**Združenie DEUS**

**Dátové centrum obcí a miest**

**Integračný manuál pre ISM:**

**Princípy integrácie**

|  |  |
| --- | --- |
| **VERZIA DOKUMEnTU** | 1.6 |
| **dáTUM VYDANIA**  **id DOKUMENTU**  **aUTOR**  **ZHOTOVITEĽ** | xx. 11. 2019  DCOM-ISM-IM\_Princpipy\_integracie  PosAm, s r.o., DATALAN, a.s.  PosAm s.r.o., DATALAN, a.s. |

# Obsah

[Obsah 2](#_Toc27404995)

[Záznam o zmenách 5](#_Toc27404996)

[1 Úvod 6](#_Toc27404997)

[2 Skratky 7](#_Toc27404998)

[3 Základné princípy 9](#_Toc27404999)

[3.1 Testovacie prostredie IS DCOM 9](#_Toc27405000)

[3.2 Dodávateľ ISM si spravuje svoje prostredie v DCOM 9](#_Toc27405001)

[3.3 Vysoká dostupnosť 9](#_Toc27405002)

[4 Spôsoby integrácie 10](#_Toc27405003)

[4.1 Asynchrónna komunikácia prostredníctvom messaging-u 10](#_Toc27405004)

[4.1.1 Správy v RabbitMQ 10](#_Toc27405005)

[4.1.2 Prístup do messaging-u 11](#_Toc27405006)

[4.1.3 Riadenie prístupu 11](#_Toc27405007)

[4.1.4 Vzorová implementácia konzumenta na strane ISM 12](#_Toc27405008)

[4.1.4.1 Java 12](#_Toc27405009)

[4.1.4.2 .Net 15](#_Toc27405010)

[4.2 Webové služby SOAP/REST 18](#_Toc27405011)

[4.2.1 Integračná adresa na zbernicu služieb 18](#_Toc27405012)

[4.2.2 Bezpečnosť rozhraní a riadenie prístupu 18](#_Toc27405013)

[4.2.2.1 Session cookie 19](#_Toc27405014)

[4.2.3 Príklad volania v jazyku Java 20](#_Toc27405015)

[4.2.3.1 Zdrojový kód príkladu 22](#_Toc27405016)

[5 Infraštruktúra a prístupy 23](#_Toc27405017)

[5.1 Pripojenie IS mesta 23](#_Toc27405018)

[5.2 Produkčné prostredie 24](#_Toc27405019)

[5.2.1 Rozhrania integračnej platformy PROD prostredia 24](#_Toc27405020)

[5.2.1.1 Zabezpečenie komunikácie 24](#_Toc27405021)

[5.2.2 Rozhrania aplikačnej platformy PROD prostredia 25](#_Toc27405022)

[5.2.2.1 Zabezpečenie komunikácie 25](#_Toc27405023)

[5.2.3 Podporné infraštruktúrne služby PROD prostredia 25](#_Toc27405024)

[5.2.3.1 DNS 25](#_Toc27405025)

[5.2.3.2 OCSP 25](#_Toc27405026)

[5.2.3.3 PROXY pre TEST prostredie 25](#_Toc27405027)

[5.3 Testovacie prostredie 26](#_Toc27405028)

[5.3.1 Rozhrania integračnej platformy TEST prostredia 26](#_Toc27405029)

[5.3.1.1 Zabezpečenie komunikácie 26](#_Toc27405030)

[5.3.2 Rozhrania aplikačnej platformy TEST prostredia 27](#_Toc27405031)

[5.3.2.1 Zabezpečenie komunikácie 27](#_Toc27405032)

[5.3.3 Podporné infraštruktúrne služby TEST prostredia 27](#_Toc27405033)

[5.3.3.1 OCSP 27](#_Toc27405035)

[5.4 Prístup reprezentantov mesta 28](#_Toc27405036)

[5.4.1 Adresácia sietí 29](#_Toc27405037)

[5.4.2 Parametre L2L tunela 30](#_Toc27405038)

[5.4.3 Zabezpečenie sieťovej komunikácie 30](#_Toc27405039)

[5.4.3.1 Zabezpečenie na strane mesta 30](#_Toc27405040)

[5.4.3.2 Zabezpečenie na strane DCOM 31](#_Toc27405041)

[5.4.4 IPS 31](#_Toc27405042)

[5.4.5 DNS komunikácia 31](#_Toc27405043)

[5.4.6 Služby pre reprezentantov mesta 32](#_Toc27405044)

[5.4.7 Nastavenie počítačov mesta 33](#_Toc27405045)

[5.4.7.1 Certifikáty 33](#_Toc27405046)

[5.4.7.2 Prístup do testovacieho prostredia 33](#_Toc27405047)

[5.4.7.3 d.Viewer 34](#_Toc27405048)

[6 Proces pripojenia mesta 34](#_Toc27405049)

[6.1 Pripojenie IS mesta 34](#_Toc27405050)

[6.2 Pripojenie reprezentantov mesta 34](#_Toc27405051)

[6.2.1 L2L tunel a zabezpečenie sieťovej komunikácie 34](#_Toc27405052)

[6.2.2 DNS komunikácia 35](#_Toc27405053)

[6.2.3 Certifikáty 35](#_Toc27405054)

[6.2.4 Nastavenie počítačov mesta 37](#_Toc27405055)

[7 Overenie funkčnosti služieb 37](#_Toc27405056)

[7.1 Otestovanie backendových služieb v TESTe 37](#_Toc27405057)

[7.1.1 Otestovanie z ISM do IS DCOM 37](#_Toc27405058)

[7.1.2 Otestovanie z IS DCOM do ISM 37](#_Toc27405059)

[7.2 Otestovanie frontendových služieb v TESTe 38](#_Toc27405060)

[7.2.1 Otestovanie dostupnosti verejného portálu s prihlásením ako obyvateľ/podnikateľ 38](#_Toc27405061)

[7.2.2 Otestovanie intranetu mesta v IS DCOM 38](#_Toc27405062)

# Záznam o zmenách

|  |  |  |  |
| --- | --- | --- | --- |
| **Verzia** | **Popis zmien** | **Autor zmeny** | **Dátum** |
| 1.0 | Prvá verzia dokumentu | Ivan Krištek | 22.3.2019 |
| 1.1 | Doplnenie informácií v kapitole 4.2.2 Bezpečnosť rozhraní a riadenie prístupu, doplnenie kapitol 4.2.3 Príklad volania v jazyku Java a 5.2 Prístup do testovacieho prostredia | Miroslav Rúčka, Ivan Krištek | 10.04.2019 |
| 1.2 | Doplnenie informácií v kapitole 4.1 aj s príkladom implementácie konzumenta správ va Java a .Net. Doplnenie kapitoly 5.2.1 Testovanie portálu s prihlásením ako obyvateľ/podnikateľ. | Miroslav Rúčka, Tomáš Mihalovič, Ivan Krištek | 03.05.2019 |
| 1.3 | Doplnenie informácií v kapitolách 2, 5.2.3, 5.3.3 a doplnenie nových kapitol 5.4. a 6.2. | Vladislav Striško | 24.10.2019 |
| 1.4 | Vyjasnenie detailov podľa pripomienok | Miloslav Hladík | 25.10.2019 |
| 1.5. | Doplnenie umiestnenia dokumentov s konfiguračnými parametrami v kapitole 5.4 a upresnenie úprav balíčka Chromium v kapitole 5.4.7.  Úprava procesu pre získanie session cookie pre volanie WS – kapitola 4.2.2.1 | Vladislav Striško, Miroslav Rúčka | 06.11.2019 |
| 1.6 | Doplnenie služby pki.test.dcom.sk v kapitole 5.3.3 | Vladislav Striško | 18.11.2019 |
| 1.6a | Úprava obrázku v kap. 5.1 | Štefan Mašura | 16.12.2019 |

# Úvod

Dokument typu „integračný manuál pre ISM“ je určený dodávateľom IS mesta (ISM), ktorí sa budú integrovať na DCOM. Cieľom tohto dokumentu popísať všeobecné princípy integrácie IS mesta s IS DCOM.

Integračné manuály sú dokumenty, ktoré sa v čase môžu vyvíjať a meniť. Podliehajú procesu „release managementu“ DEUS. Zmeny budú zohľadňovať predovšetkým požadované legislatívne zmeny a ďalšie funkčné požiadavky, vždy s ohľadom na existujúce implementácie. Pri každej zmene bude definované obdobie, dokedy sa musia integrované subjekty prispôsobiť.

Každá zmena podlieha pripomienkovému konaniu, ktoré zastrešuje prevádzkovateľ DEUS.

# Skratky

| Skratka | Význam |
| --- | --- |
| AMQP | Advanced message queueing protocol |
| CA | Certifikačná Autorita |
| CRL | Certificate Revocation List |
| DNS | Domain Name System |
| ESB | Enterprise Service Bus |
| FW | Firewall |
| GUID | Globally Unique Identifier |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | Hypertext Transfer Protocol Secure |
| IAM | Identity and Access Management |
| ISM | Informačný systém mesta |
| IKE | Internet Key Exchange |
| ISO | Informačný systém obce |
| LAN | Local Area Network |
| L2L | LAN to LAN |
| MTOM | Message Transimssion Optimization Mechanism |
| NAT | Network Address Translation |
| NFS | Network File System Protocol |
| OSCP | Online Certificate Status Protocol |
| PKI | Public Key Infrastructure |
| SFTP | Secure File Transfer Protocol |
| SOAP | Simple Object Access Protocol |
| TCP | Transmission Control Protocol |
| TLS | Transport Layer Security |
| UDP | User Datagram Protocol |
| URI | Uniform Resource Identifier |
| WS | Web Service (Webová Služba) |
| WSDL | Web Service Definitiona Language (Jazyk pre definovanie Webových Služieb) |
| XML | eXtensible Markup Language |
| XSD | XML Schema Definition |

# Základné princípy

Rozhrania sú popisované z hľadiska účelu integračných rozhraní a pre každé existuje osobitný integračný manuál.

## Testovacie prostredie IS DCOM

Testovacie prostredie slúži dodávateľom ISM predovšetkým na:

* vývoj, otestovanie, odladenie integračného „kontraktu“,
* overenie zhody s požiadavkami na integráciu ISM.

Pre uľahčenie vývojových prác na strane dodávateľov ISM budú v testovacom prostredí k dispozícii rozhrania, na základe ktorých si budú môcť „odladiť“ funkcionalitu súvisiacu s integráciou.

V rámci testovacieho prostredia bude pre každého dodávateľa k dispozícii „testovacie mesto“ (tenant) s testovacími účtami.

## Dodávateľ ISM si spravuje svoje prostredie v DCOM

Dodávateľ ISM, v závislosti od rozsahu a spôsobu integrácie, bude mať v rámci DCOM vyhradené vlastné „produkčné“ a „testovacie“ cloud prostredie, ktoré si bude spravovať – vytváranie virtuálnych serverov, ich nastavovanie, inštalácie programov na tieto virtuálne servre atď.

## Vysoká dostupnosť

Vysoká dostupnosť poskytovaných integračných služieb je zabezpečená architektonickým riešením IS DCOM. Jednotlivé služby integračného rozhrania sú nasadené na serveroch v klaster konfigurácii. Služby sú nasadené v primárnej lokalite IS DCOM ako vysoko dostupné, a taktiež v sekundárnej lokalite ako záloha v prípade výpadku primárnej lokality.

# Spôsoby integrácie

## Asynchrónna komunikácia prostredníctvom messaging-u

Interakciu medzi IS DCOM a ISM zabezpečuje pri asynchrónnej komunikácii messaging-ová platforma RabbitMQ.

Messaging sa používa pri nasledovných scenároch:

* Distribúcia zmenových dávok z Registra adries (RA)
* Distribúcia zmenových dávok z Registra právnických osôb (RPO)
* Distribúcia zmenových dávok z Registra fyzických osôb (RFO)
* Notifikácie z modulu IS DCOM Evidencia osôb
* Notifikácie z eDesk GW po prevzatí správy zo schránky mesta v eDesku ÚPVS

Vo všetkých scenároch vystupuje IS DCOM ako producent správ a IS mesta ako konzument správ.

### Správy v RabbitMQ

Štandardný tok správ je v RabbitMQ zabezpečený v zmysle návrhového vzoru producent – konzument. Správy sú odosielané do jednotlivých radov (queues). Konzument je pripojený na svoj rad a doručené správy si môže načítať a spracovať.



Obrázok 1: Príklad jednoduchého toku správy od producenta ku konzumentovi

Pre každého konzumenta (jednotlivé ISM aj špecificky IS DCOM) a scenár použitia bude k dispozícii dedikovaná queue, ku ktorej bude mať konzument prislúchajúce prihlasovacie údaje. ISM musí implementovať klienta – konzumenta správ, ktorý sa pripojí na konkrétnu queue (napr. „/ra/ra-moldava-nad-bodvou “) a bude spracovávať a potvrdzovať jednotlivé správy tak, ako budú prichádzať. **Klient musí „počúvať“ v on-line režime.** Queue má nastavený atribút „durable“, čo znamená že správy sú perzistentné. Podľa nastavených oprávnení je správy možné len čítať resp. konzumovať.

Správy sú distribuované prostredníctvom tzv. fan-out exchange, čiže pre každého konzumenta bude do príslušného queue zaradená identická kópia správy (ak je správa určená viacerým konzumentom). Po jej načítaní a potvrdení už nebude možné správu opakovane načítať.



Obrázok 2: Ilustračný príklad „fanout“ exchange spracovania správ

### Prístup do messaging-u

Všetci konzumenti z ISM budú mať dostupnú URL **amqp://messaging.intra.dcom.sk:5671***.* Bezpečnosť komunikačného kanála je zabezpečená kryptovaním dát na úrovni TLS. Pre komunikáciu sa používa protokol AMQP.

### Riadenie prístupu

Prístup na RabbitMQ server a do konkrétnej ISM queue je naviazaný na prihlasovacie meno a heslo. Každý IS mesta bude mať pridelený vlastný technický účet.

Pre jednotlivé integračné scenáre budú vytvorené v platforme samostatné logické oddelenia (napr. messaging.intra.dcom.sk:5671/ra, messaging.intra.dcom.sk:5671/rfo, atď.). Na základe zmluvy o využívaní konkrétnych integrácií budú jednotlivým ISM sprístupnené konkrétne logické oddelenia a v každom bude mať prístup do vlastného queue. Pri pripájaní sa klientom je potrebné špecifikovať konkrétny virtual host.

### Vzorová implementácia konzumenta na strane ISM

#### Java

Táto kapitola obsahuje príklad implementácie klientskej aplikácie pre konzumáciu správ z message queue. Jedná sa o Java springboot aplikáciu, ktorá sa pripája na RabbitMQ server pomocou Spring AMQP knižnice.

Pre klientský prístup jedinou potrebnou závislosťou je RabbitMQ:



čo v prípade maven projektu znamená nasledujúcu závislosť:

<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-amqp</artifactId>  
</dependency>

Základné atribúty v **application.properties**, potrebné pre prihlásenie klientom na RabbitMQ server sú:

# rabbit  
spring.rabbitmq.addresses=messaging.intra.dcom.sk:5671

spring.rabbitmq.username=ra-test-user  
spring.rabbitmq.password=ra-test-user  
spring.rabbitmq.virtual-host=/ra

# tls properties

spring.rabbitmq.ssl.algorithm=TLSv1.2

spring.rabbitmq.ssl.enabled=true

spring.rabbitmq.ssl.validate-server-certificate=true

spring.rabbitmq.ssl.verify-hostname=true

**spring.rabbitmq.addresses** - messaging.intra.dcom.sk:5671.

**spring.rabbitmq.username** a **spring.rabbitmq.password** – prihlasovacie údaje

**spring.rabbitmq.virtual-host** – určenie virtual hosta v rámci RabbitMQ servera (default je „/“)

**spring.rabbitmq.ssl.\*** - ssl parametre spojenia

Pre vytvorenie TLS spojenia je nutné pridať do Java trust-store certifikát CA DCOM, ktorá vydala certifikát servera.

Na základe parametrov v **application.properties** spring boot aplikácia automaticky vytvorí connection factory a pripojí sa pri štarte na RabbitMQ.

Ďalším potrebným krokom je vytvorenie „listenera“, ktorý sa pripojí na queue a počúva na nové správy.

Jednoduchý príklad „listenera“, ktorý spracováva všetky prichádzajúce správy vo formáte XML, prípadne aj skomprimované v gzip encodingu.

package sk.dcom.integration.ext.rachangesconsumer;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.amqp.core.Message;  
import org.springframework.amqp.core.MessageProperties;  
import org.springframework.amqp.rabbit.annotation.RabbitListener;  
import org.springframework.stereotype.Component;  
import sk.dcom.integration.ext.rachanges.service.ChangesResponse;  
  
import javax.xml.bind.JAXBContext;  
import javax.xml.bind.JAXBException;  
import javax.xml.bind.Unmarshaller;  
import java.io.ByteArrayInputStream;  
import java.io.IOException;  
import java.io.InputStream;  
import java.util.zip.GZIPInputStream;  
  
@Component  
public class RaChangeListener {

public static final String *QUEUE\_RA\_CHANGES* = "ra-ba-consumer";

static final Logger *logger* = LoggerFactory.*getLogger*(RaChangeListener.class);  
  
 public ChangesResponse unmarshallChangesResponse(InputStream from) {  
 try {  
 JAXBContext jaxbContext = JAXBContext.*newInstance*(ChangesResponse.class);  
 Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller();  
  
 ChangesResponse rawResponse = (ChangesResponse)jaxbUnmarshaller.unmarshal(from);  
 return rawResponse;  
 } catch (JAXBException e) {  
 *logger*.error(e.getMessage(), e);  
 return null;  
 }  
 }  
  
 @RabbitListener(queues = *QUEUE\_RA\_CHANGES*)  
 public void processChanges(Message message) {  
 InputStream input = new ByteArrayInputStream(message.getBody());  
 MessageProperties msp = message.getMessageProperties();  
 if (msp != null && msp.getContentEncoding() != null && msp.getContentEncoding().equalsIgnoreCase("gzip"))  
 {  
 *logger*.info("RECEIVED gzipped Message: trying to unpack it.");  
 try {  
 input = new GZIPInputStream(input);  
 } catch (IOException e) {  
 *logger*.error(e.getMessage(), e);  
 return;  
 }  
 }  
 ChangesResponse cr = unmarshallChangesResponse(input);  
 if (cr == null) {  
 *logger*.info("RECEIVED Message: " + message.getMessageProperties().toString() + " -> but fail to unmarshal it.");  
 return;  
 }  
 *logger*.info("RECEIVED CHANGE RESPONSE: ID=" + cr.getChangesId() + " generated on=" + cr.getChangesGenerated() + " register size=" + cr.getRegister().size());  
 }  
}

Potvrdenie jednotlivých správ v queue je defaultne nastavené tak, že ak metóda listenera (v príklade **processChanges**) zbehne bez výnimky, správa sa považuje za doručenú. Ak metóda vyvolá výnimku správa sa znovu zaradí do queue.

#### .Net

V tejto kapitole uvádzame príklad implementácie klientskej aplikácie pre konzumovanie správ z message queue. Jedná sa o štandardnú .NET konzolovú aplikáciu, ktorá sa pripája na RabbitMQ server pomocou officiálnej RabbitMQ.Client knižnice.

V prostredí Visual Studia je možné ju nainštalovať do projektu cez NuGet Manager:



Jedná sa o nasledovnú verziu knižnice RabbitMQ.Client (a jej závislosti):



Implementáciu jednoduchého konzumenta vidieť v nasledujúcom príklade:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using RabbitMQ.Client;

using RabbitMQ.Client.Events;

namespace RabbitConsumerTest

{

class Program

{

static void Main(string[] args)

{

String queueName = “dcom.rfo.mnb“;

ConnectionFactory factory = new ConnectionFactory

{

UserName = “test-user”,

Password = “test-password“,

VirtualHost = “/rfo“,

HostName = “messaging.intra.dcom.sk“,

Port = 5671,

Ssl = new SslOption(“messaging.intra.dcom.sk“, String.Empty, true)

};

Console.WriteLine("Connecting to {0} via TLS...", factory.HostName);

using (var connection = factory.CreateConnection())

{

using (var channel = connection.CreateModel())

{

channel.QueueDeclarePassive(queue: queueName);

var consumer = new EventingBasicConsumer(channel);

consumer.Received += (model, ea) =>

{

Console.WriteLine("Message received: ");

foreach (var prop in ea.BasicProperties.Headers)

{

Console.WriteLine(" [Header] {0} : {1}", prop.Key, prop.Value);

}

var body = ea.Body;

var message = Encoding.UTF8.GetString(body);

Console.WriteLine(" [Body] : {0}", message);

};

channel.BasicConsume(queue: queueName,

autoAck: true,

consumer: consumer);

Console.WriteLine("Hit <Enter> to exit...");

Console.WriteLine();

Console.ReadLine();

}

}

}

}

}

Aplikácia pri štarte vytvorí ConnectionFactory s požadovanými parametrami pre vytvorenie šifrovaného TLS spojenia na server messaging.intra.dcom.sk. Dôležitými parametrami sú virtual-host servera a názov queue. Metóda channel.BasicConsume a jej parameter autoAck: true hovorí o tom, že správa po prijatí v metode consumer.Received bude automaticky potvrdená a server ju bude považovať za doručenú. Potvrdená správa bude serverom z queue vymazaná a už nebude možné sa k nej vrátiť a opätovne ju načítať.

Pre úspešné vytvorenie TLS spojenia je nutné pridať do windows trust-store lokálneho počítača z ktorého bude iniciované spojenie certifikát certifikačnej autority CA DCOM, ktorá vydala certifikát servera.

V prípade potreby ďalších informácií je možné sa obrátiť na dokumentáciu RabbitMQ.Client knižnice na adrese <https://www.rabbitmq.com/dotnet.html>

## Webové služby SOAP/REST

Na synchrónnu interakciu medzi IS DCOM a ISM slúžia webové služby štandardu SOAP a REST.

### Integračná adresa na zbernicu služieb

Všetky služby sú publikované na adrese:

* <https://esbx.intra.dcom.sk/XX>

kde XX = kontext konkrétnej služby

### Bezpečnosť rozhraní a riadenie prístupu

Volanie webových služieb DCOM zo strany ISM bude vyžadovať autentifikáciu a autorizáciu.

Každý ISM bude mať pridelený vlastný klientsky certifikát, ktorý bude namapovaný na technický účet v Active Directory. V atribútoch technického účtu bude definované:

* aké mesto sa jedná - formou GUID prideleného tenanta
* ID dodávateľa ISM
* povolená IP adresa pre komunikáciu

Jednotlivým technickým používateľom budú pridelené oprávnenia na volanie konkrétnych služieb vypublikovaným na spoločnej zbernici backendovej integrácie alebo frontendovej integrácie ÚPVS.

Autentifikáciu a autorizáciu klientov bude zabezpečovať loadbalancer F5 BIG-IP modulom APM (Access Policy Manager) – po úspešnej autentifikácii klienta bude udržiavaná cez session cookie.

Proces prihlásenia klienta k službe:

* Klient nadväzuje komunikáciu na URL s publikovanou službou ESB. F5 BIG-IP uloží IP adresu klienta do *src\_IP variable* na ďalšie spracovanie.
* F5 BIG-IP vynúti autentifikáciu klientskym certifikátom.
* F5 BIG-IP cez LDAP Query v AD overí,
  + či je technický účet ku ktorému je priradený certifikát zaradený do požadovanej skupiny s oprávnením na prístup k službe,
  + či sa klient pripája z IP (*src\_IP variable*) ktorú má definovanú v AD ako atribút v rámci účtu.
* F5 BIG-IP sprístupní klientovi požadovanú službu a zároveň vygeneruje session cookie. Ďalšie requesty klienta v rámci platnej session nevyžadujú proces autentifikácie (autentifikáciu F5 BIG-IP vynúti až po vypršaní platnosti session).

#### Session cookie

Každé volanie služby bez session cookie s platným klientskym certifikátom vytvorí automaticky novú session, ktorá je platná nastavený čas (30 min). Vytvorenie session je operácia, ktorá zaťažuje F5 BIG-IP. Preto pri integrácii treba klásť dôraz na správnu manipuláciu s cookie. Ak už session bola vytvorená a je platná, je potrebné priložiť do http headru skupinu session cookie.

Pre získanie session cookie je potrebné zavolať [**https://esbx.intra.dcom.sk/login**](https://esbx.intra.dcom.sk/login)

Táto adresa vráti prázdnu odpoveď 200 a session cookie.

Jedná sa o nasledovné kľúče:

* LastMRH\_Session=xx
* MRHSession=zz

|  |  |
| --- | --- |
| LastMRH\_Session | Tracking the last 8 digits of the MRHSession session ID. For example: LastMRH\_Session=41f45923; MRHSession=d896020385383db9ece7ac6d41f45923  You can use the value of LastMRH\_Session in the Configuration utility to view the details of a particular BIG-IP APM session.  For security purposes, when processed through the Access Policy evaluation, the first 24 digits of MRHSession is rotated, but the last 8 digits of LastMRH\_Session is retained. |
| MRHSession | BIG-IP APM Session ID 32 random hex digits. |

Http hlavička, potom môže vyzerať nasledovne:

Cookie: LastMRH\_Session=0e3af881; MRHSession=6413722beea3d1c3b6dabfd00e3af881

Postup získania a použitie session cookie:

1. GET volanie na [**https://esbx.intra.dcom.sk/login**](https://esbx.intra.dcom.sk/login)
2. Uloženie session cookie v cookie store (LastMRH\_Session=xx, MRHSession=zz)
3. Volanie zvolenej služby s priloženým cookie
   1. ak je session platná, vráti sa odpoveď zo služby
   2. ak je session neplatná vráti sa 403 a je potrebné znova získať session

### Príklad volania v jazyku Java

Jednoduchý príklad volania služby na zbernice pre vyžiadanie wsdl audit služby. Dôležité sú metódy pre načítanie SSLContext-u buildCustomSSLContext. Táto metóda načíta oba spomínané certifikáty a uloží ich pre tento program do SSLContextu. Samozrejme dá sa použiť aj štandardný truststore javy, ktorý číta *Java\jdk\jre\lib\security\cacerts* a načítať iba klientsky certifikát do keystore.

package sk.dcom.integration;  
  
import org.apache.http.HttpResponse;  
import org.apache.http.client.HttpClient;  
import org.apache.http.client.methods.HttpGet;  
import org.apache.http.impl.client.HttpClients;  
import org.apache.http.util.EntityUtils;  
  
import javax.net.ssl.KeyManagerFactory;  
import javax.net.ssl.SSLContext;  
import javax.net.ssl.TrustManagerFactory;  
import java.io.FileInputStream;  
import java.io.IOException;  
import java.security.\*;  
import java.security.cert.CertificateException;  
  
*/\*\*  
 \* http client example with client certificate auth.  
 \*  
 \** ***@author*** *rucka  
 \*/*public class HttpClientWithCertExample {  
  
// public static final String TEST\_URL = "https://esbx.intra.dcom.sk/kn/knservice/odata/Districts?$filter=Name%20eq%20%27Trnava%27&$select=Id,ValidTo,Code,Name&$expand=Region%28$select=Id,Code,Name%29&$orderby=Name";  
 */\*\*  
 \* configuration request, base auth, content type etc...  
 \*/* public static final String *TEST\_URL* = "https://esbx.intra.dcom.sk/security/2.0/audit?wsdl";  
 public static final String *TLS\_VERSION* = "TLSv1.2";  
 public static final String *CONTENT\_TYPE* = "application/json";  
 public static final String *BASIC\_AUTH* = "Basic YWRtaW46MTIzNDU=";  
 public static final String *ENVIRONMENT* = "TEST";  
 */\*\*  
 \* configuration truststore  
 \*/* public static final String *TRUSTSTORE\_PASSWORD* = "changeit";  
 public static final String *TRUSTSTORE\_PATH* = "C:/opt/esbx\_auth\_test";  
 */\*\*  
 \* configuration keystore  
 \*/* public static final String *KEYSTORE\_PASSWORD* = "Servisaky#INT";  
 public static final String *KEYSTORE\_PATH* = "C:/data/tmp/cet/svc\_esbx\_forti.pfx";  
  
 public static void main(String[] args) throws IOException, NoSuchAlgorithmException, CertificateException, KeyStoreException, KeyManagementException, UnrecoverableKeyException {  
 SSLContext sslContext = *buildCustomSSLContext*();  
 HttpClient httpClient = HttpClients.*custom*().setSSLContext(sslContext).build();  
 HttpGet request = *buildRequest*(*TEST\_URL*);  
 HttpResponse response = httpClient.execute(request);  
  
 *printTitle*(" RESPONSE STATUS ");  
 System.*out*.println(response.getStatusLine());  
  
 *printTitle*(" RESPONSE BODY ");  
 System.*out*.println(EntityUtils.*toString*(response.getEntity()));  
 }  
  
 private static HttpGet buildRequest(String uri) {  
 HttpGet request = new HttpGet(uri);  
 request.setHeader("content-type", *CONTENT\_TYPE*);  
 request.setHeader("authorization", *BASIC\_AUTH*);  
 request.setHeader("ugkkhost", *ENVIRONMENT*);  
 return request;  
 }  
  
 private static SSLContext buildCustomSSLContext() throws KeyStoreException, IOException, CertificateException, NoSuchAlgorithmException, KeyManagementException, UnrecoverableKeyException {  
 KeyManagerFactory kmf = *getKeyManagerFactory*();  
 TrustManagerFactory tmf = *getTrustManagerFactory*();  
 SSLContext sslContext = SSLContext.*getInstance*(*TLS\_VERSION*);  
 sslContext.init(kmf.getKeyManagers(), tmf.getTrustManagers(), null);  
 return sslContext;  
 }  
  
 private static TrustManagerFactory getTrustManagerFactory() throws KeyStoreException, IOException, NoSuchAlgorithmException, CertificateException {  
 char[] trustPassphrase = *TRUSTSTORE\_PASSWORD*.toCharArray();  
 KeyStore tks = KeyStore.*getInstance*("JKS");  
 tks.load(new FileInputStream(*TRUSTSTORE\_PATH*), trustPassphrase);  
 TrustManagerFactory tmf = TrustManagerFactory.*getInstance*("SunX509");  
 tmf.init(tks);  
 return tmf;  
 }  
  
 private static KeyManagerFactory getKeyManagerFactory() throws KeyStoreException, IOException, NoSuchAlgorithmException, CertificateException, UnrecoverableKeyException {  
 char[] keyPassphrase = *KEYSTORE\_PASSWORD*.toCharArray();  
 KeyStore ks = KeyStore.*getInstance*("PKCS12");  
 ks.load(new FileInputStream(*KEYSTORE\_PATH*), keyPassphrase);  
 KeyManagerFactory kmf = KeyManagerFactory.*getInstance*("SunX509");  
 kmf.init(ks, keyPassphrase);  
 return kmf;  
 }  
  
 private static void printTitle(String title) {  
 System.*out*.println("|#################################################################################################|");  
 System.*out*.println("|----------------------------------------" + title + "----------------------------------------|");  
 System.*out*.println("|#################################################################################################|");  
 }  
}

#### Zdrojový kód príkladu

V prílohe sa nachádza zbalený java program s apache http clientom. Po otvorení je potrebné nainštalovať závislosti cez príkaz mvn clean package.



# Infraštruktúra a prístupy

## Pripojenie IS mesta

Komunikácia DCOM vs. ISM bude obojsmerná, bude prebiehať cez internet. Všetky služby IS DCOM pre ISM budú publikované na verejnej IP adrese a definovanom porte a bude chránená na rôznych vrstvách.

Komunikáciu ilustruje nasledovný obrázok.



## Produkčné prostredie

### Rozhrania integračnej platformy PROD prostredia

Služby integračnej platformy PROD prostredia na strane DCOM budú vystavené na verejných adresách s certifikátom od verejnej CA:

* https://esbx.intra.dcom.sk
* apmq://messaging.intra.dcom.sk:5671
* https://ismapi.intra.dcom.sk

Na strane DCOM budú bezpečne publikované na zariadení F5 BIG-IP - ADC/WAF (NP1NADC01, NP1NADC02).



#### Zabezpečenie komunikácie

##### Whitelist

Služby budú dostupné iba pre definovaný zoznam verejných IP adries (whitelist) všetkých pripojených miest. Obdobným spôsobom by mal byť limitovaný prístup aj na strane mesta, aby služba nebola verejne vystavená, kde IS DCOM bude pristupovať z verejnej IP adresy 85.248.49.200.

##### https

Služby budú publikované cez zabezpečený hypertextový prenosový protokol HTTPS použitím TLS protokolu. Čím je zabezpečená dôveryhodnosť publikovanej služby (certifikát servera) ako aj šifrovanie komunikácie. Certifikát servera bude publikovať verejná dôveryhodná certifikačná autorita.

##### Vzájomná autentifikácia

Na službách na strane DCOM je zo strany ISM vynucovaná klientska autentifikácia (klientsky certifikát). Strana ISM vyžaduje autentifikáciu menom a heslom.

##### Klientský certifikát

Každý ISM bude mať pridelený vlastný klientsky certifikát, ktorý bude namapovaný na technický účet v Active Directory. V atribútoch technického účtu bude definované o aké mesto sa jedná (formou GUID prideleného tenanta) a zároveň ID dodávateľa ISM. Oprávnenia na volanie jednotlivých vypublikovaných služieb budú prideľované technickému účtu.

### Rozhrania aplikačnej platformy PROD prostredia

Vybrané služby IS DCOM budú dostupné na sprístupnenom intranete PROD prostredia. Predpokladom dostupnosti DCOM služieb je integrácia a L2L VPN prepoj medzi mestom a IS DCOM.

#### Zabezpečenie komunikácie

##### https

Služby budú publikované cez zabezpečený hypertextový prenosový protokol HTTPS použitím TLS protokolu. Čím je zabezpečená dôveryhodnosť publikovanej služby (certifikát servera) ako aj šifrovanie komunikácie. Certifikát servera bude publikovať verejná dôveryhodná certifikačná autorita.

##### Autentifikácia

Službám na strane DCOM je zo strany ISM vynucovaná klientska autentifikácia menom a heslom.

### Podporné infraštruktúrne služby PROD prostredia

#### DNS

Služba DNS spravuje doménové záznamy interných zón DCOM, napr. zóny intra.dcom.sk. DNS služba je pre mestá sprístupnená load ballancerom prevádzkovaným na zariadení F5 BIG-IP. Prístup ku službe budú mať povolené len definované IP adresy mesta.

Služba je publikovaná a dostupná na adrese:

* 10.230.96.201 (dns.intra.dcom.sk)

#### OCSP

Služba OSCP slúži pre on-line overovanie platnosti vydaného certifikátu. V reálnom čase je možné overiť, že vydaný certifikát je skutočne platný.

Služba je publikovaná a dostupná na adrese:

* https://pki.intra.dcom.sk

#### PROXY pre TEST prostredie

Služba PROXY slúži na sprístupnenie testovacieho prostredia DCOM v produkčnom prostredí DCOM.

Služba je publikovaná a dostupná na adrese:

* http://proxytestvm.intra.dcom.sk:3128

## Testovacie prostredie

### Rozhrania integračnej platformy TEST prostredia

Služby integračnej platformy TEST prostredia na strane DCOM budú publikované do internetu rovnako ako služby PROD prostredia s tým rozdielom, že nebudú resolvovateľné z dôvodu rovnakých FQDN ako má prostredie PROD.

Budú vystavené na adresách:

* s verejnym certifikátom ako služba https://esbx.intra.dcom.sk (IP:85.248.49.216)
* služba apmq://messaging.intra.dcom.sk:5671 (IP: 85.248.49.216:5671)
* s verejnym certifikátom ako služba https://ismapi.intra.dcom.sk (IP: 85.248.49.216)

Na strane DCOM budú bezpečne publikované na zariadení F5 BIG-IP - ADC/WAF (NP1NADC01, NP1NADC02).



#### Zabezpečenie komunikácie

##### Whitelist

Služby budú dostupné iba pre definovaný zoznam verejných IP adries (whitelist) všetkých pripojených miest. Obdobným spôsobom by mal byť limitovaný prístup aj na strane mesta, aby služba nebola verejne vystavená, kde IS DCOM bude pristupovať z verejnej IP adresy 85.248.49.200.

##### https

Služby budú publikované cez zabezpečený hypertextový prenosový protokol HTTPS použitím TLS protokolu. Čím je zabezpečená dôveryhodnosť publikovanej služby (certifikát servera) ako aj šifrovanie komunikácie. Certifikát servera bude publikovať verejná dôveryhodná certifikačná autorita.

##### Vzájomná autentifikácia

Na službách na strane DCOM je zo strany ISM vynucovaná klientska autentifikácia (klientsky certifikát). Strana ISM vynucuje autentifikáciu menom a heslom.

##### Klientský certifikát

Každý ISM bude mať pridelený vlastný klientsky certifikát, ktorý bude namapovaný na technický účet v Active Directory. V atribútoch technického účtu bude definované o aké mesto sa jedná (formou GUID prideleného tenanta) a zároveň ID dodávateľa ISM. Oprávnenia na volanie jednotlivých vypublikovaných služieb budú prideľované technickému účtu.

### Rozhrania aplikačnej platformy TEST prostredia

Vybrané služby IS DCOM budú dostupné na sprístupnenej TEST PROXY služby. Predpokladom dostupnosti DCOM služieb je integrácia a L2L VPN prepoj medzi mestom a IS DCOM.

#### Zabezpečenie komunikácie

##### https

Služby budú publikované cez zabezpečený hypertextový prenosový protokol HTTPS použitím TLS protokolu. Čím je zabezpečená dôveryhodnosť publikovanej služby (certifikát servera) ako aj šifrovanie komunikácie. Certifikát servera bude publikovať verejná dôveryhodná certifikačná autorita.

##### Autentifikácia

Službám na strane DCOM je zo strany ISM vynucovaná klientska autentifikácia menom a heslom.

### Podporné infraštruktúrne služby TEST prostredia

#### OCSP

Služba OSCP slúži pre on-line overovanie platnosti vydaného certifikátu. V reálnom čase je možné overiť, že vydaný certifikát je skutočne platný.

Služba je publikovaná a dostupná na adrese:

* https://pki.test.dcom.sk

## Prístup reprezentantov mesta

Komunikácia medzi reprezentantami mesta a dátovým centrom DCOM bude využívať LAN to LAN VPN tunel prechádzajúci cez internet, súčasťou tejto komunikácie je aj súvisiaca DNS komunikácia potrebná pre reprezentantov. Všetky služby IS DCOM pre reprezentantov budú publikované na interných IP adresách DCOM a definovaných portoch a budú chránené na rôznych vrstvách.



Obrázok 3: Komunikácia reprezentanta mesta s IS DCOM

VPN koncentrátor na strane DCOM a prístupové zariadenie na strane mesta vytvárajú IPsec tunel. Týmto tunelom je prenášaná komunikácia reprezentantov mesta do DCOM, komunikácia bude prebiehať iba jedným smerom a to z mesta do prostredia DCOM.

Na prístupovom zariadení mesta musí byť zabezpečené, že prístup k službám dostupným cez IPsec tunel majú len relevantné počítače a servery mesta, t.j. komunikácia bude filtrovaná na základe zdrojových IP adries počítačov a serverov. Nakoľko siete v mestách nepoužívajú jedinečné IP rozsahy tak je potrebné IP adresy zariadení mesta nahradiť jedinečnými IP adresami v rámci DCOM. Prístupové zariadenie mesta musí zabezpečiť preklad zdrojových IP adries mesta na adresy z určeného subnetu mestu a len IP adresy z toho subnetu budú mať povolenú komunikáciu v rámci tunela.

Na strane DCOM je zabezpečené filtrovanie poskytovaných služieb na VPN koncentrátore, core FW, zariadení F5 BIG-IP a PROXY s prístupom do TEST prostredia. VPN koncentrátor bude filtrovať komunikáciu zo subnetu mesta na sprístupnené služby DCOM. Pri filtrovaní bude zohľadňovať ktorá časť subnetu je určená pre servery (DNS služba) a ktorá pre samotných reprezentantov (ostatné služby). Súčasťou VPN koncentrátora bude IPS filter, ktorý bude blokovať neželanú komunikáciu. Core FW, F5 BIG-IP a PROXY do TEST prostredia budú filtrovať komunikáciu zo subnetov miest na im sprístupnené služby DCOM.

Pre každé mesto bude konfigurovaný špecifický IPsec tunel.

### Adresácia sietí

Pre zabezpečenie prístupu miest s rôznymi aj rovnakými IP rozsahmi na LAN sieťach na služby DCOM budú lokálne subnety miest prekladané na strane mesta za IP subnety pridelené správcom DCOM infraštruktúry. Toto zabezpečí, že každé mesto bude mať pridelené jedinečné rozsahy, ktoré budú v prostredí DCOM routované na poskytované služby a základe IP sietí bude možné kontrolovať komunikáciu. Subnety budú prideľované z rozsahu 10.233.0.0/16.

Každé mesto bude mať pridelený subnet rozdelený na dve časti. Prvá časť bude určená pre DNS servery s prístupom k internej DNS službe DCOM. Druhá časť subnetu bude určená pre počítače reprezentantov s prístupom do IS DCOM. Preklad jednotlivých IP adries mesta odporúčame realizovať tak, aby každá IP adresa mesta bola reprezentovaná samostatnou IP adresou z prideleného subnetu. V DCOM prostredí je štandardne každý používateľ reprezentovaný jedinečnou IP adresou. Toto riešenie umožňuje, že pri identifikovaní podozrivej komunikácie (z jedného počítača) nemá zablokovanie jednej IP adresy negatívny dopad na viac ako jeden počítač a jedného používateľa. Usporiadanie IP adresácie v rámci subnetu zostáva plne v kompetencii mesta, je nie potrebné zabezpečiť aby mal používateľ/počítať trvalo rovnakú IP adresu.

Zoznam pridelených IP subnetov mestám bude zaznamenaný v dokumente:

* https://sps01.posam.sk/DCOM/Prev\_DEUS/2018\_Pripajanie\_Miest/ISM\_Podklady/Dokumentacia/Pripojenie\_mesta/DCOM\_VM\_L2L\_IP\_subnets.xlsx

Znázornenie sieťovej komunikácie viacerých miest je na nasledovnom obrázku.



Obrázok 4: Znázornenie sieťovej komunikácie viacerých miest s IS DCOM

### Parametre L2L tunela

Každé mesto bude mať konfigurovaný špecifický L2L tunel, pričom bude zjednotená maximálna množina parametrov. Rozdielne údaje sa týkajú verejnej IP adresy mesta pre ukončenie tunela, preshared key a crypto mapy. Preshared key bude obsahovať znaky osvedčené z iných implementácií:

* alfanumerické znaky a-z, A-Z, 0-9
* špeciálne znaky - + & ! @ # $ % ^ \* ( ) , . : ; \_ = < > { } /

Crypto mapa definuje, ktoré subnety budú prenášané tunelom.

Konfiguračné parametre L2L tunelov budú evidované v dokumentoch:

* „DCOM\_VM\_Interconnect\_information\_Názov-Mesta.xlsx“   
  („Názov-Mesta“ bude nahradený skutočným názvom mesta vo formáte zaužívanom v označovaní obcí a miest – medzery sú nahradené znakom „-“, diakritické znaky nediakritickými a duplicitne povenované mestá sú rozšírené o skratku okresu.)  
  Príklad pre mesto Ružomberok https://sps01.posam.sk/DCOM/Prev\_DEUS/2018\_Pripajanie\_Miest/ISM\_Podklady/Dokumentacia/Pripojenie\_mesta/DCOM\_VM\_Interconnect\_information\_ruzomberok.xlsx

Partner na strane mesta dopĺňa požadované vstupné (žlté podfarbenie).

### Zabezpečenie sieťovej komunikácie

#### Zabezpečenie na strane mesta

Ako bolo spomenuté v kapitole 5.4 Prístup reprezentanta mesta prístupové zariadenia mesta musí zabezpečiť filtrovanie, že cez L2L tunel budú komunikovať len určené servery a počítače mesta.

Každé mesto bude mať pridelený subnet rozdelený na dve časti. Prvá časť bude určená pre DNS servery s prístupom k internej DNS službe DCOM a druhá časť subnetu bude určená pre počítače reprezentantov s prístupom do IS DCOM. Na prístupovom zariadení mesta bude potrebné realizovať, aby preklady interných IP adries DNS serverov bol realizovaný na adresy z prvej časti subnetu a preklady počítačov reprezentantov na adresy z druhej časti subnetu. Ďalej je potrebné prvej a druhej časti subnetu nastaviť, ktoré protokoly môžu používať.

Zoznam sieťových protokolov povolených pre DNS servery:

* 53/udp (DNS)

Zoznam sieťových protokolov povolených pre počítače:

* 80/tcp (HTTP)
* 443/tcp (HTTPS)
* 3128/tcp (PROXY)

#### Zabezpečenie na strane DCOM

Ako bolo spomenuté v kapitole 5.4 Prístup reprezentantov mesta VPN koncentrátor, Core FW, F5 a proxy musia zabezpečiť filtrovanie, že z L2L tunelov budú dostupné len určené služby pre mestá.

Každé mesto bude mať pridelený subnet rozdelený na dve časti. Prvá časť bude určená pre DNS servery s prístupom k internej DNS službe DCOM a druhá časť subnetu bude určená pre počítače reprezentantov s prístupom do IS DCOM.

Na VPN koncentrátore bude potrebné realizovať aby z prvej a druhej časti subnetu boli dostupné len nižšie uvedené služby. Na Core FW, F5 a proxy bude potrebné zabezpečiť, aby subnety miest mali prístup k nižšie uvedeným službám.

Zoznam služieb povolených pre DNS servery:

* dns.intra.dcom.sk:53 (UDP)

Zoznam služieb povolených pre počítače:

* http(s)://www.intra.dcom.sk a https(s)://<mesto>.intra.dcom.sk
* http(s)://bpms.intra.dcom.sk
* http(s)://elearn.intra.dcom.sk
* http(s)://elearning.intra.dcom.sk
* http(s)://epod.intra.dcom.sk
* http(s)://kb.intra.dcom.sk
* http(s)://login.intra.dcom.sk
* http(s)://osoby.intra.dcom.sk
* http(s)://provisioning.intra.dcom.sk
* http(s)://sd.intra.dcom.sk
* http(s)://selfcarezone.intra.dcom.sk
* http://pki.intra.dcom.sk a http://pki.test.dcom.sk
* http://proxytestvm.intra.dcom.sk:3128

### IPS

Nie je limitom pre PoC, rozpracované.

### DNS komunikácia

Vo všeobecnosti mestá používajú interný DNS a tento DNS využívajú počítače na komunikáciu s IS mesta a internetom. Interné DNS-y miest potrebujú mať nastavený Conditional Forwarding pre interné záznamy/zónu DCOM (intra.dcom.sk). Počítače mesta pri vyhľadávaní DCOM služby oslovia interný DNS mesta a tento DNS sprostredkuje počítaču odpoveď na akej IP adrese je dostupná služba.

DNS služba je publikovaná a dostupná na adrese:

* https://dns.intra.dcom.sk

Prístup ku službe budú mať povolené len definované IP adresy z prideleného subnetu mesta.

### Služby pre reprezentantov mesta

Reprezentantom mesta bude sprístupnený produkčný a testovací IS DCOM. Do produkčného IS DCOM budú môcť pristupovať z ľubovoľného prehliadača nachádzajúcom sa na počítači mesta. Do testovacieho IS DCOM budú môcť pristupovať prostredníctvom proxy. Aby používatelia mohli v jednom čase pristupovať do produkčného aj testovacieho IS DCOM, a aj pre ich lepšiu orientáciu, tak budú mať na počítači ďalší prehliadač, ktorý bude komunikáciu s testovacím IS DCOM smerovať na proxy. Napríklad do produkčného IS DCOM bude reprezentant pristupovať z prehliadača Google Chrome a do testovacieho IS DCOM bude pristupovať z prehliadača Chromium s inštalovaným rozšírením Proxy SwitchyOmega.

Komunikáciu reprezentantov ilustruje nasledovný obrázok:



Zoznam dostupných služieb produkčného prostredia (pre používateľov z browsera):

* http(s)://www.intra.dcom.sk a https(s)://<mesto>.intra.dcom.sk
* http(s)://bpms.intra.dcom.sk
* http(s)://elearn.intra.dcom.sk
* http(s)://elearning.intra.dcom.sk
* http(s)://epod.intra.dcom.sk
* http(s)://kb.intra.dcom.sk
* http(s)://login.intra.dcom.sk
* http(s)://osoby.intra.dcom.sk
* http(s)://provisioning.intra.dcom.sk
* http(s)://sd.intra.dcom.sk
* http(s)://selfcarezone.intra.dcom.sk
* http://pki.intra.dcom.sk

Zoznam dostupných služieb testovacieho prostredia (pre používateľov, z browsera Chromium)

* http(s)://www.intra.dcom.sk a https(s)://<mesto>.intra.dcom.sk
* http(s)://bpms.intra.dcom.sk
* http(s)://elearn.intra.dcom.sk
* http(s)://elearning.intra.dcom.sk
* http(s)://epod.intra.dcom.sk
* http(s)://kb.intra.dcom.sk
* http(s)://login.intra.dcom.sk
* http(s)://osoby.intra.dcom.sk
* http(s)://provisioning.intra.dcom.sk
* http(s)://sd.intra.dcom.sk
* http(s)://selfcarezone.intra.dcom.sk
* http://pki.test.dcom.sk

### Nastavenie počítačov mesta

#### Certifikáty

Servery v prostredí DCOM štandardne používajú certifikáty vydávané internou certifikačnou autoritou. Pre overenie dôveryhodnosti servera prehliadač potrebuje veriť, že certifikát servera, na ktorý sa pripája, bol vydaný dôveryhodnou certifikačnou autoritou. Každý počítač s prístupom do DCOM prostredia musí mať importovaný certifikát DCOM CA medzi dôveryhodné koreňové certifikačné autority.

Certifikáty DCOM CA sú dostupné na adrese:

* <https://sps01.posam.sk/DCOM/Prev_DEUS/2018_Pripajanie_Miest/ISM_Podklady/Na_stiahnutie/DCOM_CA_cert.7z>

Certifikáty NASES sú dostupné na adrese:

* https://sps01.posam.sk/DCOM/Prev\_DEUS/2018\_Pripajanie\_Miest/ISM\_Podklady/Na\_stiahnutie/NASES\_CA\_cert.7zPrístup do produkčného prostredia

Reprezentanti budú mať prístup do produkčného prostredia IS DCOM prostredníctvom L2L tunela. Na prístup môžu použiť ľubovoľný internetový prehliadač.

Ak mesto používa na prístup do internetu nejaký proxy server, tak potrebujú na počítačoch nastaviť výnimku pre adresy začínajúce na \*.dcom.sk.

Pozn: Ak použitý prehliadač nevyužíva interné úložisko certifikátov operačného systému, tak špecifické úložisko prehliadača musí obsahovať certifikát CA produkčného prostredia DCOM.

#### Prístup do testovacieho prostredia

Reprezentanti mesta budú mať prístup do testovacieho prostredia IS DCOM prostredníctvom L2L tunela a proxy v DCOM prostredí. Na prístup budú môcť použiť prehliadač Chromium s rozšírením Proxy SwitchyOmega. Balíček s plne nakonfigurovaným prehliadačom Chromium budú mať mestá k dispozícii v 32 a 64 bitovej verzii.

Rozšírenie Proxy SwitchyOmega má nakonfigurovaný auto switch. Pre doménové mená obsahujúce reťazec \*.dcom.sk komunikáciu automaticky presmeruje na definovanú proxy. Pre doménové mená stránok testovacieho prostredia ÚPVS (\*.upvsfix.gov.sk a \*.upvsfixnew.gov.sk) komunikáciu budú smerovať do internetu, podľa nastavenia operačného systému počítača. Nastavenie je kompatibilné aj s riešením ak mesto používa pre prístup do internetu vlastný proxy server. Ostatná komunikácia bude smerovať na lokálny port počítača, t.j. zneprístupní ostatné stránky (myslené na ľubovoľné interné stránky).

Prevádzkovateľ IS DCOM bude pravidelne aktualizovať balíčky DCOM Chromium a správca počítačov zabezpečovať distribúciu nového balíčku Chromium. Zodpovednosť za aktualizácie SW Chromium zostáva na meste.

Balíčky DCOM Chromium sú dostupné na adrese:

* [https://sps01.posam.sk/DCOM/Prev\_DEUS/2018\_Pripajanie\_Miest/ISM\_Podklady/Na\_stiahnutie/DCOM\_Chromium\_32bit.7z](https://sps01.posam.sk/DCOM/Prev_DEUS/2018_Pripajanie_Miest/ISM_Podklady/Na_stiahnutie/OBSOLETE/DCOM_Chromium_32bit_79.7z)
* [https://sps01.posam.sk/DCOM/Prev\_DEUS/2018\_Pripajanie\_Miest/ISM\_Podklady/Na\_stiahnutie/DCOM\_Chromium\_64bit.7z](https://sps01.posam.sk/DCOM/Prev_DEUS/2018_Pripajanie_Miest/ISM_Podklady/Na_stiahnutie/OBSOLETE/DCOM_Chromium_64bit_79.7z)

Balíček podľa bitovej verzie Windows je potrebné rozbaliť rootu disku C:\.

#### d.Viewer

Na každý počítač s prístupom do IS DCOM je potrebné nainštalovať prehliadač dokumentov d.Viewer.

Inštalačný balík je dostupný na stránke:

* <https://www.slovensko.sk/sk/na-stiahnutie>.

# Proces pripojenia mesta

## Pripojenie IS mesta

Proces pripojenia mesta je definovaný v samostatnom dokumente.

## Pripojenie reprezentantov mesta

### L2L tunel a zabezpečenie sieťovej komunikácie

1. „Správca L2L tunelov“ pridelí mestu subnet, zapíše ho do príslušných dokumentov a zašle konfiguračný dokument kontaktnej osobe na strane mesta.
2. Partner na strane mesta vyplní kontaktné údaje za mesto a verejnú IP adresu kde bude ukončený L2L tunel na strane mesta (požadované údaje majú žlté podfarbenie). Vyplnený konfiguračný dokument vráti kontaktnej osobe na strane DCOM.
3. „Správca L2L tunelov“ na strane DCOM vygeneruje preshared key, ktorý odošle prostredníctvom sms partnerovi na strane mesta.
4. „Správca L2L tunelov“ na strane DCOM a tiež partner na strane mesta nakonfigurujú zariadenia ukončujúce tunel.
5. Ak bude L2L tunel nakofigurovaný na oboch stranách, tak by sa mal automaticky vytvoriť.
6. Partner na strane mesta nastaví NAT pravidlá pre DNS servery a počítače mesta.
7. Partner na strane mesta nastaví firewallové pravidlá, ktoré IP adresy môžu komunikovať cez L2L tunel.
8. „Správca L2L tunelov“ poloví prvej a druhej časti subnetu mesta prístup k jednotlivým DCOM službám.
9. Partner na strane mesta overí dostupnosť DNSu a web služieb publikovaných na zariadení F5 BIG-IP.

Overenie funkčnosti tunela je možné realizovať ako je uvedené v nasledovnom príklade:

C:\Windows\system32>**nslookup www.dcom.sk 10.230.96.201**

Server: server.mesto.local

Address: 192.168.1.11

Name: www.dcom.sk

Address: **10.230.96.16**

C:\Windows\system32>**telnet 10.230.96.16 80**

### DNS komunikácia

Správca DNS servera mesta musí nastaviť Conditional Forwarding pre zónu dcom.sk na adresu:

* 10.230.96.201 (dns.intra.dcom.sk)

Overenie funkčnosti je možné realizovať ako je uvedené v nasledovnom príklade:

C:\Windows\system32>**nslookup www.dcom.sk**

Server: server.mesto.local

Address: 192.168.1.11

Name: www.dcom.sk

Address: **10.230.96.16**

### Certifikáty

Správca počítačov mesta musí importovať do lokálneho interného úložiska certifikátov počítača certifikáty certifikačných autorít IS DCOM a NASES. V prípade použitia prehliadača, ktorý nevyužíva interné úložisko certifikátov operačného systému, tak certifikát CA musí importovať do špecifického úložiska prehliadača.

Import certifikátov CA IS DCOM a NASES do interného úložiska počítača je možné realizovať ako je uvedené v nasledovnom príklade:

C:\Windows\system32>**certutil -addstore Root DCOM\_PROD\_Root\_CA.cer**

Root "Dôveryhodné koreňové certifikačné autority"

Signature matches Public Key

**Certificate "DCOM Production Root CA" added to store.**

CertUtil: -addstore command completed successfully.

C:\Windows\system32>**certutil -addstore Root DCOM\_TEST\_Root\_CA.cer**

Root "Dôveryhodné koreňové certifikačné autority"

Signature matches Public Key

**Certificate "DCOM Root Test CA" added to store.**

CertUtil: -addstore command completed successfully.

C:\Windows\system32>**certutil -addstore Root ca.nases.gov.sk.crt.cer**

Root "Dôveryhodné koreňové certifikačné autority"

Signature matches Public Key

**Certificate "DCOM Root Test CA" added to store.**

CertUtil: -addstore command completed successfully.

Overenie importuje možné realizovať ako je uvedené v nasledovnom príklade:

C:\Windows\system32>**certutil -store Root 25d955ae853458ad40cba56f5f2b141b**

Root "Dôveryhodné koreňové certifikačné autority"

================ Certificate 7 ================

Serial Number: 25d955ae853458ad40cba56f5f2b141b

Issuer: CN=DCOM Production Root CA, O=DEUS, L=Bratislava, C=SK

NotBefore: 19. 5. 2014 14:21

NotAfter: 19. 5. 2034 14:31

**Subject: CN=DCOM Production Root CA**, O=DEUS, L=Bratislava, C=SK

Certifikačná autorita - verzia: V0.0

Signature matches Public Key

Root Certificate: Subject matches Issuer

Cert Hash(sha1): bb c9 3c c0 b6 a9 cb 1d 98 2f d8 b8 ef 64 f0 da 43 22 b9 78

No key provider information

Nedá sa nájsť certifikát ani osobný kľúč na dešifrovanie.

Encryption test passed

CertUtil: -store command completed successfully.

C:\Windows\system32>**certutil -store Root 564193973ce0caac4291f1646f3dd35e**

Root "Dôveryhodné koreňové certifikačné autority"

================ Certificate 8 ================

Serial Number: 564193973ce0caac4291f1646f3dd35e

Issuer: CN=DCOM Root Test CA, OU=IT, O=DCOM, L=Bratislava, S=Slovakia, C=SK

NotBefore: 12.5.2014 22:15

NotAfter: 12.5.2034 22:25

**Subject: CN=DCOM Root Test CA**, OU=IT, O=DCOM, L=Bratislava, S=Slovakia, C=SK

Certifikačná autorita – verzia: V0.0

Signature matches Public Key

Root Certificate: Subject matches Issuer

Cert Hash(sha1): 24 85 af 11 35 20 d3 cc 03 55 ff 9c 85 56 c0 68 c7 42 4b 41

No key provider information

Nedá sa nájsť certifikát ani osobný kľúč na dešifrovanie.

Encryption test passed

CertUtil: -store command completed successfully.

### Nastavenie počítačov mesta

Správca počítačov mesta musí na každý počítač s prístupom do testovacieho IS DCOM nakopírovať balíček DCOM Chromium. Balíček obsahuje plne nakonfigurovaný prehliadač.

Na počítačoch so 64-bitovou verziou operačného systému odporúčame používať 64-bitovú verziu prehliadača Chromium.

# Overenie funkčnosti služieb

## Otestovanie frontendových služieb v TESTe

### Otestovanie dostupnosti verejného portálu s prihlásením ako obyvateľ/podnikateľ

Pre otestovanie je potrebné vykonať nasledovné kroky:

1. Spustiť prehliadač Chromium, napr. C:\DCOM\Chromium\32\chrlauncher.exe
2. Otvoriť stránku <https://www.dcom.sk/pre-obcanov-a-firmy>
3. Kliknúť na „Prihlásiť sa“
4. Prehliadač by mal byť presmerovaný na stránku prihlasenie.upvsfix.gov.sk
5. Vyplniť prihlasovacie údaje. Možno použiť ľubovoľný testovací účet ÚPVS.
   1. napr. Prihlasovacie meno: E0005047171, Heslo: PopradTa3
   2. ďalšie testovacie účty: E0005047170, E0005047168
6. Ak je ponúknuté vybrať subjekt pre zastupovanie
   1. napr. vybrať „Jan Tester 2“ a kliknúť na „PRIHLÁSIŤ SA“
7. Prehliadač by mal byť presmerovaný naspäť na stránku <https://www.dcom.sk/pre-obcanov-a-firmy>

už ako prihlásený obyvateľ/podnikateľ

1. Pre test podania je možné použiť napr. tlačidlo všeobecné podanie (na stránke vpravo)

Pozn. Pri testovaní z PosAm-u (mimo L2L tunela) je potrebné použiť VPN DCOM PROD Admin a využiť govproxy.intra.dcom.sk:3128.

### Otestovanie intranetu mesta v IS DCOM

Pre otestovanie je potrebné vykonať nasledovné kroky:

1. Spustiť prehliadač Chromium, napr. C:\DCOM\Chromium\32\chrlauncher.exe
2. Otvoriť stránku <https://www.intra.dcom.sk>
3. Vyplniť prihlasovacie údaje. Možno použiť ľubovoľný testovací účet DCOM pre VM.
   1. napr. Meno: [primator.piestany@test.dcom.sk](mailto:primator.piestany@test.dcom.sk), Heslo: XDiuFRnj4XNhQxFx
4. Kliknúť na „PRIHLÁSIŤ SA“
5. Prehliadač by mal byť presmerovaný na intranetovú stránku mesta

Pozn. Pri testovaní z PosAm-u (mimo L2L tunela) je potrebné použiť VPN DCOM PROD Admin.