Standardi za zagotovitev varnosti spletnih storitev

Jurij Laznik, Matjaž B. Jurič, Ivan Rozman

Povzetek

Spletne storitve so razmeroma nova tehnologija, ki naj bi največji razmah uporabe doživele v letu 2005. Tehnologija spletnih storitev deluje kot integrator nekompatibilnih informacijskih sistemov, saj za povezovanje le-teh ni več pomembna izbira računalniške arhitekture, operacijskega sistema in programskega jezika. Spletne storitve tako prinašajo v integracijo informacijskih sistemov šibko sklopljenost. Ravno tako kot tehnologija spletnih storitev so pomembni tudi varnostni standardi v spletnih storitvah. Ti so namreč zagotovilo za množično uporabo spletnih storitev. Tako kot se intenzivno razvijajo varnostni standardi, se pojavljajo tudi alternative varnostnim standardom, ki delujejo pod okriljem organizacije WS-I. Spletne storitve in varnostni standardi so že pred napovedmi dosegle kritično maso, kar jim daje prednost pred konkurenco na področju porazdeljenega računalništva.

Abstract

Standards for Assuring Web Services Security

Web services are a young technology that should reach the peak of usage in 2005. The technology of web services acts as integrator of incompatible informatics systems. The choice of computer architecture, operating system or programming language is no longer needed. In this way web services bring loose coupling into technology of informatics systems integration. Security standards in web services are no less important than web services technology itself. They are a warranty of web services prosperity. Security standards in web services are rapidly developing and so are alternatives. One such alternative is the organization WS-I. Web services have reached their critical mass before predictions and are some steps ahead of competition in the area of distributed computing.

1 UVOD

Za prvi pojav spletnih storitev lahko štejemo konec leta 1999, ko je podjetje Hewlett – Packard predstavilo izdelek e-Speak [1]. To je bil prvi primerek komercialne verzije spletnih storitev. Od takrat se spletne storitve izjemno hitro razvijajo. Po analizah podjetja Gartner [2] bodo spletne storitve množično uporabo doživele leta 2005.

Prednost spletnih storitev je predvsem v šibki sklopljenosti aplikacij, neodvisnosti od uporabe programskih jezikov in tudi neodvisnosti od uporabe operacijskih sistemov.

Namen spletnih storitev je ponuditi učinkovito in poenoteno komunikacijo med več strankami. Najpomembnejše področje, ki ga podjetja in ponudniki spletnih storitev pri razvoju svojih aplikacij zahtevajo, je varnost. Leto 2002 je tako prineslo velik zagon na področju dela s spletnimi storitvami. Med podjetji, ki aktivno sodelujejo pri nastajanju varnostnih standardov, lahko najdemo vsa zveneča imena v svetu informatike, kot so IBM, Microsoft, Oracle, WebMethods, BEA, Intel in drugi. Krovni organizaciji, ki skrbita za standardizacijo varnostnih standardov v spletnih storitvah, pa staW3C (World Wide Web Consortium) [3] in OASIS (Organization for the Advancement of Structured Information Standards) [4].

V tem članku bomo pregledali trenutno najbolj aktualne varnostne standarde, možnosti uporabe leteh in alternative tem standardom.

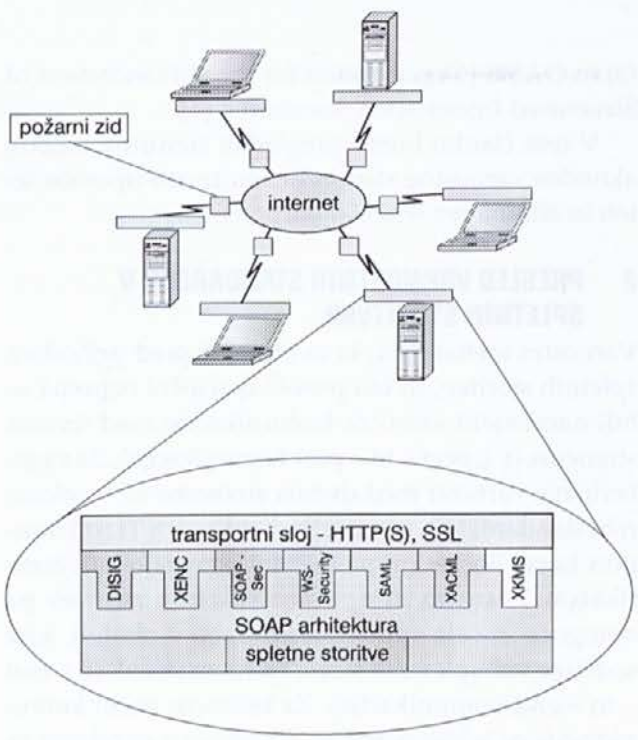
2 PREGLED VARNOSTNIH STANDARDOV V SPLETNIH STORITVAH

Varnostni mehanizmi, ki so obstajali pred prihodom spletnih storitev, so bili preveč specifični oziroma so bili namenjeni samo za komunikacijo med dvema stranema (t. i. peer – to – peer komunikacija). Za zagotavljanje varnosti med dvema stranema so v splošni rabi standardi SSL (Secure Socket Layer), TLS (Transport Layer Socket) in pa mehanizem digitalnih certifikatov. Zasnova in uporaba spletnih storitev pa omogoča in celo spodbuja uporabo v okoljih, kjer sodeluje več vpletenih strani oziroma strank (t. i. end – to – end komunikacija). Za takšnem način komuniciranja pa je bilo potrebno razviti nove standarde za varno izmenjavo podatkov med vpletenimi strankami.

Varnostni standardi se umeščajo nekje med transportni sloj in pa SOAP sloj in predstavljajo tako imenovano nadgradnjo SOAP arhitekture. Iz slike 1 lahko razberemo, da delujejo varnostni standardi kot nekakšni vmesniki, kar omogoča razvijalcem veliko fleksibilnost sistemov, hkrati pa je odprta tudi možnost za dodajanje na novo definiranih standardov. Razberemo lahko tudi, da varnostni standardi niso nadomestna tehnologija, ki bi izpodrinila obstoječe tehnologije, pač pa so samo gradnik v zagotavljanju večje varnosti informacij. Skupaj z varnostnimi standardi namreč lahko uporabljamo tudi že vse obstoječe tehnologije, ki so na voljo: HTTP(S), SSL, požarni zidovi itd.

V nadaljevanju si bomo pogledali varnostne standarde, ki so namenjeni splošni uporabi v jeziku XML. Takšna standarda sta dva, in sicer digitalno podpisovanje XML dokumentov (DSIG) in pa enkripcija XML dokumentov (XENC). Ogledali pa si bomo tudi varnostne standarde, ki so namenjeni izključno rabi v spletnih storitvah. Ti so:

- SOAP-Sec predstavlja standard za digitalno podpisovanje SOAP sporočil,
- WS-Security je nadgradnja standarda SOAP-Sec, ki vsebuje tudi dodatne funkcionalnosti. Najpomembnejša med njimi je enkripcija SOAP sporočil.
- SAML in XACML sta standarda, ki definirata XML jezika za učinkovito izmenjavo podatkov o avtentikaciji in avtorizaciji ter upravljanju dostopov do sredstev prek spleta.
- XKMS standard je namenjen upravljanju in delu z javnimi in skrivnimi ključi.



Slika 1: **Umestitev varnostnih standardov**

Pri naštevanju standardov smo večkrat zapisali besedi »SOAP sporočilo« zato namenimo nekaj stavkov še definiciji SOAP standarda in njegovemu namenu.

SOAP je protokol, ki deluje na osnovi jezika XML in omogoča izmenjavo strukturnih in tekstovnih informacij v porazdeljenem okolju. Dejansko je to protokol, ki definira format sporočil za komunikacijo med dvema računalnikoma. Takšnim sporočilom torej pravimo SOAP sporočila. Ena od možnosti uporabe SOAP protokola je tudi proženje spletnih storitev prek lokalne mreže in tudi medmrežja.

2.1 Digitalno podpisovanje XML dokumentov

Standard digitalnega podpisovanja XML dokumentov ali z uradno označbo DSIG (Digital SIGnature) [5] ima namen podpisovanja dela ali pa celote XML dokumenta. XML dokumente običajno hočemo digitalno podpisati takrat, kadar želimo, da prejemnik zagotovo sprejme nespremenjeni dokument oziroma vsebino. Digitalni podpis v XML dokumentu tako zagotavlja integriteto podatkov. Standard omogoča podpisovanje tako spletne kot tudi nespletne vsebine. Tako lahko podpišemo razne binarne slike (GIF, JPEG), različne dokumente na spletu, zvok in ostale binarne vsebine.

Standard podpira tri načine vsebovanja digitalnega podpisa:

- **ločen** (detached): digitalni podpis se nahaja zunaj meja XML dokumenta, vendar se sklicuje na reference v dokumentu.
- **ovijajoč** (enveloped): digitalni podpis je vsebovan v XML dokumentu.
- **obdajajoč** (enveloping): najbolj zunanji element XML dokumenta se zamenja. Digitalni podpis obdaja prvotni XML dokument.

Za generiranje digitalnih podpisov obstajata dva mehanizma:

- **direktni:** programer mora s pomočjo programske kode izdelati celoten XML dokument, ki je pri-

pravljen za podpis in ga nato podpisati. Takšen način ima drago vzdrževanje, saj za vsako spremembo potrebujemo programerja, ki pozna tematiko, ki se nanaša na XML dokument, poznati pa mora tudi programski jezik, v katerem je aplikacija napisana.

- **predloge:** s pomočjo predloge definiramo strukturo digitalnega podpisa. Ko pridobimo podatke, ki jih želimo podpisati, podamo te podatke skupaj s predlogo metodi, ki digitalno podpiše XML dokument. Ta način omogoča neprimerno lažje vzdrževanje aplikacije, za vzdrževanje pa potrebujemo le inženirja, ki pozna XML jezik.

Podatke o ključu, s katerim smo digitalno podpisali XML dokument, lahko vključimo v dokument, lahko pa ga tudi po kateri drugi poti dostavimo do prejemnika. Standard določa samo primer, kjer ključ vključimo v XML dokument.

Digitalno podpisovanje XML dokumentov lahko uporabimo pri podpisovanju raznih dokumentov v elektronski obliki, za izdajo različnih potrdil itd. Področja, kjer ima uporaba digitalnega podpisovanja velik potencial, je javna uprava in pa elektronsko bančništvo, vendar naj takoj omenimo, da standard nima točno določenega področja uporabe, ampak je namenjen splošni uporabi.

2.2 Enkripcija XML dokumentov

Standard XENC (Xml ENCryption) [6] je namenjen enkripciji vsebine XML dokumentov. XML dokumente običajno enkriptiramo takrat, kadar želimo prikriti njegovo vsebino ali del vsebine vsem, razen pošiljatelju in prejemniku XML dokumenta. Čeprav je na prvi pogled videti kot zamenjava standarda SSL ali TLS, pa ni tako. Standard XENC je samo eden od gradnikov v arhitekturi zagotavljanja varne izmenjave podatkov med več strankami. Enkriptiramo lahko del XML dokumenta ali pa kar celoten dokument. XML dokument, ki ga želimo enkriptirati, lahko zakodiramo tako, da isti dokument prejme več strank. Stranke lahko nato pregledujejo samo vsebino, za katero imajo ustrezne skrivne ključe.

Mehanizmov za izmenjavo ključev, ki jih potrebujemo za kriptiranje/dekriptiranje, standard ne predpisuje, ampak je to prepuščeno razvijalcem oziroma drugim standardom. Izmenjava ključev lahko poteka po elektronski ali navadni pošti, lahko pa tudi s pomočjo standarda XKMS, ki ga bomo kasneje opisali.

Uporaba standarda XENC je možna tako v javni upravi (potrjevanje vozniških dovoljenje), kot tudi v elektronskem bančništvu (potrdila o stanju na transakcijskem računu). Navedeni primeri so namenjeni zgolj prikazu primerov uporabe standarda, sicer pa je standard namenjen splošni rabi na mnogih področjih informatike.

2.3 Digitalni podpis SOAP sporočil

Standard digitalnega podpisovanja SOAP sporočil (SOAP-Sec) [7] je namenjen poenotenju standardov za digitalno podpisovanje SOAP sporočil. Sama specifikacija standarda SOAP-Sec v veliki meri uporablja že prej omenjeni DSIG standard za digitalno podpisovanje XML dokumentov. Standard uporabljamo iz podobnih razlogov kot standard DSIG. Pri standardu želimo zagotoviti integriteto podatkov XML dokumenta, pri SOAP-Sec pa želimo zagotoviti integriteto SOAP sporočil.

Standard SOAP-Sec se dotika predvsem dveh področij:

- avtentikacija ali kako prepoznamo identiteto pošiljatelja,
- digitalni podpis SOAP sporočila ali kako prepoznati, ali lahko prejetemu sporočilu zaupamo.

Zavedati se namreč moramo, da SOAP protokol temelji na HTTP protokolu. Zahteve za HTTP in s tem tudi SOAP protokol pa potekajo tudi prek meja požarnih zidov (»firewall«). Pri tem pa je zagotavljanje varnosti pri sprejemanju SOAP sporočil poglavitnega pomena.

SOAP sporočilo digitalno podpišemo tako, da v glavo (Header) sporočila zapišemo podatke o digitalnem podpisu. V glavi lahko tudi določimo, kateremu prejemniku je SOAP sporočilo namenjeno. Večkrat se namreč zgodi, da SOAP sporočila potujejo prek več strežnikov, preden prispejo do naslovnika.

Verifikacija SOAP sporočila poteka tako, da najprej iz glave SOAP sporočila izluščimo podatke o digitalnem podpisu, preverimo podpis in se nato odločimo, ali spletno storitev izvedemo ali pa zavrnemo izvršbo in pošljemo poročilo o napaki pošiljatelju.

Digitalno podpisovanje SOAP sporočil lahko uporabimo pravzaprav kjerkoli se srečamo z uporabniki, ki dostopajo do naše spletne storitve prek požarnih zidov.

Omenimo še, da se uporaba standarda SOAP-Sec opušča, saj sta podjetji IBM in Microsoft predstavili standard WS-Security, ki ga bomo opisali v naslednjem podpoglavju in v veliki meri že vključuje specifikacije standarda SOAP-Sec.

2.4 WS-Security

Standard WS-Securty (Web Services Security) [8] je nastal kot nadgradnja standarda SOAP-Sec. Dodane so tudi nekatere nove funkcionalnosti, kot so propagiranje varnostnih žetonov (security tokens propagation), zagotavljanje nespremenjenosti SOAP sporočil (SOAP message integration) in preprečevanje prestrezanja vsebine sporočil (message confidentiality).

Uporaba standarda je primerna, kadar želimo prikriti vsebino SOAP sporočila nepooblaščenim uporabnikom. SOAP arhitektura in zasnova omogočata, da pri proženju spletnih storitev sodeluje več strežnikov, kar daje nepooblaščenim uporabnikom možnost prestrezanja SOAP sporočil in ponovno oddajanje le-teh v kasnejših časovnih intervalih.

Standard definira digitalni podpis in enkripcijo SOAP sporočil ter izmenjavo uporabniških imen in gesel. Pri izmenjavi uporabniških imen in gesel še omenimo, da se le-ti običajno prenašajo po medmrežju kot navadno besedilo. To pa predstavlja varnostno luknjo v sistemu, zato je priporočena uporaba dodatne zaščite (kodiranje uporabniških imen in gesel) in pa uporaba standarda SSL za izmenjavo podatkov med dvema točkama.

Razvijalci standarda priporočajo uporabo dodatnih pristopov za učinkovito varno izmenjavo SOAP sporočil. Takšni pristopi so naslednji:

- časovna znamka: SOAP sporočilu prilepimo časovno znamko oziroma poročilo o tem, kdaj je bilo SOAP sporočilo oddano.
- številka sekvence: SOAP sporočilo vsebuje zapis o zaporedni številki, ki se mora ujemati na strežniku. Primer: številka sporočila ne sme biti manjša, kot je številka na strežniku.
- časovna omejitev: V SOAP sporočilo zapišemo, do kdaj je digitalni podpis SOAP sporočila veljaven oziroma kdaj je bilo SOAP sporočilo podpisano.

Standard WS-Security lahko uporabimo pri spletnih storitvah, ki zahtevajo določen nivo avtentikacije. Takšen primer je elektronsko poslovanje prek medmrežja. Naslednji primer je uporaba pri komunikaciji med več strankami, recimo podjetji, ki trgujejo med svojimi enotami ali z drugimi podjetji. Predstavili smo samo nekaj primerov uporabe, vendar ima standard širok spekter uporabe, podobno kot standard SOAP-Sec.

2.5 SAML in XACML

Standarda SAML (Security Assertion Markup Language) [9] in XACML (Extensible Access Control Markup Language) [10] sta nastala iz potrebe po standardizaciji upravljanja dostopov do različnih sredstev in upravljanju dostopov do različnih podatkov. Standard SAML predstavlja XML jezik za učinkovito izmenjavo podatkov o avtorizaciji in avtentikaciji, standard XACML pa predstavlja XML jezik za izmenjavo podatkov glede dostopov do sredstev prek spleta.

Standarda uporabimo takrat, ko želimo imeti podatke o avtentikaciji in avtorizaciji ter dostopih do različnih sredstev na enem mestu. Običajno dostop do takšnih informacij ponudimo prek spletne storitve in z uporabo standardov SAML in XACML.

Področja uporabe standarda SAML so predvsem tri:

- enkratna prijava oziroma registracija (Single Sign On): Uporabnik se registrira samo na enem spletnem mestu, na sorodnih straneh pa se mu ni potrebno ponovno registrirati.
- porazdeljene transakcije (Distributed Transactions): Primer uporabe porazdeljene transakcije je nakup avtomobila prek spleta. Uporabnik vnese pri prodajalcu avtomobilov svoje podatke, prodajalec pa doda podatke o prodanem avtomobilu. Ko želi lastnik avtomobila skleniti zavarovalno polico, ima zavarovalnica že vse želene podatke na voljo.
- storitev avtorizacije (Authorization Service): Primer uporabe storitve avtorizacije sta dve podjetji, kjer lahko nabavni referent prvega podjetja nabavlja izdelke glede na svoje pristojnosti. Pristojnosti se ugotavljajo na podlagi SAML zahteve.

Implementacije standarda SAML delujejo po načelu »zahteva – odgovor«. Tako poznamo tri načine podajanja zahtev, ki so:

1. zahteva po potrditvi avtentikacije: Uporabnik pošlje pooblaščeni organizaciji zahtevo po potrditvi avtentikacije, pooblaščena organizacija pa vrne uporabniku odgovor, ki vsebuje podatke o tem, ali je avtentikacija uspela ali ne.
2. zahteva po lastnostih: Uporabnik pošlje pooblaščeni organizaciji zahtevo po lastnostih želenega objekta. Pooblaščena organizacija pošlje uporabniku odgovor, v katerem izjavi, da je objekt v zahtevi povezan z naštetimi lastnostmi, poleg odgovora pa uporabnik prejme tudi vrednosti lastnosti.
3. zahteva po avtorizaciji: Pooblaščeni organizaciji uporabnik pošlje zahtevo po avtorizaciji, kjer želi izvedeti ali ima specificirani objekt dostop vnaprej določenega tipa do sredstev, do katerih želi uporabnik dostopati. Pooblaščena organizacija pošlje

odgovor, v katerem izjavi, da ima objekt v podani zahtevi odobren ali zavrnjen dostop do želenih sredstev.

Standarda SAML in XACML sta tesno povezana med seboj, saj standard XACML uporablja za podajanje zahtev in prejemanje odgovorov SAML zahteve in odgovore. Standard XACML je namenjen upravljanju in dodeljevanju pravic spletnih storitvam in objektom na osnovnem nivoju t. i. »fine-grained control«.

Razvijalci standarda XACML vidijo možnost uporabe standarda predvsem na naslednjih področjih:

a) evidenca zdravstvenih kartotek

Računalniško vodenje zdravstvenih kartotek je ena najbolj občutljivih področij informatike glede zaupnosti podatkov. Zdravstvene kartoteke namreč vsebuje osebne podatke pacientov, ki bi jih lahko nepooblaščena oseba izkoristila v pacientovo škodo.

b) bančne storitve

Bančne storitve so naslednje precej občutljivo področje, ki je podvrženo nepooblaščenem razkritju osebnih podatkov. Vemo, da zapisi o komitentu banke obsegajo podatke, kot so naslov, telefon, znesek na računu in podatki o limitu ter kreditih. Ti podatki so lahko dostopni delavcu na okencu, medtem ko morajo nekateri ostati prikriti delavcem v drugih oddelkih banke.

c) spletni strežniki

Pri zadnjem primeru se ustavimo še pri internetnih strežnikih, ki so za nas bolj domače področje. S pomočjo standarda XACML lahko določimo, do katerih storitev ima določen uporabnik dostop. S pomočjo jezika XACML lahko celo določimo, da lahko uporabnik izvede storitev, vendar sama storitev na podlagi uporabnikovih podatkov preverja, katere razrede lahko uporabnik izvede. Uporaba jezika XACML se tako pokaže kot prožna in prilagodljiva rešitev za različne probleme in zahteve.

2.6 XKMS

Standard XKMS (Xml Key Management Specification) [11] je namenjen delu in upravljanju z javnimi in skrivnimi ključi. Razdeljen je na dve logično zaključeni enoti, ki sta:

- X-KISS (Xml Key Information Service Specification) obravnava protokol za pridobivanje informacij o ključih in izmenjavo teh informacij prek medmrežja.
- X-KRSS (Xml Key Registration Service Specification) obravnava protokol za registracijo javnih ključev. Kasneje do ključev dostopamo s pomočjo protokola X-KISS.

Standard XKMS uporabimo takrat, kadar želimo imeti ključe centralno organizirane. Stopnjo organizacije si lahko izbiramo poljubno. Ključe lahko upravljamo na nivoju ene organizacije, več organizacij, države in pa globalno.

Protokol X-KISS lahko deluje na treh nivojih glede na želje in potrebe uporabnikov. Prvi nivo predstavlja procesiranja elementa RetrivalMethod v XML dokumentih. Ta element se uporablja v digitalnih podpisih XML dokumentov in SOAP sporočil. Drugi nivo predstavlja nivo storitve iskanja. Ta nivo omogoča uporabnikom iskanje podatkov o ključih. Storitev, ki podatke vrača uporabnikom, ni dolžna zagotavljati avtentičnih podatkov. Zadnji nivo, ki v največji meri izkorišča zmožnosti standarda, je storitev validacije. V tem primeru storitev poišče podatke o zahtevanih ključih in jih pošlje uporabniku. Poleg teh podatkov pa storitev uporabniku ponuja podatke o validaciji ključev in podatke o njihovi avtentičnosti.

Protokol X-KRSS je namenjen registriranju javnih ključev. Omogoča dva načina registracije:

- registracija uporabniško kreiranih ključev: Uporabnik zahteva registracijo javnega ključa pri spletni storitvi, ki opravlja registracije. Spletna storitev lahko v tem primeru zahteva dodatna potrdila o verodostojnosti ključa od uporabnika.
- registracija strežniško kreiranih ključev: Storitev registracije v tem primeru kreira par ključev (javnega in skrivnega) in javnega tudi registrira. Nato oba ključa pošlje uporabniku. Pri uporabi tega načina registriranja ključev moramo predvsem paziti, da uporabljamo dodatne načine zaščite prenosa podatkov prek medmrežja, saj se tako javni kot tudi skrivni ključ prenašata po medmrežju, zato obstaja nevarnost prestrezanja podatkov.

Standard XKMS je primeren predvsem za uporabo v organizacijah, ki želijo imeti javne ključe na enem mestu in tudi opravljati administracijo teh ključev z enega mesta. Prav tako je standard namenjen avtorizacijskim organizacijam, ki skrbijo za izdajo ključev in potrdil o pristnosti teh ključev.

3 ALTERNATIVE VARNOSTNIM STANDARDOM

Razvijalci in snovalci standardov za spletne storitve so se razdelili na dva pola. Prva skupina je mnenja, da se spletne storitve prepočasi razvijajo, druga skupina pa meni, da se spletne storitve razvijajo z ravno pravo

hitrostjo. In ravno tisti, ki pripadajo prvi skupini, so ustanovili novo organizacijo, ki naj bi skrbela za standardizacijo v spletnih storitvah. Naziv novoustvarjene organizacije je WS-I (Web Services Interoperability Organization) [12]. Kot pravijo snovalci organizacije, ki so podjetja, kot so IBM, Microsoft in drugi, njen namen ni postavljati novih standardov, ampak uporabiti najboljše standarde izmed obstoječih. Prav tako želijo ponuditi možnost preverjanja in certificiranja ustreznosti programskih rešitev. Po napovedih podjetja Sun, bo le-to ponudilo platformo J2EE (Java 2 Enterprise Edition) v verziji 1.4 skladno z specifikacijami začetnega področja standardov WS-I, ki ga bomo opisali kasneje.

Organizacija WS-I je izdelala nabor tako imenovanih »profilov«, ki ustrezajo različnim področjem dela s spletnimi storitvami. Pri tem velja omeniti, da so ravno s takšnim klasificiranjem standardov posegli v razvoj standardov, ki nastajajo pod okriljem organizacij OASIS in W3C. Drugi konflikt, ki ga je sprožila organizacija WS-I, pa je problem trženja licenc in avtorskih pravic. Medtem ko organizaciji OASIS in W3C zastopata stališča proste uporabe standardov, pa organizacija WS-I zagovarja licenciranje standardov oziroma plačilo za uporabo standardov. Sicer pa je organizacija WS-I razdelila profile na dve glavni področji, ki sta [13]:

- **začetno področje** je nabor standardov oziroma profilov, ki zagotavljajo osnovno delovanje spletnih storitev;
- **razširjeno področje** je nabor standardov oziroma profilov, ki omogočajo dodatno funkcionalnost spletnim storitvam.

Začetno področje vsebuje naslednje profile:

- **WS-Security**: Specifikacija določa, kako lahko dodajamo informacije o digitalnih podpisih in enkripciji SOAP sporočilom. Specifikacija prav tako določa, kako dodamo SOAP sporočilom podatke o ključih in certifikatih.
- **WS-Policy**: Standard je namenjen pridobivanju informacij o zahtevah glede varnosti. S pomočjo standarda je mogoče pridobiti informacije o tem, katere standarde moramo imeti, katere certifikate podpira spletna storitev in katerim drugim zahtevam moramo zadostiti, če želimo uporabljati spletno storitev.
- **WS-Trust**: Standard bo definiral ogrodje varnostnega modela, ki bo omogočalo varno medsebojno komuniciranje spletnih storitev.
- **WS-Privacy**: Standard bo definiral model, s katerimi bodo lahko uporabniki in spletne storitve izražale nastavitve glede zasebnosti.

Razširjeno področje pa vsebuje naslednje profile:

- **WS-SecureConversation:** Standard bo enolično določal, kako lahko spletna storitev avtenticira SOAP sporočila uporabnika, kot tudi, kako lahko uporabnik avtenticira želeno spletno storitev. Standard bo prav tako omogočal vzpostavljanje varnostnih kontekstov. Za izmenjavo varnih podatkov bo standard uporabljal mehanizme, ki jih zajemata standarda WS-Securty in WS-Trust.
- **WS-Federation:** Specifikacija bo definirala načine, kako s pomočjo standardov WS-Security, WS-Trust in WS-SecureConversation izdelamo scenarije za varne skupine povezav – federacij. Tako bo razvijalec lahko napisal scenarij, ki bo omogočal medsebojno zaupanja vredno sodelovanje med infrastrukturo Kerberos in ostalimi infrastrukturami.
- **WS-Authorization:** Specifikacija bo določala, kako so organizirane posamezne pravice oziroma pravila dostopa do spletnih storitev.

Trenutno na trgu ni izdelkov, ki bi podpirali standarde, ki smo jih omenjali v tem članku, pač pa jih podjetja ponujajo v obliki dodatnih paketov »toolkitov« (tabela 1). Te pakete pa lahko namestimo na skoraj vse znane aplikacijske strežnike: BEA WebLogic, IBM WebSphere, Apache AXIS, JBoss in Microsoft .NET okolje.

Podjetje	Naziv programskega paketa
IBM	Web Services Toolkit (WSTK) [14]
Microsoft	Web Services Enhancements for Microsoft .NET (WSE) [15]
Verisign	Trust Services Integratin Kit (TSIK) [16]
Apache	Apache XML Security [17]

Tabela 1: **Izdelovalci dodatnih paketov za aplikacijske strežnike**

4 SKLEP

V članku smo opisali trenutno stanje na področju varnostnih standardov v spletnih storitvah. Videli smo, da je varnostno področje precej široko in obsega elemente, kot so digitalni podpis in enkripcija ter pravice dostopov do spletnih storitev.

Ugotovili smo tudi, da nekatera podjetja niso zadovoljna z napredkom razvoja standardov, zato je

bilo ustanovljeno več organizacij, ki skrbijo za standardizacijo varnostnih mehanizmov v spletnih storitvah, kar v nekaterih pogledih pospešuje standardizacijo mnogih varnostnih mehanizmov. Glede na to, da podjetje Gartner napoveduje množično uporabo spletnih storitev v letu 2005, lahko pričakujemo, da trenutek, ko bodo spletne storitve množično uporabljane, šele prihaja.

Vsekakor lahko rečemo, da bodo spletne storitve ostale še nekaj časa »na sceni«. Preteklost tehnologij porazdeljenega računalništva (distributed computing) je namreč pokazala, da so te tehnologije v veliki meri odvisne od kritične mase, najpogosteje imenovane s tujko »network effect«. Na kratko to pomeni, da kolikor bolj so spletne storitve uporabljane in razširjene, toliko večja je verjetnost dominacije v tej veji računalništva. In spletne storitve so kritično mase že dosegle in tudi presegle, kar jim zagotavlja občutno prednost pred konkurenti na področju porazdeljenega računalništva.

5 LITERATURA

[1] *Hewlett-Packard, e-Speak:* http://www.hpmiddleware.com/SalSAPI.dll/SaServletEngine.class/products/hp_web_services/default.jsp

[2] *Gartner:* http://www3.gartner.com/Init

[3] *World Wide Web Consortium :* http://www.w3.org/

[4] *OASIS:* http://www.oasis-open.org/

[5] *XML Digital Signature Standard:* http://www.w3.org/TR/xmldsig-core/

[6] *XML Encryption Standard:* http://www.w3.org/TR/xmlenc-core/

[7] *SOAP-Sec Standard:* http://www.w3.org/TR/SOAP-dsig/

[8] *WS-Security Homepage:* http://www-106.ibm.com/developerworks/library/ws-secure/

[9] *Security Assertion Markup Language:* http://www.oasis-open.org/committees/security/

[10] *OASIS eXtensible Access Control Markup Language TC:* http://www.oasis-open.org/committees/xacml/

[11] *XML Key Management Specification (XKMS):* http://www.w3.org/TR/xkms/

[12] *Web Services Interoperability Organization:* http://www.ws-i.org/

[13] *Security in a Web Services World: A Proposed Architecture and Roadmap:* http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwssecur/html/securitywhitepaper.asp

[14] *Web Services Toolkit (WSTK):* http://www.alphaworks.ibm.com/tech/webservicestoolkit

[15] *Web Services Enhancements for Microsoft .NET (WSE):* http://msdn.microsoft.com/webservices/building/wse/default.aspx

[16] *Trust Services Integratin Kit (TSIK):* http://www.xmltrustcenter.org/developer/verisign/tsik/index.htm

[17] *Apache XML Security:* http://ws.apache.org/security/index.html

Jurij Laznik je sistemski inženir v podjetju Hermes SoftLab. Na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru je leta 2001 diplomiral na programu Računalništvo in informatika. Njegova raziskovalna področja zajemajo predvsem spletne storitve, varnost in porazdeljeno procesiranje. Študira na magistrskem podiplomskem študiju informatike na Univerzi v Mariboru.

Matjaž B. Jurič je docent na Inštitutu za informatiko Fakultete za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Ukvarja se predvsem s komponentnim razvojem, integracijo, elektronskim poslovanjem, spletnimi storitvami in optimizacijo zmogljivosti. Je soavtor knjig .NET Serialization Handbook, J2EE Design Patterns Applied, Professional J2EE EAI in Professional EJB (Wrox Press), poglavja v knjigi More Java Gems (Cambridge University Press) in Technology Supporting Business Solutions (Nova Science Publishers). Objavljal je v revijah Web Services Journal, eAI Journal, Java Report, Java Developers Journal in na konferencah, kot so OOPSLA, Java Development, BEA Forum, Wrox Conferences itd. Sodeloval je pri številnih projektih doma in v tujini, med drugim tudi pri razvoju RMI-IIOP, sestavnega dela Java 2 platforme, pred kratkim pa je bil uvrščen v seznam Techiindex Evangelist.

Marjan Heričko je izredni profesor in namestnik predstojnika na Inštitutu za informatiko na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Njegovo raziskovalno-razvojno delo obsega vse vidike objektne tehnologije in komponentnega razvoja, s poudarkom na metodologijah razvoja, sodobnih arhitekturah, metrikah in razvojnih okoljih. Svoje izkušnje je predstavil v številnih prispevkih na domačih in tujih konferencah ter revijah. Je tehnični koordinator aktivnosti Centra za objektno tehnologijo in predsednik konference OTS Objektna tehnologija v Sloveniji. Diplomiral, magistriral in doktoriral je na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru.