Valsts reģionālās attīstības aģentūra



|  |  |
| --- | --- |
|  |  |

Par Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana

-6\_15\_11\_58-VISS\_2010-DTS-PR

versija 2.06

Rīgā 2016

|  |  |  |
| --- | --- | --- |
|  | **IEGULDĪJUMS TAVĀ NĀKOTNĒ** |  |

Dokumenta identifikācija

|  |  |
| --- | --- |
| Dokumenta ID: | -6\_15\_11\_58-VISS\_2010-DTS-PR-V2.06-05.02.2016. |
| Dokumenta nosaukums: | Par Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana.  .  .  . |
| Dokumenta kods: | -6\_15\_11\_58-VISS\_2010-DTS-PR |
| Versija: | Versija 2.06, Laidiens 05.02.2016. (saīsināti V2.06 05.02.2016.) |

Saskaņojumi

|  |  |  |  |
| --- | --- | --- | --- |
| Organizācija | Vārds, uzvārds, amats | Datums | Paraksts |
|  | Atbildīgā persona no Pasūtītāja puses |  |  |
| SIA "ABC software" | J.Korņijenko, projekta vadītājs par tehniskiem jautājumiem no Izpildītāja puses |  |  |
| SIA "ABC software" | M.Pētersons, projekta vadītājs par administratīviem jautājumiem no Izpildītāja puses |  |  |
| SIA "ABC software" | E.Blumberga, projekta kvalitātes kontroles vadītāja |  |  |

Izmaiņu vēsture

|  |  |  |  |
| --- | --- | --- | --- |
| **Versija** | **Datums** | **Apraksts** | **Autors** |
| 1.00 | 22.05.2008. | Izveidota dokumenta sākotnējā versija | J.Korņijenko |
| 1.01 | 22.05.2008. | Rokasgrāmata papildināta saskaņā ar I.Misiņa jautājumiem no 22.05.2008. | J.Gekišs |
| 1.02 | 06.06.2008. | Tiek atjaunota pāreja no ZZDATS.Authotity uz TicketCentral | A.Zelikovičs |
| 1.03 | 11.06.2008. | Tiek atjaunota pāreja no ZZDATS.Authotity uz TicketCentral – papildināts | A.Zelikovičs |
| 1.04 | 17.06.2008. | Tiek atjaunota pāreja no ZZDATS.Authotity uz TicketCentral – papildināts | J.Gekišs |
| 1.05 | 14.07.2008. | Tiek pievienota pāreja IDDV formām no ZZDATS.Authotity uz TicketCentral | J.Gekišs |
| 1.06 | 21.07.2008. | Izmaiņas saskaņā ar iespēju lietotājam pievienot vairākas iestādes. | Irēna Tuleiko |
| 1.07 | 25.07.2008. | Tiek pievienota LVP e-pakalpojumu pāreja | Andrejs Jeršovs |
| 1.08 | 18.09.2008. | Papildināta LVP e-pakalpojumu pāreja | Andrejs Jeršovs |
| 1.09 | 12.01.2010. | Papildinājumi drošā Web servisu izstrādātājiem un identifikācijas piegādātāju izstrādātājiem. | J,Gekišs |
| 1.10 | 19.05.2010. | Papildinājumi MobilaisID autorizācijas provaiderim | Edžus Žeiris |
| 1.11 | 20.03.2012. | Veikti redakcionāli labojumi | Edžus Žeiris |
| 2.00 | 24.05.2013. | Dokuments aktualizēts saskaņā ar WIF un PFAS AUTH STS izstrādes rekomendācijas un praktisko pielietojumu. Dzēstas šādas sadaļas:   * Drošā Web Servisa izstrāde – nodaļa satur vadlīnijas droša Web Servisa izstrādei; * MobilaisID identifikācijas piegādātāja izstrādes vadlīnijas - nodaļa satur vadlīnijas MobilaisID identifikācijas piegādātāja izstrādei; * TicketCentral pieslēgšana pie portāla - nodaļā aprakstīts portāla autentifikācijas scenārijs; * Pāreja no ZZDATS.Authority uz TicketCentral vadlīnijas - nodaļa satur vadlīnijas pārejai no ZZDATS.Authority uz TicketCentral; * Pielikumi.   Precizēta sadaļa – Ievads. | J.Korņijenko |
| 2.01 | 08.10.2013. | Pievienota nodaļa 4.1. „Autentifikācijas datu saņemšana no bankas izmantojot tiešo linku” | J.Korņijenko |
| 2.02 | 16.07.2015 | Pievienota nodaļa 4.1.“Bankas MS VisualStudio template apraksts”, kā arī sadaļa 4.2.1. un 2.pielikums. | J.Gekišs |
| 2.03 | 29.07.2015 | Pievienota sadaļa 4.3 | M.Gūtmanis |
| 2.04 | 12.08.2015 | Papildinātas sadaļas 4.1.1. un 4.1.2. | J.Korņijenko |
| 2.05 | 25.08.2015 | Precizētas sadaļas 4.1., 4.2. un 4.3. | J.Korņijenko |
| 2.06 | 05.02.2016 | Precizējumi sadaļā 4. | M.Gūtmanis |

Saturs

[Attēlu saraksts 5](#_Toc428369405)

[1. Ievads 6](#_Toc428369406)

[1.1. Dokumenta nolūks 6](#_Toc428369407)

[1.2. Darbības sfēra 6](#_Toc428369408)

[1.3. Termini un saīsinājumi 6](#_Toc428369409)

[1.4. Saistītie dokumenti 6](#_Toc428369410)

[1.5. Dokumenta pārskats 6](#_Toc428369411)

[1.6. Problēmu ziņošana 7](#_Toc428369412)

[1.7. Vispārējais apraksts 7](#_Toc428369413)

[2. Drošības talonu servisu izmantošana 10](#_Toc428369414)

[2.1. Lietotāju tipi 10](#_Toc428369415)

[2.2. Pieprasītāju atribūtu apraksts 10](#_Toc428369416)

[2.3. PFAS AUTH STS ieejas punkti 13](#_Toc428369417)

[3. Drošā Web Servisa izsaukšana 15](#_Toc428369418)

[3.1. ws2007Federation vs ws2007FederationNoSct 15](#_Toc428369419)

[3.2. Droša Web Servisa apraksts 15](#_Toc428369420)

[3.3. Drošā Web servisa klienta konfigurācija 16](#_Toc428369421)

[4. Bankas identifikācijas piegādātāja izstrādes vadlīnijas 19](#_Toc428369422)

[4.1. Identifikācijas adaptera MS VisualStudio template apraksts 19](#_Toc428369423)

[4.2. Bankas adaptera specifikācija 19](#_Toc428369424)

[4.3. Univerālā bankas adaptera specifikācija 19](#_Toc428369425)

[4.4. HTTP plūsmas piemērs autentificējoties ar banku 21](#_Toc428369426)

[4.5. Autentifikācijas datu saņemšana no bankas izmantojot tiešo linku 23](#_Toc428369427)

[4.6. HTTP plūsmas piemērs autentificējoties no bankas 24](#_Toc428369428)

[1. Pielikums. HTML lapas saites izveidošana autentifikācijas datu saņemšanai no bankas 26](#_Toc428369429)

[2. Pielikums. Testa adaptera pamata kodi 30](#_Toc428369430)

# Attēlu saraksts

[1.attēls. Publisko Web servisu drošā izsaukuma risinājums 7](#_Toc428369431)

[2.attēls. Autentifikācijas brokera izmantošana Web servisa izsaukšanai 8](#_Toc428369432)

[3.attēls. Identifikācijas piegādātāja izmantošana Web lietojuma resursa piekļuvei 8](#_Toc428369433)

[4. attēls. Iespēja izsaukt latvija.lv portāla e-pakalpojumu ar autentifikāciju 23](#_Toc428369434)

# Ievads

## Dokumenta nolūks

Šis dokuments ir projekta „Par Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana” 1.2.7.Vienotā pieteikšanās moduļa papildinājumu nodevums.

Programmētāja rokasgrāmata ir domāta e-pakalpojumu un saistīto lietojumu izstrādātājiem, identifikācijas piegādātāju (*provider*) veidotājiem. Pirms šī dokumenta lasīšanas būtu vēlams iepazīties ar [1] un [2] dokumentiem.

Programmētāja rokasgrāmatas nolūks ir:

* izklāstīt PFAS.STS un vienotās pieteikšanās moduļa lietojumu kopējo aprakstu;
* aprakstīt drošā Web Servisa un tā klienta izstrādes procesu;
* aprakstīt bankas identifikācijas piegādātāja izstrādes procesu.

## Darbības sfēra

Dokuments paredzēts lietošanai projekta izstrādes grupas ietvaros, kā arī visām tām personām, kuru darba pienākumi ir tieši saistīti ar projekta realizāciju.

## [Termini un saīsinājumi](#_Toc65487056)

Visi šajā dokumentā izmantotie termini un saīsinājumi ir apkopoti [3] dokumentā.

## Saistītie dokumenti

Instrukcija ir lietojama kopā ar šādiem dokumentiem:

1. A GUIDE TO CLAIMS-BASED IDENTITY AND ACCESS CONTROL. Authentication and Authorization for Services and the Web. Dominick Baier. Vittorio Bertocci. Keith Brown. Eugenio Pace. Matias Woloski.
2. Microsoft Windows Identity Foundation (WIF) Whitepaper for Developers. Keith Brown. Sesha Mani.
3. IVIS „Terminu un saīsinājumu indekss.” (EPS-31/2005-IVIS-OLIMPS-TSI-V0.02 07.03.2006.).
4. STS konfigurēšana. Administratora rokasgrāmata (VRAA-6\_15\_11\_58-VISS\_2010-STS-AR)

## Dokumenta pārskats

Dokuments sastāv no 4 (četriem) nodalījumiem:

* Ievads – nodaļa satur dokumenta izmantošanas aprakstu, terminu un saīsinājumu definīcijas un saistīto dokumentu sarakstu, kā arī vispārējais apraksts;
* Drošības talonu servisa izmantošana – nodaļa satur PFAS AUTH STS izmantoto deklarāciju apraksts.
* Drošā Web Servisa izsaukšana – nodaļa satur vadlīnijas droša Web Servisa izstrādei un izsaukšanai;
* Bankas identifikācijas piegādātāja izstrādes vadlīnijas – nodaļa satur vadlīnijas Bankas identifikācijas piegādātājam.

## Problēmu ziņošana

Gadījumos, kad lietotāja dokumentācijā vai programmatūrā ir pamanītas problēmas, par tām jāpaziņo uzturošajam personālam.

## Vispārējais apraksts

Lai projektētu, realizētu un uzstrādātu drošos Web servisus, nepieciešams izmantot jaunas tehnoloģijas, kas ļauj nodrošināt funkcionalitāti potenciāli nedrošos tīklos. 1.attēlā parādīts publisko Web servisu droša izsaukuma risinājums, risinājuma detalizēts apraksts ir pieejams [1] un [2] dokumentā.



1.attēls. Publisko Web servisu drošā izsaukuma risinājums

Publisko Web servisu drošā izsaukuma risinājums izveidots sekojoši:

* Publiskais Web serviss izmanto servera sertifikātu drošo savienojumu (SSL) uzstādīšanai, izmantojot HTTPS.
* Klienta aplikācija nosūta drošības talonu, ar kuru tā apstiprina savu identitāti.
* Drošības talona akreditācijas dati tiek izmantoti klienta autorizācijai.
* Publiskais Web serviss izmanto drošo sistēmu piekļuvei pie kataloga datiem.

VISS (bija IVIS) vidē lietojamiem Web servisiem nepieciešams noteikt klienta autentifikāciju heterogēnās vidēs, lai varētu veikt autorizāciju un auditāciju. Šim nolūkam tiek izmantots autentifikācijas brokeri, lai nodrošinātu kopējo piekļuvi aplikāciju grupai. Autentifikācijas brokeris veic sarunas starp klientu aplikācijām un Web servisiem, tas noņem taisnās saites. Autentifikācijas brokeris izdod drošības talonus, kuri tiek izmantoti identifikācijai.

Situācija, kad klients izsauc Web servisu starp autentifikācijas brokeri parādīta 2.attēlā.



2.attēls. Autentifikācijas brokera izmantošana Web servisa izsaukšanai

Autorizācijas brokeris Web servisa izsaukšanai tiek izmantots sekojoši:

* Autentifikācijas brokeris pārbauda klienta akreditācijas datus un izdod drošības talonu, kuru izmanto serviss klienta autentifikācijai. Drošības talons tipiski tiek pārbaudīts uz servisa, bet servisam nav obligāti jāgriežas pie autorizācijas brokera klienta akreditācijas datu pārbaudes gadījumā, jo serviss uzticas drošības talonam, kuru izdeva autorizācijas brokeris.

VISS vidē Web lietojumiem nepieciešama autentifikācija. Šim nolūkam izmantojam identifikācijas piegādātājus. Situācija, kad klientam nepieciešana autentificēta pieeja pie Web resursa izmantojot identifikācijas piegādātāju parādīta 3.attēlā.



3.attēls. Identifikācijas piegādātāja izmantošana Web lietojuma resursa piekļuvei

Piekļuve pie resursa realizēta sekojoši:

* Klients griežas pie resursa. Piekļuvei nepieciešama autentifikācija. Lietotājs tiek pāradresēts uz Identifikācijas piegādātāju.
* Identifikācijas piegādātājā tiek pārbaudīti lietotāja akreditācijas dati. No identifikācijas piegādātāja tiek atgriezts drošības talons, ar to klients griežas pie resursa atkārtoti.

# Drošības talonu servisu izmantošana

## Lietotāju tipi

Pavisam tiek izšķirti dažāda veida drošības taloni, kur katrs satur atribūtu komplektu saskaņā ar 1.tabulu, kur ir aprakstīti šādi lietotāju tipi:

* Izmantojot PFAS AUTH autentifikācijas nodrošinātāju:
* SU – sistēma (lietotāja vārds un parole);
* SS – sistēma (sertifikāts);
* DU – iestādes darbinieks (lietotāja vārds un parole);
* DS – iestādes darbinieks (sertifikāts).
* Izmantojot PFAS AUTH STS ar delegāciju no *IdentitySelector*:
* I – iedzīvotājs;
* LE – juridiskās personas pilnvarots pārstāvis;
* UN – neapliecināta identitāte.
* *IdentitySelector* (Vienotās pieteikšanas modulis) ar delegāciju no banku identifikācijas nodrošinātājiem:
* I\_LVP – iedzīvotāja pilnvarotais pārstāvis;
* LE\_LVP – juridiskās personas pilnvarots pārstāvis;
* UN\_LVP – neapliecināta identitāte.
* banku identifikācijas nodrošinātājs:
* I\_B – iedzīvotājs.

## Pieprasītāju atribūtu apraksts

Pieprasītāju atribūtu saraksts dots 1.tabulā, kur ar M – atzīmētas *mandatory* (obligātās) tiesības, bet O-atzīmētas – *optional* (izvēles) tiesības.

1.tabula

Pieprasītāju atribūtu apraksts

| Atribūts | Vārdtelpa | Apraksts | SU | SS | DU | DS | I | LE | UN | I\_LVP | LE\_LVP | UN\_LVP | I\_B |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| nameidentifier | http://schemas.xmlsoap.org/ws/2005/05/identity/claims/ | Iekšējais kods, kas viennozīmīgi identificē autorizāciju drošības talonā (skat. 2.tabulā). | M | M | M | M | M | M | M | M | M | M | M |
| authenticationmethod | http://schemas.microsoft.com/ws/2008/06/identity/claims/ | IdP URN identifikators | M | M | M | M | M | M | M | M | M | M | M |
| authenticationinstant | http://schemas.microsoft.com/ws/2008/06/identity/claims/ | Autentifikācijas apgalvojuma izsniegšanas laiks | M | M | M | M | M | M | M | M | M | M | M |
| privatepersonalidentifier | http://schemas.xmlsoap.org/ws/2005/05/identity/claims/ | Personas kods |  |  | O | O | O | O |  | M | M |  | M |
| emailaddress | http://schemas.xmlsoap.org/ws/2005/05/identity/claims/ | E-pasta adrese |  |  | O | O |  |  |  |  |  | O |  |
| USER\_AUTHORITY | http://www.oasis-open.org/RSA2004/attributes/ | Lietotāja autentifikācijas iestāde kur darbojas lietotājs (šobrīd netiek izmantots) |  |  | O | O |  |  |  |  |  |  |  |
| sid | http://schemas.xmlsoap.org/ws/2005/05/identity/claims/ | Lietotāja autentifikācijas unikāls identifikators PFAS AUTH | M | M | M | M |  |  |  |  |  |  |  |
| AUTHORITY | http://www.oasis-open.org/RSA2004/attributes/ | Lietotāja piešķirtās autentifikācijas iestāde - īpašnieks | O | O | O | O |  |  |  |  |  |  |  |
| name | http://schemas.xmlsoap.org/ws/2005/05/identity/claims/ | Vārds | O | O | O | O |  |  |  |  |  | O |  |
| surname | http://schemas.xmlsoap.org/ws/2005/05/identity/claims/ | Uzvārds |  |  | O | O | O | O | O | M | M | O | M |
| givenname | http://schemas.xmlsoap.org/ws/2005/05/identity/claims/ | Vārds |  |  | O | O | O | O | O | M | M | O | M |
| action | http://docs.oasis-open.org/wsfed/authorization/200706/claims/ | Lietotājam piešķirtas operācijas PFAS AUTH (aktuālas) | O | O | O | O |  |  |  |  |  |  |  |
| role | http://schemas.microsoft.com/ws/2008/06/identity/claims/ | Lietotājam piešķirtas lomas PFAS AUTH (aktuālas) | O | O | O | O |  |  |  |  |  |  |  |
| legalentity | http://ivis.eps.gov.lv/schema/identity/claims/ | Juridiskās personas, kuru pārstāv lietotājs, UR 11-zimju kods |  |  |  |  | O |  |  |  | M |  |  |
| legalentityname | http://ivis.eps.gov.lv/schema/identity/claims/ | Juridiskās personas, kuru pārstāv lietotājs, nosaukums (no UR DB) |  |  |  |  | O |  |  |  | M |  |  |
| legalentityaddress | http://ivis.eps.gov.lv/schema/identity/claims/ | Juridiskās personas, kuru pārstāv lietotājs, juridiskā adrese (no UR DB) |  |  |  |  | O |  |  |  | O |  |  |
| legalentityposition | http://ivis.eps.gov.lv/schema/identity/claims/ | Juridiskās personas, kuru pārstāv lietotājs, amats (no UR DB) |  |  |  |  |  |  |  |  | O |  |  |
| 109x34 | http://ivis.eps.gov.lv/schema/media/image/ | Identifikācijas piegādātāja bildes interneta adrese |  |  |  |  |  |  |  | O | O | O |  |
| employeegroup | http://ivis.eps.gov.lv/schema/identity/claims/ | Lietotāja grupu identifikatori no PFAS AUTH |  |  | O | O |  |  |  |  |  |  |  |
| systemcode | http://ivis.eps.gov.lv/schema/identity/claims/ | VISS informācijas sistēmas identifikators, kas veic pieprasījumu | O | O |  |  |  |  |  |  |  |  |  |
| grantorname | http://ivis.eps.gov.lv/schema/identity/claims/ | Pilnvaras izdevēja nosaukums |  |  |  |  |  |  |  | M | M |  |  |
| grantor | http://ivis.eps.gov.lv/schema/identity/claims/ | Pilnvaras izdevēja identifikators (reģistrācijas numurs – iestādes kods) |  |  |  |  |  |  |  | M | M |  |  |

*nameidentifier* pielaides lietotāju kodu veido viens vai vairāki identifikatori, kopā ar atsauci uz identifikācijas sistēmu, piemēram:

2.tabula

Lietotāja nameidentifier apraksts

| Personas tips | Identifikatora piemērs | nameidentifier atribūts – format |
| --- | --- | --- |
| Iedzīvotājs | *PK:10098610000* | urn:ivis:100001:name.id-viss |
| Iedzīvotājs ar neapstiprinātu identitāti (pašreiz tikai e-mail) | *jurijs@abcsoftware.lv* | urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress |
| Iestādes darbinieks (no IVIS klasifikatora) | *AU:100001-PK:10098610000* | urn:ivis:100001:name.id-viss |
| Uzņēmuma paraksttiesīgā persona (Uzņēmumu reģistra DB) | *PK:07017010000-UR:40003627089* | urn:ivis:100001:name.id-viss |
| Sistēma | *AU:100001* | urn:ivis:100001:name.id-viss |
| Iedzīvotājs, kuru   * pilnvarojis cits iedzīvotājs * pilnvarojis uzņēmums * pilnvarojusi valsts iestāde | *DP:01127612344-*  *PK:07017010000*  *DP:40003627089-*  *PK:07017010000*  *DP:100001-*  *PK:07017010000* | urn:ivis:100001:name.id-viss |

*nameidentifier* pielaides saņemšanas WIF3.5 koda piemērs:

ClaimsIdentity cIdentity = claimsPrincipal.Identity as ClaimsIdentity;

                    if (cIdentity != null)

                        var nameId = GetClaimValue(cIdentity, ClaimTypes.NameIdentifier);

private string GetClaimValue(IClaimsIdentity identity, string claimType) {

            Claim claim = identity.Claims.FirstOrDefault<Claim>(c => c.ClaimType == claimType);

            if (claim != null && !string.IsNullOrEmpty(claim.Value)) {

                return claim.Value;

            }

            return null;

        }

## PFAS AUTH STS ieejas punkti

Drošības talonu servisa adrese sastāv no

.../Issue.svc/trust/{version}/{name}

3.tabula

{version} atribūta iespējamās vērtības un tā atbilstība SAML un WS-\* standartiem

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | SOAP | WS-Trust | WS-Addressing | WS-Security |
| 2005 | V1.2 | V1.2 (2005) | 2005/08 | V1.1 |
| 13 | V1.2 | V1.3 | 2005/08 | V1.1 |
| wse3 | V1.2 | V1.2 (2005) | 2004/08 | V1.1 |

4.tabula

{name} atribūta iespējamās vērtības un tā atbilstība WS-\* standartiem

| Name | WS-Security |
| --- | --- |
| certificate | Certificate profile |
| certificatemixed | Certificate profile over transport |
| username | UserName profile |
| usernamemixed | UserName profile over transport |
| issuedtokensymmetricbasic256 | IssedToken(SAML) profile |
| issuedtokenmixedsymmetricbasic256 | IssedToken(SAML) profile over transport |

VISS infrastruktūrā drošie servisi tiek izmantoti tikai ar SSL („*over transport*”), PFAS AUTH STS tiek uzturēti šādas sasaistes (*bindings*) .NET 4.0 ietvaram:

5.tabula

Sasaistes

|  |  |
| --- | --- |
| Name | Binding configuration |
| 13/certificatemixed | <ws2007HttpBinding>  <binding name="certificateMixed">  <security mode="TransportWithMessageCredential">  <message clientCredentialType="Certificate" establishSecurityContext="false" />  </security>  </binding>  </ws2007HttpBinding> |
| 13/usernamemixed | <ws2007HttpBinding>  <binding name="usernameMixed">  <security mode="TransportWithMessageCredential">  <message clientCredentialType="UserName" establishSecurityContext="false" />  </security>  </binding>  </ws2007HttpBinding> |
| 2005/certificatemixed | <wsHttpBinding>  <binding name="certificateMixed">  <security mode="TransportWithMessageCredential">  <message clientCredentialType="Certificate" establishSecurityContext="false" />  </security>  </binding>  </wsHttpBinding> |
| 2005/usernamemixed | <wsHttpBinding>  <binding name="usernameMixed">  <security mode="TransportWithMessageCredential">  <message clientCredentialType="UserName" establishSecurityContext="false" />  </security>  </binding>  </wsHttpBinding> |

# Drošā Web Servisa izsaukšana

Drošie Web servisi tiek izstrādāti balstoties uz konsorcija OASIS izstrādātiem protokoliem *WS-Security, WS-Trust, WS-Converation*. Protokoli tika implementēti Windows *Identity* *Foundation* .NET klašu bibliotēkā 3.5 vai vienkārši .NET 4.0 ietvara versiju.

Drošo web servisu izsaukšana ir detalizēti aprakstīta [4] sadaļā 8.4. WS-Federatoin Active Profile.

## ws2007Federation vs ws2007FederationNoSct

No biznesa procesa viedokļa var būt dažādi scenāriji. Piemēram, datu saņemšanas operācija sastāv no vairākiem soļiem: iegūt datus no DIT servisa pa vienam ziņojumam, kopa vairāk nekā ~10 vaicājumi. Šajā gadījuma lietderīgi izmantot WS-SecureConversation, kas nodrošina drošas sesijas uzturēšanu:

Izraksts no wikipedia: WS-SecureConversation is to establish security contexts for multiple SOAP message exchanges, reducing the overhead of key establishment.

6.tabula

Ziņojumu plūsma atkarībā no *binding* nosaukuma

|  |  |
| --- | --- |
| Name | Message Flow |
| ws2007FederationNoSct | CallServiceRequest  CallServiceResponse |
| ws2007Federation | IssueSCTRequest  IssueSCTResponse  CallServiceRequest  CallServiceResponse  CallServiceRequest  CallServiceResponse  …  CallServiceRequest  CallServiceResponse  CancelSCTRequest  CancelSCTResponse |

ws2007Federation izmantošanas rekomendācija:

ws2007Federation gadījumā nedrīkst turēt sesiju atvērtu ilgu laiku, jo sesiju skaits ir ierobežots. Tas nozīme, ka citiem klientiem varētu būt atteikums, pieprasot sesiju.

## Droša Web Servisa apraksts

VISS WCF droša servisa adrese sastāv no

...Service.svc/{name}

7.tabula

{name} atribūta iespējamās vērtības un to atbilstība WS-\* standartiem

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | SOAP | WS-Trust | WS-Addressing | WS-Security | WS-SecureConversation |
| ws2007Federation | V1.2 | V1.3 | 2005/08 | V1.1 | V1.3 |
| ws2007FederationNoSct | V1.2 | V1.3 | 2005/08 | V1.1 |  |

ws2007Federation vai ws2007FederationNoSct izvēle ir atkarīga no biznesa procesa. Ja klientam jāpieslēdzas pie servisa un jāizsauc vairāk pār ~10 vaicājumiem, tad lietderīgāk izmantot SCT. Ja serviss jāizsauc vienu reizi, tad bez SCT.

8.tabula

Drošiem web servisiem jānodrošina šādas sasaistes (*bindings*) .NET 4.0 ietvaram

|  |  |
| --- | --- |
| Name | Binding configuration |
| ws2007Federation | <ws2007FederationHttpBinding>  <binding name="ws2007Federation">  <security mode="TransportWithMessageCredential" />  </binding>  </ws2007FederationHttpBinding> |
| ws2007FederationNoSct | <ws2007FederationHttpBinding>  <binding name="ws2007FederationNoSct">  <security mode="TransportWithMessageCredential">  <message establishSecurityContext="false" >  </message>  </security>  </binding>  </ws2007FederationHttpBinding> |

9.tabula

Drošiem Web servisiem jānodrošina šādas sasaistes (bindings) .NET 3.5 ietvaram

|  |  |
| --- | --- |
| Name | Binding configuration |
| ws2007FederationNoSct | <customBinding>  <binding name="ws2007FederationNoSct">  <security authenticationMode="IssuedTokenOverTransport" messageSecurityVersion="WSSecurity11WSTrust13WSSecureConversation13WSSecurityPolicy12BasicSecurityProfile10" >  </security>  <textMessageEncoding />  <httpsTransport />  </binding>  </customBinding> |

## Drošā Web servisa klienta konfigurācija

Klienta konfigurācija atkarīga no biznesa procesa. Ja klientam jāpieslēdzas pie servisa un jāizsauc vairāk pār ~10 vaicājumiem, tad lietderīgāk izmantot SCT. Ja serviss jāizsauc vienu reizi, tad bez SCT.

10.tabula

Drošā servisa tipiskie izsaucēji

| Caller | Description | Lietotāju tips |
| --- | --- | --- |
| System | Drošo servisu izsauc sistēma. Par sistēmu uzskatam programmatūru, kura darbojās pati pa sevi (Windows Serviss, Web Lietojums, Web Serviss un citi). Sistēmai drošības talonu servisā jāautentificējas ar sertifikāta palīdzību. Sertifikāta reģistrēšana notiek pēc sekojoša scenārija: sistēma pasūta vai noģenerē sertifikātu ar privāto atslēgu, tad publisko daļu nosūta drošības servisa talonam, drošības servisa konfigurācijā reģistrējas nodotais sertifikāts un tam izdod attiecīgas tiesības. | SU, SS, |
| User | Drošo servisu izsaukumu iniciē pats lietotājs. To viņš paveic ar attiecīgu programmatūru (Windows Forms). Lietotājam jāautentificējas ar sertifikātu vai lietotāju vardu, paroli.  Ja autentificējas ar lietotāja vārdu un paroli, paroli nedrīkst saglabāt. | DU, DS, I\_LVP, LE\_LVP, UN\_LVP, I\_B |
| System + Delegation | Drošo servisu izsauc sistēma, bet ar lietotāja deleģētām tiesībām. | I, LE, UN, |

Atkarībā no izsaucējā mainās klienta konfigurācijas *bindings*, .NET4.0.

11.tabula

Klienta konfigurācijas *bindings*

|  |  |
| --- | --- |
| =Izsaucējs | system.serviceModel configuration (piemēri) |
| System | <system.serviceModel>  <behaviors>  <endpointBehaviors>  <behavior name="certificatemixed">  <clientCredentials>  <clientCertificate findValue="..." storeLocation="LocalMachine" x509FindType="FindByThumbprint" />  </clientCredentials>  </behavior>  </endpointBehaviors>  </behaviors>  <bindings>  <ws2007HttpBinding>  <binding name="certificateMixed">  <security mode="TransportWithMessageCredential">  <message clientCredentialType="Certificate" establishSecurityContext="false" />  </security>  </binding>  </ws2007HttpBinding>  <ws2007FederationHttpBinding>  <binding name="ws2007FederationNoSct">  <security mode="TransportWithMessageCredential" >  <message establishSecurityContext="false" >  <issuer address=".../Issue.svc/trust/13/certificatemixed" binding="ws2007HttpBinding" bindingConfiguration="certificatemixed" />  </message>  </security>  </binding>  </ws2007FederationHttpBinding>  </bindings>  <client>  <endpoint address="...Service.svc/ws2007Federation" behaviorConfiguration="certificatemixed" binding="ws2007FederationHttpBinding" bindingConfiguration="ws2007Federation" contract="Contract" name="Name"/>  </client>  </system.serviceModel> |
| User | <system.serviceModel>  <bindings>  <ws2007HttpBinding>  <binding name="usernameMixed">  <security mode="TransportWithMessageCredential">  <message clientCredentialType="Certificate" establishSecurityContext="false" />  </security>  </binding>  </ws2007HttpBinding>  <ws2007FederationHttpBinding>  <binding name="ws2007FederationNoSct">  <security mode="TransportWithMessageCredential" >  <message establishSecurityContext="false" >  <issuer address=".../Issue.svc/trust/13/usernamemixed" binding="ws2007HttpBinding" bindingConfiguration="usernameMixed" />  </message>  </security>  </binding>  </ws2007FederationHttpBinding>  </bindings>  <client>  <endpoint address="...Service.svc/ws2007Federation" binding="ws2007FederationHttpBinding" bindingConfiguration="ws2007Federation" contract="Contract" name="Name"/>  </client>  </system.serviceModel>  Papildus koda izmaiņas:  client.Credentials.UserName.UserName = "User"  client.Credentials.UserName.Password = "password" |

# Bankas identifikācijas piegādātāja izstrādes vadlīnijas

Bankas identifikācijas piegādātāja (adaptera) izstrādei jāizmanto MS *VisualStudio template*, kas pieejams VISS piemēru bibliotēkā.

## Identifikācijas adaptera MS VisualStudio template apraksts

Identifikācijas adapteriem (bankas, e-pasts utt) var izmantot specializētu VisualStudio šablonu, kas darbojas kā “proxy” no identifikācijas piegādātāja uz vienotās pieteikšanās moduli. Adaptera koda apraksts pieejams 2. pielikumā.

## Bankas adaptera specifikācija

VISS infrastruktūrā veidojamamajiem bankas adapteriem ir jānodrošina šādu operāciju izpildi izmantojot HTTP komandas:

1. Autentifikācijas pieprasījums

GET url?parameters HTTP/1.1

CT=string - obligāts parametrs, norada uz autentifikācijas kontekstu, jāatgriež atpakaļ.

1. Autentifikācijas atgriešana

POST url?parameters HTTP/1.1

PK=string – obligāts parametrs, personas kods;

FN=string – obligāts parametrs, vārds;

LN=string – obligāts parametrs, uzvārds;

CT=string – obligāts parametrs, konteksta informācijā, kas tika sūtīta pieprasījumā.

1. Autentifikācijas pieprasījuma atcelšana

POST url?parameters HTTP/1.1

Cancel=true

GET izmanto query string, POST izmanto body

## Univerālā bankas adaptera specifikācija

Adapteris nodrošina universālu saskarni lietotāju autentifikācijai.

Adaptera specifikācija:

1. Informācijas apmaiņa starp identifikācijas nodrošinātāju un universālo adapteri tiek veikta, izmantojot HTTPS protokolu.
2. Portāls Latvija.lv sagatavo pieprasījumu „4002”, kas nodrošina lietotāja piekļuvi autentifikācijas nodrošinātāja autentifikācijai. Pieprasījums ved uz interneta lapas adresi, kura norādīta konfigurācijas failā.
3. Autentifikācijas nodrošinātājs veic pilnu lietotāja autentifikāciju.
4. Autentifikācijas nodrošinātājam jāsaņem lietotāja piekrišana personas datu nosūtīšanai uz portālu Latvija.lv:
   1. Ja lietotājs piekrīt personas datu nosūtīšanai, autentifikācijas nodrošinātājs sagatavo pieprasījumu „3002”, kas nodrošina lietotāja piekļuvi LVP. Pieprasījums tiek nosūtīts uz interneta lapas adresi: <https://www.latvija.lv/Default.aspx> kuru uztur Valsts reģionālās attīstības aģentūra;
   2. Ja lietotājs nepiekrīt personas datu nosūtīšanai, Internetbanka pāradresē lietotāju uz interneta lapas adresi: https://www.latvija.lv/Default.aspx, kuru uztur Valsts reģionālās attīstības aģentūra;
   3. Atbilde tiek nodota ar POST metodi.
5. Sūtījumu parakstīšana un parakstu atšifrēšana:
   1. Digitālā paraksta iegūšanai tiek izmantoti sekojoši lauki: type, version, sender\_id, nonce, info, user, date, time. Ja nosūtāmie vai saņemamie dati nesatur kādu no laukiem, tas netiek izmantots. Lauku kārtība ir atbilstoši norādītajai kārtībai;
   2. Kodējumam izmanto UTF-8 kodējumu;
   3. Digitālā paraksta МАС008 vērtības aprēķins tiek veikts izmantojot publiskās atslēgas RSA algoritmu un SHA-1 "hash" algoritmu. Aprēķinā tiek ņemta vērā pieprasījuma parametru garums, tā saucamie pieprasījuma tukšie lauki:

MAC008(x1,x2,...,xn) := RSA( SHA-1(x1| | x2 | |... | |xn),e,n)

kur:

| |– Simbolu rindas konkatenācija;

x1, x2, ..., xn – Pieprasījumа parametri;

е, n – RSA parametri;

* 1. Iegūtā simbolu virkne tiek parakstīta ar privāto atslēgu lietojot RSA un SHA-1 funkcijas;
  2. Privātajai atslēgai jābūt 1024 bitus garai;
  3. Iegūtā vērtība ir pieprasījuma elektroniskais paraksts;
  4. Aprēķinātā digitālā paraksta vērtība tiek pārveidota simbolu virknē, izmantojot BASE64 kodu, un nosūtīta darījuma pretējai pusei pieprasījuma parametrā “signature”.

1. Pieprasījuma saņēmējs pārbauda, vai pieprasījuma dati atbilst parakstam:
   1. Ģenerē saņemtajam pieprasījumam atbilstošo simbolu virkni izmantojot algoritmu, kas aprakstīts 5.punktā;
   2. Lietojot Publisko atslēgu, salīdzina iegūto simbolu virkni ar saņemto elektronisko parakstu.
2. Pieprasījumu parakstīšanas un paraksta verifikācijas funkcijas ir OpenSSL bibliotēkas vai analogas bibliotēkas funkcijas.

12.tabula

Nododamie parametri

| Nosaukums | Tips | obligāts | Apraksts |
| --- | --- | --- | --- |
| type | string(4) | jā | Pieprasījumа tips (4002) |
| version | string(3) | jā | Izmantotais paraksta algoritms. Konstante „008” |
| sender\_id | string(15) | jā | Uzņēmuma identifikators |
| nonce | string(50) | jā | Unikāla atslēga, kuru ģenerē pieprasījuma nosūtītājs (izmanto, lai garantētu operācijas svaigumu) |
| returnURL | string(60) | jā | URL, uz kuru jāsūta atbilde |
| charset | string(10) | nē | Ziņojuma kodējums. Neobligāts parametrs.  Pieļaujams ISO-8859-1 (noklusētā vērtība) vai UTF-8 |
| signature | string(300) | jā | Digitālais paraksts |

13.tabula

Saņemamie parametri

| Nosaukums | Tips | obligāts | Apraksts |
| --- | --- | --- | --- |
| type | string(4) | jā | Pieprasījumа tips (3002) |
| version | string(3) | jā | Izmantotais paraksta algoritms. Konstante „008” |
| user | string(16) | jā | Lietotāja identifikators |
| date | string(10) | jā | Paketes ģenerācijas datums. Datuma formāts „dd.MM.yyyy” |
| time | string(8) | jā | Paketes ģenerācijas laiks. Laika formāts „HH:mm:ss” |
| sender\_id | string(15) | jā | Bankas identifikators |
| info | string(300) | jā | Lauks, kas satur autentificējamās personas personīgos datus – uzvārdu un vārdu, atdalītus ar tukšuma zīmi, un personas kodu, atdalītu ar „;”. Par personas uzvārdu tiek uzskatīts teksts līdz pēdējai tukšuma zīmei pirms „;”. Personas vārds ir pēdējais vārds no pēdējās tukšuma zīmes pirms „;” līdz „;”.  „UZVARDS VARDS;PERSONAS\_KODS” |
| charset | string(10) | nē | Ziņojuma kodējums. Neobligāts parametrs.  Pieļaujams ISO-8859-1 (noklusētā vērtība) vai UTF-8 |
| signature | string() | jā | Digitālais paraksts |

## HTTP plūsmas piemērs autentificējoties ar banku

Piemērs izmantojot Uri

|  |  |
| --- | --- |
| Item | URI |
| Bankas adapters | https://host/bankadapter/ |
| Banka | https://bank/ |

**Solis 1. Pieprasījums uz bankas adapteri**

GET [https://host/bankadapter/](https://host/bankadapter/Default.aspx)?wa=wsignin1.0&wtream=urn:realm HTTP/1.1

**Solis 2. Pāradresācija uz banku**

GET <http://bank/?CT=0DFA559C-81DD-4D38-9D63-4BD358B128E0>

**Solis 3. Bankas lietotāja interfeiss**

[Implementation Specific Traffic]

**Solis 4. Atgriešana no bankas**

HTTP/1.1 200 OK

…

<html xmlns="https://www.w3.org/1999/xhtml">

<head>

<title>Working...</title>

</head>

<body>

<form method="post" action="<https://host/bankadapter/SignIn.aspx>">

<p>

<input type="hidden" name="PK" value="1111111111" />

<input type="hidden" name="CT" value="0DFA559C-81DD-4D38-9D63-4BD358B128E0" />

<input type="hidden" name="FN" value="Custom" />

<input type="hidden" name="LN" value="Tester" />

<button type="submit">POST</button> <!-- included for requestors that do not support javascript -->

</p>

</form>

<script type="text/javascript">

setTimeout('document.forms[0].submit()', 0);

</script>

</body>

</html>

**Solis 5. POST**

POST <https://host/bankadapter/SignIn.aspx> HTTP/1.1

…

PK=111111111

CT=0DFA559C-81DD-4D38-9D63-4BD358B128E0

FN=Custom

LN=Tester

**Solis 6. Testa adapteris atgriež talonu**

HTTP/1.1 200 OK

…

<html xmlns="https://www.w3.org/1999/xhtml">

<head>

<title>Working...</title>

</head>

<body>

<form method="post" action="https://resource">

<p>

<input type="hidden" name="wa" value="wsignin1.0" />

<input type="hidden" name="wresult" value="&lt;RequestSecurityTokenResponse&gt;...&lt;/RequestSecurityTokenResponse&gt;" />

<button type="submit">POST</button> <!-- included for requestors that do not support javascript -->

</p>

</form>

<script type="text/javascript">

setTimeout('document.forms[0].submit()', 0);

</script>

</body>

</html>

## Autentifikācijas datu saņemšana no bankas izmantojot tiešo linku

Dažreiz nepieciešams pārsūtīt autentifikācijas informāciju no bankas uz portālu (noteiktu portāla sadaļu) izmantojot banka internetbankā pieejamo linku, sk. 4. attēla.



4. attēls. Iespēja izsaukt latvija.lv portāla e-pakalpojumu ar autentifikāciju

Autentifikācijas datu nodošana autentifikācijas adapteram notiek izmantojot saskaņoto protokolu. Izņēmums ir izsaukuma konteksta nodrošināšana (wctx). Gadījumā, kad portāls iniciē autentifikācijas datu pieprasījums, adapteris pārsuta kontekstu uz banku un saņēm nemainītā veida atpakaļ. Bet gadījumā, kad bankas internetbanka iniciē portāla izsaukums, notiek konteksta datu formēšana atbilstoši šādam kodam:

CT=**URLencoded**(

pr=wsfederation

&rm=**URLencoded**(<http://www.latvija.lv/sts>)

[&cx=**URLencoded**(appContext)], kur

appContext=**URLencoded**(

pr=wsfederation

&rm=**URLencoded**(<http://www.latvija.lv/2.0>)

[&cx=**URLencoded**(/lv/epakalpojumi/EP00)]

Rezultātā jābūt šādam kontekstam un notiek autentifikācijas datu nodošana uz 0.e-pakalpojumu:

CT=rm%3Dhttp%253A%252F%252Fwww.latvija.lv%252Fsts%26pr%3Dwsfederation%26cx%3Drm%253Dhttps%25253A%25252F%25252Fwww.latvija.lv%25252F2.0%2526pr%253Dwsfederation%2526cx%253D%25252Flv%25252FEpakalpojumi%25252FEP00

Pielikumā ir HTML lapas teksts kas nodrošina SignIn konteksta izveidošanu.

## HTTP plūsmas piemērs autentificējoties no bankas

Piemērs, izmantojot Uri

|  |  |
| --- | --- |
| Item | URI |
| Bankas adapters | https://host/bankadapter/ |
| Banka | https://bank/ |

**Solis 1. Bankas lietotāja interfeiss**

[Implementation Specific Traffic]

**Solis 2. Atgriešana no bankas**

HTTP/1.1 200 OK

…

<html xmlns="https://www.w3.org/1999/xhtml">

<head>

<title>Working...</title>

</head>

<body>

<form method="post" action="<https://host/bankadapter/SignIn.aspx>">

<p>

<input type="hidden" name="PK" value="1111111111" />

<input type="hidden" name="CT" value=" rm%3Dhttp%253A%252F%252Fwww.latvija.lv%252Fsts%26pr%3Dwsfederation%26cx%3Drm%253Dhttps%25253A%25252F%25252Fwww.latvija.lv%25252F2.0%2526pr%253Dwsfederation%2526cx%253D%25252Flv%25252FEpakalpojumi%25252FEP00" />

<input type="hidden" name="FN" value="Custom" />

<input type="hidden" name="LN" value="Tester" />

<button type="submit">POST</button> <!-- included for requestors that do not support javascript -->

</p>

</form>

<script type="text/javascript">

setTimeout('document.forms[0].submit()', 0);

</script>

</body>

</html>

**Solis 3. POST**

POST <https://host/bankadapter/SignIn.aspx> HTTP/1.1

…

PK=111111111

CT= rm%3Dhttp%253A%252F%252Fwww.latvija.lv%252Fsts%26pr%3Dwsfederation%26cx%3Drm%253Dhttps%25253A%25252F%25252Fwww.latvija.lv%25252F2.0%2526pr%253Dwsfederation%2526cx%253D%25252Flv%25252FEpakalpojumi%25252FEP00

FN=Custom

LN=Tester

**Solis 4. Testa adapters atgriež talonu**

HTTP/1.1 200 OK

…

<html xmlns="https://www.w3.org/1999/xhtml">

<head>

<title>Working...</title>

</head>

<body>

<form method="post" action="https://resource">

<p>

<input type="hidden" name="wa" value="wsignin1.0" />

<input type="hidden" name="wctx" value=" rm%3Dhttps%253A%252F%252Fwww.latvija.lv%252F2.0%26pr%3Dwsfederation%26cx%3D%252Flv%252FEpakalpojumi%252FEP00" />

<input type="hidden" name="wresult" value="&lt;RequestSecurityTokenResponse&gt;...&lt;/RequestSecurityTokenResponse&gt;" />

<button type="submit">POST</button> <!-- included for requestors that do not support javascript -->

</p>

</form>

<script type="text/javascript">

setTimeout('document.forms[0].submit()', 0);

</script>

</body>

</html>

# Vienota pieteikšanas moduļa izmantošana

Vienoto pieteikšanas moduļa izmantošana detalizēti ir aprakstīta [4] sadaļā 8.5. WS-Federation Passive Profile.

# Pielikums. HTML lapas saites izveidošana autentifikācijas datu saņemšanai no bankas

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>IdpInitiated SignOn Context Generator</title>

<style type="text/css">

.textbox {width: 500px;}

</style>

<script type="text/javascript">

function SubmitMeADFS(stsId, stsProt, appId, appProt, appCtx) {

if (!stsId || !appId) {

alert("The textbox should not be empty");

return;

}

var ApplicatoinId = encodeURIComponent(appId);

var ApplicatoinContext = encodeURIComponent(appCtx);

var ApplicatoinProtocol = encodeURIComponent(appProt);

var LvpStsId = encodeURIComponent(stsId);

var stsCtx = "RPID=" + ApplicatoinId;

if (ApplicatoinProtocol = "wsfederation") {

stsCtx = stsCtx + "&wctx=" + ApplicatoinContext;

}

else {

alert("Not Supported protocol.");

return

}

var LvpStsContext = encodeURIComponent(stsCtx);

var bankCtx = "RPID=" + LvpStsId;

if (ApplicatoinProtocol = "wsfederation") {

bankCtx = bankCtx + "&wctx=" + LvpStsContext;

}

else {

alert("Not Supported protocol.");

return

}

document.getElementById('lblDisplay').innerHTML = bankCtx;

document.getElementById('lblBank').innerHTML = "SignIn.aspx?ctx=" + encodeURIComponent(bankCtx)

}

function SubmitMeACS(stsId, stsProt, appId, appProt, appCtx) {

if (!stsId || !appId) {

alert("The textbox should not be empty");

return;

}

var ApplicatoinId = encodeURIComponent(appId);

var ApplicatoinContext = encodeURIComponent(appCtx);

var ApplicatoinProtocol = encodeURIComponent(appProt);

var LvpStsId = encodeURIComponent(stsId);

var stsCtx = "rm=" + ApplicatoinId + "&pr=" + ApplicatoinProtocol;

if (ApplicatoinContext) {

stsCtx = stsCtx + "&cx=" + ApplicatoinContext;

}

var LvpStsContext = encodeURIComponent(stsCtx);

var LvpStsProt = encodeURIComponent(appProt);

var bankCtx = "rm=" + LvpStsId + "&pr=" + LvpStsProt + "&cx=" + LvpStsContext;

document.getElementById('lblDisplay').innerHTML = bankCtx;

document.getElementById('lblBank').innerHTML = "SignIn.aspx?ctx=" + encodeURIComponent(bankCtx)

}

</script>

</head>

<body style="background-color: blue; color: white">

<p><br />

<strong>Context Generator for IDP Initiated Signon </strong></p>

<table>

<tr>

<td>

LVP.STS Identifier

<label style="color:red">\*</label>

</td>

<td>

<input type="text" value="http://www.latvija.lv/sts" class="textbox" id="txtLVPSTSRPID" />

</td>

<td>

<select id="optLVPSTSPROT" >

<option value="wsfederation">Ws-Federation</option>

</select>

</td>

</tr>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>

App Identifier

<label style="color:red">\*</label>

</td>

<td>

<input type="text" value="https://www.latvija.lv/2.0" class="textbox" id="txtLVPRPID" />

</td>

<td>

<select id="optLVPPROT" >

<option value="wsfederation">Ws-Federation</option>

</select>

</td>

</tr>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>App Context</td>

<td>

<input type="text" value="/lv/Epakalpojumi/EP00" class="textbox" id="txtContext" />

</td>

<td></td>

</tr>

</table>

<br />

<strong>

<input type="button" onclick="SubmitMeACS(document.getElementById('txtLVPSTSRPID').value, document.getElementById('optLVPSTSPROT').value, document.getElementById('txtLVPRPID').value, document.getElementById('optLVPPROT').value, document.getElementById('txtContext').value)" value="Generate URL" /></strong>

<br />

<hr /><strong>Context:</strong><br /><br />

<label id="lblDisplay"></label><br /><br />

<strong>Bank Sample:</strong><br /><br />

<label id="lblBank"></label>

</body>

</html>

# Pielikums. Testa adaptera pamata kodi

// Izveidojam klasi bāzētu uz klases BankAdapterBase

public class MyBankAdapter : BankAdapterBase {

// Piešķiram autentifikācijas metodi formātā URN:IVIS:100001:AM.BANK-<name>

protected override string AuthenticationMethod {

get { return "URN:IVIS:100001:AM.BANK-MYBANKA"; }

}

// Funkcija kas pārbauda, ka tas ir atbilde no testa bankas

// Mūsu gadījumā atbilde atnāk kā POST

private static bool IsBankResponse(HttpRequestBase request) {

return request.HttpMethod == "POST";

}

// Izveidojam funkciju kas pārbauda atbildi no bankas un atgriež AuthorizationState, ja lietotājs bija autentificēts, vai CancelAuthorizationStatem ja tika autentifikācijā bija atcelta

public override IAuthorizationState ProcessUserAuthorization(HttpRequestBase request = null) {

if (request == null) {

request = this.GetRequestFromContext();

}

if (!IsBankResponse(request)) {

return null;

}

if (request.Form["Cancel"] == null) {

return new AuthorizationState(request.Form["PK"], request.Form["FN"], request.Form["LN"]) { Context = request["CT"] };

}

else {

return new CancelAuthorizationState() { Context = request["CT"] };

}

}

/// <inheritdoc/>

public override BankRequestSpecification GetBankRequestSpecification(string context, Uri returnTo) {

if (returnTo == null) {

returnTo = new Uri(this.GetRequestFromContext().Url.GetLeftPart(UriPartial.Path));

}

// Izveidojam funkciju kas veido pieprasījumu uz banku

// Klase BankRequestSpecification izmantojas pieprasījuma izveidei un aizpildās no // bankas pieprasījumā specifikācijas. Mūsu gadījumā metode GET, ar parametru CT

var myBankUrl = this.GetRequestFromContext().Url.GetLeftPart(UriPartial.Path);

myBankUrl = myBankUrl.Substring(0, myBankUrl.LastIndexOf('/')) + "/STS/Abc.Bank.MyBankAdapter.html";

return new BankRequestSpecification("GET", new Uri(myBankUrl), new KeyValuePair<string, string>[] { new KeyValuePair<string, string>("CT", context), new KeyValuePair<string, string>("ret", returnTo.AbsoluteUri) });

}

}

Datnē SignIn.apsx tiek pielietota testa adaptera klase

private static readonly MyBankAdapter client = new MyBankAdapter();

protected void Page\_Load(object sender, EventArgs e) {

// Pārbaudām vai izsaukums nāk no bankas

IAuthorizationState authorization = client.ProcessUserAuthorization(null);

if (authorization == null) {

// Save current passive context

if (!this.PassiveContext.IsRequest) {

throw new Microsoft.IdentityModel.SecurityTokenService.RequestFailedException((string)this.GetGlobalResourceObject("SR", "ID002"));

}

// Tiek saglabāts sakuma pieprasījums (kookies) un noģenerēta unikālā konteksta vērtība

var ctx = Abc.IdentityModel.Web.PassiveAuthentication.SavePassiveContext();

// Ar medoti GetBankRequestSpecification jāizveido pieprasījumu pie bankas.

// Piemēram parametrs ‘nonce’ Swedbankai.

// returnTo parametrs nav obligāts, domāts tam ka banka var atgriezt atbildi uz citu URL.

// Kick off authorization request

client.RequestUserAuthorization(ctx, null);

}

else {

// No banka atbildes dabūjam to pašu contexta vērtību ko sūtījām uz banku.

// ‘nonce’ Swedbankai

// Restore passive context

var ctx = authorization.Context;

Abc.IdentityModel.Web.PassiveAuthentication.RestorePassiveContext(ctx, true);

if (!this.PassiveContext.IsRequest) {

throw new Microsoft.IdentityModel.SecurityTokenService.RequestFailedException((string)this.GetGlobalResourceObject("SR", "ID002"));

}

// No banka atbildes (personas kods, vārds, uzvārds) dabūjam Principal un autentificējam pēc ta.

if (authorization is AuthorizationState) {

var principal = client.Authorize(authorization);

PassiveAuthentication.SignIn(new SessionSecurityToken(principal));

}

else {

PassiveAuthentication.ReturnError(null);

}

}

}