



Referenční klient RVO

Verze 1.0.0

1. 4. 2020



Obsah

1	Úvod.....	3
1.1	Účel dokumentu	3
1.2	Slovník pojmů.....	3
2	Obecné informace	3
2.1	Distribuce a licence	4
2.2	Mapování JSON a XML.....	6
2.3	Poskytované operace	7
3	Referenční klient RVO Java	7
3.1	Spouštění.....	8
3.2	Konfigurace	9
3.3	Zdrojový kód	11
3.4	Build.....	11
4	Referenční klient .NET.....	12
4.1	Spouštění.....	12
4.2	Konfigurace	13
4.3	Zdrojový kód	14
4.4	Build.....	15



1 Úvod

1.1 Účel dokumentu

Tento dokument navazuje na „Kompletní technickou specifikaci informačního systému provozování hazardních her“ týkající se rejstříku fyzických osob vyloučených z účasti na hazardních hrách podle ustanovení § 16 zákona č. 186/2016 Sb., o hazardních hrách, ve znění pozdějších předpisů (dále jen „zákon o hazardních hrách“) a ověření totožnosti a věku osoby podle § 46 odst. 1 a § 77 odst. 6 zákona o hazardních hrách, vyhlášenou Ministerstvem financí ČR.

Dokument popisuje způsoby práce s Referenčním klientem RVO, který je publikován Ministerstvem financí ČR spolu se spuštěním neprodukčního prostředí informačního systému provozování hazardních her týkající se rejstříku fyzických osob vyloučených z účasti na hazardních hrách a ověření totožnosti a věku osoby (dále také jako „Playground RVO“ nebo „Neprodukční prostředí RVO“).

Dokument zároveň navazuje na dokument „Koncepce testovacích dat pro neprodukční prostředí RVO (Playground RVO)“ a jeho přílohy, které popisují testovací data a certifikáty využívané v distribuci Referenčního klienta RVO.

Tento dokument je součástí distribuce Referenčního klienta RVO.

1.2 Slovník pojmů

Pojem / Zkratka	Popis
Datové rozhraní	Datové rozhraní AISG pro výměnu informací a dat s informačními systémy provozovatelů hazardních her zajišťující zejména dálkovým způsobem ověření totožnosti a věku osob žádajících o registraci a ověření, zda konkrétní osoba je nebo není zapsána v rejstříku fyzických osob vyloučených z účasti na hazardních hrách
„MF“ nebo „ministerstvo“	Ministerstvo financí České republiky
ROB	Registr obyvatel
RVO	Rejstřík fyzických osob vyloučených z účasti na hazardních hrách
Referenční klient RVO	Konzolová aplikace, která umožňuje jednotlivě volat všechny služby datového rozhraní
„Playground RVO“ nebo „neprodukční prostředí RVO“	Neprodukční prostředí informačního systému provozování hazardních her týkající se rejstříku fyzických osob vyloučených z účasti na hazardních hrách a ověření totožnosti a věku osoby pro testování a podporu vývoje informačních systémů zejména provozovatelů hazardních her. Umožňuje testování služeb datového rozhraní pro ověření totožnosti a věku osob a ověření, zda konkrétní osoba je nebo není zapsána v RVO

2 Obecné informace

Referenční klient RVO slouží jako vzorová implementace klienta webových služeb datového rozhraní. Na základě této implementace lze vyrobit vlastní klienty nebo použít fragmenty kódu Referenčního klienta RVO pro integraci do informačních systémů zejména provozovatelů hazardních her.



Referenční klient RVO je publikován jako Java a .NET implementaci. V této kapitole jsou uvedeny informace společné pro Java i .NET implementace.

Referenční klient RVO je předkonfigurován pro prostředí Playground RVO a jeho distribuce obsahuje i vybrané testovací případy Playgroundu RVO ve formátu JSON. Po úpravě konfigurace je Referenčního klienta RVO možno použít i pro volání služeb produkčního prostředí.

2.1 Distribuce a licence

Referenční klient RVO je k dispozici ke stažení formou GitHub repository na adresách <https://github.com/Ministerstvo-financi/hazard-rovo-java> a <https://github.com/Ministerstvo-financi/hazard-rovo-dot-net>, kde jsou uloženy jak zdrojové kódy, tak i spustitelné verze (releases).

Referenční klient RVO je distribuován na základě následující open source licence:

Copyright [2020] [Ministerstvo financí České republiky]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Součástí distribuce Referenčního klienta RVO jsou následující komponenty s open source licencemi, které Referenční klient RVO pouze používá, nemění a jsou redistribuovány pod jejich původními licencemi.

Referenční klient RVO Java

GroupId:ArtifactId:Version	Licence
ch.qos.logback:logback-classic:1.2.3	Eclipse Public License - v 1.0 GNU Lesser General Public License
ch.qos.logback:logback-core:1.2.3	Eclipse Public License - v 1.0 GNU Lesser General Public License
com.fasterxml.jackson.core:jackson-annotations:2.9.0	The Apache Software License, Version 2.0
com.fasterxml.jackson.core:jackson-core:2.9.8	The Apache Software License, Version 2.0
com.fasterxml.jackson.core:jackson-databind:2.9.8	The Apache Software License, Version 2.0
com.fasterxml.jackson.datatype:jackson-datatype-jdk8:2.9.8	The Apache Software License, Version 2.0
com.fasterxml.jackson.datatype:jackson-datatype-jsr310:2.9.7	The Apache Software License, Version 2.0
com.fasterxml.jackson.module:jackson-module-parameter-names:2.9.8	The Apache Software License, Version 2.0
com.fasterxml.woodstox:woodstox-core:5.0.3	The Apache License, Version 2.0
com.google.guava:guava:19.0	The Apache Software License, Version 2.0
com.jayway.jsonpath:json-path:2.4.0	The Apache Software License, Version 2.0
com.vaadin.external.google:android-json:0.0.20131108.vaadin1	Apache License 2.0
commons-codec:commons-codec:1.11	Apache License, Version 2.0
io.dropwizard.metrics:metrics-core:4.0.5	Apache License 2.0
javax.annotation:javax.annotation-api:1.3.2	CDDL + GPLv2 with classpath exception



javax.validation:validation-api:2.0.1.Final	Apache License 2.0
javax.xml.stream:stax-api:1.0-2	COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0 GNU General Public Library
joda-time:joda-time:2.10.1	Apache 2
junit:junit:4.12	Eclipse Public License 1.0
net.bytebuddy:byte-buddy:1.9.10	The Apache Software License, Version 2.0
net.bytebuddy:byte-buddy-agent:1.9.10	The Apache Software License, Version 2.0
net.minidev:accessors-smart:1.2	The Apache Software License, Version 2.0
net.minidev:json-smart:2.3	The Apache Software License, Version 2.0
net.shibboleth.utilities:java-support:7.3.0	The Apache Software License, Version 2.0
org.apache.commons:commons-lang3:3.8.1	Apache License, Version 2.0
org.apache.cxf:cxf-core:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-bindings-soap:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-bindings-xml:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-databinding-jaxb:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-features-logging:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-frontend-jaxws:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-frontend-simple:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-security:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-security-saml:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-transports-http:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-ws-addr:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-ws-policy:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-ws-security:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-rt-wsdl:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-spring-boot-autoconfigure:3.2.1	The Apache Software License, Version 2.0
org.apache.cxf:cxf-spring-boot-starter-jaxws:3.2.1	The Apache Software License, Version 2.0
org.apache.geronimo.javamail:geronimo-javamail_1.4_mail:1.8.4	The Apache Software License, Version 2.0
org.apache.logging.log4j:log4j-api:2.11.2	Apache License, Version 2.0
org.apache.logging.log4j:log4j-to-slf4j:2.11.2	Apache License, Version 2.0
org.apache.neethi:neethi:3.1.0	The Apache Software License, Version 2.0
org.apache.santuario:xmlsec:2.1.0	The Apache Software License, Version 2.0
org.apache.ws.xmlschema:xmlschema-core:2.2.2	The Apache Software License, Version 2.0
org.apache.wss4j:wss4j-bindings:2.2.0	The Apache Software License, Version 2.0
org.apache.wss4j:wss4j-policy:2.2.0	The Apache Software License, Version 2.0
org.apache.wss4j:wss4j-ws-security-common:2.2.0	The Apache Software License, Version 2.0
org.apache.wss4j:wss4j-ws-security-dom:2.2.0	The Apache Software License, Version 2.0
org.apache.wss4j:wss4j-ws-security-policy-stax:2.2.0	The Apache Software License, Version 2.0
org.apache.wss4j:wss4j-ws-security-stax:2.2.0	The Apache Software License, Version 2.0
org.assertj:assertj-core:3.11.1	Apache License, Version 2.0
org.bouncycastle:bcprov-jdk15on:1.54	Bouncy Castle Licence
org.codehaus.woodstox:stax2-api:3.1.4	The BSD License
org.codehaus.woodstox:woodstox-core-asl:4.4.1	The Apache Software License, Version 2.0
org.cryptacular:cryptacular:1.1.1	Apache 2 GNU Lesser General Public License
org.hamcrest:hamcrest-core:1.3	New BSD License
org.hamcrest:hamcrest-library:1.3	New BSD License
org.jasypt:jasypt:1.9.2	The Apache Software License, Version 2.0
org.mockito:mockito-core:2.23.4	The MIT License
org.objenesis:objenesis:2.6	Apache 2
org.opensaml:opensaml-core:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-profile-api:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-saml-api:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-saml-impl:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-security-api:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-security-impl:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-soap-api:3.3.0	The Apache Software License, Version 2.0



org.opensaml:opensaml-xacml-api:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-xacml-impl:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-xacml-saml-api:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-xacml-saml-impl:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-xmlsec-api:3.3.0	The Apache Software License, Version 2.0
org.opensaml:opensaml-xmlsec-impl:3.3.0	The Apache Software License, Version 2.0
org.ow2.asm:asm:5.2	BSD
org.skyscreamer:jsonassert:1.5.0	The Apache Software License, Version 2.0
org.slf4j:jul-to-slf4j:1.7.25	MIT License
org.slf4j:slf4j-api:1.7.25	MIT License
org.springframework:spring-aop:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework:spring-beans:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework:spring-context:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework:spring-core:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework:spring-expression:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework:spring-jcl:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework:spring-test:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework:spring-web:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework:spring-webmvc:5.1.5.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot:2.1.3.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot-autoconfigure:2.1.3.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot-starter:2.1.3.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot-starter-json:2.1.3.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot-starter-logging:2.1.3.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot-starter-test:2.1.3.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot-starter-web:2.1.3.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot-test:2.1.3.RELEASE	Apache License, Version 2.0
org.springframework.boot:spring-boot-test-autoconfigure:2.1.3.RELEASE	Apache License, Version 2.0
org.xmlunit:xmlunit-core:2.6.2	The Apache Software License, Version 2.0
org.yaml:snakeyaml:1.23	Apache License, Version 2.0
wsdl4j:wsdl4j:1.6.3	CPL
xml-resolver:xml-resolver:1.2	The Apache Software License, Version 2.0

Referenční klient RVO .NET

Komponenta	Licence
NDesk.Options	MIT/X11 license.
Newtonsoft.Json	MIT License (MIT)
.NET Framework 4.8	Bez licence

2.2 Mapování JSON a XML

Referenční klient RVO používá pro zadávání testovacích dat formát JSON. Tento formát je dobře čitelný a jednoduše upravitelný, je vhodný pro přípravu testovacích dat. Formát JSON vychází ze stejných konvencí, jako jazyk JavaScript (Java, C#), proto jsou elementy pojmenované shodně. Příklad fragmentu JSON:

```
{  
  "cisloPozadavku": "3fba99be-cbfe-48eb-b632-c2a8bd9e3661",  
  "icovcp": "123456"  
}
```

Webová služba je postavená na technologii SOAP a formátech XML. Formát XML je určený pro přenosy strukturovaných dat mezi systémy. Popis struktury dat obsahuje soubor XSD. Data jsou přenášena ve formě **elementů** a volitelně jejich **atributů**. Formát XML je při



pojmenování elementů a jejich atributů poměrně benevolentní. Sice doporučuje použití UpperCamelCase pro elementy a lowerCamelCase pro názvy atributů, připouští ovšem i zkratky a oddělovače jako podtržítka (_) nebo pomlčku (-).

Zpracování XML probíhá v prostředí Java nebo C#. Obě prostředí podle svých konvencí pracují výhradně s názvy s použitím lowerCamelCase. Přitom automaticky přejmenovávají elementy na své proměnné tak, aby splnily konvenci:

- oddělovače odstraňují a další písmeno za nimi mění na velké,
- pokud je název elementu celý velkými písmeny, změni je na malá,
- pokud je v názvu skupina velkých písmen, ale před nimi jsou malá, velká písmena ponechají.

Takto například bude přejmenován:

XML Element	Java proměnná
ICO_VCP	icovcp
CisloPozadavku	cisloPozadavku
HID	hid
NalezenaROB	nalezenaROB

Nejmarkantnější rozdíl je v ICO_VCP, viz výše. Jde o jediný případ s oddělovačem. Ostatní elementy se liší pouze ve velikosti písmen podle pravidel výše.

2.3 Poskytované operace

Referenční klient RVO představuje funkční příklad klienta webové služby používané zejména provozovateli hazardních her. Jaká operace webové služby datového rozhraní bude volána, určuje parametr klienta „operace“.

Operace	Popis
OveritOsobu	Ověření totožnosti a věku osob nebo ověření, zda konkrétní osoba je nebo není zapsána v RVO
OveritOsobyHromadne	Hromadné ověření plnoletosti osob, jejich nalezení v ROB či RVO a zpravidla k získání jejich HID
ZiskatVysledkyOveritOsobyHromadne	Získání výsledku požadavku zaslání prostřednictvím operace OveritOsobyHromadne
ZmenitUdajeOsoby	Zaslání informací o změně některého z údajů osoby
Test	Ověření spojení s datovým rozhraním a kontrole nastavení zabezpečení služeb

3 Referenční klient RVO Java

Referenční klient RVO Java je aplikace v jazyce Java verze 8, která kompletně implementuje klientské rozhraní webové služby zejména pro provozovatele hazardních her a umožňuje volat všechny její operace. Referenční klient RVO je určený jako vzor pro implementaci klientů webové služby používané zejména provozovateli hazardních her, nicméně je plně funkční a použitelný.



Aplikace funguje tak, že naváže spojení s HTTPS serverem, podepíše a odešle zprávu a zkontroluje podpis přijaté zprávy. Číslo požadavku se generuje pro každý požadavek zvlášť.

Pozn: Pro zjednodušení identifikace provozovatele (nebo jeho doodavatele) na prostředí Playground RVO je doporučeno nastavit v konfiguraci ICO_VCP, který umožní jeho identifikaci a cílenou podporu ze strany MF.

3.1 Spouštění

Referenční klient RVO se spouští z příkazové řádky např. následovně:

```
java -jar HazardRefClient.jar --operace=Test
```

Dodatečné parametry pro spouštění a logování:

operace	Test, OveritOsobu, OveritOsobyHromadne, ZiskatVysledkyOveritOsobyHromadne, ZmenitUdajeOsoby – jaká operace webové služby má být volána
input	cesta k souboru, ve kterém je JSON reprezentace požadavku, tj. vstupní parametry operace - defaurávyt: \$operace.json
outputXMLRequest	cesta k souboru pro uložení XML požadavku
outputXMLResponse	cesta k souboru pro uložení XML odpovědi
outputJSONRequest	cesta k souboru pro uložení dat požadavku
outputJSONResponse	cesta k souboru pro uložení dat odpovědi

Logování XML

Na standardní výstup se loguje kompletní request a response včetně hlaviček.

XML požadavek a XML odpověď jsou v následující struktuře:

```
2020-03-07T20:36:05.864Z - PrintWriterEventSender - RESP_IN
Content-Type: text/xml; charset=UTF-8
ResponseCode: 200
ExchangeId: 6a45cdc6-b530-4c71-aa1c-634444c90271
ServiceName: AisgPortService
PortName: AisgPortPort
PortTypeName: AisgPort
Headers: {Strict-Transport-Security=max-age=604800;...}
Payload: <soap:Envelope...>/soap:Envelope>
```

RESP_IN – příznak, jestli jde o požadavek (REQ_OUT) nebo odpověď (RESP_IN). V případě chyby může být ještě FAULT_IN

Content-type – MIME typ zprávy

ResponseCode – HTTP kód odpovědi

ExchangeId – identifikátor, který spojuje požadavek s odpovědí

ServiceName – jméno klienta webové služby

PortName – vygenerovaný název portu klienta webové služby

PortTypeName – název vygenerované obslužné třídy klienta



Headers – HTTP hlavičky požadavku/odpovědi

Payload – XML obsah požadavku/odpovědi

Logy jsou v kódování UTF-8.

Logování JSON

Odesílaná a přijímaná data jsou dále logována ve formátu JSON. Je to z důvodu zjednodušení testování klienta při implementaci vlastní verze.

Oproti XML logům mají JSON logy několik rozdílů:

- Atribut Duvod – důvod může být v JSON logu uvedený velkými písmeny. Je to dáno pouze serializací logu, data jsou webovou službou odesílána správně, tedy s prvním písmenem velkým a ostatními malými
- Data – datumové položky mohou být uvedeny včetně časového údaje, přestože např. u data narození nemá význam. To je dáno tím, že JSON formát nezná typ DATE, ani DATETIME a všechny časové údaje zobrazuje ve formátu včetně času. Webové službě jsou data opět předávána správně, bez časového rozvoje.

3.2 Konfigurace

Defaultní konfigurace webové služby je uložena přímo v jar souboru vzorového klienta. Výchozí hodnoty jsou nastaveny tak, aby posílaly zprávy na Playground RVO s ICP_VCP = 123456 a s použitím certifikátu fiktivní organizace Hazard bez rizika, o.p.s.

Konfiguraci lze změnit použitím vlastního souboru `application.yml`, umístěného v classpath Referenčního klienta RVO, nebo nastavením parametru `spring.config.location`. Příklad:

```
java -jar -Dspring.config.location=my_applcfg.yml HazardRefClient.jar --  
operace=Test
```

Obsah konfiguračního souboru je následující:

```
client:  
  address: https://api-pg.hazard.mfcr.cz/rovo/v1  
  sign:  
    keystoreLocation: cert/hbr-pfx.pfx  
    keystorePassword: unsafe  
    keystoreType: PKCS12  
    keyAlias: 1  
    keyPassword: unsafe  
  trust:  
    keystoreLocation: cert/truststore.jks  
    keystorePassword: changeit  
    keystoreType: JKS  
  icoVCP: 123456
```

Význam jednotlivých parametrů:

address

URL adresa webové služby. Výchozí hodnota je <https://api-pg.hazard.mfcr.cz/rovo/v1>.

sign

Skupina parametrů pro konfiguraci elektronického podpisu odchozích zpráv.



keystoreLocation

Cesta k uloženému certifikátu s privátním klíčem, který má být použitý pro podepsání odchozí zprávy. Cesta může být absolutní nebo relativní k aktuálnímu adresáři. Pokud klient nenajde soubor s klíčem na disku počítače, pokusí se certifikát vyhledat interně v jar souboru. Na cestě cert/<zkratka>-pfx.pfx jsou uloženy vzorové certifikáty.

keystorePassword

Heslo k souboru s uloženým certifikátem a privátním klíčem. Pro všechny testovací certifikáty je použito heslo „unsafe“.

keystoreType

Typ souboru, který obsahuje certifikáty. Jde o standardní úložiště klíčů, se kterými umí pracovat Java, typicky JKS a PKCS12.

keyAlias

Alias klíče v úložišti. Pro PKCS12 je to typicky hodnota „1“.

keyPassword

Heslo k privátnímu klíči v úložišti. Pro všechny testovací certifikáty je použito heslo „unsafe“.

trust

Skupina parametrů pro konfiguraci SSL komunikace klienta.

keystoreLocation

Cesta k uloženému certifikátu serveru pro ověření SSL komunikace. Cesta může být absolutní nebo relativní k aktuálnímu adresáři. Pokud klient nenajde soubor na disku počítače, pokusí se certifikát vyhledat interně v jar souboru. Na cestě cert/truststore.jks je uložen certifikát serveru Playgroundu RVO.

keystorePassword

Heslo k souboru s uloženým certifikátem serveru. Pro cert/truststore.jks je použito heslo „changeit“.

keystoreType

Typ souboru, který obsahuje certifikát serveru. Jde o standardní úložiště klíčů, se kterými umí pracovat Java, typicky JKS a PKCS12.

icovcp

Hodnota ICO_VCP provozovatele, která se bude automaticky přidávat do zpráv.

Hodnotu konfiguračního parametru icovcp je třeba zadávat jako řetězec (v uvozovkách), jinak ji Referenční klient přečte podle specifikace jako číslo. Počáteční znak 0 v takovém případě neinterpretuje jako znak 0, ale jako označení oktalového čísla.

Práce s certifikáty

Certifikát pro podpisy zpráv klientem je dán nastavením parametru **sign/keystoreLocation**. Typicky jde pfx soubor s certifikátem včetně privátního klíče.

Certifikáty pro ověřování zpráv jsou uloženy v úložišti klíčů daném parametrem **trust/keystoreLocation**. Úložiště je standardní java keystore, doporučujeme formát JKS. Úložiště musí obsahovat certifikát serveru pro zabezpečení SSL komunikace nebo certifikát autority, která tento certifikát vydala, a certifikát, na základě kterého je podepisována odpověď webové služby serverem (nebo certifikát autority, která tento certifikát vydala).



Úložiště distribuované s Referenčním klientem RVO obsahuje požadované certifikáty pro prostředí Playground RVO.

3.3 Zdrojový kód

Zdrojový kód Referenčního klienta RVO je spravovaný nástrojem maven. Struktura kódu je následující:

src/main/java	Java zdrojové kódy
src/main/resources/cert	Obsahuje certifikáty truststore a keystore
src/main/resources/data	Výchozí vstupní parametry pro jednotlivé služby
src/main/resources/wsdl	Definice rozhraní webové služby
src/main/resources/application.yml	Konfigurační parametry (např. URL pro spojení)

Složky resources se při sestavení aplikace přenášejí do výsledného jar souboru. Obsahují výchozí nastavení směřované na prostředí Playgroundu RVO.

3.4 Build

Referenční klient RVO na platformě Java je sestavitelný nástrojem maven verze 3.6 a výše. Sestavení používá Java 8.

Postup sestavení:

1. Rozbalte zdrojové soubory aplikace na disk počítače.
2. Přejděte do složky Java.
3. Konfigurace sestavení je v souboru pom.xml. Sestavení provedte příkazem
`mvn clean package.`

Výsledkem sestavení je spustitelný jar soubor ve složce **target**. Soubor je pojmenovaný **HazardRefClient-<verze>.jar**.

Sestavený spustitelný jar má přibližně 30 MB.

Vedle tohoto souboru v rámci sestavení vzniká ve složce target soubor HazardClient.zip s následující strukturou:

HazardRefClient – kořenová složka

\certs – vzorové certifikáty

\data – vzorové JSON datové soubory

HazardRefClient.jar – spustitelný jar

README.md – stručný návod na spuštění klienta

application.yml – konfigurace klienta

Zip soubor je možné rozbalit na disk počítače, doplnit vlastní certifikáty, příslušným způsobem upravit konfiguraci a spustit.



4 Referenční klient .NET

Referenční klient RVO .NET byl implementován v jazyce C# s použitím .NET frameworku verze 4.8.

Implementace vychází z možností a standardů tohoto prostředí. Referenční klient RVO je určený jako vzor pro implementace klientů provozovatelů na platformě .NET, nicméně je plně funkční a použitelný.

Aplikace funguje tak, že naváže spojení s HTTPS serverem, podepíše a odešle zprávu a zkontroluje podpis přijaté zprávy. Číslo požadavku se generuje pro každý požadavek zvlášť.

Pozn: Pro zjednodušení identifikace provozovatele (nebo jeho dodavatele) na prostředí Playground RVO je doporučeno nastavit v konfiguraci ICO_VCP, který umožní jeho identifikaci a cílenou podporu ze strany MF.

4.1 Spouštění

Referenční klient RVO se spouští z příkazové řádky např. následovně:

```
HazardRefClient.exe --operace=Test
```

Dodatečné parametry pro spouštění a logování:

Operace	Test, OveritOsobu, OveritOsobyHromadne, ZiskatVysledkyOveritOsobyHromadne, ZmenitUdajeOsoby – jaká operace webové služby má být volána
Input	cesta k souboru, ve kterém je JSON reprezentace požadavku, tj. vstupní parametry operace - default: \$operace.json
outputJSONRequest	cesta k souboru pro uložení dat požadavku
outputJSONResponse	cesta k souboru pro uložení dat odpovědi

Logování

Na standardní výstup se loguje kompletní request a response včetně hlaviček. XML struktura zpráv je logována prostředky Windows. Konfigurační soubor obsahuje položku **configuration/system.diagnostics/sharedListeners** s nastavením logování do souboru *.svclog. Logovací soubor lze prohlížet v nástroji Microsoft Service Trace Viewer. Jednotlivé operace klienta jsou logovány v aktivitách „Process Action ,<operace>“. Záznamy logu, které obsahují přenášené zprávy jsou uvedeny popisem Message Log Trace:



The screenshot shows the Microsoft Service Trace Viewer interface. The top menu includes File, Edit, View, Activity, and Help. Below the menu is a search bar with 'Look For' and 'Find What' fields. The 'Look In' dropdown is set to 'All Activities'. The main pane displays a list of activities with columns for Activity, # Traces, and Duration. The 'Process action T...' activity is selected, showing 18 traces. The right pane shows a detailed view of a 'Message Log Trace' with columns for Description, Level, Thread ID, Process Name, Time, and Trace Identifier. The 'Description' column contains a list of events, including 'Wrote to the EventLog', 'Message Log Trace', 'Identity was determined for an EndpointReference', 'Identity verification succeeded', 'The security protocol secured the outgoing message', 'Message Log Trace', 'Sent a message over a channel', 'Received a message over a channel', 'Message Log Trace', and 'Received reply over request channel'. The 'Level' column shows 'Information' for all events. The 'Thread ID' column shows '1' for all events. The 'Process Name' column shows 'HazardRefC...' for all events. The 'Time' column shows '09.03.2020 09:30:16...' for all events. The 'Trace Identifier' column shows 'https://docs.micr...' for all events. The bottom pane shows the XML representation of the selected trace, with a 'Message' tab selected. The XML is a SOAP envelope containing a 'MessageLogTraceRecord' element. The 'MessageLogTraceRecord' element has a 'Header' element with 'Action' and 'ActivityId' attributes. The 'Body' element contains a 'TestRequest' element with a 'CisloPozadavku' attribute.

Odesílaná a přijímaná data jsou dále logována ve formátu JSON. Je to z důvodu zjednodušení testování klienta při implementaci vlastní verze.

Oproti XML logům mají JSON logy několik rozdílů:

- Atribut Duvod – důvod může být v JSON logu uvedený velkými písmeny. Je to dáno pouze serializací logu, data jsou webovou službou odesílána správně, tedy s prvním písmenem velkým a ostatními malými
- Data – datumové položky mohou být uvedeny včetně časového údaje, přestože např. u data narození nemá význam. To je dáno tím, že JSON formát nezná typ DATE, ani DATETIME a všechny časové údaje zobrazuje ve formátu včetně času. Webové službě jsou data opět předávána správně, bez časového rozvoje.

4.2 Konfigurace

Defaultní konfigurace klienta vychází ze souboru app.config a je při sestavení uložena do souboru Client.exe.config. Výchozí hodnoty jsou nastaveny tak, aby posílaly zprávy na Playground RVO s ICP_VCP = 123456 a s použitím certifikátu fiktivní organizace Hazard bez rizika, o.p.s.

Konfiguraci lze přímo v souboru HazardRefClient.exe.config změnit.

Konfigurační soubor má standardní strukturu konfiguračního souboru .NET App.config. Soubor je ve formátu XML. Pro ilustraci je uveden fragment souboru:

```
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.8" />
  </startup>
  <appSettings>
    <add key="serviceSigningCertificateSubjectName" value="MF - Dozorová část AISG - Playground" />
  </appSettings>
</configuration>
```



```
<add key="clientSigningCertificateSubjectName" value="hranice.cz" />
<add key="serviceDns" value="api-pg.hazard.mfcr.cz" />
<add key="icovcp" value="123456" />
</appSettings>
<system.diagnostics>

<sources>
```

Nastavení konfiguračních parametrů klienta:

Sekce configuration/appSettings

serviceSigningCertificateSubjectName

Název certifikátu, který služba používá pro podpis vracených zpráv. Certifikát musí být uložený v úložišti certifikátů Windows v sekci Místní počítač/Osobní.

clientSigningCertificateSubjectName

Název certifikátu, které klient používá pro podpis odchozích zpráv. Certifikát musí být uložený v úložišti certifikátů Windows v sekci Místní počítač/Osobní.

serviceDns

Název serveru, na které služba běží (api-pg.hazard.mfcr.cz).

icovcp

Hodnota ICO_VCP provozovatele, která se bude automaticky přidávat do zpráv.

Sekce configuration/system.serviceModel/client

endpoint, atribut address

URL adresa webové služby. Výchozí hodnota je <https://api-pg.hazard.mfcr.cz/rovo/v1>.

Práce s certifikáty

Jméno certifikátu pro podpisy zpráv klientem je nastaveno v parametru **configuration/appSettings/serviceSigningCertificateSubjectName**. Klient vyhledá v úložišti certifikátů operačního systému Windows certifikát s tímto jménem a použije jej pro podpis zprávy. Certifikát musí být uložen v sekci Místní počítač/Osobní a musí obsahovat privátní klíč.

Certifikát pro ověřování podpisu zpráv došlých ze serveru klient hledá v úložišti certifikátů operačního systému Windows podle jména uloženého v parametru **configuration/appSettings/serviceSigningCertificateSubjectName**. Certifikát musí být uložen v sekci Místní počítač/Osobní.

Certifikát používaný serverem Playgroundu RVO pro elektronický podpis odpovědí je distribuován spolu s ostatními certifikáty v souboru certs\aisg.cer.

4.3 Zdrojový kód

Zdrojový kód Referenčního klienta RVO je spravovaný nástrojem Microsoft Visual Studio 2019. Struktura kódu je následující:

Client	Kořenová složka projektu, obsahuje zdrojové kódy
Client/App.config	Konfigurační parametry (např. url pro spojení)
Client/resources/data	Výchozí vstupní parametry pro jednotlivé služby



Client/resources/cert	Obsahuje certifikáty klienta a služby
------------------------------	---------------------------------------

Při sestavení projektu vznikají ve složce Client/bin/Debug (resp. Client/bin/Release) spustitelné soubory klienta. Ve složce je dále pod názvem Client.exe.config umístěn původní app.config. Obsah konfigurace je popsán níže.

4.4 Build

Vývoj a sestavení Referenčního klienta RVO probíhá kompletně v prostředí nástroje Microsoft Visual Studio 19, stačí edice Community.

Postup sestavení:

1. Rozbalte zdrojové soubory aplikace na disk počítače.
2. Otevřete v Microsoft Visual Studiu soubor Client.sln ze složky .NET.
3. Spusťte sestavení volbou menu Sestavit -> Sestavit řešení.

Výsledek sestavení je uložený ve složce Client\bin\debug.

Soubory aplikace mají přibližnou velikost 1,6 MB.