 12. 201~~19~~

Historie verzí

Verze	Datum	Popis
0.8	13._9._2010	Předběžná verze
0.81	20._9._2010	Verze pro testování vývojářů HPN v komunitním prostředí
0.9	7._10._2010	Aktualizovaná verze (doplnění popisu VREP, doplnění informací k webovým službám PVS a VREP, aktualizace informací - spuštění v produkčním prostředí)
1.0	13._10._2010	Rozšíření a doplnění informací pro podání přes VREP a ISDS
1.1	18._10._2010	Doplnění informací pro web
1.2	12._11._2010	Doplnění ID produkční datové schránky
1.21	13._12._2010	Oprava adres VREPu v testovací větvi
1.22	22._03._2011	Oprava hodnot TimestampValue pro ggdisg
1.23	31._03._2011	Doplnění informací ohledně používání elementu PollInterval
1.24	8._6._2011	Nová testovací datová schránka
1.3	~~xx~~31. 12~~xx~~.~~xxxx~~ 2011	Aktualizace při ukončení provozu transakční části PVS
1.4	16. 12. 2019	Aktualizace o nové typy podání, podpora více příloh pro kanál ISDS, preferované URL pro VREP, začištění textu

Obsah

Seznam zkratek

IČPE	identifikační číslo pracoviště pro elektronická podání (identifikátor ošetřujících lékařů)
ISDS	informační systém datových schránek
PVS	portál veřejné správy (zde myšlena transakční část), již není provozován
SSZ	správa sociálního zabezpečení
VREP/APEP	veřejné rozhraní pro e-Podání
VS	variabilní symbol (identifikátor zaměstnavatele, OSVČ,..)

Úvod

Tento dokument popisuje rozhraní, formáty zpráv a podávací a dotazovací protokol pro zasílání e-podání na ČSSZ.

Informace v tomto dokumentu jsou plně platné pro komunikační kanál VREP (nově též označován jako APEP, v tomto dokumentu nadále používáme původní název VREP). ~~Pro PVS jsou doplňkové (aktuální platné informace pro PVS publikuje provozovatel PVS, v případě odlišnosti informací o PVS v tomto dokumentu a v dokumentaci PVS platí vždy aktuální informace PVS).~~ Pro ISDS jsou informace v tomto dokumentu závazné pro aplikační protokol (viz dále v tomto dokumentu), informace ohledně metod a rozhraní ISDS jsou doplňkové (aktuální platné informace o protokolu a rozhraní pro ISDS publikuje provozovatel ISDS, v případě odlišnosti informací o ISDS v tomto dokumentu a v dokumentaci ISDS platí vždy aktuální informace ISDS). Informace o rozhraní B2B webových služeb pro eNeschopenku (od 1.1.2020) určené pro lékařský SW jsou zveřejněny v samostatných dokumentech ke stažení na webu ČSSZ.

V případě komunikačních kanálů, které používají i jiné orgány veřejné správy, je tento dokument použitelný pouze pro komunikaci s ČSSZ, tj. informace obsažené v tomto dokumentu jsou specializované pro komunikaci s ČSSZ a nelze je aplikovat na komunikaci s jinými orgány veřejné správy, byť by se jednalo o stejný komunikační kanál (~~PVS,~~ ISDS).

Pro zasílání e-podání na ČSSZ jsou použity běžné standardy (http, ssl/tls, XML, XSD, SOAP, PKCS, CMS, X.509), které umožňují komunikaci s ČSSZ bez závislosti na platformě (operačním systému, vývojovém prostředí, programovacím jazyku atd.).

Fragmenty kódu, uvedené v tomto dokumentu, slouží pouze jako ukázky pro upřesnění informací ohledně sledu prováděných operací apod. Jedná se o ukázky, které jsou zaměřeny zejména na funkčnost e-podání ČSSZ, nemusí tedy nutně reprezentovat obecně uznávaná doporučení v jiných oblastech (jmenné konvence, práce s pamětí, ošetření výjimek apod.). Uvedené příklady též nemusí představovat nejvhodnější řešení pro všechny aplikace (např. načítání certifikátů ze souboru či výběr z úložiště certifikátů apod.), vhodnost pro konkrétní použití je závislá na požadavcích na aplikaci. Jakékoliv použití ukázkového kódu v aplikacích je plně na zodpovědnosti vývojáře, včetně posouzení výkonnostních a jiných dopadů na vlastní aplikaci, která takový kód využije.

Ukázky kódu jsou v jazyce C# (Microsoft .NET Framework). Doplnění ukázek v jiných programovacích jazycích je ponecháno vývojářské komunitě.

Při implementaci protokolu do aplikací pro elektronické podávání u kanálůu ~~PVS a~~ VREP je důležité věnovat pozornost doporučenému nastavení intervalu pro vyzvedávání odpovědí a ukončování transakcí. Dodržením přispějete k plynulému zpracování e-podání.

Aktuality o e-Podání pro SW vývojáře jsou dostupné na https://www.cssz.cz/web/cz/informace-pro-sw-vyvojare.

e-Podání

Elektronické podání je elektronická datová struktura. Jedno podání může obsahovat 1-1500 tzv. formulářů (stejného typu e-podání, v jednom podání není možné kombinovat formuláře různých typů e-podání). Formulář je část elektronického podání, která nese informace odpovídající vyplněnému jednotlivému papírovému formuláři (tiskopisu). Základní technické předpoklady k e-Podání naleznete na https://www.cssz.cz/web/cz/technicke-predpoklady-k-e-podani.

Druhy e-podání

~~Kapitola bude postupně doplňována (zejména pro nové druhy e-Podání), informace jsou v současné době zveřejněny na webových stránkách~~

Zaměstnavatelé podávají:

- Evidenční list důchodového pojištění (ELDP)
- Oznámení o nástupu do zaměstnání (ONZ)
- Přehled o výši pojistného (PVPOJ)
- Potvrzení o studiu (o teoretické a praktické přípravě) pro účely důchodového pojištění (POS)
- Příloha k žádosti o dávku nemocenského pojištění (NEMPRI)
- Hlášení zaměstnavatele při ukončení pracovní neschopnosti (HZUPN)
- Dotaz zaměstnavatele na DPN (DZDPN)

Osoby samostatně výdělečně činné podávají:

- Přehled o příjmech a výdajích osob samostatně výdělečně činných (OSVC)
- Hlášení osoby dobrovolně nemocensky pojištěné při ukončení pracovní neschopnosti (HZUPN)

Ošetřující lékaři podávají

- Hlášení pracovní neschopnosti (HPN) – od 1.1.2020 primárně skrz nové rozhraní B2B WS

Systémové podání

- Registrace nového certifikátu pro VREP (USRCERT)

Aktuální informace o podporovaných typech e-Podání jsou zveřejněny na webových stránkách ČSSZ v sekci pro SW vývojáře.

~~**Evidenční listy (ELDP, RELDP)**~~

~~***Formáty a verze podání***~~

~~**Oznámení o nástupu do zaměstnání (ONZ, PRIHL)**~~

~~***Formáty a verze podání***~~

~~**Hlášení pracovní neschopnosti (HPN)**~~

~~Podání HPN slouží k zasílání formulářů hlášení pracovní neschopnosti (HPN, jsou elektronickou podobou papírových formulářů rozhodnutí o dočasné pracovní neschopnosti, tj. tzv. „neschopenky") ze~~

~~zdravotnických zařízení na ČSSZ. Pro elektronickou komunikaci jsou připraveny definice tří typů formulářů: díl 1, díl 2 a hlášení ošetřujícího lékaře.~~

~~Podání HPN zasílá zdravotnické zařízení a předává informace o dočasně práce neschopných (pojištěncích), tj. v datech podání není zasílán variabilní symbol žádné organizace, není ani na současném papírovém tiskopise (variabilní symbol zdravotnického zařízení není pro ČSSZ relevantní a nelze předpokládat, že lékař či pojištěnec znají variabilní symbol pojištěncova zaměstnavatele). Podání HPN tedy nepoužívá VS, ve zprávě submission request tedy část GovTalkDetails/Keys nemá obsahovat klíč s typem „vars" a variabilním symbolem.~~

~~Formáty a verze podání~~

~~HPN1.0~~

~~HPNDS3~~
~~Datový standard Ministerstva zdravotnictví (DASTA, http://ciselniky.dasta.mzcr.cz/) verze 3 (DS3) (připravuje se).~~

~~HPNDS4~~
~~Datový standard Ministerstva zdravotnictví (DASTA, http://ciselniky.dasta.mzcr.cz/) verze 4 (DS4) (implementace zvažována dle rozšíření použití verze 4).~~

~~Příloha k žádosti o dávku (NEM PRI)~~

~~Formáty a verze podání~~

~~Přehled OSVČ (OSVC PRE)~~
~~Podání musí obsahovat právě jeden formulář.~~

~~Formáty a verze podání~~

~~POSTP (POSTP)~~

~~Formáty a verze podání~~

~~PVPOJ (PVPOJ)~~
~~Podání musí obsahovat právě jeden formulář.~~

~~Formáty a verze podání~~

Aplikační komunikační protokol podání-odpověď

Aplikační komunikační protokol je sada komunikačních pravidel na aplikační úrovni (na rozdíl od protokolů na nižších vrstvách síťového modelu komunikace). Pro e-podání ČSSZ je aplikační komunikační protokol definován dvojicí podání-odpověď. Podání je zpráva, která obsahuje data elektronických formulářů či data požadavku/požadavků. Na každé podání je právě jedna odpověď, odpověď obsahuje potvrzení o přijetí podání do zpracování příp. seznam chyb, které je nutné opravit, nebo data vyžádaná požadavkem/y.

Vzhledem ke komplexnímu způsobu zpracování podání v informačním systému ČSSZ není možné garantovat synchronní odpověď (tj. v rámci jednoho http spojení jako reakci na zaslané podání). Pro běžná podání jsou odpovědi dostupné v řádu jednotek minut, nicméně běžně nastávají situace (např. plánovaná údržba systému či el. sítě apod.), které vedou k tomu, že odpovědi jsou dostupné po několika hodinách či dokonce druhý den. Aplikace, implementující komunikační protokol pro e-podání ČSSZ, tedy musí počítat s tím, že mohou být přerušeny před možností vyzvednout odpověď, a musí umět po opětovném spuštění opakovat požadavek na vyzvednutí odpovědi. Jedním z možných (a ~~doporučených – za~~doporučených – za podmínek zajištění přístupu k uloženým datům pouze autorizovaným osobám) způsobů řešení tohoto způsobu komunikace je ukládání seznamu odeslaných podání a jeho načítání při opětovném spuštění.

Přijetí či zamítnutí podání, částečné přijetí

Odpověď na podání obsahuje informac~~í~~ o přijetí či zamítnutí podání jako celku. Aplikace nutně musí pro každé podání odpověď zpracovat na úrovni vyhodnocení výsledku zpracování jednotlivých formulářů.

Pro některé typy podání (RELDP, ~~ONZ, ELDP,~~ POSTP, ~~PVPOJ~~OSVC) je podání jako celek zamítnuto v případě, že se v kterémkoliv formuláři vyskytuje chyba. V takovém případě je nutné zaslat po opravě chyby znovu všechny formuláře.

Pro jiné typy podání (NEM-PRI, HPN10 a HPN18) je podání zamítnuto jako celek pouze tehdy, pokud neobsahuje ani jeden formulář, který by bylo možné zpracovat. V opačném případě je podání označeno jako přijaté, ovšem zpracovány budou pouze formuláře, které nevykazují chyby – tato situace se označuje jako částečné přijetí podání. Je nezbytné, aby aplikace, implementující komunikační protokol pro e-podání ČSSZ, vyhodnotila informace v odpovědi na úrovni jednotlivých formulářů (tj. nikoliv pouze na úrovni celého podání) a v případě, že se jedná o částečné přijetí, umožnila opravu chybných formulářů a jejich následné opakované zaslání. Jedním z možných (a doporučených – za podmínek zajištění přístupu k uloženým datům pouze autorizovaným osobám) způsobů řešení je ukládat originály odesílaných podání (ev. Šifrovat podání kromě šifrování pro ČSSZ též vlastním klíčem) a umožnit načtení dat pro opravu z těchto originálů.

Pro většinu nových podání (ELDP09, PVPOJ13 a novější atd.) lze prostřednictvím atributu partialAccept / castecnePrijeti v datové větě podání zvolit mezi režimem částečného přijetí nebo zamítnutí jako celku.

Opravné podání

V případě, že dojde k omylu při zaslání nesprávných údajů, a je třeba zaslat opravné podání, je třeba zaslat znovu kompletní (tj. ne pouze opravované údaje) všechny opravované formuláře. Některé druhy podání (RELDP/ELDP, PVPOJ, HPN) umožňují označit formulář jako opravný. To je nutné pouze

v případě, že formulář podání s nesprávnými údaji byl přijat. Pokud byl formulář zamítnut, je třeba zaslat formulář po opravě znovu, nejedná se však o opravné podání.

Opravné podání tedy slouží pro opakované zaslání opravených údajů přijatých formulářů.

Opakované zaslání opravených údajů zamítnutých formulářů není opravným podáním, ale podáním běžným.

Storno podání

Je možné pouze u podání PVPOJ, a to na úrovni podání (tj. požadavek na storno podání je obsažen v informacích vyplňovaných do formuláře), nikoliv komunikačního protokolu (není definována žádná specializovaná zpráva, požadavek na storno se posílá jako zpráva submission request).

Žádný z ostatních současně používaných druhů e-podání nedefinuje možnost elektronického zaslání požadavku na storno předchozího podání, zejména z důvodů neexistující podpory pro takovou akci v cílových agendách ČSSZ. Požadavek na storno podání je tedy u dalších druhů podání nutné uplatnit jiným způsobem – telefonicky, osobně, žádostí přes e-podatelnu či datové schránky apod.

Komunikační kanály

e-Podání mohou být na ČSSZ doručena ~~několika~~ dvěma elektronickými komunikačními kanály: ~~přes Portál veřejné správy (PVS),~~ přes Veřejné rozhraní pro e-podání (VREP) či přes Informační systém datových schránek (ISDS).

Pozn. Pro eNeschopenku od 1.1.2020 je jenom pro lékařský SW vystaveno samostatné rozhraní B2B webových služeb. Rozhraní je popsáno v samostatném dokumentu.

Pro každý kanál jsou popsány požadavky (registrace, potřebné identifikátory apod.), autentizace v rámci komunikačního kanálu, komunikační vzor (message exchange pattern, tj. implementace aplikačního komunikačního protokolu na úrovni zpráv daného kanálu) a rozhraní.

~~Aktuální nastavení kanálů pro příjem e-podání v komunitním prostředí ke dni 18.10.2010~~xx.xx.xxxx ~~31.12.2011 pro VREP a ISDS je uvedeno v následující tabulce.~~

~~e-podání~~	~~kanál~~			
	~~**PVS**~~	~~**VREP**~~	~~**ISDS (XMLs podpisem a šifrováním)**~~	~~**ISDS (holé XML)**~~
~~**RELDP/ELDP09**~~	~~Ano~~	~~Ano~~	~~Ano~~	~~Ano~~
~~**POSTP**~~	~~Ano~~	~~Ne~~	~~Ne~~	~~Ne~~
~~**ONZ/PRIHL**~~	~~Ano~~	~~Ano~~	~~Ano~~	~~Ano~~
~~**OSVC_PRE**~~	~~Ano~~	~~Ano~~	~~Ano~~	~~Ano~~
~~**PVPOJ/PVPOJ09**~~	~~Ano~~	~~Ano~~	~~Ano~~	~~Ano~~
~~**NEM_PRI**~~	~~Ano~~	~~Ano~~	~~Ano~~	~~Ano~~
~~**HPN**~~	~~Ne~~	~~Ano~~	~~Ano~~	~~Ano~~

Aktuální nastavení kanálů pro příjem e-podání v ~~produkčním prostředí~~produkčním prostředí ~~—~~ke dni ~~18.10.2010~~xx.xx.xxxx ~~31~~16.12.201~~9~~9 pro VREP a ISDS je uvedeno v následující tabulce.

e-podání	kanál		
	VREP	**ISDS (~~XMLs~~GovTalk XML ~~podpisem a šifrováním~~)**	**ISDS (holé XML)**
PRIHL / ONZ	Ano	Ano	Ano
RELDP / ELDP~~09~~	~~Ne~~Ano	~~Ne~~Ano	~~Ne~~Ano
PVPOJ	Ano	Ano	Ano
POSTP	~~Ne~~Ano	~~Ne~~Ano	~~Ne~~Ano
~~**ONZ/PRIHL**~~	~~Ne~~Ano	~~Ne~~	~~Ne~~
~~**OSVC_PRE**~~	~~Ne~~Ano	~~Ne~~	~~Ne~~
~~**PVPOJ/PVPOJ09**~~	~~Ne~~Ano	~~Ne~~	~~Ne~~
NEM_PRI	~~Ne~~Ano	~~Ne~~Ano	~~Ne~~Ano
~~**HPN**~~	~~Ano~~	~~Ano~~	~~Ano~~

HZUPN	Ano	Ano	Ano
ZZVDPN	Ano	Ano	Ano
DZDPN	Ano	Ano	Ano
HPN (původní)	Ano	Ano	Ano
OSVC	Ano	Ano	Ano
USRCERT	Ano	Ano	Ne

Společné a souhrnné informace

Vlastnosti, požadavky a doporučená nastavení jednotlivých komunikačních kanálů

Jednotlivé komunikační kanály vyžadují pro e-podání splnění rozdílných požadavků daných specifiky daného komunikačního kanálu (ať už technickými, či legislativními). Nejvýznamnější požadavky jsou shrnuty v následující tabulce:

	~~PVS~~ VREP	ISDS
Kvalifikovaný certifikát	Vyžadován	~~Doporučen~~Nevyžadován
Datová schránka	Není vyžadována	Vyžadována
Specializovaný software pro komunikaci*	Vyžadován **	~~Vyžadován~~ Doporučen ***
Běžný čas dodání odpovědi ****	5 minut – 1 hodina	1 hodina – 1 den
Doporučený dotazovací interval***	12x5 minut, poté 1 hodina	1 hodina

* Ačkoliv je použitý formát zpráv (XML) čitelný ~~(a ev. je možné sestavit požadovanou XML zprávu s použitím obecných sw nástrojů, např. Poznámkový blok)~~, pro sestavení samotného podání je vzhledem k použité komprimaci, šifrování a podepisování vyžadován specializovaný software. Obdobně pro zpracování odpovědi je vzhledem k vhodnosti ověření podpisu odpovědi a vzhledem k použití šifrování a komprimace nutný specializovaný software.

** ~~PVS ani~~ VREP ~~nemají~~ nemá uživatelské rozhraní pro zaslání podání. Pro zaslání podání přes ~~tato rozhraní~~ VREP je vyžadován software, který umožní uživateli zaslat podání (musí implementovat podávací a dotazovací protokol daného kanálu).

*** ISDS má uživatelské rozhraní, přístupné z prohlížeče, které umožňuje zaslat podání. ~~Ke své funkčnosti vyžaduje specializovaný software (602 XML Filler), instalovaný jako ActiveX komponenta v prostředí prohlížeče.~~ Pomocí tohoto rozhraní je možné poslat předem připravené podání ve formátu „holého" XML (tj. bez komprimace, šifrování, podepisování). Toto XML je však nutné předem sestavit v jiném nástroji.~~Vzhledem k tomu, že i prop~~přes kanál ISDS ~~bude podávajícímu zaslána odpověď s výsledkem zpracování ve tvaru XML, doporučujeme používat pro účely e-podání používat podepsané a šifrované podání, je k sestavení podání vyžadován specializovaný software.~~, ~~který samozřejmě může zajistit i komunikační funkci (pokud implementuje podávací a dotazovací protokol kanálu ISDS).~~

**** Jedná se o čas pro podání s maximálně 300 formuláři (hodnota je daná aktuální konfigurací a může být měněna na základě aktuální zátěže systému zpracovávajícího e-podání). Zpracování podání s větším počtem formulářů je odkládáno na dobu mimo špičku, tj. na večerní hodiny.

***** Pro plynulost zpracování na straně ČSSZ je důležité v podávající aplikaci tento Doporučený dotazovací interval nezkracovat.

Vlastnosti a požadavky jednotlivých druhů podání

Jednotlivé druhy e-podání vyžadují různé identifikační údaje (variabilní symbol VS, identifikační číslo pro e-podání IČPE), mají různé požadavky a přijetí či zamítnutí ~~celého~~ podání je celkové či částečné. Informace pro jednotlivé druhy podání jsou shrnuty v následující tabulce:

	VS	IČPE	Registrace na ČSSZ	Kvalifikovaný certifikát	Datová schránka	~~Přijetí/zamítnutí~~Částečné přijetí
~~ONZ (PRIHL) /~~ ONZ	V~~v~~yžadován	Není použito	Vyžadována (VS, certifikát nebo databox)	Vyžadován pro ~~PVS a~~ VREP	Vyžadována pro ISDS	PRIHL ne, ONZ ~~Vždy všechny formuláře~~dle atributu v DV
RELDP / ~~(RELDP)~~	Vyžadován~~v yžadován~~	Není použito	Vyžadována (VS, certifikát nebo databox)	Vyžadován pro ~~PVS a~~ VREP	Vyžadována pro ISDS	RELDP ne, ELDP dle atributu v DV~~Vždy všechny formuláře~~
PVPOJ	Vyžadován	Není použito	Vyžadována (VS, certifikát nebo databox)	Vyžadován pro VREP	Vyžadována pro ISDS	Do PVPOJ12 ne, od PVPOJ13 dle atributu v DV
POSTP	Vyžadován ~~Vyžadován~~	Není použito	Vyžadována (VS, certifikát nebo databox)	Vyžadován pro ~~PVS a~~ VREP	Vyžadována pro ISDS	Ne~~Vždy všechny formuláře~~
~~PVPOJ~~	~~Vyžadován~~	~~Není použito~~	~~Vyžadována (VS, certifikát nebo databox)~~	~~Vyžadován pro PVS a VREP~~	~~Vyžadována pro ISDS~~	~~Vždy všechny formuláře~~
NEM_PRI	Vyžadován	Není použito	Vyžadována (VS, certifikát nebo databox)	Vyžadován pro ~~PVS a~~ VREP	Vyžadována pro ISDS	~~Jednotlivé formuláře podání~~Ano (vždy)
~~HPN~~	~~Není použit~~	~~Vyžadováno~~	~~Vyžadována (certifikát nebo databox a IČPE)~~	~~Vyžadován pro VREP **~~	~~Vyžadována pro ISDS~~	~~Jednotlivé formuláře podání~~
~~OSVC_PRE~~	~~Vyžadován~~	~~Není použito~~	~~Není vyžadována~~	~~Vyžadován pro PVS a VREP *~~	~~Vyžadována pro ISDS~~	~~Vždy všechny formuláře~~
HZUPN	Vyžadován pro zaměstnavatele	Není použito	Vyžadována pro zaměstnavatele (VS, certifikát nebo databox)	Vyžadován pro VREP *	Vyžadována pro ISDS *	Ano (vždy)
ZZVDPN	Ne (podává pojištenec)	Není použito	Není vyžadována	Vyžadován pro VREP *	Vyžadována pro ISDS *	Ne (jen jeden formulář)
DZDPN	Vyžadován	Není použito	Vyžadována (VS, certifikát nebo databox)	Vyžadován pro VREP	Vyžadována pro ISDS	Ne (jen jeden formulář)
HPN	Není použit	Vyžadováno	Vyžadována (certifikát nebo databox a IČPE)	Vyžadován pro VREP	Vyžadována pro ISDS	Ano (vždy)

OSVC	Vyžadován	Není použito	Není vyžadována	Vyžadován pro VREP *	Vyžadována pro ISDS *	Ne (jen jeden formulář)
USRCERT	Vyžadován	Není použito	Vyžadována (VS, certifikát nebo databox)	Vyžadován pro VREP	Vyžadována pro ISDS	Dle atributu v DV

* Pro podání zasílaná přímo pojištencem ~~přehledu OSVČ~~ není vyžadována předchozí registrace certifikátu či datové schránky v registračním systému ČSSZ. ~~Při registraci na PVS je vyžadován pouze VS, který má každá OSVČ přidělen, není vyžadováno registrační číslo, není tedy nutná registrace na ČSSZ, stačí pouze registrace na PVS.~~

~~** HPN není možné zasílat přes PVS~~

~~Portál veřejné správy (PVS)~~

~~PVS byl prvním komunikačním kanálem pro e-podání. PVS je provozován Ministerstvem vnitra ČR (http://www.mvcr.cz/clanek/portal-verejne-spravy.aspx). Podrobná aktuálně platná dokumentace podávacího a dotazovacího protokolu PVS je publikována provozovatelem PVS, pokud se rozchází informace v tomto dokumentu a informace dokumentace PVS, jsou platné informace v dokumentaci PVS.~~

~~Prerekvizity~~

~~Pro zaslání e-podání pro ČSSZ přes PVS je vždy nutný kvalifikovaný certifikát vydaný akreditovanou certifikační autoritou. Dále je vždy nutná registrace ke službám ČSSZ na PVS, která pro většinu druhů e-podání ČSSZ (všechny kromě OSVC PRE) vyžaduje registrační číslo, vydané při registraci na ČSSZ. Pro všechna e-podání je pro zaslání přes PVS třeba variabilní symbol, vydaný ČSSZ (HPN neobsahuje VS, ale podání HPN se přes PVS nezasílá).~~

~~*Registrace*~~

~~Registrace ke službám ČSSZ na PVS je nezbytná pro zaslání podání přes PVS na ČSSZ. Pro registraci na PVS je pro většinu druhů e-podání (výjimkou je podání přehledu OSVČ) nutné registrační číslo, které je přidělováno při registraci na ČSSZ.~~

~~Registrace na PVS~~

~~Pro registraci ke službám jsou nutné tzv. známé údaje. Jedním z nich je variabilní symbol, který přiděluje ČSSZ (všem organizacím automaticky), a který je vyžadován pro všechny služby. Druhým je registrační číslo, které též přiděluje ČSSZ.~~

~~Registrace na PVS je jednorázovou záležitostí.~~

~~*Registrace na ČSSZ*~~

~~Pro ověření oprávnění podávat podání daného typu a pro přidělení registračního čísla, nutného pro registraci na PVS, je nutné navštívit územní pracoviště SSZ a na základě pověření organizace požádat o registraci. Pro registraci je nutné předložit kvalifikovaný certifikát (veřejnou část certifikátu na el. nosiči nebo výpis vydaný vystavitelem certifikátu, resp. přesné plné jméno vystavující certifikační autority a sériové číslo certifikátu v dekadické nebo hexadecimální podobě) vydaný akreditovanou certifikační autoritou, který bude používán k podepisování podání (ČSSZ provádí autentizaci jednotlivých podání bez ohledu na komunikační kanál).~~

~~Registrace na ČSSZ je jednorázovou záležitostí, před vypršením platnosti registrovaného certifikátu je možné registraci obnovit elektronicky (lze vzdáleně), jinak je nutná opakovaná návštěva územního pracoviště.~~

Autentizace

PVS provádí autentizaci na úrovni komunikačního kanálu ověřením registrace podávajícího ke službám ČSSZ na PVS. Autentizace je možná jménem a heslem či klientským certifikátem.

PVS nepředává autentizační informace, použité pro přihlášení, ani identifikaci autentizovaného uživatele, na ČSSZ. ČSSZ provádí autentizaci a autorizaci zaslaných podání z informací v podpisu podání.

Komunikační vzor

Komunikační protokol PVS přímo podporuje aplikační komunikační protokol podání – odpověď. Pro zajištění garantovaného dokladovatelného doručení podání a potvrzeného převzetí odpovědi a asynchronní (neblokující) komunikace nad sadou běžných standardních síťových protokolů (http, ssl/tls) a vzhledem k časům zpracování je aplikační komunikační protokol implementován následující sadou zpráv:

Podání (submission request, zasílá podávající) – doručenka (submission acknowledgement, zasílá PVS), Dotaz (submission poll, zasílá podávající) – Odpověď (submission response, zasílá PVS), Ukončení transakce (delete request, zasílá podávající) – Potvrzení (delete response, zasílá PVS).Toto je hlavní (ideální) větev zpracování. V případě výskytu chyby či v případě prodloužení zpracování jsou vraceny zprávy submission error resp. submission acknowledgement, zprávy a variantní cesty zpracování jsou popsány v části Formáty zpráv.

Služby, definované na PVS pro zasílání podání na ČSSZ, vyžadují předávání variabilního symbolu organizace, za kterou je podání zasíláno. Tento musí být vždy vyplněn v sekci GovTalkDetails/Keys v klíči s nastaveným typem „vars". PVS používá tento VS spolu s autentizačními údaji při autentizaci, na rozdíl od autentizačních údajů však VS v podání zůstává a ČSSZ jej též používá pro kontrolu obsahu podání (dle jednotlivých druhů podání).

Rozhraní

PVS má dvě rozhraní: rozhraní POX („plain old XML", XML over HTTPS) a rozhraní webových služeb.

Obě rozhraní jsou na úrovni http protokolu bezestavová, nepoužívají tedy ani cookies ani session cookies ani proměnné v URL.

POX

Rozhraní XML přes http PVS přímo podporuje komunikační vzor implementující aplikační komunikační protokol podání-odpověď, tj. obsahuje koncové body pro podání a pro dotaz na výsledek zpracování. Klient (podávací software) posílá sestavenou zprávu, která obsahuje potřebné struktury podání či požadavku na vyzvednutí odpovědi, v těle standardního http požadavku (metodou POST). Reakce na zprávy (opět ve formátu XML) jsou vraceny PVS synchronně v rámci jednoho http spojení v těle odpovědi protokolu http. Další zprávy v rámci komunikačního vzoru (dotaz na výsledek) jsou opět zasílány v těle dalšího požadavku v rámci dalšího http spojení.

Jedná se o rozhraní postavené nad základními protokoly (http, ssl/tls).

Hlavička Content-type protokolu HTTP musí být nastavena na hodnotu text/xml.

WS

Rozhraní webových služeb PVS přímo podporuje komunikační vzor implementující aplikační komunikační protokol podání-odpověď. Služby tohoto rozhraní jsou nasazeny na třech koncových bodech (endpointech) dle způsobu autentizace: koncový bod pro anonymní autentizaci (pouze

~~metody, které nevyžadují autentizaci), koncový bod pro autentizaci jménem a heslem a koncový bod pro autentizaci certifikátem. Samotné služby rozhraní implementují metody pro podání (submit), dotaz na výsledek zpracování (poll) a požadavek na uzavření transakce (dispose). Klient (podávací software) volá přes tzv. proxy třídy metody webové služby s parametry, ve kterých předává jednotlivé části zprávy (tělo, hlavičky apod.).~~

~~Rozhraní webových služeb je implementováno s využitím protokolu SOAP nad protokolem http (se zabezpečením kanálu ssl/tls). Jednotlivé zprávy komunikačního vzoru jsou předávány jako samostatná volání metod webových služeb, s daty a parametry předávanými v polích struktur definovaných v rámci kontraktu služby.~~

~~Interface a kontrakt rozhraní jsou popsány v kapitole WS PVS a WS VREP.~~

~~Metoda Submit může být volána na koncovém bodu /anonymous, /username i /certificate (koncový bod /anonymous se v tomto případě využije pouze u služeb/transakcí nevyžadující registraci).~~

~~Metoda Poll může být volána na koncovém bodu /anonymous, /username i /certificate.~~

~~Metoda Dispose může být volána na koncovém bodu /anonymous, /username i /certificate.~~

Veřejné rozhraní pro e-podání (VREP)

VREP je nový komunikační kanál, který má za cíl co největší kompatibilitu s původním~~e~~ ~~stávajícím~~ rozhraním PVS~~,~~ tak, aby bylo ~~možné~~možné ~~pokud možno~~ použít stávající komunikační komponenty s minimem nutných změn. Jedná se o přímé rozhraní ČSSZ, vyžaduje pro většinu podání registraci v registračním systému ČSSZ ~~(pro většinu podání kromě přehledu OSVČ)~~, nevyžaduje webovou registraci (ve srovnání s PVS).

Prerekvizity

Pro zaslání podání přes VREP je vždy nutný kvalifikovaný certifikát vydaný akreditovanou certifikační autoritou. Pro většinu druhů e-podání (kromě HPN) je třeba variabilní symbol, vydaný ČSSZ. Pro podání hlášení pracovní neschopnosti (HPN, e-neschopenky) není variabilní symbol vyžadován, ale je nutné identifikační číslo pro e-podání (IČPE, nejedná se o registrační číslo).

Registrační číslo, ač je při registraci vždy implicitně přiděleno, není v průběhu další registrace ani následně pro odeslání podání~~, na rozdíl od PVS,~~ již dále třeba.

Registrace na ČSSZ

Pro ověření oprávnění podávat podání daného typu je nutné ~~navštívit územní pracoviště SSZ a na základě pověření organizace~~na místně příslušné správě sociálního zabezpečení~~S~~ požádat o registraci. Pro registraci je nutné předložit/zaslat vyplněný formulář ~~*Oznámení*~~*Oznámení o pověření k zajištění všech úkonů souvisejících s e-Podáním* resp. *Sdělení doplňujících údajů* (určeného pro ~~staturární~~statutární orgány apod.) a kvalifikovaný certifikát (veřejnou část certifikátu na el. nosiči nebo výpis vydaný vystavitelem certifikátu, resp. přesné plné jméno vystavující certifikační autority a sériové číslo certifikátu v dekadické nebo hexadecimální podobě) vydaný akreditovanou certifikační autoritou, který bude používán k podepisování podání. ~~(ČSSZ provádí autentizaci jednotlivých podání bez ohledu na komunikační kanál).~~

Pro zasílání podání, která obsahují VS, je nutné při registraci předložit potvrzení organizace o oprávnění zasílat podání uvedeného typu. Na základě tohoto potvrzení je možné provést registraci pro jednotlivé druhy podání.

Pro zasílání podání, která obsahují IČPE (HPN), bude IČPE přiděleno (vygenerováno) při registraci na pracovišti SSZ.

Registrace na ČSSZ je jednorázovou záležitostí, před vypršením platnosti registrovaného certifikátu je možné registraci obnovit elektronicky (lze vzdáleně), jinak je nutná opakovaná návštěva územního pracoviště.

Autentizace

Autentizace je prováděna na úrovni jednotlivých zpráv (podání) spolu s autorizací (ověření oprávnění zasílat podání), na úrovni komunikačního kanálu není autentizace prováděna (není tedy třeba jméno a heslo či použití certifikátu v rámci sestavení zabezpečeného kanálu).

Komunikační vzor

Komunikační protokol VREP přímo podporuje aplikační komunikační protokol podání – odpověď. Pro zajištění garantovaného dokladovatelného doručení podání a potvrzeného převzetí odpovědi a asynchronní (neblokující) komunikace nad sadou běžných standardních síťových protokolů (http, ssl/tls) a vzhledem k časům zpracování je aplikační komunikační protokol implementován následující sadou zpráv:

Podání (submission request, zasílá podávající) – doručenka (submission acknowledgement, zasílá VREP), Dotaz (submission poll, zasílá podávající) – Odpověď (submission response, zasílá VREP), Ukončení transakce (delete request, zasílá podávající) – Potvrzení (delete response, zasílá VREP). Toto je hlavní (ideální) větev zpracování. V případě výskytu chyby či v případě prodloužení zpracování jsou ~~vraceny~~ vráceny zprávy submission error resp. submission acknowledgement, zprávy a variantní cesty zpracování jsou popsány v části Formáty zpráv.

Výše uvedený sled je hlavní cestou zpracování. Podávající software (klient) zasílá podání ve zprávě submission request, na kterou VREP (pokud je podání správně formátováno) odpovídá (synchronně, v rámci jednoho http spojení) doručenkou submission acknowledgement. Doručenka obsahuje unikátní identifikátor podání (VREP Correlation ID), který je nutný pro další kroky. Doručenka též obsahuje časovou značku VREPu, tj. datum a čas převzetí podání ke vstupnímu zpracování, opatřené podpisem technologické komponenty. Časová značka slouží jako „podací lístek" k prokázání zaslání podání v daném čase a zároveň umožňuje ověřit identitu příjemce podání (VREP). Následně se musí podávací software dotázat na stav vstupního zpracování. Doporučeným intervalem mezi podáním a dotazem na stav zpracování je interval, který se vrátí jako hodnota elementu „PollInterval". Tato hodnota je uvedena ve vteřinách. V ostatních případech je **doporučeným intervalem mezi podáním a dotazem na stav zpracování 5 minut**, stejný interval je doporučeno zachovat i pro ev. opakované dotazy, případně je vhodné interval prodloužit, **ovšem není doporučeno interval zkracovat**. Dotaz na stav vstupního zpracování (submission poll) musí obsahovat VREP Correlation ID, VREP na dotaz odpovídá synchronně buď odpovědí (submission response) (pokud už vstupní zpracování proběhlo a alespoň jeden formulář podání je v pořádku) nebo chybou (pokud už vstupní zpracování proběhlo a žádný formulář v podání nelze přijmout) či doručenkou (v případě, že vstupní zpracování ještě nebylo dokončeno). Pokud VREP odpoví doručenkou, je třeba po určitém časovém intervalu (viz výše) opakovat dotaz na stav zpracování. Pokud VREP odpoví informací o přijetí (submission response) či zamítnutí (submission error) podání, je vyžadováno, aby klientská aplikace zaslala požadavek na uzavření transakce (delete request), na který VREP odpoví potvrzením uzavření transakce (delete response). **Aplikace, které nebudou korektně uzavírat ~~transakce, budou~~transakce, budou považovány za aplikace porušující pravidla pro zasílání e-podání, neboť tím mohou působit provozní problémy.** Pokud VREP na požadavek na uzavření transakce

odpoví doručenkou (delete acknowledgement), je třeba, aby aplikace opakovala po uplynutí určitého časového intervalu (viz výše) požadavek na ukončení transakce.

V případě, že se v kterékoliv části zpracování vyskytnou problémy, je vrácena chyba (submission error), tj. sekvence zpráv pak neběží hlavní cestou zpracování (podání – doručenka, dotaz – odpověď, ukončení transakce – potvrzení ukončení transakce), ale běží jednou z níže uvedených cest

- podání – chyba
- podání – doručenka, dotaz – chyba
- podání – doručenka, dotaz – odpověď, ukončení transakce - chyba

Chyba jako reakce na dotaz na výsledek zpracování (tj. v sekvenci zpráv podání – doručenka, dotaz - chyba) může být logická chyba aplikačního komunikačního protokolu (dotaz na výsledek neexistujícího podání), syntaktická chyba aplikačního komunikačního protokolu (nesprávně formátovaná zpráva dotazu na výsledek, zpráva bez CorrelationID) aj., může se ale jednat i o chybu na úrovni dat podání (tj. nikoliv na úrovni aplikačního komunikačního protokolu) (tzv. odmítnutí podání v případě, že podání obsahuje chyby, kvůli kterým není možné jej převzít do zpracování). Vyhodnocení může na straně podávající aplikace vyžadovat interakci uživatele.

Chyba jako reakce na podání (tj. v sekvenci ~~podání – chyba~~podání – chyba) či požadavek na uzavření transakce (v sekvenci podání – doručenka, dotaz – odpověď, požadavek na ukončení transakce - chyba) je vždy chybou aplikačního komunikačního protokolu (např. nesprávný formát zprávy, nenaplnění povinných polí či naplnění nesprávnými hodnotami (VS, email) či neexistující podání apod.), tj. nejedná se o chybu v datech podání.

VREP vzhledem k následné autentizaci a autorizaci jednotlivých podání nevyžaduje autentizaci pro komunikaci. Z důvodu zachování stejného způsobu validace zaslaných dat však pro stávající druhy e-podání, které používají variabilní symbol, používá stejnou logiku kontroly VS z obálky podání proti datům formulářů, jaká je použita pro podání z PVS. VREP tedy nevyžaduje plnění Header/SenderDetails (autentizační údaje), nicméně pro podání, která se vážou k organizaci (v současné době všechna kromě HPN), vyžaduje předání VS v GovTalkDetails/Keys.

Rozhraní

VREP má dvě rozhraní: rozhraní POX („plain old XML", XML over HTTPS) a rozhraní webových služeb.

~~Obě rozhraní jsou dostupná na dvou adresách pro základní zajištění možnosti komunikace v případě výpadku jedné z lokalit, ve kterých má ČSSZ rozhraní nasazeno (tzv. „backup transport" pattern). Je vhodné, aby podávající komponenty v případě chyby síťové komunikace automaticky vyzkoušely záložní adresu).~~

Obě rozhraní jsou na úrovni http protokolu bezestavová, nepoužívají tedy ani cookies ani session cookies ani proměnné v URL.

POX

Rozhraní XML přes http VREP přímo podporuje komunikační vzor implementující aplikační komunikační protokol podání-odpověď, tj. obsahuje koncové body pro podání a pro dotaz na výsledek zpracování. Klient (podávací software) posílá sestavenou zprávu, která obsahuje potřebné struktury podání či požadavku na vyzvednutí odpovědi, v těle standardního http požadavku (metodou POST). Reakce na zprávy (opět ve formátu XML) jsou vraceny VREPem synchronně v těle http odpovědi v rámci jednoho http spojení. Další zprávy v rámci komunikačního vzoru (dotaz na výsledek) jsou opět zasílány v těle dalšího požadavku v rámci dalšího http spojení.

Jedná se o rozhraní postavené nad základními protokoly (http se zabezpečením kanálu ssl/tls).

Hlavička Content-type protokolu HTTP musí být nastavena na hodnotu text/xml.

WS

Rozhraní webových služeb VREP přímo podporuje komunikační vzor implementující aplikační komunikační protokol podání-odpověď. Služba tohoto rozhraní je nasazena na ~~jednom~~ koncovém bodě (endpoint) s anonymní autentizací (neboť VREP autentizaci na úrovni komunikačního kanálu nepoužívá, viz výše), implementuje metody pro podání (submit), dotaz na výsledek zpracování (poll) a požadavek na uzavření transakce (dispose). Klient (podávací software) volá přes tzv. proxy třídy metody webové služby s parametry, ve kterých předává jednotlivé části zprávy (tělo, hlavičky apod.).

Rozhraní webových služeb je implementováno s využitím protokolu SOAP nad protokolem http (se zabezpečením kanálu ssl/tls). Jednotlivé zprávy komunikačního vzoru jsou předávány jako samostatná volání metod webových služeb, s daty a parametry předávanými v polích struktur definovaných v rámci kontraktu služby.

Interface a kontrakt rozhraní jsou popsány v kapitole WS VREP~~WS VREPWS VREPWS VREPWS PVS a WS VREP~~.

~~WS PVS a~~ WS VREP

Webová služba pro podávání ~~je implementována na platformě Windows Communication Foundation a~~ podporuje standardy WS-Security a WS-Addressing. Služba je umístěna pod symbolickým názvem Public.svc a implementuje jediné veřejné rozhraní IBusinessTransaction.

Implementace služby VREP v maximální možné míře zachovává kompatibilitu s implementací služby na PVS.

WSDL služby je dostupné na adrese: https://t-epodani.cssz.cz/VREP/ws/APEP.wsdl

Rozhraní IBusinessTransaction

Rozhraní IBusinessTransaction má následující veřejné metody:

Název metody	Popis
PollResponse Submit(string tclass, BindingList<BodyPart> bodies, BindingList<OptionalParameter> optionals)	Odešle podání
PollResponse Poll(string correlationid, BindingList<OptionalParameter> optionals)	Dotáže se na výsledek zpracování (resp. odpověď)
void Dispose(string correlationid, BindingList<OptionalParameter> optionals)	Ukončí transakci

Metoda Submit

```
PollResponse Submit(string tclass, BindingList<BodyPart> bodies,
BindingList<OptionalParameter> optionals)
```

Metoda zajistí předání podání na ~~PVS či~~ VREP. Návratovou strukturou je instance třídy PollResponse s informací o stavu podání.

Parametry

Název	Datový typ	Popis
tclass	String	Typ/identifikátor transakce
bodies	BindingList<BodyPart>	Seznam struktur BodyPart (pro e-

		podání ČSSZ vždy právě jedna)
Optionals	BindingList<OptionalParameter>	Viz. prvek OptionalParameter

Pokud při předání podání ke zpracování nedojde k chybám, je vrácena instance třídy PollResponse, která obsahuje kromě základních identifikačních údajů podání (correlationID, tj. jedinečný identifikátor podání (transakce), nezbytný pro následující dotazy na výsledek zpracování) i tělo (element Body) zprávy SUBMISSION_ACK, které obsahuje podepsanou časovou značku ~~PVS či~~ VREP.

Výjimky

Pokud při předání podání ke zpracování dojde k chybám, je vyvolána výjimka FaultException, která ve vlastnosti Detail nese instanci třídy GGErrorException s popisem chyby.

```
[FaultContract(typeof(GGErrorException))]
```

Chyba při zpracování požadavku.

~~MessageSecurityException~~

~~Chyba při kontrole autentizace, např. nesprávné heslo. Může nastat v případě použití koncových bodů /username a /certificate (pouze PVS; VREP nepoužívá).~~

Ukázka použití

```
public void SendSubmission(byte[] data)
{
    XmlDocument xBodyData = new XmlDocument();
    xBodyData.LoadXml(Encoding.UTF8.GetString(data));

    BodyPart body = new BodyPart();
    body.Id = "0";
    body.Body = xBodyData.DocumentElement;

    BindingList<BodyPart> bodyList = new BindingList<BodyPart>();
    bodyList.Add(body);

    BindingList<OptionalParameter> optionalPars = new
BindingList<OptionalParameter>();
    optionalPars.Add(new OptionalParameter() { ParameterName = "emailaddress",
ParameterValue = email });
    optionalPars.Add(new OptionalParameter() { ParameterName = "key", ParameterType
= "vars", ParameterValue = vars });
    optionalPars.Add(new OptionalParameter() { ParameterName = "timestampversion",
ParameterValue = "xmldsig" });

    ServiceClient proxy = CreateProxyClient(address);
    try
    {
        PollResponse resp = proxy.Submit("CSSZ_ONZ", bodyList, optionalPars);
    }
    catch (FaultException vrepEx)
    {
        GGErrorException ggex =
vrepEx.CreateMessageFault().GetDetail<GGErrorException>();
        //TODO
    }
    catch (XmlException xmlEx)
    {
        //TODO
    }
    catch (System.ServiceModel.Security.MessageSecurityException mesEx)
    {
        //TODO
```

```
    }
}
```

Metoda Poll

```
PollResponse Poll(string correlationid, BindingList<OptionalParameter> optionals)
```

Metoda vyžádá informaci o výsledku vstupního zpracování podání (identifikovaného parametrem correlationid, tedy identifikátorem podání (transakce) ~~PVS či~~ VREP, získaného po odeslání podání metodou Submit) resp. odpověď.

Parametry

Název	Datový typ	Popis
correlationid	String	Identifikátor transakce
optionals	BindingList<OptionalParameter>	Viz. prvek OptionalParameter

Pokud při zpracování požadavku nedojde k chybám, je vrácena instance třídy PollResponse, která obsahuje kromě základních identifikačních údajů podání i těla (elementy Body) zprávy GovtalkMessage. V případě, že již bylo dokončeno vstupní zpracování podání v systému ČSSZ, se jedná o odpověď (submission response či submission error), která v těle nese zprávu CSSZ Message s protokolem o zpracování či s daty odpovědi. Pokud vstupní zpracování ještě probíhá, jedná se o doručenku (submission ackowledgement).

Výjimky

Pokud při zpracování požadavku dojde k chybám, je vyvolána výjimka FaultException, která ve vlastnosti Detail nese instanci třídy GGErrorException s popisem chyby.

```
[FaultContract(typeof(GGErrorException))]
```

Chyba při zpracování požadavku.

~~MessageSecurityException~~

~~Chyba při kontrole autentizace, např. nesprávné heslo. Může nastat v případě použití koncových bodů /username a /certificate (PVS).~~

Ukázka použití

```
public void PollForResponse(byte[] data)
{
    ServiceClient proxy = CreateProxyClient(address);

    BindingList<OptionalParameter> optionalPars = new
BindingList<OptionalParameter>();
    optionalPars.Add(new OptionalParameter() { ParameterName = "key", ParameterType
= "vars", ParameterValue = vars });

    try
    {
        PollResponse resp = proxy.Poll(correlationID, optionalPars);
    }
    catch (FaultException vrepEx)
    {
        GGErrorException ggex =
vrepEx.CreateMessageFault().GetDetail<GGErrorException>();
        //TODO
    }
    catch (XmlException xmlEx)
    {
```

```
        //TODO
    }
    catch (System.ServiceModel.Security.MessageSecurityException mesEx)
    {
        //TODO
    }
}
```

Metoda Dispose

```
void Dispose(string correlationid, BindingList<OptionalParameter> optionals)
```

Metoda zajistí ukončení transakce v systému ~~PVS či~~ VREP.

Parametry

Název	Datový typ	Popis
correlationid	String	Identifikátor transakce
optionals	BindingList<OptionalParameter>	Viz. prvek OptionalParameter

V případě úspěšného ukončení transakce je potvrzení volání pouze na úrovni komunikačního protokolu (http status), nevrací se žádná hodnota (void).

V případě ukončení transakce dojde k vyvolání výjimky FaultException. Tato pak ve vlastnosti Detail obsahuje objekt typu GGErrorException s popisem chyby.

Výjimky

```
~~MessageSecurityException~~
```

~~Chyba při kontrole autentizace, např. nesprávné heslo. Může nastat v případě použití koncových bodů /username a /certificate (PVS).~~
Nemá specifické výjimky.

Ukázka použití

```
public void Dispose()
{
    ServiceClient proxy = CreateProxyClient(address);

    BindingList<OptionalParameter> optionalPars = new
BindingList<OptionalParameter>();
    optionalPars.Add(new OptionalParameter() { ParameterName = "key", ParameterType
= "vars", ParameterValue = vars});

    try
    {
        proxy.Dispose(correlationID, optionalPars);
    }
    catch (FaultException vrepEx)
    {
        GGErrorException ggex =
vrepEx.CreateMessageFault().GetDetail<GGErrorException>();
        //TODO
    }
    catch (System.ServiceModel.Security.MessageSecurityException mesEx)
    {
        //TODO
    }
}
```

Klient

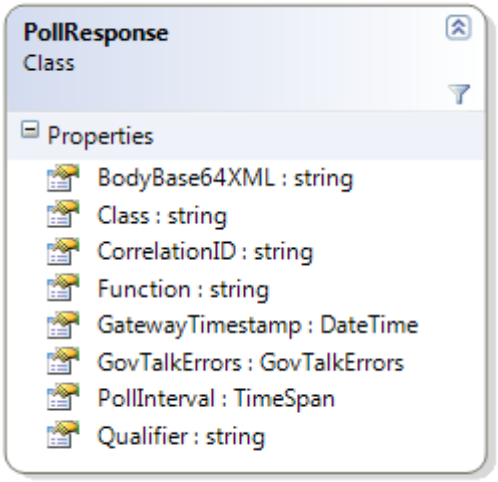
Příklad pomocné metody pro inicializaci klienta služby:

```
private ServiceClient CreateProxyClient(string address)
{
    WSHttpBinding basicHttp;
    EndpointAddress epAddress;
    ServiceClient proxy;
    basicHttp = new WSHttpBinding(SecurityMode.Transport);
    basicHttp.UseDefaultWebProxy = true;
    basicHttp.Security.Message.EstablishSecurityContext = false;
    basicHttp.Security.Message.NegotiateServiceCredential = false;
    basicHttp.MaxReceivedMessageSize = 512000L;
    basicHttp.ReaderQuotas.MaxStringContentLength = 64000;
    basicHttp.Security.Message.ClientCredentialType = MessageCredentialType.None;
// anonymous
    epAddress = new EndpointAddress(address);
    proxy = new ServiceClient(basicHttp, epAddress);
    return proxy;
}
```

Kontrakt služby

PollResponse

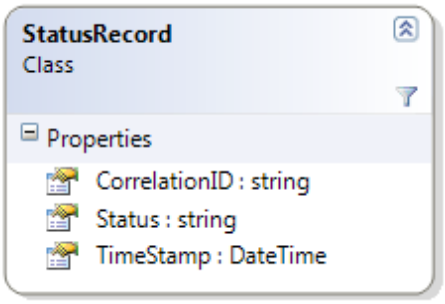
Datová struktura popisující stav podání.



Název	Datový typ	Popis
BodyBase64XML	String	Tělo XML zprávy v kódování Base64 (CSSZ Message)
Class	String	Typ transakce
CorrelationID	String	Identifikátor transakce
Function	String	Název funkce
Qualifier	String	Název typu odpovědi
GatewayTimestamp	DateTime	Časová značka přijetí podání
PollInterval	TimeSpan	Časový interval pro další dotaz POLL.
GovTalkErrors	GovTalkErrors	Viz. GovTalkErrors

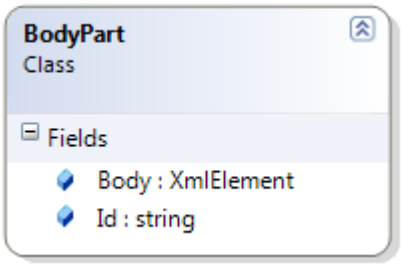
StatusRecord

Datová struktura obsahující stav podání.



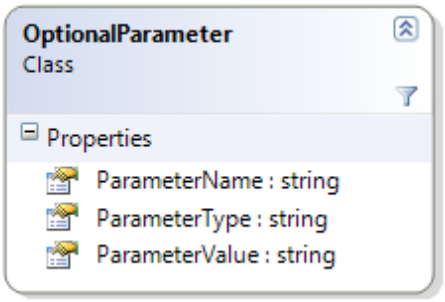
Název	Datový typ	Popis
CorrelationID	String	Identifikátor transakce
Status	String	Stav podání
TimeStamp	DateTime	Časová značka ~~PVS či~~ VREP s časem a datem přijetí podání

BodyPart



Název	Datový typ	Popis
Body	String	XML struktura CSSZ Message.
Id	String	Pořadové číslo sekce Body (indexováno od 0).

OptionalParameter



Název	Datový typ	Popis
ParameterName	String	Název parametru
ParameterType	String	Typ parametru (pouze u známého údaje)
ParameterValue	String	Hodnota parametru.

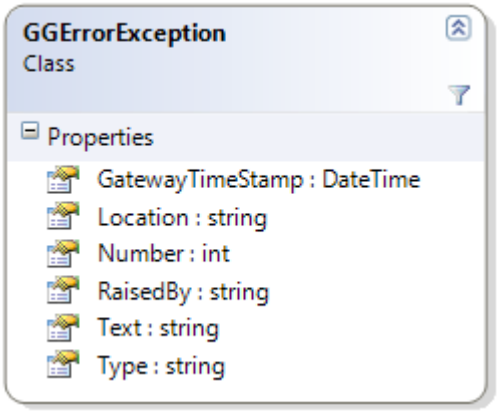
Nepovinné parametry jsou následující (case insensitive):

ParameterName	ParameterType	Popis
emailaddress	not used	Emailová adresa (popř. nomail)
formcount	not used	Statistický údaj počtu formulářů
key	Identifikátor známého údaje (vars etc.)	Známý údaj
timestampversion	not used	Definuje, jaká verze elektronické značky bude vrácena. V případě, že parametr timestampversion není uveden, bude vrácena verze elektronické značky, která je nastavena v systému jako výchozí. Možné hodnoty parametru jsou uvedeny v následující tabulce:

Hodnota	Význam
none	Elektronická značka nebude vrácena.
ggdsig	Elektronická značka bude ve formátu PVS.
xmldsig	Elektronická značka bude vrácena ve formátu dle doporučení W3C XMLDSIG - http://www.w3.org/TR/xmldsig-core/.

GGErrorException

Datová struktura obsahující popis chyby.



Název	Datový typ	Popis
GatewayTimeStamp	DateTime	Datum a čas zpracování požadavku
Location	String	Kde nastala chyba
Number	Int	Číslo chyby
RaisedBy	String	Chybu vyvolal
Text	String	Textové znění chyby
Type	String	Typ chybového hlášení

ISDS

Informační systém datových schránek je pro e-podání ČSSZ nový komunikační kanál. ISDS je provozován Ministerstvem vnitra ČR (http://www.mvcr.cz/clanek/informacni-system-datovych-schranek-isds.aspx). Podrobná aktuálně platná dokumentace rozhraní ISDS je publikována

provozovatelem ISDS, pokud se rozchází informace v tomto dokumentu a informace dokumentace ISDS, jsou platné informace v dokumentaci ISDS.

Prerekvizity

Pro zaslání podání přes ISDS je nezbytná datová schránka ISDS, ze které bude podání odesláno https://www.mojedatovaschranka.cz/PortalDS/ . ~~Je též pro zajištění integrity zprávy doporučeno použít kvalifikovaný certifikát vydaný akreditovanou certifikační autoritou pro podepsání podání.~~ Použití kvalifikovaného certifikátu není podmínkou, pokud budou podání předávána ve formě tzv. „holého" XML~~, který nevyžaduje (ani neumožňuje) přidat elektronický podpis~~. Pro většinu druhů e-podání ~~(kromě přehledu OSVČ)~~ je dále nutná registrace na ČSSZ pro ověření oprávnění zasílat podání daného typu – je vyžadována registrace datové schránky ~~a v případě použití certifikátu i registrace certifikátu~~. Pro většinu druhů e-podání (kromě HPN) je třeba variabilní symbol, vydaný ČSSZ. Pro podání hlášení pracovní neschopnosti (HPN, e-neschopenky) není variabilní symbol vyžadován, ale je nutné identifikační číslo pro e-podání (IČPE, nejedná se o registrační číslo).

Registrace

Pro ověření oprávnění podávat podání daného typu, vyjma OSVC_PRE, je nutná registrace na ČSSZ.

Registrace na ČSSZ

~~Pro registraci je nutné navštívit provést na územním pracovišti~~ a ~~N~~na základě pověření organizace je nutné na územním pracovišti požádat o registraci. Pro registraci je nutné předložit/zaslat formulář *Oznámenní o pověření k zajištění všech úkonů souvisejících s e-Podáním* resp. *Sdělení doplňujících údajů* ~~—~~(určeného pro staturární orgány apod.) ~~—~~s uvedením ~~kvalifikovaný certifikát vydaný akreditovanou certifikační autoritou, pokud bude používán k podepisování podání. Vždy je nutné předložit~~ identifikátoru datové schránky, ze které budou podání zasílána.

~~Registrace na ČSSZ je jednorázovou záležitostí, před vypršením platnosti registrovaného certifikátu je možné registraci obnovit elektronicky (vzdáleně), jinak je nutná opakovaná návštěva územního pracoviště.~~

Autentizace

ISDS provádí autentizaci na úrovni komunikačního kanálu dle přístupového koncového bodu. Autentizace je prováděna ~~buď na základě volání první metody pro „přihlášení" (DummyOperation) a předáváním získaných cookies ve všech následujících voláních (stavové rozhraní), nebo~~ předáváním autentizačních informací s každým voláním (jméno a heslo, klientský uživatelský či systémový certifikát a jejich kombinace; bezestavové rozhraní~~). V době psaní tohoto dokumentu (9/2010) je v testovacím prostředí ISDS verze, která již stavové rozhraní nepodporuje.~~

ISDS nepředává na ČSSZ autentizační informace, použité pro přihlášení, ani identifikaci autentizovaného uživatele. ČSSZ provádí autentizaci a autorizaci zaslaných podání ~~z informací~~na základě ~~v podpisu podání (pokud je přítomen) či z~~ identifikace datové schránky odesílatele, čímž plně respektuje aktuální výklad legislativních ustanovení a prováděcích předpisů ISDS, ~~která~~ které odpovědnost za přidělení přístupu pověřeným osobám a zajištění, že budou provádět pouze úkony v rámci rozsahu jejich pověření, plně přenáší na oprávněnou osobu.

Komunikační vzor

Datová zpráva ISDS (podrobnější informace viz Provozní řád ISDS a jeho přílohy) je tvořena obálkou a obsahem. Obsahem zprávy může být jedna či více příloh, formát (typ) příloh je dán výčtem v příloze prováděcí vyhlášky ISDS.

E-podání pro ČSSZ je zpráva ISDS, jejímž obsahem je ~~právě~~ jedna nebo vícejedna ~~příloha~~ ve formátu XML~~, odpovídající schématu GovTalk, tj. zpráva submission request (v případě použití šifrování a elektronického podpisu) či XML~~ odpovídající schématu některého z typů podání uvedených na http://www.cssz.cz/cz/e-podani/druhy-e-podani/ ~~http://www.cssz.cz/cz/e-podani/pro-vyvojare/struktura-datove-vety/~~ https://www.cssz.cz/web/cz/informace-pro-sw-vyvojare (tzv. „holé XML podání"). Pouze takové datové zprávy ISDS budou zpracovány automaticky systémem pro zpracování e-podání na ČSSZ. Obsah polí v obálce datové zprávy (těch, která může odesílatel nastavovat) není pro zpracování relevantní (pole věc, zmocnění, k rukám apod.), mohou být nastavena libovolně (v souladu s požadavky ISDS a povinností polí dle dokumentace ISDS).

E-podání ~~Jakékoliv jiné datové zprávy (zprávy s přílohou, která nebude ve formátu XML, či nebude odpovídat schématu GovTalk či schématu jednotlivých druhů podání, nebo zprávy s více přílohami – a to i v případě, že jednou z nich bude příloha ve formátu XML) nemohou být zpracovány automaticky.~~

~~Pro automatické zpracování v systému pro zpracování e-podání ČSSZ je dále~~ možno ~~nezbytné~~ zaslat do specializované datové schránky pro e-podání ČSSZ~~do schránky~~ nebo do datové schránky místně příslušné OSSZ/PSSZ/MSSZ~~, která je vyhrazena pro e-podání.~~ . ~~Její i~~Identifikátor specializované schránky pro e-podání ČSSZ ~~je~~ pro jednotlivá prostředí je uveden v další části tohoto dokumentu.

Na rozdíl od ~~PVS a~~ VREP (což jsou rozhraní pro implementaci komunikace podání - odpověď) je primárním zaměřením ISDS doručování dokumentů (http://www.mvcr.cz/clanek/datove-schranky-co-jsou-datove-schranky.aspx), tj. zrovnoprávnění listinné a elektronické podoby dokumentů. Pro využití ISDS pro zasílání strukturovaných automaticky zpracovatelných dat formulářů e-podání pro ČSSZ je nutné nad komunikačním rozhraním ISDS implementovat logiku aplikačního komunikačního protokolu pro e-podání.

Zaslání podání je provedeno vytvořením nové zprávy voláním metody CreateMessage. Zpráva musí mít jednu nebo více příloh~~u~~ ve formátu XML dle ~~schématu GovTalk, s obsahem dle popisu zprávy submission request (ev. XML dle~~ schématu pro „holé" podání~~)~~, a musí být doručena do datové schránky ČSSZ~~ pro e-podání~~. ČSSZ zajistí zaslání datové zprávy s odpovědí do datové schránky, ze které bylo odesláno podání. Zpráva odpovědi je datová zpráva, která má ~~právě~~ jednu nebo více příloh~~u~~ ve formátu XML dle schématu GovTalk a s obsahem submission response nebo submission error (s šifrovaným protokolem zpracování či odpovědí, pokud je k dispozici certifikát příjemce pro šifrování a jedná se o podání, které šifrovanou odpověď či protokol podporuje). Pro získání odpovědi musí aplikace podávajícího stáhnout seznam došlých zpráv voláním metody GetListOfReceivedMessages (bohužel aktuální verze rozhraní ISDS neumožňuje vyhledávat došlé zprávy podle pole věc či podle spisové značky či čísla jednacího příjemce, takže je nutné stažený seznam projít a vyhledat dle pole „věc" odpovědi na podání – viz dále v této kapitole). Následným procházením získaného seznamu a vyhledáváním dle obsahu pole „věc" lze identifikovat zprávu ISDS, která nese odpověď. Tuto zprávu je nutné následně stáhnout voláním metody SignedMessageDownload (či MessageDownload, ovšem vhodnější je dříve uvedená) a z obsahu získat přílohyu ~~u~~ ve formátu XML. ~~(doporučeným postupem je ověřit, zda je ve zprávě skutečně pouze jedna příloha, a ta je ve formátu XML).~~ Přílohyu ~~u~~ ve formátu XML je třeba zpracovat jako přijatou odpověď submission response či submission error, tj. vyhledat data protokolu zpracování či odpovědi, příp.

rozšifrovat a dekomprimovat, a zpracovat chyby na úrovni jednotlivých formulářů původního podání.[JB1]

[x2]Pro usnadnění párování zpráv obsahujících podání a odpověď garantuje systém pro zpracování e-podání ČSSZ uvedení ID datové zprávy ISDS s podáním v poli „věc" v datové zprávě s odpovědí, a to ve formátu "ČSSZ - Odpověď na e-Podání. [{0}-{1}-{2}]" (kde prvkem {0} je transakce/classname, prvkem {1} je unikátní identifikátor podání a prvkem {2} je identifikátor původní zprávy s podáním). Obdobně název přílohy je ve zprávě s odpovědí ve formátu "ČSSZ_Protokol_o_zpracování_e-Podání_{0}-{1}-{2}.xml", význam prvků je stejný (classname, correlationId, dmId). Dále jsou v datové zprávě odpovědi nastavena pole RecipientOrgUnit, RecipientOrgUnitNum, RecipientRefNumber (číslo jednací příjemce), RecipientIdent (spisová značka příjemce) na stejné hodnoty, jaké byly ve zprávě podání nastaveny v odpovídajících polích odesílatele (SenderRefNumber resp. SenderIdent).

Jiné zprávy aplikačního protokolu pro e-podání než submission request, submission response a submission error nejsou pro kanál ISDS použity (tj. není použita zpráva submission poll, submission acknowledgement, delete request a delete response). Tyto zprávy nejsou třeba, neboť rozhraní ISDS zajišťuje funkce obdobné těm, které poskytují, např. submission poll je nahrazena získáním seznamu zpráv a procházením obsahu pole „věc" v získaném seznamu. Delete request pro uzavření transakce není třeba, neboť pro případná opakovaná stažení je zpráva v odpovědi k dispozici v datové schránce po dobu 90 dnů, na ČSSZ je tedy transakce považována za uzavřenou.

Z důvodu zachování stejného způsobu validace zaslaných dat pro stávající druhy e-podání, které používají variabilní symbol, používá systém ČSSZ stejnou logiku kontroly VS z obálky podání formátu GovTalk proti datům formulářů pro všechny kanály ~~(PVS, ISDS, VREP)~~. Podání ve formátu GovTalk přes ISDS, která se vážou k organizaci (v současné době všechna kromě HPN), tedy vyžadují předání VS v GovTalkDetails/Keys. Podání ve formátu „holé XML" nevyžadují validaci VS proti žádným dalším údajům.

Rozhraní

ISDS má rozhraní webových služeb, implementované s využitím protokolu SOAP nad protokolem http (se zabezpečením kanálu ssl/tls). Jednotlivé zprávy komunikačního vzoru jsou předávány jako samostatná volání metod webových služeb, s daty a parametry předávanými v polích struktur definovaných v rámci kontraktu služby.

~~V době psaní tohoto dokumentu (9/2010) je R~~rozhraní ISDS je na úrovni http protokolu ~~implementováno dvojím způsobem: stavové (používá session cookies) a~~ bezestavové (s autentizací uživatelským jménem a heslem a/nebo klientským certifikátem pro každou zprávu). ~~V testovacím prostředí je aktuálně verze, která již stavové rozhraní nepodporuje.~~

Rozhraní webových služeb ISDS nepodporuje komunikační vzor implementující aplikační komunikační protokol podání-odpověď. Rozhraní je tvořeno několika službami, pro zaslání e-podání na ČSSZ je třeba použít služby dm_operations a dm_info.

Služby tohoto rozhraní jsou nasazeny na koncových bodech (endpoint) dle autentizace: s basic autentizací (jméno a heslo), s basic autentizací a klientským certifikátem, s klientským systémovým certifikátem a s klientským systémovým certifikátem provozovatele hostované spisové služby~~, v kombinaci s použitím bezestavového a stavového rozhraní~~.

Klient (podávací software) volá přes tzv. proxy třídy metody webové služby s parametry, ve kterých předává jednotlivé části zprávy (tělo, hlavičky apod.).

Podrobný popis rozhraní ISDS je přílohou Provozního řádu ISDS, dostupného (v ~~září 2010~~říjen 2011) na http://www.datoveschranky.info/dokumenty/ (aktuálně včetně WSDL a několika ukázek kódu).

Příklad odeslání zprávy systému datových schránek s podáním v příloze:

```
public void SendSubmission(byte[] data)
{
    WSOps.dmOperationsWebService svcOps = new WSOps.dmOperationsWebService();
    svcOps.UserAgent = "CSSZSubmissionDemo(.NET)";
    svcOps.AllowAutoRedirect = true;
    svcOps.PreAuthenticate = true;
    svcOps.Proxy = WebRequest.DefaultWebProxy;
    svcOps.Url = address;
    svcOps.Credentials = new NetworkCredential(login, password);

    WSOps.tMessageCreateOutput tmco = svcOps.CreateMessage(new
WSOps.tMessageCreateInput()
    {
        dmEnvelope = new WSOps.tMessageEnvelopeSub()
        {
            dbIDRecipient = CSSZePodaniDataBoxId,
            dmAnnotation = String.Format("Podani {1} {0:yyyyMMddHHmmss}",
DateTime.Now, className),
            dmLegalTitleLaw = "",
            dmLegalTitlePar = "",
            dmLegalTitlePoint = "",
            dmLegalTitleSect = "",
            dmLegalTitleYear = "",
            dmRecipientIdent = "",
            dmRecipientOrgUnit = "",
            dmRecipientOrgUnitNum = "",
            dmRecipientRefNumber = "",
            dmSenderOrgUnit = "",
            dmSenderOrgUnitNum = "",
            dmSenderRefNumber = cisloJednaci,
            dmSenderIdent = spisovaZnacka,
            dmToHands = ""
        },
        dmFiles = new WSOps.tFilesArrayDmFile[] {
            new WSOps.tFilesArrayDmFile() {
                dmFileMetaType = WSOps.tFilesArrayDmFileDmFileMetaType.main,
                dmMimeType = "application/xml",
                dmFileDescr = String.Format("Podani-{1}-
{0:yyyyMMddHHmmss}.xml",DateTime.Now, className),
                Item = data
            }
        }
    });
    //tmco.dmID;
}
```

Pozn.: důležité je správné nastavení požadovaných vlastností přílohy, tj. FileMetaType, MimeType atd., podrobnosti viz dokumentace ISDS.

Příklad stažení zpráv systému datových schránek:

```
public override void PollForResponse()
{
    WSOps.dmOperationsWebService svcOps = new WSOps.dmOperationsWebService();
    svcOps.UserAgent = "CSSZSubmissionDemo(.NET)";
```

```
    svcOps.AllowAutoRedirect = true;
    svcOps.PreAuthenticate = true;
    svcOps.Proxy = WebRequest.DefaultWebProxy;
    svcOps.Url = address1;
    svcOps.Credentials = new NetworkCredential(login, password);

    WSInfo.dmInfoWebService svcInfo = new WSInfo.dmInfoWebService();
    svcInfo.UserAgent = "CSSZSubmissionDemo(.NET)";
    svcInfo.AllowAutoRedirect = true;
    svcInfo.PreAuthenticate = true;
    svcInfo.Proxy = WebRequest.DefaultWebProxy;
    svcInfo.Url = address2;
    svcInfo.Credentials = new NetworkCredential(login, password);

    WSInfo.tListOfMessOutput tlmo = svcInfo.GetListOfReceivedMessages(new
WSInfo.tListOfFReceivedInput() { dmFromTime = DateTime.Now.Date.AddDays(-5),
dmToTime = DateTime.Now.Date.AddDays(1) });
    if (tlmo != null&& tlmo.dmRecords != null && tlmo.dmRecords.dmRecord != null)
    {
        foreach (var dm in tlmo.dmRecords.dmRecord)
        {
            if (dm.dmAnnotation.Contains(submissionID) == true)
            {
                WSOps.tSignedMessDownOutput tsmo = svcOps.SignedMessageDownload(new
WSOps.tIDMessInput() { dmID = dm.dmID });
                byte[] rsp = tsmo.dmSignature;
                //rsp
            }
        }
    }
}
```

Pozn.: výsledkem je stažená podepsaná datová zpráva ISDS, která má v příloze odpověď ČSSZ. Podrobnosti o formátu a zpracování podepsané datové zprávy viz dokumentace ISDS.

Podepsaná časová značka ISDS

Podepsaná časová značka ISDS je součástí datové zprávy ISDS (v elementu dmQTimestamp).

Podepsaná časová značka ISDS není v aplikační komunikaci e-podání pro ČSSZ významově nijak využívána. Informace o struktuře a možnosti ověření viz. dokumentace ISDS.

Formáty zpráv

Podání je datová struktura, která obsahuje data formulářů, odpovídající vyplněným papírovým tiskopisům. ČSSZ používá pro podání formát XML, struktura se liší pro jednotlivé typy podání a je popsána na ~~http://www.cssz.cz/cz/e-podani/druhy-e-podani/~~ https://www.cssz.cz/web/cz/informace-pro-sw-vyvojare~~http://www.cssz.cz/cz/e-podani/pro-vyvojare/struktura-datove-vety/~~.

Elektronický přenos podání má svá specifika, která mají analogii u papírových podání (podpis, obálka). Účelem jejich využití je

- zabezpečení dat podání před zpřístupněním neoprávněným osobám (k tomu slouží zejména šifrování)
- zabezpečení dat podání před modifikací neoprávněnými osobami a zajištění jednoznačné identity autora (podávajícího, původce) podání (k tomu slouží zejména elektronický podpis dat podání
- zabezpečení ověření oprávnění podávat podání daného typu (k tomu slouží registrace a identifikace na základě certifikátu použitého k podpisu podání)
- jednoznačná identifikace podání, zajištění odpovědi na každé podání v rámci komunikačního protokolu aj.

Pro přenos těchto specifických dat je definována struktura zprávy, která umožňuje zabezpečeným způsobem přenášet samotná data podání a informace pro zajištění jejich zpracování (data podpisu, identifikace apod.). ČSSZ používá pro vyměňované zprávy formát XML dle schématu GovTalk, tělo zprávy je tvořeno XML zprávou ČSSZ Message.

Pozn.: Base64 kódovaná data jsou v příkladech nahrazena komentářem, např. <!-- telo podani --> apod., z důvodů zachování alespoň částečné přehlednosti příkladů.

Pozn.: XML vyžaduje rozlišování mezi malými a velkými písmeny v názvech elementů a atributů. Pořadí elementů a atributů je dáno schématem a musí být dodrženo.

Kódování znaků

Pro všechna data, tj. jak data podání, tak zprávy (obálky) jsou podporována kódování UTF-8, Windows 1250, ISO Latin 2, ostatní kódování nejsou testována (podání v jiných kódováních mohou být úspěšně zpracována, ale v případě chyby ve zpracování, která bude souviset s kódováním, bude řešením změna kódování na straně podávajícího). **Doporučeným kódováním je UTF-8. Pro zajištění správného zpracování je doporučeno explicitně uvádět použité kódování v XML deklaraci (a to jak pro zprávy GovTalk, tak pro ev. zašifrovaná data podání)**. Pokud při zpracování ukládáte XML do souborů, je nutné ukládat ve stejném kódování, jaké je v XML deklaraci; pokud používáte Unicode (UTF-8, UTF-16 aj.), doporučujeme soubor zapisovat s tzv. byte-order-mask (BOM).

Pozor: jednotlivá podání definují pro obsah elementů a atributů povolené znaky

GovTalk Message

GovTalk Message je zpráva ve formátu XML, která v systému e-podání ČSSZ plní následující úlohy:

- obsahuje informace pro řízení aplikačního komunikačního protokolu (podání-odpověď, viz dále v tomto dokumentu) nezávisle na komunikačním kanálu

- umožňuje asynchronní (navzájem nezávislá) volání odeslání podání a dotaz na výsledek zpracování nezávisle na komunikačním kanálu
- obsahuje elementy pro přenos vlastních dat podání (slouží tedy jako jakási přenosová obálka)
- může obsahovat informace specifické pro daný komunikační kanál (autentizační údaje pro PVS, požadovaná verze časové značky v doručence pro VREP aj.)

Příklad zprávy GovTalk Message (submission request pro podání PRIHL přes VREP):

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_PRIHL</Class>
<Qualifier>request</Qualifier>
<Function>submit</Function>
<CorrelationID></CorrelationID>
</MessageDetails>
</Header>
<GovTalkDetails>
<Keys>
<Key Type="vars">1111234567</Key>
</Keys>
<GatewayAdditions>
<Flags>
<TimestampVersion>xmldsig</TimestampVersion>
</Flags>
</GatewayAdditions>
</GovTalkDetails>
<Body>
<!-- telo podani -->
</Body>
</GovTalkMessage>
```

Struktura

V tabulce níže je uveden seznam elementů a atributů XML struktury schématu GovTalk. Šedivě označené řádky reprezentují prvky, které se vyskytují pouze ve zprávách, odesílaných z ČSSZ. Tučně jsou zvýrazněny bloky, ke kterým se vztahují následující prvky. Prvky v tabulce nejsou v pořadí dle XSD.

Prvek	Význam a povinnost
http://www.govtalk.gov.uk/CM/envelope	Jmenný prostor, povinný.
GovTalkMessage	Kořenový element, povinný.
GovTalkMessage/EnvelopeVersion	Verze protokolu/formátu, povinný, požadována hodnota „2.0"
GovTalkMessage/Header	Povinný
GovTalkMessage/Header/MessageDetails	Povinný
./Class	Druh podání, povinný.
./Qualifier	Rozlišení funkce (request, poll, acknowledgement, response, error), povinný.
./Function	Povinný, funkce (submit, delete)
./TransactionID	Nepovinný
./AuditID	Nepovinný, není implementováno.
./CorrelationID	Povinný * (pro podání musí být prázdná hodnota, v doručence je vrácen unikátní

	identifikátor podání, pro dotaz na stav zpracování je vyžadována hodnota)
./ResponseEndPoint	Nepovinný
./Transformation	Nepovinný, není implementováno
./GatewayTest	Nepovinný, není implementováno
./GatewayTimestamp	Nepovinný, jen vrácené zprávy
GovTalkMessage/Header/SenderDetails	Informace o odesílateli, povinné pouze pokud jsou plněny některé podřízené elementy
./IDAuthentication/SenderID	Identifikátor odesílatele, ~~povinné pro PVS~~VREP nevyužívá
./IDAuthentication/Authentication/Method	Metoda autentizace, ~~povinné pro PVS,~~ možnosti Clear, MD5 a W3Csigned, VREP nevyužívá
./IDAuthentication/Authentication/Value	~~Povinné pro PVS pro Clear či MD5.~~ Heslo (pro Clear) či MD5 hash hesla v Base64 kódování (pro MD5), VREP nevyužívá
./IDAuthentication/Authentication/Signature	~~Povinné pro PVS pro W3Csigned~~VREP nevyužívá~~.~~
./X509Certificate	~~Povinné pro PVS pro W3Csigned~~VREP nevyužívá~~.~~[x3][JB4]
./EmailAddress	Emailová adresa pro notifikaci o zpracování, nepovinný
GovTalkMessage/GovTalkDetails	Detaily protokolu, povinný pouze pokud jsou plněny některé podřízené elementy
GovTalkMessage/GovTalkDetails/Keys	Sada polí klíč-hodnota, povinný pouze pokud jsou plněny některé podřízené elementy
./Key[@Type='vars']	Variabilní symbol, povinný ~~pouze~~ pro druhy podání které se vážou k organizaci ~~vyžadující VS~~ (~~PVS využívá~~kontroluje se shoda s VS uvedeném v datové větě podání ~~ke kontrole oprávnění, využíváno i přes ostatní kanály ke zpracování~~)
./Key[Type='spokename']	Identifikátor služby ~~PVS~~, jen vrácené zprávy
GovTalkMessage/GovTalkDetails/GatewayAdditions	Rozšíření protokolu, povinný pouze pokud jsou plněny některé podřízené elementy
./Flags/TimestampVersion	Požadovaná verze časové značky ~~PVS či~~ VREP, xmldsig pro vrácení časové značky ve struktuře xmldsig s využitím SHA-2, ~~neuvedeno či~~ ggdsig pro původní verzi struktury časové značky PVS s SHA-1
GovTalkMessage/GovTalkDetails/GovTalkErrors	Chyby podání, jen vrácené zprávy
./Error/RaisedBy	Zdroj chyby
./Error/Number	Číslo chyby
./Error/Type	Typ chyby
./Error/Text	Text chyby
GovTalkMessage/Body	Tělo zprávy~~.~~

Body

Pokud jsou v těle zprávy GovTalk Message elementy, musí být z jiného jmenného prostoru (namespace), než je namespace GovTalk.

V odeslaných zprávách (z pohledu podávajícího) musí být v těle zprávy ČSSZ Message (v submission request) nebo musí být tělo prázdné (v submission poll či delete request). Ve vrácených zprávách VREP ~~či PVS~~ je v těle podepsaná časová značka ~~PVS či~~ VREP (submission acknowledgement), ČSSZ Message (submission response, submission error) nebo je tělo prázdné (submission error, delete response).

CSSZ Message

Popsáno v samostatné kapitole.

~~Podepsaná časová značka PVS~~

~~Podepsaná časová značka PVS je vracena v těle zprávy submission acknowledgement. PVS vrací časovou značku ve dvou formátech – ggdsig a xmldsig. Původní ggdsig formát je specifický pro PVS, xmldsig je formát odpovídající standardu W3C XMLDSIG. Formát xmldsig podporuje použití algoritmů SHA-2. Podávající SW může vyžádat konkrétní formát vracené časové značky. Vzhledem k doporučenému resp. vyžadovanému přechodu na algoritmy rodiny SHA-2 je vhodné **přejít na využití formátu xmldsig**.~~

~~*XMLDSIG*~~

~~Pokud submission request či submission poll obsahoval TimestampVersion=xmldsig, je vrácena struktura xmldsig – příklad uveden níže. Pro ověření je nutná podpora algoritmů SHA-2. Podrobnosti viz. dokumentace PVS.~~

```
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#" Id="Acknowledgement">
<SignedInfo>
<CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"
/>
<SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha256" />
<Reference URI="#gg.properties"
Type="http://www.w3.org/2000/02/xmldsig#SignatureProperty">
<Transforms>
<Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
</Transforms>
<DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
<DigestValue>Qn5ICwRyCuNwFsfWmfAcsfa+ukGYG1P8vC5NHNGVbFI=</DigestValue>
</Reference>
</SignedInfo>
<SignatureValue><!-- --></SignatureValue>
<Object>
<SignatureProperties Id="gg.properties">
<SignatureProperty Target="#Acknowledgement">
<TimeStamp xmlns="urn:gg:ts.v2">2010-09-08 07:22:59</TimeStamp>
</SignatureProperty>
<SignatureProperty Target="#Acknowledgement">
<CorrelationID
xmlns="urn:gg:ts.v2">298D72D48D90404FA10C371749D99B6B</CorrelationID>
</SignatureProperty>
</SignatureProperties>
</Object>
<Object>
<SignatureProperties Id="gg.x509">
<SignatureProperty Target="#Acknowledgement">
<SignerCertificate xmlns="urn:gg:ts.v2">
<!-- -->
</SignerCertificate>
```

```
</SignatureProperty>
</SignatureProperties>
</Object>
</Signature>
```

GGDSIG

Pokud submission request či submission poll neobsahoval TimestampVersion=xmldsig, je vrácena specifická XML struktura podpisu. Pro ověření je nutné využít stejný postup canonizace a stejným způsobem převádět řetězce na byty (kódování), jako na straně PVS. Podrobnosti viz. dokumentace PVS.

Podepsaná časová značka VREP

Podepsaná časová značka VREP je vracena v těle zprávy submission acknowledgement. Implementace podepsané časové značky VREP byla navržena pro maximální kompatibilitu s podepsanou časovou značkou PVS, takže i . PVSVREP vrací časovou značku ve dvou formátech – ggdsig a xmldsig. Původní ggdsig formát je specifický pro PVS, xmldsig je formát odpovídající standardu W3C XMLDSIG a je použit na VREP. Formát xmldsig podporuje použití algoritmů SHA-2. Podávající SW může vyžádat konkrétní formát vracené časové značky. Vzhledem k doporučenému resp. vyžadovanému přechodu na algoritmy rodiny SHA-2 je vhodnédoporučujeme vývojářům podávacího SW **přejít na využití formátu xmldsig**, tj. zajistit ve všech zasílaných zprávách předávání TimestampVersion=xmldsig a validaci vrácené časové značky.[x5][JB6]

XMLDSIG

Pokud submission request či submission poll obsahoval TimestampVersion=xmldsig, je vrácena struktura xmldsig – příklad uveden níže. Pro ověření je nutná podpora algoritmů SHA-2.

```
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#" Id="Acknowledgement">
<SignedInfo>
<CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"
/>
<SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha256" />
<Reference URI="#gg.properties"
Type="http://www.w3.org/2000/02/xmldsig#SignatureProperty">
<Transforms>
<Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
</Transforms>
<DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
<DigestValue>B7S+FBNoUmf8frrJQqnJP+cGwfUCWVh4J/TDkDLWSIg=</DigestValue>
</Reference>
</SignedInfo>
<SignatureValue><!-- --></SignatureValue>
<Object>
<SignatureProperties Id="gg.properties">
<SignatureProperty Target="#Acknowledgement">
<TimeStamp xmlns="urn:gg:ts.v2">2010-09-10 07:57:30</TimeStamp>
</SignatureProperty>
<SignatureProperty Target="#Acknowledgement">
<CorrelationID
xmlns="urn:gg:ts.v2">D966D6549F4546979FA35CFF34473D9B</CorrelationID>
</SignatureProperty>
</SignatureProperties>
</Object>
<Object>
<SignatureProperties Id="gg.x509">
<SignatureProperty Target="#Acknowledgement">
<SignerCertificate xmlns="urn:gg:ts.v2"><!----></SignerCertificate>
</SignatureProperty>
</SignatureProperties>
</Object>
```

```
</Signature>
```

Pro validaci v prostředí .NET lze použít metodu CheckSignature třídy System.Security.Cryptography.Xml.SignedXml; struktura xmldsig obsahuje informace o použitých algoritmech a canonizaci, takže validace nevyžaduje kromě nalezení elementu Signature žádné speciální postupy.

Na úrovni konfigurace systému je nutné zajistit rozpoznání použitých SHA-2 algoritmů, výchozí konfigurace operačních systémů Windows v době psaní tohoto materiálu tyto definice neobsahuje. Podrobnější informace jsou popsány na http://blogs.msdn.com/b/shawnfa/archive/2008/12/02/cryptoconfig.aspx.

GGDSIG[JB7]
Pokud submission request či submission poll ~~ne~~obsahoval TimestampVersion=gg~~xml~~dsig, je vrácena specifická XML struktura podpisu, která je implementována z důvodů zpětné kompatibility a noví klienti by ji neměli používat. Pro ověření je nutné využít stejný postup canonizace a stejným způsobem převádět řetězce na byty (kódování), jako na straně VREPu.

Funkce pro canonizaci, použitá na PVS, a tudíž i VREPu, je uvedena níže:

```
private static string PVSStripWhitespaces(string xml)
{
    int beg, end, idx = 0;
    while ((idx = xml.IndexOf(">", idx)) != -1)
    {
        if (xml[idx - 1] == '/')
        {
            end = idx - 2;
            beg = end;
            while (Char.IsWhiteSpace(xml[beg])) beg--;
            xml = xml.Remove(beg + 1, end - beg);
            idx = beg + 2;
        }
        beg = idx + 1;
do
        {
            idx++;
        } while (idx < xml.Length && Char.IsWhiteSpace(xml[idx]));
        if ((idx - beg) > 0)
        {
            xml = xml.Remove(beg, idx - beg);
            idx = beg;
        }
    }
    return xml;
}
```

Zprávy

Vzhledem k vlastnostem použitých technologií je výše uvedený aplikační komunikační protokol podání-odpověď implementován použitými kanály více zprávami. Využití jednotlivých zpráv jednotlivými kanály je zachyceno v popisu konkrétních komunikačních kanálů dále v tomto dokumentu (tj. ne všechny zprávy jsou využívány všemi kanály).

Použití zpráv jednotlivými kanály

~~PVS~~	VREP	ISDS

Submission request	~~Ano~~	Ano	Ano
Submission acknowledgement	~~Ano~~	Ano	Ne
Submission poll	~~Ano~~	Ano	Ne
Submission response	~~Ano~~	Ano	Ano
Delete request	~~Ano~~	Ano	Ne
Delete acknowledgement	~~Ano~~	Ano	Ne
Delete response	~~Ano~~	Ano	Ne
Submission error	~~Ano~~	Ano	Ano

Zprávy zasílané podávajícím sw

Submission request (podání)

Submission request je zpráva, která nese data podání. Platnou odpovědí na zprávu submission request je zpráva submission acknowledgement (úspěšné provedení, hlavní cesta zpracován), submission response (v případě, že mezi podáním a přípravou odpovědi došlo ke zpracování podání, a výsledkem je přijetí podání) či submission error (v případě chyby podávacího protokolu či v případě, že mezi podáním a přípravou odpovědi došlo ke zpracování podání, a výsledkem je odmítnutí podání).

V hlavičce musí být vyplněny elementy Class, Qualifier a Function, CorrelationID musí být prázdné. V těle musí být CSSZ Message.

Variabilní symbol v GovTalkDetails/Key musí být plněn pro ty druhy podání, které se vážou k organizaci (v současné době všechny kromě HPN).

Je doporučeno plnit TimestampVersion hodnotou xmldsig pro vyžádání podepsané časové značky VREPu ~~či PVS~~ ve formátu xmldsig.

Pro VREP a ISDS je minimální nutná struktura následující:

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_PRIHL</Class>
<Qualifier>request</Qualifier>
<Function>submit</Function>
<CorrelationID></CorrelationID>
</MessageDetails>
</Header>
<GovTalkDetails>
<Keys>
<Key Type="vars">1111234567</Key>
</Keys>
<GatewayAdditions>
<Flags>
<TimestampVersion>xmldsig</TimestampVersion>
</Flags>
</GatewayAdditions>
</GovTalkDetails>
<Body>
<!-- telo podani -->
</Body>
```

```
</GovTalkMessage>
```

~~PVS vyžaduje autentizaci, proto je navíc nutné plnit strukturu SenderDetails (podrobněji k metodám autentizace viz dokumentace PVS).~~

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_PRIHL</Class>
<Qualifier>request</Qualifier>
<Function>submit</Function>
<CorrelationID></CorrelationID>
</MessageDetails>
<SenderDetails>
<IDAuthentication>
<SenderID>login</SenderID>
<Authentication>
<Method>MD5</Method>
<Value>heslo</Value>
</Authentication>
</IDAuthentication>
<EmailAddress>email@hotmail.com</EmailAddress>
</SenderDetails>
</Header>
<GovTalkDetails>
<Keys>
<Key Type="vars">1111234567</Key>
</Keys>
<GatewayAdditions>
<Flags>
<TimestampVersion>xmldsig</TimestampVersion>
</Flags>
</GatewayAdditions>
</GovTalkDetails>
<Body>
<!-- telo podani -->
</Body>
</GovTalkMessage>
```

Submission poll (dotaz na výsledek zpracování)

Submission poll je dotaz na výsledek zpracování. Platnou odpovědí na zprávu submission poll je zpráva submission acknowledgement, submission response (úspěšné provedení, hlavní cesta zpracování) či submission error.

V hlavičce musí být vyplněny elementy Class, Qualifier, Function a CorrelationID (získané ze zprávy submission acknowledgement). Tělo musí být prázdné.

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_RELDP</Class>
<Qualifier>poll</Qualifier>
<Function>submit</Function>
<CorrelationID>163CB7BFC921495CAAA0C28DDE89335B</CorrelationID>
</MessageDetails>
<SenderDetails>
```

```
<IDAuthentication>
<SenderID></SenderID>
<Authentication>
<Method>MD5</Method>
<Value></Value>
</Authentication>
</IDAuthentication>
</SenderDetails>
</Header>
<GovTalkDetails>
<Keys>
<Key Type="vars">1111234567</Key>
</Keys>
<GatewayAdditions>
<Flags>
<TimestampVersion>xmldsig</TimestampVersion>
</Flags>
</GatewayAdditions>
</GovTalkDetails>
<Body></Body>
</GovTalkMessage>
```

Delete request (požadavek na uzavření transakce)

Delete request je požadavek na uzavření transakce. Platnou odpovědí na zprávu delete request je zpráva delete response (úspěšné provedení, hlavní cesta zpracování), delete acknowledgement či submission error (v případě chyby podávacího protokolu).

V hlavičce musí být vyplněny elementy Class, Qualifier, Function a CorrelationID (získané ze zprávy submission acknowledgement). Tělo musí být prázdné.

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_RELDP</Class>
<Qualifier>request</Qualifier>
<Function>delete</Function>
<CorrelationID>163CB7BFC921495CAAA0C28DDE89335B</CorrelationID>
</MessageDetails>
<SenderDetails>
<IDAuthentication>
<SenderID></SenderID>
<Authentication>
<Method>MD5</Method>
<Value></Value>
</Authentication>
</IDAuthentication>
</SenderDetails>
</Header>
<GovTalkDetails>
<Keys>
<Key Type="vars">1111234567</Key>
</Keys>
<GatewayAdditions>
<Flags>
<TimestampVersion>xmldsig</TimestampVersion>
</Flags>
</GatewayAdditions>
</GovTalkDetails>
<Body></Body>
```

```
</GovTalkMessage>
```

Zprávy vracené DIS systémem ČSSZ

Submission acknowledgement (doručenka, podací lístek)

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_PRIHL</Class>
<Qualifier>acknowledgement</Qualifier>
<Function>submit</Function>
<TransactionID />
<CorrelationID>298D72D48D90404FA10C371749D99B6B</CorrelationID>
<ResponseEndPoint
PollInterval="35">https://bezpecne.dev.gov.cz/poll</ResponseEndPoint>
<GatewayTimestamp>2010-09-08T07:22:59.247</GatewayTimestamp>
</MessageDetails>
<SenderDetails />
</Header>
<GovTalkDetails>
<Keys />
</GovTalkDetails>
<Body>
<!-- casova znacka -->
</Body>
</GovTalkMessage>
```

Submission response (odpověď, přijetí podání)

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope"
xmlns:xsig="http://www.w3.org/2000/09/xmldsig#">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_RELDP</Class>
<Qualifier>response</Qualifier>
<Function>submit</Function>
<TransactionID />
<CorrelationID>163CB7BFC921495CAAA0C28DDE89335B</CorrelationID>
<ResponseEndPoint
PollInterval="30">https://bezpecne.dev.gov.cz/submission</ResponseEndPoint>
<Transformation>XML</Transformation>
<GatewayTimestamp>2010-10-01T15:50:34.000</GatewayTimestamp>
</MessageDetails>
</Header>
<Body xmlns="http://www.govtalk.gov.uk/CM/envelope" Id="0">
<Message xmlns="http://www.cssz.cz/XMLSchema/envelope" version="1.2"
eType="response">
<Header>
<Signature xmlns="http://www.cssz.cz/emp/timestamp" Version="1.0">
<DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
<TimeStamp>
<date>20101001</date>
<time>12:45:40</time>
</TimeStamp>
<SignatureValue>
<!-- -->
</SignatureValue>
</Signature>
</Header>
<Body>
<!-- -->
```

```
</Body>
</Message>
</Body>
</GovTalkMessage>
```

Submission error (chyba či odmítnutí podání)

Pokud jsou Class a CorrelationID vyplněny, jedná se o chybu zpracování podání. Pokud ne, jedná se o chybu komunikačního protokolu.

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope"
xmlns:xsig="http://www.w3.org/2000/09/xmldsig#">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_PRIHL</Class>
<Qualifier>error</Qualifier>
<Function>submit</Function>
<TransactionID />
<CorrelationID>298D72D48D90404FA10C371749D99B6B</CorrelationID>
<ResponseEndPoint
PollInterval="30">https://bezpecne.dev.gov.cz/submission</ResponseEndPoint>
<Transformation>XML</Transformation>
<GatewayTimestamp>2010-09-08T09:24:25.000</GatewayTimestamp>
</MessageDetails>
<SenderDetails>
<IDAuthentication>
<SenderID>*****</SenderID>
<Authentication>
<Method>clear</Method>
<Value>*****</Value>
</Authentication>
</IDAuthentication>
</SenderDetails>
</Header>
<GovTalkDetails>
<Keys>
<Key xmlns="http://www.govtalk.gov.uk/CM/envelope"
Type="SpokeName">CSSZ_1_ORG</Key>
</Keys>
<GovTalkErrors>
<Error Id="0">
<RaisedBy>CSSZDIS</RaisedBy>
<Number>305</Number>
<Type>business</Type>
<Text>Počet formulářů v podání musí být větší než 0. E-podání mohlo být zasláno
zastaralou verzí programu nebo v podání byla použita nesprávná či starší ČSSZ
obálka nebo se e-podání nepodařilo dešifrovat.</Text>
</Error>
</GovTalkErrors>
</GovTalkDetails>
<Body xmlns="http://www.govtalk.gov.uk/CM/envelope" Id="0">
<!-- -->
</Body>
</GovTalkMessage>
```

Delete acknowledgement

Delete response (potvrzení uzavření transakce)

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
```

```
<MessageDetails>
<Class>CSSZ_RELDP</Class>
<Qualifier>response</Qualifier>
<Function>delete</Function>
<TransactionID></TransactionID>
<CorrelationID>163CB7BFC921495CAAA0C28DDE89335B</CorrelationID>
<ResponseEndPoint
PollInterval="35">https://bezpecne.dev.gov.cz/submission</ResponseEndPoint>
<GatewayTimestamp>2010-10-01T14:08:45.450</GatewayTimestamp>
</MessageDetails>
<SenderDetails />
</Header>
<GovTalkDetails>
<Keys />
</GovTalkDetails>
<Body></Body>
</GovTalkMessage>
```

XSD schéma

CSSZ Message ~~(pro VREP)~~ [JB8]

CSSZ Message (též „obálka ČSSZ“) je XML struktura, která je pro e-podání ČSSZ určena ~~shodná pro všechny komunikační kanály (PVS, ISDS a VREP)~~ pro komunikační kanál VREP. Je použita v podání (submission request – VREP vyžaduje, ISDS umožňuje jak tento formát, tak „holé XML“ – viz dále) a v odpovědi (submission response či submission error – VREP i ISDS). Tato struktura umožňuje v jedné zprávě předávat několik různých informací (podání - data podání a data podpisu, informace o použitých algoritmech).

Příklad CSSZ Message je uveden níže.

```
~~Příklad CSSZ Message je uveden níže.~~<Message
xmlns="http://www.cssz.cz/XMLSchema/envelope" version="1.2" eType="PRIHL">
————<Header>
<Signature xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="bin.base64">
<!-- podpis -->
</Signature>
<Vendor productName="CSSZSubmissionDemo" version="1.0.0" />
</Header>
<Body xmlns:dt="urn:schemas-microsoft-com:datatypes" encrypted="yes"
contentEncoding="gzip" dt:dt="bin.base64">
<!-- data -->
</Body>
</Message>
```

CSSZ Message je vkládána do těla zprávy GovTalkMessage, příklad podání je uveden níže:

```
<?xml version="1.0" encoding="utf-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope">
<EnvelopeVersion>2.0</EnvelopeVersion>
<Header>
<MessageDetails>
<Class>CSSZ_PRIHL</Class>
<Qualifier>request</Qualifier>
<Function>submit</Function>
<CorrelationID></CorrelationID>
</MessageDetails>
</Header>
<GovTalkDetails>
<Keys>
```

```
<Key Type="vars">1111234567</Key>
</Keys>
<GatewayAdditions>
<Flags>
<TimestampVersion>xmldsig</TimestampVersion>
</Flags>
</GatewayAdditions>
</GovTalkDetails>
<Body>
<Message xmlns="http://www.cssz.cz/XMLSchema/envelope" version="1.2" eType="PRIHL">
<Header>
<Signature xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="bin.base64">
<!-- podpis -->
</Signature>
<Vendor productName="CSSZSubmissionDemo" version="1.0.0" />
</Header>
<Body xmlns:dt="urn:schemas-microsoft-com:datatypes" encrypted="yes"
contentEncoding="gzip" dt:dt="bin.base64">
<!-- data -->
</Body>
</Message>
</Body>
</GovTalkMessage>
```

Struktura

Prvek	Význam
http://www.cssz.cz/XMLSchema/envelope	Namespace, povinný
Message	kořenový element, povinný
./@version	Verze obálky, povinná. Pro nové implementace je doporučeno používat verzi 1.2
./@eType	Rozlišení podtypu či verze podání (např. při změně formuláře)—, rozlišení podání a odpovědi
Message/Header/Signature	Podpis dat podání či podepsaná časová značka ČSSZ v odpovědi
http://www.cssz.cz/emp/timestamp	Namespace podepsané časové značky ČSSZ
./@version	Verze podepsané časové značky ČSSZ
./DigestMethod/@Algorithm	Algoritmus použitý při výpočtu dat pro podpis (~~SHA-1 či~~ SHA-2)
./ TimeStamp/date	Datum podpisu
./ TimeStamp/time	Čas podpisu
./ SignatureValue	Data podpisu
Message/Header/Vendor	Informace o podávající aplikaci
./@productName	Název aplikace, doporučeno pro pomoc při řešení příp. problémů
./@version	Verze aplikace, doporučeno pro pomoc při řešení příp. problémů
Message/Body	Tělo zprávy – podání či data odpovědi
./@encrypted	Příznak šifrování dat v těle zprávy (yes či no, pro podání vyžadováno šifrování)
./@contentEncoding	Příznak komprimace dat v těle zprávy (gzip pro komprimovaná data, raw pro nekomprimovaná), pro podání vyžadována

	komprimace
./	Komprimovaná zašifrovaná data podání zakódovaná do Base64
./GatewayTimestamp	Podepsaná časová značka VREP ~~či PVS~~
~~./ProcessingResponse~~	~~Odpověď (verze 1.1 v případě, že data odpovědi lze šifrovat)~~
http://www.cssz.cz/XMLSchema/envelope	Namespace struktury odpovědi
~~./ProcessingResponse/Data/@encryptionAlgorithm~~	~~Šifrovací algoritmus odpovědi~~
~~./ProcessingResponse/Data/@compression~~	~~Příznak komprimace odpovědi~~
~~./ProcessingResponse/Data/@contentEncoding~~	~~Příznak kódování odpovědi~~
~~./ProcessingResponse/Data/.~~	~~Data odpovědi~~
./ProcessingResult	Odpověď (verze 1.0, nešifrovaná)
./ZpracovaniProtokol	Odpověď (verze 1.1 v případě, že data odpovědi nelze šifrovat)
./ProcessingResponse	Odpověď (verze 1.1 v případě, že data odpovědi lze šifrovat)
./ProcessingResponse/Data/@encryptionAlgorithm	Šifrovací algoritmus odpovědi
./ProcessingResponse/Data/@compression	Příznak komprimace odpovědi
./ProcessingResponse/Data/@contentEncoding	Příznak kódování odpovědi
./ProcessingResponse/Data/.	Data odpovědi
http://schemas.cssz.cz/epodani/protokol/1.0.0	Namespace ~~protokolu~~protokolu ZpracovaniProtokol

Zprávy

Podání

Zpráva CSSZ Message typu podání musí nést data podání a podpis.

eType

Zpráva typu podání musí mít správně nastaven podtyp podání platný pro daný druh podání.

version

Atribut version musí být nastaven na číslo verze obálky ČSSZ, která je použita. Pro všechna podání je doporučeno používat aktuální verzi, tj. 1.2.

Data

Data podání musí být XML struktura. Jsou definovaná pro každý typ podání (viz https://www.cssz.cz/web/cz/informace-pro-sw-vyvojare a https://www.cssz.cz/web/cz/datova-veta~~/.tam~~). Data podání musí být komprimována, zašifrována (v tomto pořadí, neboť šifrováním by vznikla binární data se špatným kompresním poměrem), výsledek musí být zakódován pro přenos v textové podobě.

Komprimace

Komprimace dat podání je založena na algoritmu LZ77 (GZip). V prostředí .NET je tento algoritmus implementován třídou GZipStream, příklad využití je zde:

```
protected byte[] Compress(ref byte[] content)
{
    using (MemoryStream ms = new MemoryStream())
    {
        using (GZipStream gz = new GZipStream(ms, CompressionMode.Compress))
        {
            gz.Write(content, 0, (int)content.Length);
```

```
            gz.Close(); //IMPORTANT: Close GZipStream
        }
        return ms.ToArray();
    }
}
```

Důležité: Ujistěte se, že zavřete komprimační stream předtím, než začnete číst komprimovaná data. Zavření komprimačního streamu zapíše potřebné hlavičky, bez nich nebudou komprimovaná data zpracovatelná a vaše podání budou odmítána.

Šifrování

Šifrování dat chrání data ve zprávě před zpřístupněním neoprávněným uživatelům. Příjemcem dat je ČSSZ, je tedy třeba zašifrovat data tak, aby data podání mohl číst systém pro zpracování e-podání na ČSSZ, a aby data nebyla přístupná neoprávněným uživatelům. Šifrování e-podání ČSSZ je založeno na standardu PKCS/CMS a využívá asymetrických klíčů X.509 ~~certifikátů.Šifrovací~~certifikátů. Šifrovací certifikát ČSSZ, tj. certifikát, pomocí kterého je třeba data zašifrovat, aby je bylo možné v systému pro zpracování e-podání rozšifrovat, je umístěn na webu ČSSZ (~~http://www.cssz.cz/cz/e-podani/sifrovani-datovych-zprav/~~https://www.cssz.cz/web/cz/ke-stazeni~~http://www.cssz.cz/cz/e-podani/pro-vyvojare/~~).

V prostředí .NET je pro šifrování dle uvedeného standardu k dispozici třída EnvelopedCms. Příklad kódu pro zašifrování dat je uveden níže.

```
protected byte[] Encrypt(ref byte[] content, ref X509Certificate2[] recipients)
{
    ContentInfo ci = new ContentInfo(content);
    EnvelopedCms cms = new EnvelopedCms(ci);
    CmsRecipientCollection rcps = new CmsRecipientCollection();
    foreach (var recipient in recipients)
    {
        rcps.Add(new CmsRecipient(recipient));
    }
    cms.Encrypt(rcps);
    return cms.Encode();
}
```

Příklad neřeší načtení certifikátu pro šifrování. To je možné provést více způsoby, např. nechat uživatele v konfiguraci nastavit cestu k souboru s certifikátem či vybrat certifikát z úložiště apod. či je možné zajistit stažení certifikátu z webu ČSSZ. Třída X509Certificate2 poskytuje bohaté možnosti načtení certifikátu, které pokrývají široké spektrum scénářů, zvolte ten, který je pro vaši aplikaci vhodný. Načtení certifikátu ze souboru lze provést např. níže uvedeným voláním:

```
new X509Certificate2(CertificateFilePath);
```

Pozn.: pro šifrování je třeba certifikát příjemce, tj. v tomto případě certifikát ČSSZ. Při zašifrování dat klíčem z certifikátu (veřejným klíčem) bude moci data rozšifrovat pouze vlastník privátního klíče, náležejícího k veřejnému klíči certifikátu, tedy systém ČSSZ. Pro zašifrování tedy není třeba privátní klíč.

Pozn.: PKCS/CMS umožňuje šifrování pro více příjemců. To můžete (ale nemusíte) využít ve svých aplikacích např. k tomu, abyste si uložili originály odesílaných podání v zašifrované podobě a mohli je ev. později rozšifrovat. K tomu je třeba, abyste měli vlastní certifikát (s privátním klíčem), který použijete k zašifrování (veřejný klíč) a ev. v budoucnu dle potřeby k rozšifrování (privátní klíč) dat. Vždy je nutné, aby byl právě jeden z příjemců definován certifikátem ČSSZ, jinak nebude podání možné zpracovat v systému ČSSZ; ostatní příjemce můžete definovat dle potřeb vaší aplikace. V případě více příjemců na pořadí nezáleží.

Kódování pro přenos

Výsledkem komprimace a šifrování jsou binární data, která je nutno pro přenos v textové podobě zakódovat. Pro kódování je nutné použít algoritmus Base64, v prostředí .NET lze využít metodu ToBase64String třídy Convert.

```
Convert.ToBase64String(content);
```

Podpis

ČSSZ provádí autentizaci na úrovni jednotlivých podání bez ohledu na použitý komunikační kanál (tj. pro všechny kanály, specificky pro jednotlivé druhy podání). Autentizace je založena na identifikaci na základě certifikátu, použitého k podpisu podání. Identifikace je prováděna vůči registrační databázi, stejně jako následná autorizace (např. ověření oprávnění podávat podání konkrétního typu za danou organizaci).

Podpis musí být typu detached signature, data podpisu musí být zakódována pro přenos v textové podobě. **Podepisují se původní data podání před komprimací, šifrováním a kódováním pro přenos, v podobě pole bytů**. Dbejte, abyste korektně načítali data pro podepsání stejným způsobem, jakým jsou načítána pro komprimaci a šifrování (kódování, načtení včetně BOM apod.). Lze jen doporučit, abyste data připravili v paměti či v souboru, načetli je a pro podepsání i komprimaci použili stejnou instanci objektu (stream či pole apod.).

Podepisování e-podání ČSSZ je založeno na standardu PKCS/CMS a využívá X.509 certifikáty. Pro podepsání musí být k dispozici certifikát s privátním klíčem, ČSSZ pro e-podání akceptuje pouze kvalifikované certifikáty dle ZoEP (zákon o elektronickém podpisu) vydané akreditovanými certifikačními autoritami.

```
protected byte[] Sign(ref byte[] content, ref X509Certificate2 signer)
{
    ContentInfo ci = new ContentInfo(content);
    SignedCms scms = new SignedCms(ci, true);
    CmsSigner sgnr = new CmsSigner(signer);
    scms.ComputeSignature(sgnr);
    return scms.Encode();
}
```

Příklad neřeší načtení certifikátu pro podepisování. To je možné provést více způsoby, např. nechat uživatele v konfiguraci nastavit cestu k souboru s certifikátem a privátním klíčem (a vyžadovat při použití heslo) či vybrat certifikát z úložiště, čipové karty apod. Třída X509Certificate2 poskytuje bohaté možnosti načtení certifikátu, které pokrývají široké spektrum scénářů, zvolte ten, který je pro vaši aplikaci vhodný. Načtení certifikátu ze souboru PFX s heslem lze provést např. níže uvedeným voláním:

```
new X509Certificate2(CertificatePFXFilePath, CertificatePFXFilePassword);
```

Odpověď

Zpráva CSSZ Message typu odpověď nese informace o výsledku zpracování příp. vyžádaná data (dle druhu podání).

Pro starší verze podání (PRIHL, RELDP) není v odpovědi (submission response) zpráva CSSZ Message, ale přímo odpovědní protokol. Vzhledem k tomu, že tato podání již mají aktualizované platné verze (ONZ, ELDP), není popis starších verzí odpovědí v tomto dokumentu dále rozpracován. Dokumentace formátu odpovědi je k dispozici na adrese ~~http://www.cssz.cz/cz/e-podani/druhy-e-podani/~~

https://www.cssz.cz/web/cz/datova-vetahttp://www.cssz.cz/cz/e-podani/pro-vyvojare/struktura-datove-vety/ .

eType
Atribut eType je pro odpověď nastaven na hodnotu „response".

version
Atribut version nese informaci o verzi struktury odpovědi.

Protokol či data odpovědi
V těle zprávy CSSZ Message je protokol, tj. struktura s informacemi o výsledku zpracování, či samotná data odpovědi. V případě speciálního podání DZDPN pak oboje (jak protokol tak data odpovědi). Existují následující verze odpovědi:

ProcessingResult
Struktura, která obsahuje informace o výsledku zpracování (přijetí či zamítnutí) a informace o chybách v jednotlivých formulářích.

Používá se pro většinu podání.

```
<ProcessingResult type="CSSZ_RELDP" version="1.0" result="OK" errMsg=""
errNumber="0" count="1" countErr="0" countWar="0">
<Error>
<RaisedBy />
<Number>0</Number>
<Type>CSSZ_RELDP</Type>
<Text />
</Error>
<Details>
<Item sqnr="" identifier="" subtype="ELDP09" period="" result="OK" errMsg=""
errNum="" />
<Item sqnr="1" identifier="1111111111" subtype="ELDP09" period="" result="OK"
errMsg="" errNum="" />
</Details>
</ProcessingResult>
```

ProcessingResponse
Struktura, která umožňuje přenášet zašifrované informace o výsledku zpracování (tzv. protokol o zpracování) či zašifrovaná data odpovědi. Aktuálně se pPoužívá se pro podání HPN pro šifrování protokolu o zpracování a pro podání DZDPN pro šifrovaní dat odpovědi. Vyžaduje obálku ČSSZ verze 1.2.

```
<ProcessingResponse xmlns="http://www.cssz.cz/XMLSchema/envelope">
<Data encryptionAlgorithm="3des192aes256"" compression=""gzip""
contentEncoding=""base64""></Data>
</ProcessingResponse>
```

Použití ProcessingRepsonse pro šifrování informace o výsledku zpracování (např. pro HPN)
Ty druhy podání, které používají šifrovaný výsledek zpracování (např. HPN), umožňují spolu s daty podání zaslat certifikát(y) (bez privátních klíčů, nemusí se jednat o kvalifikované certifikáty od akreditovaných certifikačních autorit), pro které má být provedeno šifrování výsledku zpracování. Systém pro zpracování e-podání na ČSSZ v takovém případě zašifruje výsledek zpracování pro všechny zaslané certifikáty (tj. bude možné rozšifrovat pomocí privátního klíče kteréhokoliv z certifikátů).

V případě, že v DV nejsou uvedeny žádné certifikáty pro šifrování, je výsledek zpracování zašifrován pro certifikát, který byl použit k podpisu podání. [PS9]Výsledek zpracování je též vždy šifrován pro certifikát systému pro zpracování e-podání ČSSZ, takže oprávnění pracovníci ČSSZ mohou z archivované odpovědi ověřit, jak přesně vypadala struktura zaslaná zpět podávajícímu.

V případě, že při zpracování podání dojde k chybě, která neumožňuje zašifrovat výsledek zpracování (např. podání není podepsáno a neobsahuje v datech šifrovací certifikáty apod.), použije se struktura ZpracovaniProtokol (viz. dále), tj. informace o výsledku zpracování jsou vloženy bez šifrování do těla zprávy CSSZ Message.

Použití ProcessingRepsonse pro šifrování dat odpovědi (např. pro DZDPN)

Ty druhy podání, které vracejí data ze systémů ČSSZ, obsahují v těle CSSZ Message jak element ProcessingResult s nešifrovaným protokolem, tak element ProcessingResponse s šifrovanými datami odpovědi. Datová věta podání obsahuje certifikát(y) (bez privátních klíčů, nemusí se jednat o kvalifikované certifikáty od akreditovaných certifikačních autorit), pro které má být provedeno šifrování dat odpovědi. Systém pro zpracování e-podání na ČSSZ v takovém případě zašifruje data odpovědi pro všechny zaslané certifikáty (tj. bude možné rozšifrovat pomocí privátního klíče kteréhokoliv z certifikátů). [PS10]Data odpovědi jsou též vždy šifrována pro certifikát systému pro zpracování e-podání ČSSZ, takže oprávnění pracovníci ČSSZ mohou z archivované odpovědi ověřit, jak přesně vypadala struktura zaslaná zpět podávajícímu.

~~Pokud je k dispozici certifikát, kterým je možné výsledek zpracování zašifrovat, bude výsledek zpracování zašifrován. Ty druhy podání, které používají šifrovaný výsledek zpracování (HPN), umožňují spolu s daty podání zaslat certifikáty (bez privátních klíčů, nemusí se jednat o kvalifikované certifikáty od akreditovaných certifikačních autorit), pro které má být provedeno šifrování výsledku zpracování. Systém pro zpracování e-podání na ČSSZ v takovém případě zašifruje výsledek zpracování pro všechny zaslané certifikáty (tj. bude možné rozšifrovat pomocí privátního klíče kteréhokoliv z certifikátů). V případě, že v datech podání nejsou žádné certifikáty pro šifrování, je výsledek zpracování zašifrován pro certifikát, který byl použit k podpisu podání. Výsledek zpracování je též vždy šifrovánpro certifikát systému pro zpracování e-podání ČSSZ, takže oprávnění pracovníci ČSSZ mohou z archivované odpovědi ověřit, jak přesně vypadala struktura zaslaná zpět podávajícímu.~~

~~V případě, že při zpracování podání dojde k chybě, která neumožňuje zašifrovat výsledek zpracování (např. podání není podepsáno a neobsahuje v datech šifrovací certifikáty apod.), použije se struktura ZpracovaniProtokol (viz. dále), tj. informace o výsledku zpracování jsou vloženy bez šifrování do těla zprávy CSSZ Message.~~

~~Pro podání, která v odpovědi nesou data (tj. ne pouze výsledek zpracování), je nemožnost odeslat odpověď zašifrovanou vyhodnocena jako chyba a v odpovědi je odeslán výsledek zpracování s touto chybou, nikoliv data odpovědi.~~

ZpracovaniProtokol

Nová verze struktury s informacemi o výsledku zpracování. Je přenášena zašifrovaná v ProcessingResponse či nezašifrovaná v těle CSSZ Message (pokud nelze zašifrovat). Používá se zatím jenom pro podání HPN.

```
<ns0:ZpracovaniProtokol typ="CSSZ_HPN" verze="1.0.0"
xmlns:ns0="http://schemas.cssz.cz/epodani/protokol/1.0.0">
<ns0:PodaniZpracovaniVysledek>
```

```
<ns0:FormulareCelkemPocet>1</ns0:FormulareCelkemPocet>
<ns0:FormulareOdmitnutiPocet>1</ns0:FormulareOdmitnutiPocet>
<ns0:FormulareUpozorneniPocet>0</ns0:FormulareUpozorneniPocet>
<ns0:Kod>ODMITNUTO</ns0:Kod>
<ns0:HlavniChyba>
<ns0:Cislo>300</ns0:Cislo>
<ns0:Text>text chyby</ns0:Text>
</ns0:HlavniChyba>
</ns0:PodaniZpracovaniVysledek>
<ZpracovaniVysledky xmlns="http://schemas.cssz.cz/epodani/protokol/1.0.0" />
</ns0:ZpracovaniProtokol>
```

Zpracování odpovědi

Data odpovědi jsou XML struktura. Může se jednat o obecnou informaci o výsledku zpracování podání (protokol), ~~či~~ nebo odpověď na požadavek (v takovém případě je struktura definována pro daný typ podání, viz~~.~~ tam), nebo oboje (tj. protokol i data odpovědi).

Data odpovědi mohou být vložena přímo v těle CSSZ Message, nebo mohou být komprimována, šifrována a zakódována pro přenos (v tomto pořadí). V případě, že se jedná o komprimovaná šifrovaná data, je postup zpracování uveden níže:

Dekódování přenesených dat

Data jsou pro přenos v textové podobě zakódována algoritmem Base64, pro dekódování je v prostředí .NET možné použít metodu FromBase64String třídy Convert.

```
Convert.FromBase64String(b64data);
```

Rozšifrování

Výsledkem dekódování je pole bytů, které obsahuje zašifrovaná data podání. Ta je možné rozšifrovat, pokud má aplikace, provádějící zpracování, přístup k privátnímu klíči alespoň jednoho z certifikátů, pro které byla data odpovědi či výsledku zpracování zašifrována. V prostředí .NET se rozšifrování provádí pomocí třídy EnvelopedCms.

```
protected byte[] Decrypt(ref byte[] data)
{
    EnvelopedCms cms = new EnvelopedCms();
    cms.Decode(data);
    try
    {
        cms.Decrypt();
        return cms.ContentInfo.Content;
    }
    catch (Exception ex)
    {
        //TODO
    }
}
```

Třída EnvelopedCms umožňuje automatické vyhledání certifikátu pro rozšifrování z úložišť systému Windows (hledá certifikát na základě informací z PKCS/CMS, ke kterému má uživatel privátní klíč). Pokud žádný certifikát odpovídající požadavkům nenalezne, nemůže rozšifrovat data a vyhodí výjimku. Třída podporuje i možnost předání certifikátu (resp. odkazu na otevřené úložiště certifikátů), ve kterém má (kromě výchozích úložišť) hledat certifikát pro rozšifrování.

```
cms.Decrypt(store);
```

Pozn.: některé výklady zákona o elektronickém podpisu (ZoEP) a souvisejících legislativních předpisů dovozují, že kvalifikovaný certifikát vydaný akreditovanou certifikační autoritou, resp. jeho tzv. párová data (tj. privátní a veřejný klíč), nesmí být použit k jinému účelu, než pro elektronický podpis. Striktně vzato by tedy neměl být takový certifikát použit k ~~roz~~šifrování dat. Pokud chcete, aby vaše aplikace takový scénář podporovala, je třeba při rozšifrovávání explicitně řídit, kterým certifikátem budou data rozšifrována. Struktura PKCS/CMS obsahuje informace o příjemcích (certifikátech), které lze pro tyto účely použít. V prostředí .NET jsou tyto informace dostupné přes kolekci RecipientInfos instance třídy EnvelopedCms.

```
EnvelopedCms cms = new EnvelopedCms();
cms.Decode(data);
foreach (RecipientInfo ri in cms.RecipientInfos)
{
       //…
}
```

Samotné rozšifrování po vyhledání požadovaného příjemce ze seznamu je pak možné provést následujícím voláním:

```
cms.Decrypt(ri);
//cms.Decrypt(ri, store);
```

Dekomprimace

Protokol či data odpovědi jsou před šifrováním komprimovány, pro zpracování je tedy po rozšifrování nutné dekomprimovat. Komprimace využívá algoritmus LZ77 (GZip), dekomprimace v prostředí .NETvyužívá třídu GZipStream.

```
protected byte[] Decompress(ref byte[] data)
{
    using (MemoryStream ms = new MemoryStream(data))
    {
        using (GZipStream gz = new GZipStream(ms, CompressionMode.Decompress))
        {
            using (MemoryStream dec = new MemoryStream())
            {
                int iBuffSize = 2048;
                byte[] buff = new byte[iBuffSize];
                int iSizeRead = gz.Read(buff, 0, iBuffSize);
                while (iSizeRead > 0)
                {
                    dec.Write(buff, 0, iSizeRead);
                    iSizeRead = gz.Read(buff, 0, iBuffSize);
                }
                if (dec.Length > 0)
                {
                    return dec.ToArray();
                }
                else
                {
                    return null;
                }
            }
        }
    }
}
```

Vyhodnocení chyb

Protokol o zpracování je XML struktura. Je definován pro jednotlivé typy podání.

Důležité: ~~Zejména u nových typů podání (HPN, NEM PRI) se z~~Zpracování podání na straně ČSSZ se snaží maximálně využít zaslaná data, tj. pro typy podání podporující tzv. částečné přijetí, pokud podání obsahuje několik platných formulářů a nějaký neplatný, není zamítnuto jako celek, ale platné formuláře jsou zpracovány. Neplatné formuláře není možné zpracovat, proto je velmi důležité správně zpracovat protokol a informovat podávajícího o nutnosti opravy chybných formulářů a jejich opakovaného zaslání.

Podepsaná časová značka odpovědi ČSSZ (CSSZ TimeStamp)

Odpověď (tj. přijetí podání submission response či odmítnutí podání submission error) v těle ve struktuře CSSZ Message (obálka ČSSZ) nese podepsanou časovou značku odpovědi ČSSZ. Jedná se o strukturu, která umožňuje validovat, že odpověď byla odeslána z ČSSZ a že nebyla změněna.

Podepsanou částí XML je CSSZ Message (tj. obsah těla GovTalkMessage), resp. hash těchto dat. Pro ověření podpisu je tedy třeba:

1. vyhledat data podpisu, dekódovat je (Base64), načíst a získat hash původních dat (obsah podpisu)
2. spočítat hash aktuálních dat, tj.
    a. načíst podepsaný obsah (CSSZ Message), odstranit hodnotu podpisu (SignatureValue) a převést do tzv. kanonické podoby (dle W3C doporučení pro Canonical XML)
    b. spočítat hash (SHA-1 či SHA-2)
3. porovnat hodnoty hash
4. ověřit identitu podepisujícího, tj. certifikát

Ukázka kódu pro validaci podepsané časové značky odpovědi ČSSZ v prostředí .NET používá XPath pro vyhledání podpisu a zjištění použitého algoritmu hash. Poté smaže v XML obsah podpisu, provede canonizaci (pomocí třídy XmlDsigC14Transform) a spočítá hash. Následně dekóduje podpis, provede ověření podpisu a načte původní hash a poté hash hodnoty porovná.

```
protected bool ValidateCSSZResponseSignature(string content)
{
    bool retval = true;
    XmlDocument xDoc = new XmlDocument();
    xDoc.LoadXml(content);
    XmlNode nod = xDoc.SelectSingleNode("//*[local-name()='Message' and namespace-
uri()='http://www.cssz.cz/XMLSchema/envelope']");

    string signatureDigestAlg = "";
    signatureDigestAlg = xDoc.SelectSingleNode("//*[local-
name()='Message']/*[local-name()='Header']/*[local-name()='Signature']/*[local-
name()='DigestMethod']").Attributes["Algorithm"].Value;

    XmlNode signatureNode = xDoc.SelectSingleNode("//*[local-name()='Message' and
namespace-uri()='http://www.cssz.cz/XMLSchema/envelope']/*[local-
name()='Header']/*[local-name()='Signature']/*[local-name()='SignatureValue']");
    string signatureValue = signatureNode.InnerText.Replace("\n", "").Replace("\r",
"").Replace("\t", "");
    signatureNode.InnerText = "";

    XmlDocument txDoc = new XmlDocument();
    txDoc.LoadXml(nod.OuterXml);
    XmlDsigC14NTransform t = new XmlDsigC14NTransform();
    t.LoadInput(txDoc);
    //Stream sCanonized = (Stream)t.GetOutput(typeof(Stream));

    Byte[] hash = null;
    if (signatureDigestAlg == "http://www.w3.org/2000/09/xmldsig#sha1")
```

```
    {
        hash = t.GetDigestedOutput(new SHA1Managed());
    }

    SignedCms signed = new SignedCms();
    signed.Decode(Convert.FromBase64String(signatureValue));
    try
    {
        signed.CheckSignature(true);
    }
    catch (Exception ex)
    {
        //TODO
        retval = false;
    }

    Byte[] data = signed.ContentInfo.Content;
    for (int i = 0; i < hash.Length; i++)
    {
        if (hash[i] != data[i]) retval = false;
    }
    return retval;
}
```

V současné době je v produkčním prostředí používán pro výpočet hash algoritmus ~~SHA-1, který bude v souladu doporučením Ministerstva vnitra ČR pravděpodobně do konce roku 2010 nahrazen algoritmem z rodiny~~ SHA-2.[x11][JB12]

Validaci certifikátu je možné provést zároveň s ověřením podpisu:

```
signed.CheckSignature(false);
```

Validaci certifikátu je možné provést pro všechny certifikáty postupně.

```
foreach (var si in signed.SignerInfos)
{
si.Certificate.Verify();
}
```

~~**XSD schéma**~~[x13][JB14]

„Holé" XML

„Holé" XML (též „nahaté" XML, ne úplně přesně též ~~občas nazývané~~ „rozšifrované XML"~~,~~ ~~trochu méně nepřesněči~~ „nezašifrované podání") je XML struktura samotného podání (data podání dle definice pro jednotlivé druhy podání) bez jakýchkoliv obálek (CSSZ Message, GovTalk).

Taková data (XML) nejsou komprimována ani šifrována (nejsou tedy mimo zabezpečený komunikační kanál a mimo zabezpečené informační systémy chráněna před zobrazení neoprávněnými osobami), není připojen elektronický podpis podávajícího (nejsou tedy mimo chráněný komunikační kanál a mimo zabezpečené informační systémy chráněna před neoprávněnou modifikací). Z tohoto důvodu dbejte v případě, že ve své aplikaci umožníte zasílání „holého" XML, zvýšené pozornosti v případě ukládání odesílaných podání.

„Holé" XML je možné zaslat pouze přes ISDS (přes ~~PVS a~~ VREP to možné není). Podání s „holým" XML je tedy datová zpráva ISDS, která obsahuje ~~právě jednu~~jednu nebo více příloh~~u~~ ve formátu XML, která odpovídá schématu pro data jednoho z druhů podání.

Takové podání nenese informace o identitě podávající osoby, pouze informaci o schránce, ze které bylo podáno. Pro zasílání podání přes ISDS ve formátu „holého" XML je pro většinu podání nutná registrace datové schránky v systému ČSSZ ~~(kromě podání OSVC_PRE).~~ [PS15]

Podávací a dotazovací protokol přes ISDS je shodný pro „holé" XML i pro podání ve formátu GovTalk, tj. podání je datová zpráva ISDS s příloh~~ou~~ami XML zaslaná voláním CreateMessage do ~~vyhrazené~~ datové schránky ČSSZ. ~~pro e-podání.~~ Odpověď je datová zpráva ISDS, zaslaná do schránky podávajícího. Datová zpráva odpovědi nese ~~jednu~~ přílohy~~u~~ ve formátu XML se zprávou GovTalk submission response či submission error (jinými slovy: **odpověď na podání ve formátu „holého" XML je zpráva ve formátu GovTalk, stejně jako na podání ve formátu GovTalk**), která v protokolu nese informace o výsledku vstupního zpracování podání. Pokud daný typ podání umožňuje zasílání certifikátů pro šifrování odpovědi (HPN, DZDPN) a v podání ve formátu „holého" XML byl zaslán alespoň jeden certifikát pro zašifrování odpovědi, bude výsledek zpracování v CSSZ Message v těle GovTalk zprávy odpovědi šifrován.

Prostředí

Pozn: pro šifrování podání je pro produkční i ~~komunitní~~ testovací prostředí použit stejný šifrovací certifikát.

Produkční prostředí

Aktuální informace naleznete na https://www.cssz.cz/web/cz/komunikacni-kanaly-e-podani.

~~PVS~~

~~PVS má v produkčním prostředí dvě rozhraní: POX a WS.~~

~~POX~~

~~Rozhraní POX má adresu https://bezpecne.podani.gov.cz/submission pro podání a https://bezpecne.podani.gov.cz/poll pro dotaz na stav zpracování.~~

~~WS~~

~~Rozhraní WS má adresy (koncové body, tj. endpointy) dle použité autentizace: adresu pro anonymní komunikaci (pro zprávy, které nevyžadují autentizaci), adresu pro autentizaci uživatelským jménem a heslem a adresu pro autentizaci certifikátem).~~

~~https://bezpecne.podani.gov.cz/ws/submission/public.svc/anonymous~~

~~https://bezpecne.podani.gov.cz/ws/submission/public.svc/username~~

~~https://bezpecne.podani.gov.cz/ws/submission/public.svc/certificate~~

VREP

VREP má v produkční větvi dvě rozhraní: POX a WS.~~ Obě rozhraní jsou publikována ve dvou lokalitách pro základní zajištění dostupnosti při ev. odstávce jedné z lokalit, tj. pro POX i WS existuje primární adresa a záložní adresa (backup transport).~~

POX

Rozhraní POX má dvě adresy (pro podání a pro dotaz na stav zpracování)~~ v každé lokalitě:.~~

https://epodani~~vrep1~~.cssz.cz/VREP/submission

~~https://vrep2.cssz.cz/VREP/submission~~

https://epodani~~vrep1~~.cssz.cz/VREP/poll

~~https://vrep2.cssz.cz/VREP/poll~~

WS

Rozhraní WS má jednu adresu~~ v každé lokalitě:.~~

https://~~vrep1~~epodani.cssz.cz/VREP/ws/public.svc

~~https://vrep2.cssz.cz/VREP/ws/public.svc~~

ISDS

Produkční prostředí ISDS je dostupné na adrese https://www.mojedatovaschranka.cz. Adresy webových služeb pro jednotlivé typy autentizace v kombinaci se stavovou či bezestavovou komunikací jsou v dokumentaci ISDS.

Podávající mají možnost zaslat e-Podání prostřednictvím ISDS do specializované datové schránky e-Podání ČSSZ:

- ID schránky: **5ffu6xk**
- jméno schránky: e-podani ČSSZ

a/nebo do datových schránek místně příslušné OSSZ/PSSZ/MSSZ.

~~Datová schránka pro příjem **produkčních e-podání** pro ČSSZ:~~

~~ID schránky: **5ffu6xk**~~

~~jméno schránky: e-podání ČSSZ[x16]~~[JB17]

~~Pro dodavatele sw aplikací~~Testovací prostředí

Pro dodavatele aplikací zajišťuje ČSSZ provoz ~~tzv.~~ testovacího ~~větve neboli komunitní~~ prostředí pro testování e-podání. ~~Komunitní~~ Toto prostředí vyžaduje samostatnou registraci, a je napojeno na odpovídající testovací větve ~~PVS~~ VREP a ISDS. Více informací naleznete na https://www.cssz.cz/web/cz/testovani-a-fiktivni-udaje.

~~**PVS**~~

~~PVS má v komunitní větvi stejně jako v produkční dvě rozhraní: POX a WS.~~

~~***POX***~~

~~Rozhraní POX má adresu https://bezpecne.dev.gov.cz/submission pro podání a https://bezpecne.dev.gov.cz/poll pro dotaz na stav zpracování.~~

~~***WS***~~

~~Rozhraní WS má adresy (koncové body, tj. endpointy) dle použité autentizace: adresu pro anonymní komunikaci (pro zprávy, které nevyžadují autentizaci), adresu pro autentizaci uživatelským jménem a heslem a adresu pro autentizaci certifikátem).~~

~~https://bezpecne.dev.gov.cz/ws/submission/public.svc/anonymous~~

~~https://bezpecne.dev.gov.cz/ws/submission/public.svc/username~~

~~https://bezpecne.dev.gov.cz/ws/submission/public.svc/certificate~~

VREP

VREP má v ~~komunitní~~ testovacím prostředí ~~větvi stejně~~ stejně jako v produkčním dvě rozhraní: POX a WS. ~~Obě rozhraní jsou publikována ve dvou lokalitách pro základní zajištění dostupnosti při ev. odstávce jedné z lokalit, tj. pro POX i WS existuje primární adresa a záložní adresa (backup transport).~~

POX

Rozhraní POX má dvě adresy (pro podání a pro dotaz na stav zpracování) ~~v každé lokalitě:~~.

https://~~vrep1-~~t-epodani.cssz.cz/VREP/submission

~~https://vrep2-t.cssz.cz/VREP/submission~~

https://~~vrep1-~~t-epodani.cssz.cz/VREP/poll

~~https://vrep2-t.cssz.cz/VREP/poll~~

WS

Rozhraní WS má jednu adresu~~ v každé lokalitě~~:~~.~~

https://~~vrep1-~~t-epodani.cssz.cz/VREP/ws/public.svc

~~https://vrep2-t.cssz.cz/VREP/ws/public.svc~~

ISDS

~~Komunitní~~ Testovací prostředí ČSSZ pro e-podání je napojeno na testovací prostředí ISDS (https://www.czebox.cz~~ , http://www.datoveschranky.info/clanek/99/~~). Adresy pro jednotlivé typy autentizace v kombinaci se stavovou či bezestavovou komunikací jsou v dokumentaci ISDS.

Datová schránka pro příjem **testovacích e-podání** pro ČSSZ:

ID schránky: **9tsaf6s**

jméno schránky: e-podání TEST

Přílohy

Seznam relevantních standardů

HTTP
http://www.ietf.org/rfc/rfc2616.txt

SSL/TLS
http://www.rfc-editor.org/rfc/rfc2818.txt

XML
http://www.w3.org/TR/xml11/

XSD
http://www.w3.org/TR/xmlschema-0/

XML Namespaces
http://www.w3.org/TR/REC-xml-names/

XMLSignature
http://www.rfc-editor.org/rfc/rfc3075.txt

Canonical XML
www.w3.org/TR/xml-c14n

http://www.rfc-editor.org/rfc/rfc3076.txt

Base64
http://www.ietf.org/rfc/rfc3548.txt

X.509
http://www.rfc-editor.org/rfc/rfc5280.txt

PKCS#7/CMS
http://www.rfc-editor.org/rfc/rfc2315.txt

GZip
http://www.rfc-editor.org/rfc/rfc1952.txt

UTF-8
http://www.rfc-editor.org/rfc/rfc3629.txt

GovTalk

e-Government Interoperability Framework (e-GIF)

SOAP
http://www.w3.org/TR/soap12-part1/

WS-Addressing
http://www.w3.org/Submission/ws-addressing/

WS-Security
http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf