KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Giedrius Budrys

Elektroninių tranzakcijų dokumentų pasirašymo žiniatinklio aplinkoje metodo sukūrimas ir tyrimas

Magistro darbas

Darbo vadovas:
doc. dr. N. Morkevičius

KAUNAS
2011

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Giedrius Budrys

Elektroninių tranzakcijų dokumentų pasirašymo žiniatinklio aplinkoje metodo sukūrimas ir tyrimas

Magistro darbas

Recenzentas:
doc. dr. P. Kanapeckas
2011-05-25

Darbo vadovas:
doc. dr. N. Morkevičius
2011-05-23

Atliko:
IFN-9/3 gr. studentas
Giedrius Budrys
2011-05-23

KAUNAS
2011

SANTRAUKA

Šiame darbe apžvelgiami el. paslaugų sistemų, kuriose disponuojama elektroninių tranzakcijų dokumentais, veikimo principai, iškeliamos tokių dokumentų apsaugos problemos, kai tranzakcijos vyksta „debesies“ principu veikiančioje infrastruktūroje: tarp dviejų, tranzakcijoje dalyvaujančių šalių, veikia trečioji, kuria nepasitikima ir nenorima atskleisti per ją persiunčiamos ar jai patikėtos saugoti informacijos konfidencialumo, pažeisti informacijos autentiškumo. Darbe apžvelgiami šiuo metu naudojami el. dokumentų apsaugos žiniatinklio aplinkoje metodai, jų panaudojimo galimybės, privalumai, trūkumai. Toliau pristatomas galimas el. tranzakcijų dokumentų apsaugos žiniatinklio aplinkoje metodas, pagrįstas el. parašo technologija – subjektas, norėdamas, užsakyti kokią nors prekę ar paslaugą internete, užpildo atitinkamą el. tranzakcijos dokumentą, pasirašo savo skaitmeniniu parašu, užšifruoja ir siunčia paslaugos ar prekės tiekėjui. Kitas metodo panaudojimo atvejis – dokumentų pasirašymas žiniatinklio aplinkoje veikiančioje dokumentų valdymo sistemoje. Tokiu būdu užtikrinami visi el. dokumentų perdavimo saugumo reikalavimai: konfidencialumas, vientisumas, autentiškumas, neišsigynimas. Kartu pateikiama tokio metodo specifikacija, programinis sprendimo prototipas, su kuriuo buvo atlikti tyrimai. Pasiūlytas metodas vertintas tiriant jo greitaveiką bei saugos ypatybes ir lyginant su kitais apžvelgtais el. dokumentų apsaugos metodais. Šiame darbe pristatoma tyrimo metodika, rezultatai bei išvados.

Research on method for signing electronic transaction documents in web environment

SUMMARY

This work presents operation principles of e-service systems, based on electronic transaction documents and problems, related to security of transaction documents, when transactions are performed in a cloud based IT infrastructure. In such case, a transaction is performed between a customer and a service provider, while there is a third actor - an untrustworthy cloud service provider between them (for example, application and storage service provider). The work presents a review of existing methods for securing e-transaction documents in web environment, exposing their features, advantages and disadvantages. Then a new possible e-transaction documents' security method is presented: a subject fills a transaction document, signs it with his private key, encrypts it and sends back to service provider. The other possible usage of this method is signing e-documents in web-based document management system. This way all security requirements for e-document based transactions are satisfied (confidentiality, integrity, authentication and non repudiation). Besides that, a specification of this method and a working prototype, used for experiment, presented in this work. The method has been evaluated by running performance tests, analyzing and comparing security features of other, reviewed methods. Finally, the results of all tests and evaluation are presented at the end of this work.

TURINYS

ĮVADAS

Tobulėjant šiandieninėms informacinėms technologijoms, atsiranda galimybė vis daugiau paslaugų teikti internetu, kalbama apie informacinę visuomenę, žinių ekonomiką ir panašiai. Informacinėje visuomenėje galima išskirti dvi pagrindines struktūras, teikiančias savo paslaugas piliečiams – elektroninę valdžią ir elektroninį verslą. Tam, kad šios struktūros būtų sukurtos ir tinkamai funkcionuotų, reikia įdiegti esamas ir kurti naujas informacijos saugos technologijas. Jungiamoji el. valdžios ir el. verslo su visuomene grandis – internetas bei jo prieiga, kuria naudodamasis visuomenės narys gali pats atlikti visas jam reikalingas operacijas, gauti norimas prekes ar paslaugas.

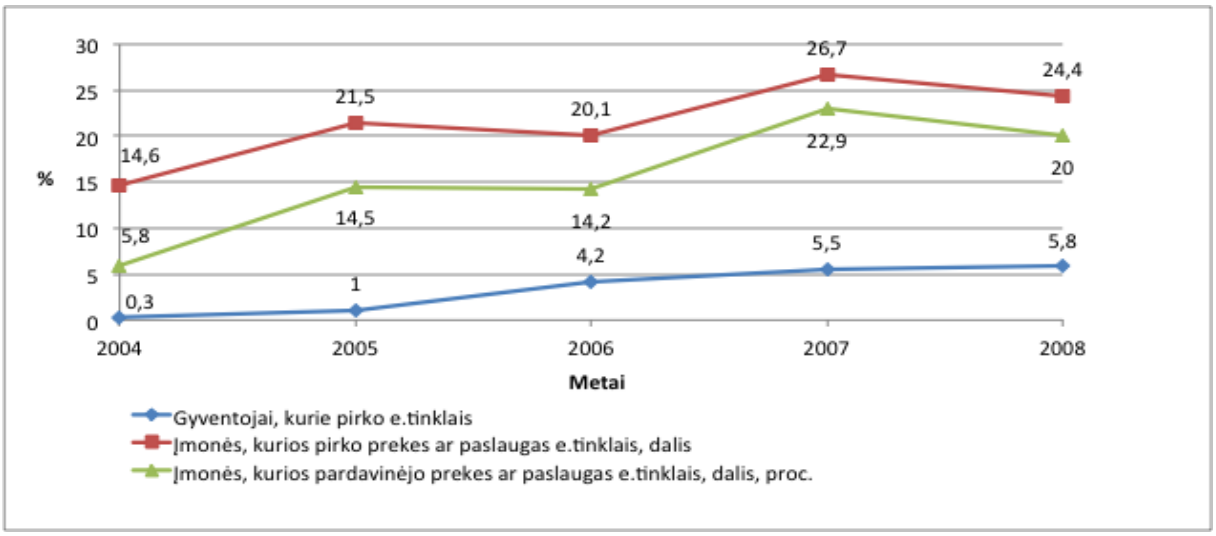
Paslaugos internetu teikiamos vykdant elektroninių dokumentų (įskaitant žiniatinklyje pateikiamus dokumentus, tinklalapius) tranzakcijas tarp skirtingų paslaugų sistemų ar vartotojo ir paslaugų sistemos. Šie dokumentai turi tokią pat juridinę galią kaip ir popieriniai dokumentai (pažymos, prašymai, liudijimai ir pan.). Šiandien populiarinama duomenų laikymo ir apdorojimo „debesyje“ technologija (*angl. cloud computing*), išorinių duomenų centrų paslaugos atveria ne tik naujas el. paslaugų teikimo galimybes, tačiau ir iškelia el. dokumentų saugos problemą. Paslaugos teikėjas negali valdyti el. tranzakcijų dokumentų ir priverstas visiškai pasitikėti perkamų duomenų apdorojimo ir kaupimo paslaugų teikėjais. Taigi iškyla el. tranzakcijų dokumentų klastojimo, neteisėto paviešinimo, nesąžiningo panaudojimo ir išsigynimo problemos vykdant jų manus tarp kliento ir paslaugų tiekėjo tarpininkaujant duomenų apdorojimo ir kaupimo centrams.

El. tranzakcijų dokumentų saugos problemoms spręsti yra sukurta daug skirtingų metodų, protokolų, tačiau nė vienas jų nėra absoliučiai pilnavertis sprendžiant dokumentų apsaugos problemą „debesies“ principu veikiančioje IT infrastruktūroje. Tokiu atveju skirtingi saugos metodai, taikomi kompleksiškai, papildydami vienas kitą, gali užtikrinti geresnę perduodamos informacijos saugą. Todėl šiuo darbu siekiama giliau pažvelgti į el. tranzakcijų dokumentų, kuriais keičiamasi tarpininkaujant duomenų apdorojimo ir kaupimo centrui apsaugos problemą, išanalizuoti el. paslaugų teikimo modelį, pačių dokumentų ir jų perdavimo saugos priemones, pasiūlyti naują ir kitus metodus papildantį žiniatinklyje pateikiamų el. dokumentų saugos metodą bei jį ištirti saugumo ir greitaveikos aspektais.

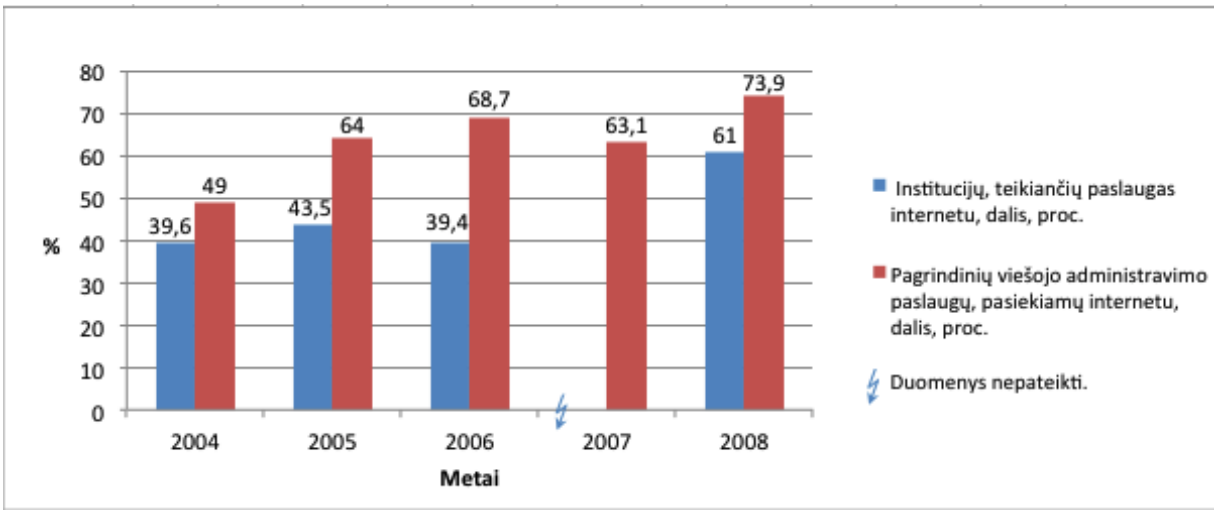
1. EL. TRANZAKCIJŲ DOKUMENTŲ APSAUGOS GALIMYBIŲ ANALIZĖ

1.1 El. paslaugų Lietuvoje ir pasaulyje plėtros apžvalga

Pastarąjį dešimtmetį sparčiai tobulėjančios ir pinančios informacinės technologijos lėmė itin spartų interneto vartotojų augimą, o kartu ir naujų paslaugų teikimo internetu poreikį. Remiantis JAV Informacinių Technologijų Asociacijos duomenimis, 2002 metais pasaulyje buvo apie 600 milijonų interneto vartotojų, 2008 metais šis skaičius gerokai viršijo milijardą. Per aštuonis metus JAV interneto vartotojų padidėjo tiek, kad visoje šalyje interneto prieigą turėjo 58 proc. namų ūkių [11]. Naujausi Statistikos Departamento prie Lietuvos Respublikos Vyriausybės pateikiami duomenys rodo, kad 2009 metais internetu Lietuvoje buvo apsirūpinę 54,7% namų ūkių (2000 metais – 2,3%) ir 95% įmonių (2000 metais – 58%). Vartotojų naudojimosi tam tikromis interneto paslaugomis bei paslaugų teikimo internetu kiekio augimą atspindi 1 ir 2 paveiksluose pavaizduotos kreivės [22].



1 pav. Elektroninės prekybos statistiniai rodikliai



2 pav. IT panaudojimas viešojo administravimo institucijose

El. paslaugas galima skirstyti į dvi dideles grupes – el. valdžios bei el. komercijos. Šios dar gali būti skirstomos į keletą kitų, pavyzdžiui el. komercijos paslaugos gali būti skirstomos taip:

- El. rinkodara ir prekyba internete (pvz. el. parduotuvės, reklama);
- Finansų ir informacijos teikimo paslaugos (pvz. el. bankininkystė, informacijos paieškos paslaugos);
- Pagalbos ir priežiūros ir aptarnavimo paslaugos (pvz. įmonės kompiuterių, informacinių sistemų priežiūra nuotoliniu būdu, gedimų fiksavimo ir šalinimo tarnybos, įvairių konsultacijų teikimo interaktyviai paslaugos) [11].

El. valdžios paslaugos daugiausiai susijusios su informacijos, konsultacijų teikimu piliečiams, įvairių dokumentų priėmimu, išdavimu, visuotinių paslaugų, tokių kaip el. balsavimas teikimu ir panašiai. Pavyzdžiui, šiuo metu Lietuvoje galimos pajamų, gyvenamosios vietos deklaravimo internetu, informacijos apie valstybinį socialinį draudimą gavimo, informacijos iš valstybinės ligonių kasos gavimo, transporto registravimo, įvairių pažymų, išrašų gavimo, prašymų ir paraiškų teikimo elektroniniu būdu paslaugos [11, 3].

Augant paslaugų, teikiamų internetu ir jų vartotojų kiekiui, susiduriama su perduodamos informacijos saugos problemomis. Statistiniai duomenys rodo, kad 2008 m. informacijos saugos problemų turėjo 41,7% įmonių (2004m. – 37,4%) [22]. Kompleksiškai informacijos saugos problemos ir informacijos saugos priemonės nagrinėjamos [9, 11, 18] literatūroje. Bendrai kalbant, informacijos saugos priemonės yra skirtos užtikrinti perduodamos informacijos konfidencialumą (kad informacija nebūtų paviešinta), vientisumą (kad informacija nebūtų iškraipyta), autentiškumą (kad informacija nebūtų suklastota). Bene populiariausia terpė, kurioje teikiamos tiek el. verslo, tiek el. valdžios paslaugos yra žiniatinklis [18]. Todėl darbe analizuojami žiniatinklio technologijos pagrindu veikiančių el. paslaugų teikimo modeliai, el. tranzakcijų dokumentų apsaugos technologijos ir metodai, pačių el. tranzakcijų dokumentų, naudojamų paslaugų sistemose, tipai ir standartai, apžvelgiami jau sukurti sprendimai.

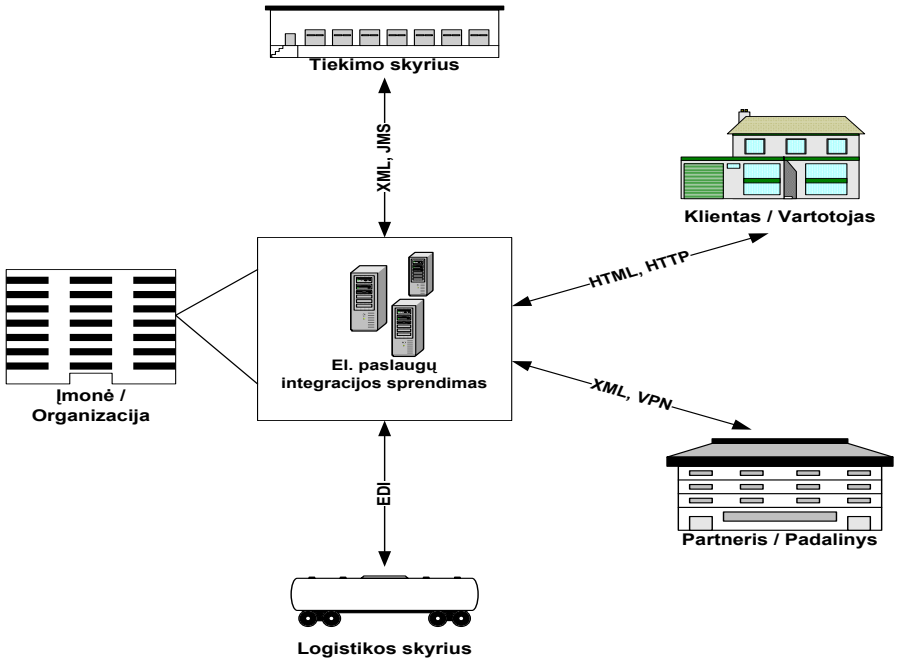
1.2 El. paslaugų saugos organizavimo priemonės ir problemos

Elektroninėje komercijoje yra priimtas paslaugų skirstymas į „verslas vartotojui“ (*angl. business to consumer, B2C*), „verslas verslui“ (*angl. business to business, B2B*). Šių dviejų modelių panašumai ir skirtumai aprašyti 1-oje lentelėje.

1 lentelė. Paslaugų teikimo modelių „verslas vartotojui“ ir „verslas verslui“ palyginimas

Modelis / Kriterijai	„Verslas vartotojui“	„Verslas verslui“
Veiklos modelis	Siekiama betarpiškai kontaktuoti su vartotoju	Orientuojamasi į bendravimą tarp skirtingų (pvz. užsakymų, tiekimo) sistemų
Veiklos	Vartotojui pateikiami paslaugų katalogai, dėmesys skiriamas užsakymo, apmokėjimo, pirkimo būsenos pateikimo procedūroms.	Sprendžiamos skirtingų tipų duomenų perdavimo tarp skirtingų sistemų (užsakymų, tiekimo ir pan. perdavimo ir suderinamumo problemos.)
Tranzakcijų pobūdis	Užsakymai (pirkimai) impulsyvūs ir trumpalaikiai. Svarbus teisingas tranzakcijos (užsakymo patvirtinimo) dokumentas.	Tranzakcijos vyksta pagal sudėtingus scenarijus, jose dalyvauja visos šalys (paslaugos teikėjai, klientai, tiekėjai) nuo tranzakcijos inicijavimo iki pabaigos.
Lankstumas santykių su klientais arba partneriais pokyčiams	Santykiai su klientais yra lankstūs. Vartotojas bet kada gali laisvai rinktis tiekėją.	Partnerių keitimas yra brangus, nes jau su tiekėjais ir pirkėjais suderinti veiklos vykdymo procesai, organizacijų sistemos jau būna sujungtos tarpusavy.
Vartotojų santykiai	Santykiai su klientais dažnai būna vienpusiai: veiklos pobūdis ir modelis apibrėžia ir palaiko santykius su klientais.	Santykiai ilgalaikiai, paremti sutartimis ir įsipareigojimais, susitarimais dėl bendradarbiavimo planuojant ir vystant veiklą.

Nepaisant skirtumų, šie modeliai el. komercijoje ir apskritai el. paslaugų srityje yra tarpusavy integruoti ir sudaro el. paslaugų teikimo sistemos visumą. Tai iliustruoja 3 pav. pavaizduota schema.



3 pav. El. paslaugų teikimo organizavimo bendroji schema

Šioje schemoje valstybinės įstaigos ar verslo kompanijos bendradarbiavimas su kiekvienu iš partnerių ar kitų padalinių gali būti grindžiamas „verslas verslui“ modeliu, o paslaugų tiekimas vartotojui – „verslas vartotojui“ modeliu. Ryšys tarp šalių gali būti organizuojamas naudojant skirtingas technologijas, keičiantis skirtingų formatų el. tranzakcijų dokumentais, kuriems gali būti taikomi skirtingi saugos reikalavimai. Pagrindinė problema tokiu atveju – sprendimo integracija. Tarp partnerių derinami šie aspektai:

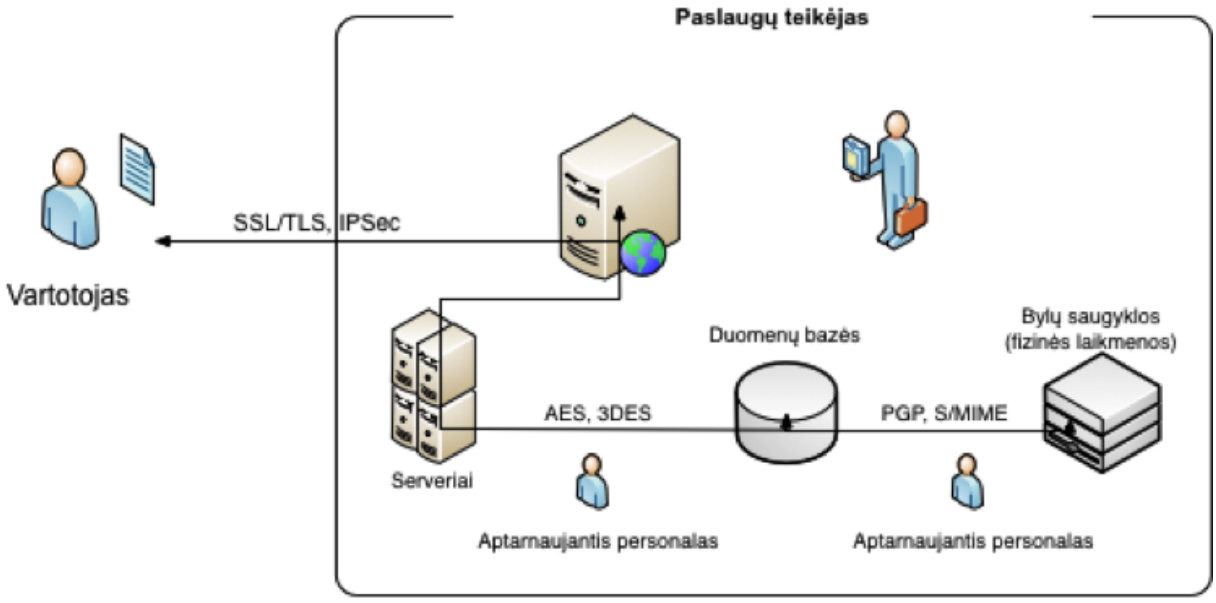
- Programinė įranga ir dokumentų formatai (XML, X.12, SAP, Oracle ir pan.);
- Ryšio kanalai (Internetas, VAN ir pan.);
- Bendravimo protokolai (FTP, HTTP, HTTPS ir pan.);
- Saugos technologijos (SSL/TLS, PGP, IPSec ir pan.);
- Reikalavimai paslaugoms (archyvavimo galimybės, maršrutizavimas, veiklos stebėjimas (*angl. monitoring*) ir pan.).

Kai kuriais atvejais, kai įmonė ar organizacija yra smulki ar vidutinė, santykiai su partneriu ar kitu organizacijos padaliniu taip pat gali būti „verslas klientui“ tipo. Tokiu atveju klientas yra pati įmonė ar organizacija, kuri naudojasi partnerio ar kito padalinio paslaugomis.

Vienas iš būdų (labiausiai paplitęs) teikti el. paslaugas yra organizacijos nuosava IT infrastruktūra (4 pav.). Ją gali sudaryti prieigos per žiniatinklį serveriai, taikomųjų procesų vykdymo serveriai, duomenų bazės, failų direktorijos, duomenų saugyklos. Tuomet el. paslaugų teikėjas pats rūpinasi organizacijos viduje apdorojamų ir saugomų duomenų bei dokumentų sauga.

Vidaus saugos politika gali nusakyti ar apdoroti duomenys bus saugomi duomenų bazėje ar saugykloje atvirai ar šifruota forma, kokiais metodais ir algoritmais bus šifruojami, kur bus saugomi raktai ir panašiai.

Ryšiui su vartotoju apsaugoti dažniausiai naudojamos SSL/TLS, IPSec technologijos. SSL/TLS saugo duomenis transporto lygmeniu, IPSec – tinklo lygmeniu (slepiamas kliento kompiuterio IP adresas). Pastaruoju būdu gali būti organizuojami virtualūs privatūs tinklai, jei vartotojas ir paslaugos teikėjas yra partneriai, ar tranzakcijos vyksta įmonės viduje (pvz. tarp padalinių).



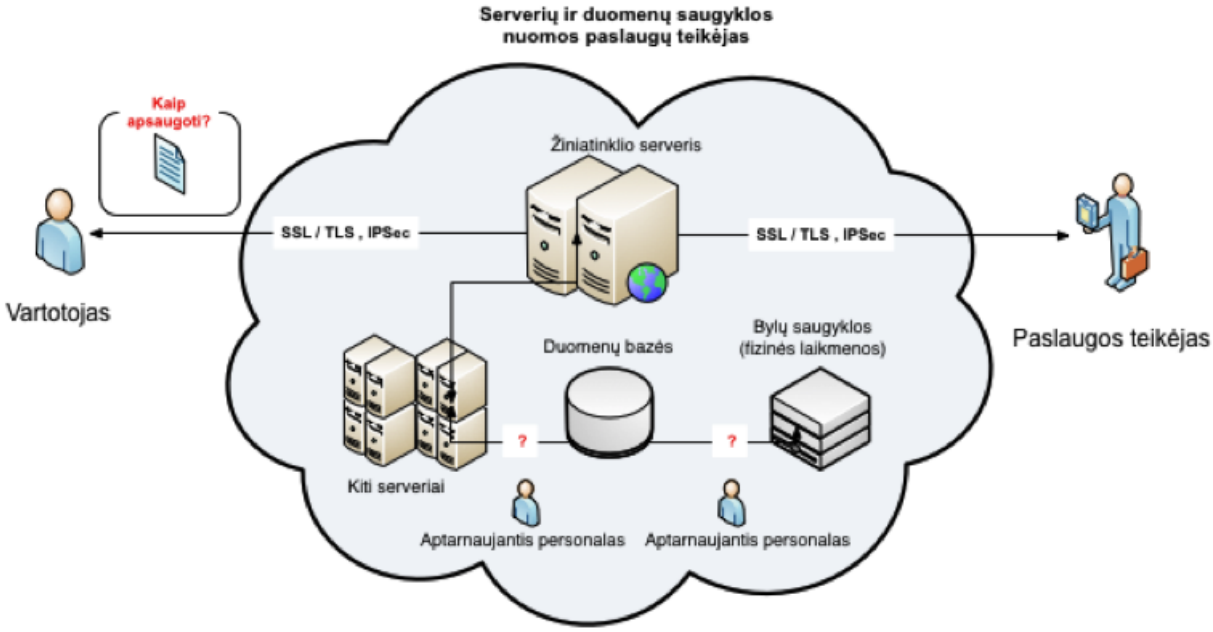
4 pav. Paslaugos tiekėjas naudojasi savo IT infrastruktūra

Augant paslaugų, teikiamų internetu ir jų vartotojų kiekiui, lokalus duomenų ir el. dokumentų saugojimas ir apdorojimas organizacijoje tampa vis sudėtingesnis (reikalauja daugiau išteklių), dėl to brangsta. Šią problemą sprendžia populiarėjanti „debesies“ technologija (*angl. cloud computing*) ir duomenų centrai (5 pav.) [3].

„Debesis“ – tai dinamiškai plečiama, dažnai naudojanti virtualizavimo technologijas kompiuterinių išteklių sistema (serveriai, duomenų saugyklos), išoriškai „matoma“ kaip viena paslauga ar jų rinkinys. Paslaugų rinkinį gali sudaryti tiek interneto svetainės (portalo) talpinimas, prieiga per žiniatinklį, FTP, duomenų apdorojimo serveriai, duomenų bazės, failų direktorijos ir duomenų saugyklos. Organizacijos šias paslaugas dažniausiai perka iš trečiųjų šalių.

Prieigai prie duomenų ir dokumentų ir jų saugai „debesies“ infrastruktūroje gali būti organizuojama įvairiai, gali būti taikomos visos anksčiau išvardintos saugos technologijos. Problema ta, kad pats paslaugos teikėjas, negali žinoti kur tiksliai ir kokiu būdu saugomi jo duomenys, kokie šifravimo metodai ir algoritmai naudojami infrastuktūros viduje.

Nuomos paslaugų teikėjas ryšį su klientais gali organizuoti apsaugotu kanalu (SSL/TLS, IPSec), tačiau šios technologijos nesaugo dokumento infrastruktūros viduje. Taip teoriškai gali būti pažeistas perduodamų duomenų ir dokumentų konfidencialumas, vientisumas, autentiškumas. Pagrindiniai „debesies“ infrastruktūros privalumai ir trūkumai išvardinti žemiau.



5 pav. Paslaugos teikėjas naudojasi „debesies“ infrastruktūra

Pagrindiniai išskiriami šios paslaugos privalumai:

- Duomenų prieinamumas iš išorės (ne tik iš organizacijos vidinio tinklo).
- Mokama tik už naudojamus resursus (sprendžiama kaštų problema, kai organizacijos serveriai ar kiti ištekliai veikia tam tikrą laiką nenaudojami).
- Sumažėja darbo organizacijos IT specialistams, nes už duomenų apdorojimo ir saugojimo organizavimą atsakingas paslaugos teikėjas.
- Nesunkiai plečiama aplinka, jei organizacijai prireikia daugiau vietos duomenims ir dokumentams saugoti ar daugiau išteklių duomenims apdoroti.

Pagrindiniai trūkumai:

- Organizacija negali kontroliuoti duomenų ir dokumentų organizavimo „debesyje“ (apsunkina duomenų ir dokumentų gyvavimo ciklo organizavimą).
- Duomenų ir informacijos konfidencialumas, vientisumas ir autentiškumas priklauso nuo trečiosios šalies – paslaugų teikėjo.
- Netikėtai sutrikus „debesies“ darbui kyla duomenų praradimo grėsmė. Dėl to ilgesniam laikui gali sutrikti organizacijos darbas.
- Ne visiškai neaišku, kas nutiks su duomenimis, jei šios paslaugos teikėjas nutrauks veiklą.

Dėl išvardintų privalumų, „debesies“ technologija yra viena išeičių smulkaus ar vidutinio verslo įmonėms ar viešojo sektoriaus organizacijoms sumažinti veiklos kaštus, tačiau būtina įvertinti saugomų duomenų svarbą, pasirinkti tinkamus jų saugos metodus [14].

Atlikti tyrimai rodo, jog vidutiniškai tik apie 27% organizacijų atskirai rūpinasi savo duomenų apsauga prieš talpinant juos „debesyje“ [6]:

- 20% respondentų teigė turintys IT saugos padalinius, kurie rūpinasi papildomų saugos priemonių organizavimu;
- 25% teigė, jog jų IT saugos padaliniai tuo nesirūpina;
- 30% teigė, jog tiesiog įvertina visus galimus paslaugų tiekėjus prieš išsirinkdami tinkamiausią;
- 23% teigė, jog reikalauja nuomos paslaugų teikėjo pateikti kokius nors jo patikimumą ar standartų laikymąsi patvirtinančius dokumentus (sertifikatus ar pan.);
- 75% mano, kad „debesies“ technologija nėra idealus sprendimas duomenų saugos požiūriu;
- 19% organizuoja mokymus duomenų saugos klausimais.

Remiantis statistika galima teigti, jog duomenų ir el. dokumentų, kuriais manipuliuojama duomenų apdorojimo ir kaupimo centre, sauga yra viena prioritetinių vystymo sričių. Tad darbe koncentruojamasi į metodus ir technologijas, skirtas apsaugoti šioje infrastruktūroje apdorojamą ir saugomą informaciją, analizuojamos el. tranzakcijų dokumentų apsaugos technologijos ir metodai, pačių el. tranzakcijų dokumentų, naudojamų paslaugų sistemose, tipai ir standartai, apžvelgiami sukurti sprendimai.

1.3 El. dokumentų ir duomenų apsaugos technologijų apžvalga

Paslaugų teikimo sistemos yra nutolusios vartotojo atžvilgiu, tad sudėtinga taikyti žemesnius nei tinklo lygmens saugos metodus. Dažniausiai organizuojama transporto ir taikomo lygmens el. dokumentų sauga. Todėl šiame skyriuje apžvelgiami populiarūs šių lygmenų saugos metodai, vertinami jų privalumai ir trūkumai, tinkamumas atsižvelgiant į nagrinėjamą problemą.

1.3.1 PGP

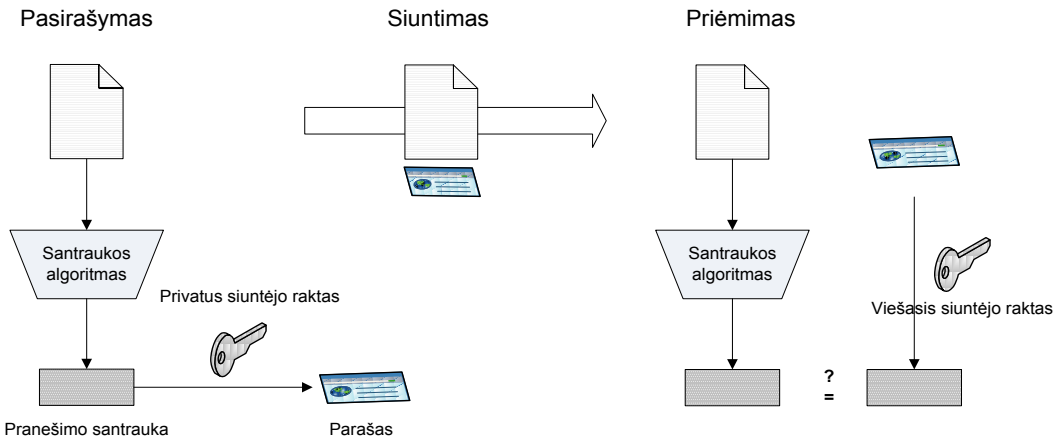
PGP (*angl. Pretty Good Privacy*) daugiausia naudojama el. laiškų ir failų apsaugai. PGP turi visus kriptografinei sistemai būdingus komponentus: simetrinius ir asimetrinius šifravimo algoritmus, pranešimo santraukos algoritmus, raktus, protokolus ir būtinus programinės įrangos elementus. Raktams valdyti ši technologija naudoja RSA viešojo rakto šifravimą, o didelio kiekio duomenis šifruoja su IDEA simetriniu šifru. PGP konfidencialumui užtikrinti naudoja IDEA šifravimo algoritmą, vientisumui – MD5 ar SHA-1 maišos algoritmą, autentifikacijai – viešojo rakto sertifikatus ir neišsigynimo funkciją,

pasitelkdama kriptografiniu būdu pasirašytus el. pašto pranešimus. PGP turi savus skaitmeninius sertifikatus (skirtingus nei PKI atveju). Šia technologija apsaugomi tik taikomojo lygmens duomenys [12].

Veikimo principas

Atvira informacija iš pradžių suspaudžiama tam, kad būtų pašalintas informacijos perteklius. Taip užtikrinamas didesnis kriptografinis atsparumas. Programa generuoja atsitiktinį simetrinio šifravimo sesijos raktą ir juo užšifruoja suspaustą informacijos failą. Informacijos gavėjo viešuoju raktu užšifruojamas šifravimo sesijos raktas. Jis kartu su užšifruotu failu jau gali būti pasiųstas gavėjui atviru neapsaugotu kanalu.

Gavėjas savo privačiu raktu dešifruoja sesijos raktą. Tada, turėdamas sesijos raktą, gavėjas simetriniu algoritmu gali iššifruoti pranešimą. Siuntėjo identifikavimo ir informacijos autentifikacijos funkcijas PGP technologijoje atlieka skaitmeninis parašas (6 pav.).



6 pav. El. dokumento perdavimo PGP protokolu schema.

PGP technologija gali būti naudojama el. tranzakcijų dokumentams šifruoti ir pasirašyti atjungties režimu (*angl. off-line*), t.y. kai dokumentas pirmiau parsisiunčiamas, redaguojamas, po to vartotojo kompiuteryje užšifruojamas, pasirašomas ir siunčiamas el. paštu, FTP ar žiniatinklio protokolu. Pastarasis atvejis taikomas įvairiose tinkle veikiančiose dokumentų ir bylų valdymo bei mainų sistemose.

Ypatybės, privalumai ir trūkumai

Pagrindiniai PGP privalumai:

- Naudojami stiprūs kriptografiniai algoritmai dokumentų šifravimui (AES, CAST, 3DES, IDEA, Twofish), santraukų sudarymui (MD5, SHA-1) bei pasirašymui (rakto formatai OpenPGP, DSA ir RSA parašo algoritmai);
- Vienareikšmiškai gali identifikuoti ir autentifikuoti vartotoją kaip asmenį;

- Nebūtinas saugus duomenų perdavimo kanalas tarp vartotojo ir serverio;
- Tinkamas naudoti el. dokumentų valdymo, el. pašto sistemose, failų sistemos šifravimui.

Pagrindiniai trūkumai:

- Nėra sertifikavimo tarnybos, tvirtinančios raktų galiojimą (nepalaiko X.509).
- Reikalinga papildoma speciali programinė įranga dokumentų šifravimui ir pasirašymui;
- Negali veikti ir šifruoti siunčiamų duomenų tiesiogiai prijungties režimu (*angl. on-line*);

Dėl aukščiau išvardintų privalumų, PGP yra populiarus el. pašto žinutėms, dokumentams, saugomiems vartotojo kompiuteryje ir bendrinamiems dokumentams šifruoti nedideliame vartotojų rate. PGP el. parašų tikrinimas yra paremtas pasitikėjimo tinklu, nėra šakninės sertifikavimo tarnybos (*angl. Certification Authority, CA*), todėl sunku suvaldyti sukompromituotus, nebegaliojančius vartotojų raktus. Būtent šis trūkumas užkerta kelią platesniam technologijos naudojimui pasirašinėti ir šifruoti dokumentus viešųjų el. paslaugų sistemose [6].

1.3.2 S/MIME

MIME (*angl. Multipurpose Internet Mail Extensions*) yra laisvai prieinamas papildinių specifikacijų rinkinys, kuris leidžia atpažinti ir žiniatinklyje bei el. pašte naudoti bei apdoroti tekstinius dokumentus, paveikslėlius, vaizdo medžiagą ir panašiai. Nors MIME buvo sukurtas pašto sistemoms, jis plačiai taikomas ir žiniatinklyje. Kiekvienam MIME tipui klientas pats turi sugebėti atvaizduoti informaciją (pvz., .GIF, .JPEG failus) arba įdiegti ir sukonfigūruoti išorines peržiūros priemones.

Saugus universalus interneto pašto papildymas (*angl. trumpinys S/MIME*) – tai standartas, skirtas saugiai perduoti elektroninius duomenis, šifruoti el. pašto laiškus ir juos pasirašyti. Taip pat S/MIME leidžia šifruoti prie el. laiškų prisegtas bylas. Šifras ir maišos algoritmai nėra iš anksto nustatyti ir gali būti specifikuoti el. pašto vartotojo. [19]

S/MIME laiško vientisumui užtikrinti naudojami atitinkami maišos algoritmai SHA-1, MD5, autentiškumui naudojamas elektroninis parašas, konfidencialumui naudojamas šifravimas [19].

S/MIME paremtas viešojo rakto infrastruktūra PKI (*angl. Public Key Infrastructure*) ir jos teikiamais skaitmeniniais sertifikatais. Sertifikatas gali būti gautas iš vidinio PKI arba iš išorinės infrastruktūros. S/MIME sujungia tris algoritmus, naudojančius viešuosius raktus. Skaitmeninio parašo standartas (DSS algoritmas) yra būtinas skaitmeninio parašo sukūrimo

algoritme. Seansų raktams šifruoti naudojamas ElGamal'o algoritmas. Kaip parašų ir seansų raktų šifravimo alternatyvą galima naudoti RSA algoritmą.

Veikimo principas

S/MIME standarte naudojama keletas specialių turinio tipų. Visi šie turinio tipai ir potipiai turi PKCS ženklinimus.

2 lentelė. S/MIME tipai ir potipiai

Tipas	Potipis	S/MIME parametras	Apibrėžimas
Daugiakomponentis (*angl. Multipart*)	Pasirašytas (*angl. Signed*)		Viešasis pasirašytas pranešimas, susidedantis iš 2 dalių: pranešimo ir jo parašo.
Programa (*angl. Application)*	pkcs7-mime	*signedData*	Pasirašytas S/MIME objektas.
	pkcs7-mime	*envelopedData*	Šifruotas S/MIME objektas.
	pkcs7-mime	*degenerateSignedData*	Objektas, turintis tik viešųjų raktų sertifikatus.
	pkcs7-signature	-	Parašo tipas, esantis pranešimo multipart/signed tipo dalis
	pkcs10-mime	-	Sertifikato registracijos užklausos pranešimas.

Viešasis pasirašytas pranešimas suformuojamas tuomet, kai turiniui naudojamas *Multipart* tipas ir *Signed* potipis. Pranešimas *Multipart/Signed* sujungia dvi dalis. Pirma dalis gali būti bet kokio MIME tipo, tačiau paruošta taip, kad ji keliaudama nuo siuntėjo iki gavėjo nebūtų pakeista. Tai reiškia, kad jeigu pirma dalis nėra užkoduota 7 bitų koduote, tai duomenis reikia koduoti base-64 formatu. Pirmoje dalyje yra viešasis pranešimo tekstas. Antra dalis – tai atskiras pasirašymas. Ji formuojama pagal objekto *signedData* algoritmą. Kaip rezultatas sukuriamas objektas *signedData*, kurio turinio laukas yra tuščias. Po to šis objektas yra koduojamas base64 formatu, kad taptų antra daugiakomponenčio pranešimo dalimi. MIME tipo antrai daliai priskiriama *Application* reikšmė, o potipiui – pkcs7-*Signature*.

Ypatybės, privalumai ir trūkumai

Pagrindiniai S/MIME privalumai:

- Naudojami pakankamai stiprūs kriptografiniai algoritmai dokumentų šifravimui (3DES, RC2/40), santraukų sudarymui (SHA-1, MD5) ir pasirašymui (X.509 parašo infrastruktūra, raktų formatai RSA, DSS);
- Geras suderinamumas su populiariausia el. pašto kliento programine įranga;
- Naudojami plačias taikymo galimybes turintys X.509 standarto sertifikatai, kuriuos prižiūri sertifikavimo tarnyba;
- Duomenims perduoti nebūtinas apsaugotas ryšio kanalas.

Pagrindiniai trūkumai:

- Nepritaikytas el. tranzakcijų dokumentams prijungties režimu (*angl. on-line*) šifruoti ir pasirašinėti žiniatinklio aplinkoje;
- Kai kurios internetinės el. pašto sistemos nesuderinamos su šiuo standartu.

S/MIME pagrindinis privalumas prieš PGP el. pašto sistemose – hierarchinė sertifikavimo centrų infrastruktūra bei geresnis suderinamumas su skirtinga el. pašto programine įranga [10]. Kadangi šis standartas buvo kuriamas specialiai el. paštui, pirmasis paminėtas trūkumas yra neesminis, tačiau parodo standarto ribotumą, negalėjimą tiesiogiai taikyti 1.2 skyriuje aprašytai problemai spręsti.

1.3.3 XMLDSig

X.509 – tai tarptautinės telekomunikacijos organizacijos (*angl. International Telecommunication Union, The Telecommunication Standardization Sector, ITU-T*) sukurtas standartas, apibrėžiantis viešojo rakto infrastruktūrą, kaip sertifikavimo tarnybų ir raktų-sertifikatų griežtą hierarchinę sistemą kaip priemonę tinklu perduodamų dokumentų ir duomenų autentiškumui užtikrinti. Tai yra ne konkretus metodas, o bendra, rekomendacinio pobūdžio metodika. XMLDsig – el. parašo XML dokumentams standartas. Kartu šie du standartai sudaro galimybes kompleksiškai saugoti XML dokumentus (šifruojant X.509 standartu pagrįstomis priemonėmis ir pasirašyti XMLDSig standartą realizuojančiomis priemonėmis) [21].

Veikimo principas

Dokumentų pasirašymo ir tikrinimo procesas toks pat, kaip pavaizduotasis 6 paveiksle, skirtumas toks, kad pasirašymui naudojamas X.509 formato saugykloje (ar faile) saugomas raktas, o tikrinimui – atitinkamas sertifikatas. XMLDsig apibrėžia, kaip X.509

priemonėmis sukurtas parašas komponuojamas į XML dokumentą (apvelkantis (*angl. enveloping*), apvilktas (*angl. enveloped*) arba atskirtas (*angl. detached*) parašas).

X.509 viešojo rakto infrastruktūra gali būti naudojama ir PGP ir S/MIME bei XML dokumentų šifravimui. Sertifikatai (failai) gali būti kelių tipų:

- .pem - (*Privacy Enhanced Mail*) Base64 koduotas DER sertifikatas;
- .cer, .crt, .der – paprastai dvejetainės DER formos, Base64 koduotas sertifikatas.
- .p7b, .p7c - PKCS#7 *SignedData* struktūra be duomenų, tik sertifikatai arba CRL(s) atšaukimo sąrašai;
- .p12 - PKCS#12 konteineris, gali talpinti sertifikatus ir privačius raktus (apsaugotas slaptažodžiu) ;
- .pfx - PFX, buvęs PKCS#12 (dažniausiai informacija laikoma PKCS#12 formatu, pvz. PFX failuose sugeneruotose IIS).

Sertifikato struktūrą sudaro tokie duomenys: X.509 standarto versija, sertifikato serijos numeris, sertifikavimo tarnybos pavadinimas, galiojimo laikas (pradžios ir pabaigos), viešasis raktas, plėtiniai. Naudojant XMLDSig, šie duomenys įtraukiami į dokumentą naudojant atitinkamas žymas [21].

Plėtiniuose gali būti nurodoma, kaip šifruojami sesijos raktai, raktų apsikeitimo protokolas, sertifikatų tarnybos sertifikatas. Taip pat galima nurodyti, kokiu tikslu gali būti naudojamas privatus raktas. Raktai gali būti naudojami serverio autentifikacijai, kliento autentifikacijai, programos kodo pasirašymui, el. pašto žinučių pasirašymui ar šifravimui, laiko žymoms.

Ypatybės, privalumai ir trūkumai

Pagrindiniai standarto privalumai:

- Plati taikymo sritis, parašo realizacijos galimybės;
- Hierarchinė sertifikatų ir sertifikavimo tarnybų struktūra, leidžianti pakankamai greitai patikrinti sertifikato tikrumą ir galiojimą;
- Galimybės kontroliuoti sertifikato naudojimą (leisti naudoti tik tam tikriems tikslams), taip išvengiant netikslingo naudojimo, naudojančio sistemų resursus.

Pagrindiniai trūkumai:

- Sudėtingas ir nepraktiškas taikymas el. komercijos sprendimuose mokėjimams (kur svarbu ne mokėtojo tapatybė, o mokėjimo instrumento duomenys, ryšio

autentiškumas, esamiems patikimiems verslo ryšiams trečios šalies pastovus dalyvavimas yra tik nepatogumas);

- Nelankstus duomenų valdymas (keičiantis vartotojo el. pašto adresui ar pavardei, sertifikatas atšaukiamas ir iš duodamas naujas, keičiantis verslo procesams, turi būti kuriama laikinai lygiagreti sertifikatų sistema – tai lėtina operacijas, kuriose naudojami sertifikatai.

Dėl savo privalumų X.509 standartas plačiai taikomas specifinėse srityse, kur svarbi vartotojo, kaip asmens identifikacija ir autentifikacija, sertifikatai gali būti pritaikyti tik tam tikroms naudojimo sritims taip lanksčiai paskirstant vartotojams atitinkamas prieigos prie tam tikru paslaugu teises. Šios savybės naudingos taikant viešojo rakto kriptografiją viešojo sektoriaus paslaugoms teikti. Dėl trūkumų šiuo standartu paremti saugos metodai paprastai netaikomi el. verslo, mokėjimų sprendimams. Čia populiarūs yra SSL/TLS, VPN, SET protokolai [2].

1.3.4 Saugos technologijų palyginimas

Saugos technologijų palyginimas atliekamas remiantis kriterijais, kurie yra esminiai aspektai vertinant nagrinėjamos technologijos tinkamumą duomenų saugos problemai spręsti juos perduodant ir saugant „debesies“ architektūros aplinkoje [2]:

- Galimybė saugoti duomenis ne tik perdavimo bet ir saugojimo metu (5 pav.).
- El. parašo galimybė – kaip rodo tyrimai ir situacija šalyje, atsiranda vis daugiau paslaugų, kurioms reikalinga vartotojo kaip asmens identifikacija ir autentifikacija. Tam naudojamas el. parašas.
- Galimybė šifruoti ir pasirašyti žiniatinklio aplinkoje pateikiamus dokumentus – daugumą paslaugų internete vartotojas pasiekia per interneto naršyklę, dokumentų atskirai nekopijuodamas į savo kompiuterį, tad svarbu, kad šie dokumentai galėtų būti apsaugoti prijungties (*angl. on-line*) režimu.
- Atskiros programinės įrangos poreikis – jei saugumo technologiją ar protokolą palaiko interneto naršyklė ar el. pašto kliento programa, vartotojui paprasčiau ir patogiau ta technologija naudotis.

Pagal šiuos kriterijus technologijos tarpusavyje palygintos 3 lentelėje.

3 lentelė. Duomenų ir dokumentų apsaugos technologijų palyginimas

Technologija / Kriterijus	PGP	S/MIME	XMLDsig
El. parašo galimybė	T	T	T
Galimybė šifruoti žiniatinklio aplinkoje pateikiamus dokumentus	N	T	T
Dalinio šifravimo ir pasirašymo galimybė	N	N	T
Galimybė nuskaityti ir tiesiogiai apdoroti pasirašytą ir šifruotą dokumentą	N	N	T

Iš lentelės matyti, kad visus kriterijus geriausiai atitinka S/MIME ir XMLDSig technologijos. Tačiau S/MIME technologija labiau pritaikyta el. pašto žinučių apsaugai. Ją naudojant sukuriamas „konteineris“ su šifruota el. pašto žinute. Šis konteineris gali būti pasirašytas ir parašas siunčiamas atskirai, arba pasirašyta el. pašto žinutė su parašu gali būti patalpinta į konteinerį ir išsiųsta kaip failas. Tokiu būdu sudėtinga operuoti siunčiamais duomenimis ar dokumentais prijungties (*angl. online*) režimu, nes kiekvienu atveju turi būti išpakuojamas „konteineris“ ir tik tada žinutė atvaizduojama. Tuo tarpu XMLDsig standartais paremta technologija sudaro galimybę realizuoti pasirašymo metodą pritaikytą konkretiems poreikiams ar problemai spręsti, pavyzdžiui XML ar kitų tipų tranzakcijų dokumentų pasirašymui ir šifravimui žiniatinklio aplinkoje. Toliau darbe bus koncentruojamasi būtent į šios technologijos panaudojimą kuriant el. tranzakcijų dokumentų pasirašymo žiniatinklio aplinkoje metodą.

1.4 El. tranzakcijų dokumentai ir jų saugos ypatybės

Bendru atveju elektroninių tranzakcijų dokumentai žiniatinklio aplinkoje – tai statiški arba dinamiškai generuojami dokumentai su galimybe juose nurodyti tam tikras tranzakcijos vykdymo sąlygas, norint gauti tam tikras paslaugas.

El. tranzakcijų dokumentai, priklausomai nuo sistemos, kurioje jais disponuojama, architektūros ir veikimo principo, gali būti įvairių tipų: sąveikaujančioje el. komercijos sistemose dažnai keičiamasi cXML, ebXML ir kitų tipų dokumentais. Šių tipų dokumentai suformuojami iš pradinių įvestų duomenų ir toliau jais automatiškai disponuojama be vartotojo įsikišimo. Tokiais dokumentais dažniausiai keičiamasi organizacijų vidiniuose tinkluose ar VPN tinkluose naudojant atitinkamas transporto ar tinklo lygmens saugos technologijas.

Sprendžiant el. tranzakcijų dokumentų saugos problemą, kai jais disponuojama „debesyje“ ir ne tarp sistemų, o galima sakyti tarp sistemos vartotojų – asmenų (kliento ir paslaugos teikėjo), aukščiau išvardintų tipų dokumentai nėra tinkamas sprendimas dėl savo specializuotos struktūros (turinčios specialias žymas, pritaikytos automatiniam apdorojimui gavėjo sistemoje ir pan.). Tokiu atveju tinkamas sprendimas yra formuoti tiesiog bazinius XML dokumentus (be specialių žymų, kaip ebXML ar cXML). Šio standarto dokumentai lengvai gali būti transformuojami į HTML, yra galimybė šifruoti atskirus XML dokumento elementus, palaiko gimiringus el. parašo standartus XMLDsig ir XAdES (šie veikia pagal X.509 standarto schemą), kurie naudojami Lietuvoje el. valdžios paslaugoms teikti [7]. Paslaugos užsakymui ar prašymui suteikti kokią nors paslaugą, dažniausiai pateikiamos HTML formos (gali būti transformuota iš XML), kuriose į laukus įvedami reikiami duomenys. Iš šių formoje įvestų duomenų, naudojant formos laukus atitinkančias žymas, gali būti formuojamas XML dokumentas.

XML dokumentams elektroninėje erdvėje apsaugoti dažniausiai naudojamas dokumento ar jo elemento šifravimas bei skaitmeninis parašas.

1.4.1 XML skaitmeninis parašas

XML skaitmenis parašas yra skaitmeninio parašo forma, apibrėžta XML standarto taisyklėmis. XML skaitmeninis parašas leidžia autentifikuoti XML dokumentą ir užtikrinti jame pateiktos informacijos vientisumą vykdant XML ir žiniatinklio paslaugų tranzakcijas (naudojamas SOAP, SAML technologijų). Savo funkcionalumu XML parašas yra panašus į PKCS#7 (naudojamas S/MIME), tačiau turi didesnes plėtimo galimybes ir pritaikytas XML dokumentams. XML skaitmeninis parašas turi keletą parametrų, nurodančių, kokio formato ir kokius algoritmus naudojant jis buvo suformuotas. Galima pasirašyti visą dokumentą arba keletą skirtingų (netgi skirtingo turinio – XML arba dvejetainio) dokumento elementų. Pasirašomi duomenys identifikuojami pagal nurodytus identifikatorius (*angl. URI, Uniform Resource Identifiers*).

XML parašai gali būti trijų tipų:

- Atskirtasis parašas (*angl. Detached signature*). Parašas XML elementui. esančiam *Signature* elemento išorėje (gali būti visai kitame dokumente ar duomenų bazėje, arba tame pačiame dokumente gretimas konkretus elementas (ar keli elementai, identifikuojami pagal *URI*)).
- Gaubiamasis parašas (*angl. Enveloping signature*). Parašas, skirtas visiems XML elementams, kurie yra *Signature* elemento viduje.

- Apgaubtasis parašas (*angl. Enveloped signature*). Parašas skirtas *Signature* elemento tėviniam (esančiam vienu hierarchiniu lygmeniu aukščiau, *angl. parent element*) XML elementui.

XML parašų formavimą ir naudojimą apibrėžia standartai. Vienas iš jų, W3C konsorciumo parengtas ir rekomenduojamas – XMLDSig [23].

XMLDSig

XML dokumente šio standarto parašą atvaizduoja *Signature* elementas, kurio hierarchinė struktūra pavaizduota 7 paveiksle.

```
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
        <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
        <ds:Reference URI="">
            <ds:Transforms>
                <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
            </ds:Transforms>
            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue/>
        </ds:Reference>
        <ds:Reference URI="http://www.w3.org/TR/xml-stylesheet">
            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue/>
        </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue/>
    <ds:KeyInfo>
        <ds:X509Data>
            <ds:X509Certificate/>
        </ds:X509Data>
        <ds:KeyValue>
            <ds:RSAKeyValue>
                <ds:Modulus/>
                <ds:Exponent/>
            </ds:RSAKeyValue>
        </ds:KeyValue>
    </ds:KeyInfo>
</ds:Signature>
```

7 pav. XMLDSig parašo struktūra

Elemento *SignedInfo* turinį sudaro nuorodos į pasirašytą turinį ir elementas, skirtas nurodyti, koks algoritmas naudotas suformuoti parašą.

Elementuose *SignatureMethod* ir *CanonizationMethod* nurodomas naudotas parašo formavimo metodas (pvz. RSA su SHA1 santrauka) ir *SignedInfo* elemento kanonizavimo algoritmas prieš įterpiant šį elementą į dokumentą.

Toliau, elementais *Reference* nurodomi pasirašomi resursai (nuoroda į resursą, reikalinga transformacija, santraukos algoritmas bei sudaryta resurso santrauka).

Elemento *Signaturevalue* turinys – Base64 koduota parašo reikšmė, sugeneruota pagal *SignatureMethod* nurodytą metodą, prieš tai pasirašomą turinį transformavus *CanonizationMethod* nurodytu metodu.

Elemente *KeyInfo* pasirinktinai galima nurodyti viešąjį raktą (paprastai X.509 standarto sertifikatų forma), kuriuo gavėjas galėtų patikrinti dokumento autentiškumą.

Jei naudojamas gaubiamasis parašas, papildomai įterpiamas *Object* elementas, kuriame patalpinamas visas pasirašytas turinys.

Tikrinant, ar XML aprašas galioja, atliekamos tokios procedūros:

- Nuorodų turinio tikrinimas. Kiekviena *References* elemente nurodyta santrauka yra patikrinama gaunant URI adresu nurodytą resursą, pritaikant jam nurodytą santraukos formavimo algoritmą ir sutikrinant santraukų reikšmes. Dokumento turinys yra nepakitęs, jei gauta santraukos reikšmė sutampa su įterptąja į dokumentą.
- Parašo tikrinimas. *SignedInfo* turinys yra apdorojamas naudojant nurodytą kanonizavimo metodą ir pritaikius *SignatureMethod* naudojamą metodą, *KeyInfo* elemente naudotu viešuoju raktu yra patikrinama, ar parašas atitinka nurodytą santrauką. Parašas galioja, jei iššifruota santrauka sutampa su įterptąja į dokumentą.

Dokumentas laikomas autentišku, jei dokumente nurodyta santrauka sutampa su iššifruotąja parašo tikrinimo metu.

XMLDSig yra plačiai naudojamas elektroninių tranzakcijų dokumentams apsaugoti, tačiau šiame standarte neapibrėžti kiti parametrai, reikalingi dokumentų įsigaliojimui teisiškai (t.y. kad jie galėtų teisiškai atitikti rašytinius dokumentus ir juos pakeisti). Šiai problemai spręsti, 1999 m. buvo išleista Europos Sąjungos Direktyva, kurioje numatyta sukurti ir e-valdžios infrastruktūroje naudoti pažangius skaitmeninius parašus [5] . Europos Standartizacijos Komitetas (CEN) ir Europos Telekomunikacijų Standartų Institutas (ETSI) parengė grupę standartų, aprašančių techninius ir programinius pažangių skaitmeninių parašų sudarymo principus ir priemones, Bendrai šie parašai vadinami CAdES (*angl. CMS Advances Electronic Signatures*). XML dokumentams saugoti pažangus skaitmeninių parašų standartas vadinamas XAdES. Tai XMLDSig plėtinys, apibrėžiantis XML schemą su naujais elementais, papildančiais XMLDSig parašą [4].

XAdES

Šiame XML parašo standarte numatytos tokios papildomos dokumentų apsaugos priemonės (*angl. Qualifying properties*):

- Parašo naudojimo politika (*angl. Signature policy identifier*). Papildomi įrašai, nurodantys parašo naudojimo aplinkybes.
- Papildomos duomenys ar žymos parašo tikrinimui (*angl. Validation data properties*). Į parašą įtraukiama visa galima informacija dokumento autentiškumui patikrinti. Tai sertifikatų grandinėlės, nuorodos į sertifikatų sąrašus (*angl Certificate Revocation Lists, CRL*). XAdES leidžia įterpti į dokumentą tiek nuorodas į tikrinimui naudojamus resursus, tiek pačius resursus.
- Laiko žymos. Ši priemonė skirta sudaryti galimybę dokumento patikrai ilgalaikiam laikotarpyje. Įterpta parašo suformavimo laiko žyma nurodo, ar parašo sertifikatas galiojo pasirašymo metu. Laiko žyma gali būti taikoma atskiroms dokumento dalims (atskiros dokumento dalys gali būti pasirašomos skirtingais laiko momentais).

Šios papildomos apsaugos priemonės į XML dokumentą įterpiamos kaip *Object* elementas. XAdES standarto parašo *Signature* elementas pavaizduotas 8 paveiksle.

```
<ds:Signature Id="SignatureId" xmlns:ds="http://www.w3.org/2000/09/xmldsig#>
    <ds:SignedInfo >
        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
        <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
        <ds:Reference URI="http://www.sample.org/docToBeSigned">
            <ds:DigestMethodAlgorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue >..<./ds:DigestValue >
        </ds:Reference >
        <ds:Reference URI="#SignedPropertiesId"Type="http://uri.etsi.org/01903#SignedProperties">
            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue >..<./ds:DigestValue >
        </ds:Reference >
    </ds:SignedInfo >
    <ds:SignatureValue >..<./ds:SignatureValue >
    <ds:KeyInfo >..<./ds:KeyInfo >
    <ds:Object >
        <QualifyingProperties Target="#SignatureId"
            xmlns="http://uri.etsi.org/01903/v1.3.2#"
            xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
            <SignedProperties Id="SignedPropertiesId">
                <SignedSignatureProperties >
                    <SigningTime >..<./SigningTime >
                    <SigningCertificate >..<./SigningCertificate >
                </SignedSignatureProperties >
            </SignedProperties >
        </QualifyingProperties >
    </ds:Object >
</ds:Signature >
```

8 pav. XAdES standarto parašas

Elementas *QualifyingProperties* į dokumentą gali būti įterpiamas tiesiogiai į elementą *Object*, arba netiesiogiai - įterpiant nuorodą *QualifyingPropertiesReference* [4].

XML dokumento parašo suformavimas patikrinimas atliekamas tokia pat tvarka, kaip ir XMLDSig, tačiau po parašo suformavimo vykdomos papildomos procedūros, suformuojančios ir į dokumentą įterpiančios papildomus elementus su tikrinimui reikalinga informacija.

1.4.2 XML šifravimas

Ši priemonė skirta užtikrinti XML dokumento turinio konfidencialumą. XML dokumentų šifravimas yra apibrėžtas standartu XMLEnc, aprašytu W3C dokumente [24]. Šio standarto specifikacijoje apibrėžta šifruoto XML dokumento schema (pagal kurią atliekamos dokumento transformacijos), numatytos galimybės šifruoti tik elemento turinį arba visą elementą su atributais. Šifravimas gali būt atliekamas tiek asimetriniu, tiek simetriniu raktu.

Užšifruota XML dokumento dalis talpinama elemente *EncryptedData*. Šio elemento struktūra pavaizduota 9 paveiksle.

```
<xenc:EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Content"
        xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
  <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
  <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <xenc:EncryptedKey xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
      <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p"/>
      <xenc:CipherData>
        <xenc:CipherValue>..</xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedKey>
  </ds:KeyInfo>
  <xenc:CipherData>
    <xenc:CipherValue>...</xenc:CipherValue>
  </xenc:CipherData>
</xenc:EncryptedData>
```

9 pav. XML dokumento šifruotą dalį vaizduojantis elementas

Elemento *EncryptionMethod* atributu *Algorithm* nurodoma, koks šifravimo algoritmas naudotas turiniui užšifruoti.

Kaip ir XMLDSig, elemente *KeyInfo* įterpta informacija apie naudotą ir kartu su dokumentu pateikiamą šifravimo raktą. Paveiksle pavaizduotu atveju simetrinis dokumento šifravimo raktas užšifruotas asimetriniu gavėjo viešuoju raktu. Informacija apie šifruotą raktą pateikta elemente *EncryptedKey*. Tai rakto šifravimo metodas (elementas *EncryptionMethod*) ir rakto kriptograma (elementas *CipherData*).

Užšifruotas dokumento turinys pateikiamas elemente *CipherData*.

XML šifravimas – viena iš priemonių, galinti pakeisti SSL/TLS, kai reikalingas konfidencialumo užtikrinimas, jį perduodant gavėjui per keletą tarpininkų.

Toliau darbe pateikiama keletas metodų ir jais pagrįstų sprendimų, kaip el. tranzakcijų dokumentai (taip pat ir XML tipo, su XAdES parašu) gali būti pasirašyti prijungties režimu (*angl. online*) ir kaip gali būti panaudoti el. paslaugų sistemose, apžvelgta 1.5 skyriuje.

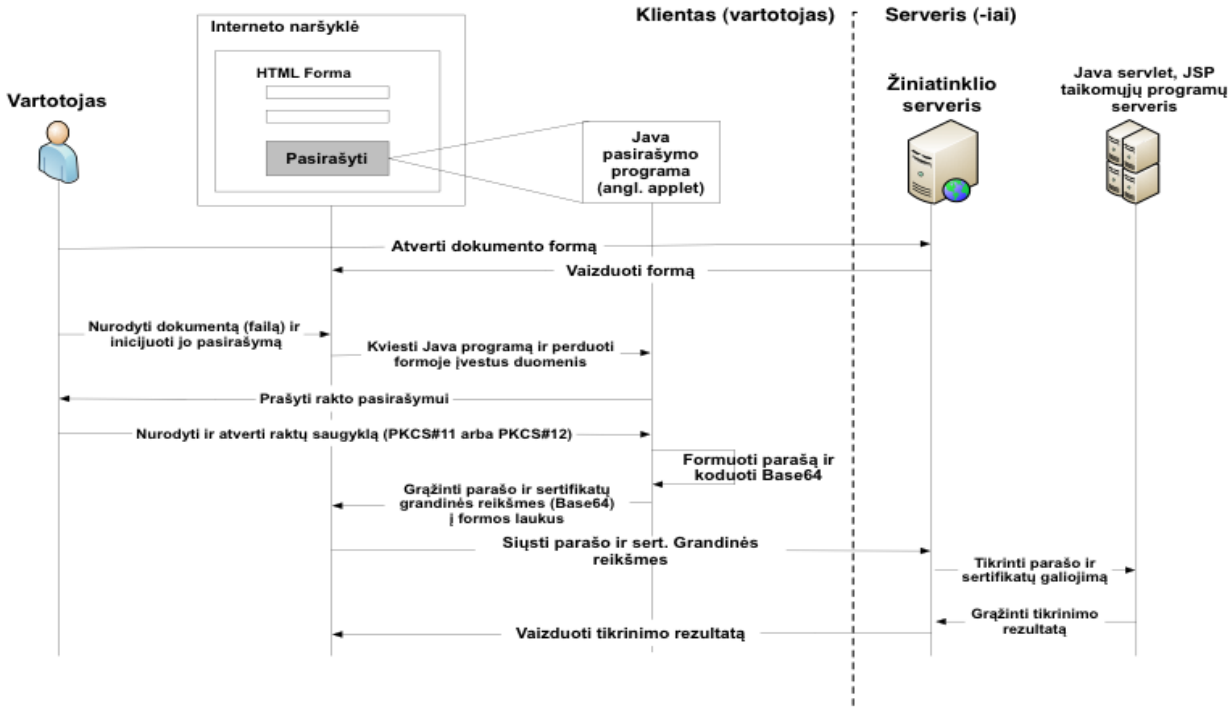
1.5 Egzistuojančių el. dokumentų pasirašymo žiniatinklio aplinkoje sprendimų apžvalga

Šioje dalyje apžvelgta keletas el. dokumentų pasirašymo žiniatinklio aplinkoje sprendimų, paremtų X.509 standartu sukurtomis technologijomis. Vertinamas jų tinkamumas iškeltai problemai spręsti, privalumai, trūkumai. Į tai bus atsižvelgta kuriant savo metodą ir juo paremtą sprendimą.

1.5.1 Nakov Document Signer

Tai sprendimas, kuriuo galima pasirašinėti nurodytus dokumentus (failus), ir kurį galima integruoti į sistemas, veikiančias Java žiniatinklio programų pagrindu. Sprendimas yra atviro kodo (autorius Svetlin Nakov, Sofijos St. Kliment Ohridski Universitetas), paremtas viešojo rakto kriptografija. Jį sudaro dvi dalys: Java pasirašymo programa (*angl. applet*) dokumento pasirašymui (nurodyto failo) ir žiniatinklio programa parašo ir sertifikato tikrinimui. Pasirašymui gali būti naudojama PKCS #12 formato raktų saugykla arba sumanioji kortelė (*angl. smart card*, PKCS #11 formato raktų saugykla). Visi raktai ir sertifikatai atitinka X.509 standartą [15, 16].

Metodo, kuriuo pagrįstas sprendimas, schema pavaizduota 10 paveiksle.



10 pav. Nakov Document Signer veikimo principas

Šis pasirašymo metodas išnaudoja žiniatinklio dokumentų formų laukus kaip galimą terpę parašui ir sertifikatų grandinei grąžinti. Tokiu būdu Java programai nereikia kurti jungčių su serverio dalyje „budinčia“ serverio programa, kad perduotų parašą ir informaciją apie pasirašantįjį [15, 16].

Viena pagrindinių metodo savybių, dėl kurių jis nėra tinkamas žiniatinklio paslaugų sistemoms – XML parašo nepalaikymas. Kitaip sakant, sprendimas tiesiog formuoja nurodyto HTML formoje failo SHA1 santrauką, vykdo jos pasirašymą nurodytu privačiu raktu ir grąžina šifruotą parašo bei sertifikatų grandinės reikšmę į atitinkamus formos laukus.

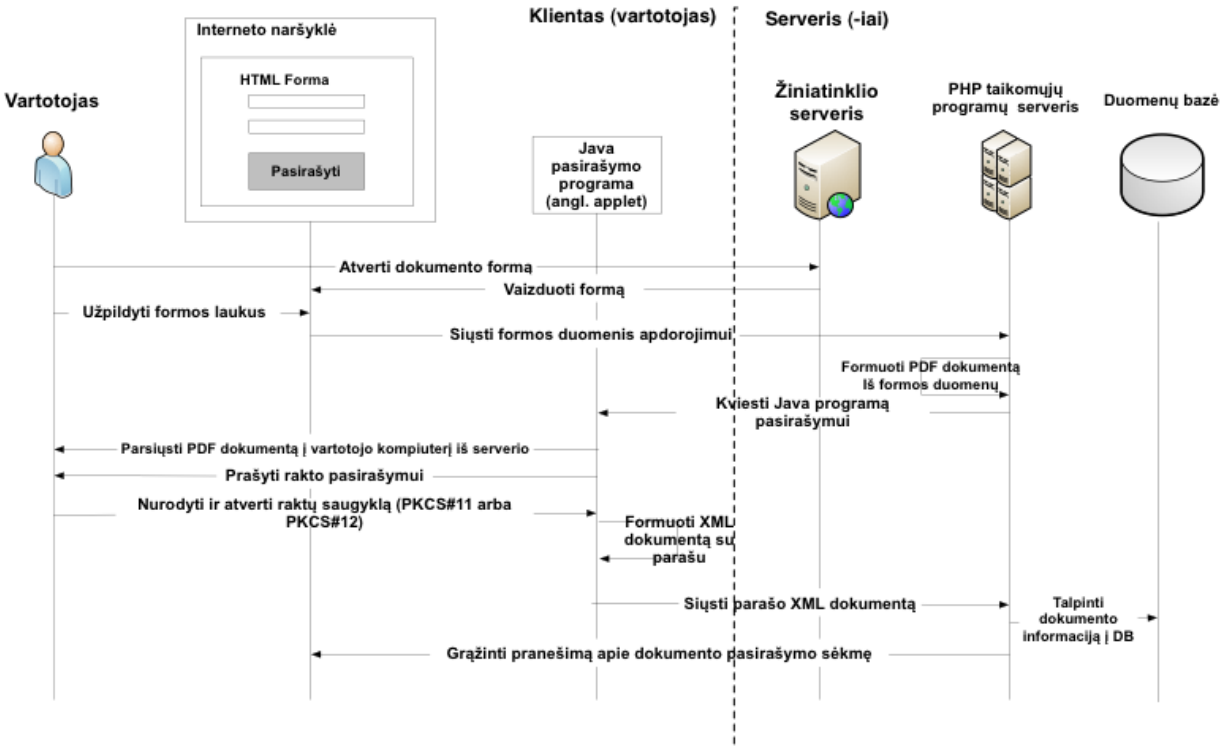
Kita savybė – tik Base64 kodavimas, kuris neužtikrina reikiamo tranzakcijų dokumentų konfidencialumo. Perėmus srautą, dokumentai gali būti lengvai iškoduoti.

Apskritai pats sprendimas nėra išbaigtas ir labiau naudotinas kaip karkasas tam tikroms funkcijoms realizuoti ir panaudoti kuriamoje sistemoje (pvz. raktų paėmimui iš saugyklos arba sumaniosios kortelės). Pats metodas iš esmės labiau tinkamas dokumentų valdymo sistemoms.

1.5.2 WebSign Project

Tai el. dokumentų pasirašymo žiniatinklio aplinkoje sprendimas, sukurtas R. Cardon de Lichtbuer (Belgijos Karališkoji karo akademija). Šiuo įrankiu galima pasirašinėti el. dokumentus pateikiamus interneto naršyklėje vartotojo privačiu raktu (saugomu Windows raktų saugykloje, saugyklos PKCS #12 formato byloje arba sumaniojoje belgiškoje tapatybės kortelėje). Formuojamas parašas yra XML formato ir atitinka XAdES standartą (tokių XML dokumentų apdorojimui naudojama *Apache XML Security* biblioteka). Sprendimui adaptuotas *Microsoft CryptoAPI* karkasas, turintis reikiamas funkcijas ir sąsajas su bibliotekomis, kurios sudaro galimybę naudotis Windows raktų ir sertifikatų saugykla bei sumaniosiomis kortelėmis.

Metodo, kuriuo pagrįstas sprendimas, schema pavaizduota 11 paveiksle.



11 pav. WebSign Project veikimo principas

Šiame metode daugiausiai užduočių atlieka Java programa: parsiunčia PDF dokumentą, sudarytą iš formos duomenų, skaito raktus ir sertifikatus iš nurodytos saugyklos, kurią taip pat galima pasirinkti (Java raktų saugykla, Windows raktų saugykla, PKCS #12 formato failas arba sumanioji kortelė), Formuoja XML XAdES standartą atitinkantį parašo dokumentą, jungiasi su serverio programa ir persiunčia jai tik suformuotą parašą (paties dokumento duomenys jau būna išlikę serveryje). Toliau jau serverio programa tikrina parašą, talpina informaciją apie tranzakciją į duomenų bazę ir praneša vartotojui apie tranzakcijos sėkmę.

Pagrindinis šio metodo trūkumas – resursų eikvojimas PDF dokumento formavimui ir siuntimui į vartotojo kompiuterį. Be to, patį siuntimą atlieka Java programa, nenaudodama jokių saugos priemonių, pats dokumentas ir jo parašas nėra šifruojamas. Viena išeičių tokiu atveju, organizuoti saugų ryšį tarp kliento ir serverio naudojant kurį nors transporto ar tinklo lygmens saugos protokolą.

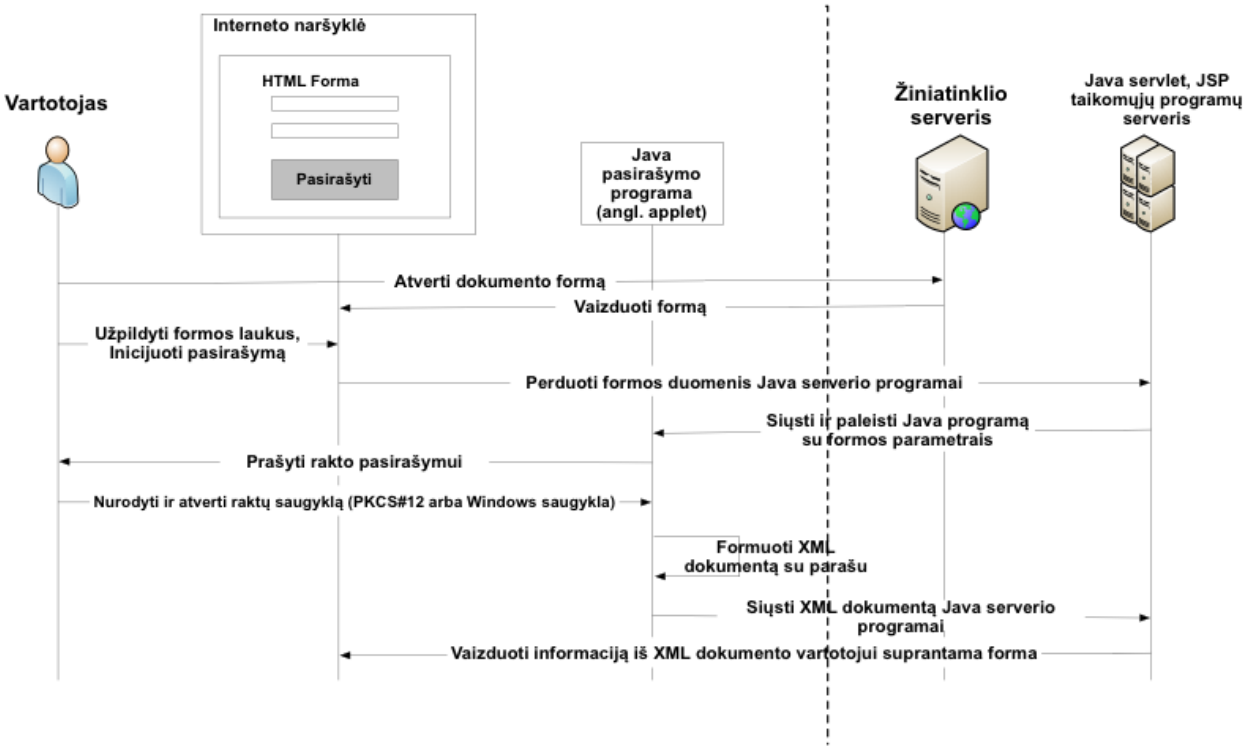
Taigi šiame sprendime formuojamas pažangus XML XAdES standarto parašas, naudinga ir lanksti galimybė pasirinkti parašo saugyklą. Dėl savo trūkumų metodas naudotinas kartu su transporto ar tinklo lygmens saugos protokolais organizacijų ar jų tinklų viduje, bet ne plačiai viešosioms paslaugoms teikti [25].

1.5.3 OpenSign

Tai *OpenCES* projekto, skirto padėti pagrindus sertifikatais paremtų sprendimų kūrimui Danijos piliečiams ir kuruojamo Danijos valstybinės IT ir telekomunikacijų agentūros, dalis. *OpenSign* sudaro dvi Java programos (*angl. applet*), kurių viena skirta vartotojo autentifikacijai jungiantis prie interneto svetainės, kita – el. dokumentų pasirašymui žiniatinklio aplinkoje [G26]. Pastaroji ir buvo apžvelgta.

Java programa dirba su X.509 standarto sertifikatais, raktus ir sertifikatus gali skaityti iš PKCS#12 formato saugyklos, Windows raktų ir sertifikatų saugyklos (tam naudojama sąsaja su *CryptoAPI*). Vienas pagrindinių kūrėjų tikslų buvo realizuoti „ką matau, tą pasirašau“ principą (*angl. what you see is what you sign*), tad programa turi galimybę formuoti parašą ne tik formoje įvestam tekstui, bet ir prisegtiems failams (jei kartu į serverį siunčiamas koks nors failas). Tokiu atveju tekstas ir siuntinys laikomi vienu dokumentu.

Metodo, kuriuo pagrįstas demonstracinis sprendimas, schema pavaizduota 12 pav.



12 pav. OpenSign veikimo principas

Iš schemos matyti, kad pasirašymo programą valdo (paleidžia ir perduoda darbinius nustatymus) serverio programa. Į formą įvesti duomenys prieš tai serverio programai perduodami atvirai, tad šiuo atveju reikalinga papildoma duomenų sauga perdavimo metu. Tam galima naudoti kurį nors transporto ar tinklo lygmens saugos protokolą.

Skirtingai nuo *WebSign Project*, šis sprendimas formuoja pilną XML dokumentą (su formos duomenimis, prisegtais failais ir parašu), todėl suformuotu XML failu galima pilnai

operuoti kaip su tranzakcijos dokumentu (*WebSign* formuoja XML dokumentą, kuriame yra tik parašas). Dabartine *OpenSign* versija galima formuoti parašą tik grynam tekstui (*angl. plain text*), tačiau to pakanka el. tranzakcijų dokumentams, kurių turinys paprastai būna grynasis tekstas, pirmiausiai įvedamas žiniatinklio formoje.

OpenSign demonstraciniame sprendime nėra realizuotas XML dokumento parašo tikrumo patikrinimas, tačiau *OpenCES* karkasas turi reikiamas bibliotekas tikrinimo komponento realizacijai serverio dalyje.

Taigi *OpenSign* iš principo gali būti (ir yra) naudojamas kaip el. dokumentų pasirašymo modulis el. paslaugų sistemose. Demonstracinis sprendimas nėra pilnavertis, tačiau *OpenCES* karkasas sudaro galimybę sprendimą tobulinti, pritaikyti savo reikmėms, tam tikrai probleminei sričiai [26].

1.5.4 Sprendimų apžvalgos išvados

Apžvelgus tris skirtingus el. tranzakcijų dokumentų pasirašymo žiniatinklio aplinkoje sprendimus, galima daryti šias išvadas:

- Java technologija yra pakankamai išvystyta (turi reikiamas bibliotekas, pasirašytos Java programos gali skaityti vartotojo failų sistemą ir tiesiogiai bendrauti su serverio dalimi) ir gali būti tinkamas pasirinkimas panašiems sprendimams kurti;
- Išbandžius sprendimų realizacijas, įsitikinta, kad XML standartas yra tinkamas tranzakcijų dokumentams formuoti ir saugoti, nes turi lanksčias el. parašo (XMLDsig ir XAdES) ir šifravimo priemonių taikymo galimybes (dalinis dokumento šifravimas, keletas parašų ir pan.);
- Kiekvienas iš apžvelgtų metodų turi savų privalumų, kuriais verta pasinaudoti kuriant metodą nagrinėjamai el. tranzakcijų dokumentų apsaugos „debesies“ architektūros sistemoje problemai spręsti.

Ši apžvalga ir jos išvados svarbios formuluojant darbo tikslą ir uždavinius, nes nurodo sprendimo kūrimo gaires.

1.6 Darbo tikslas ir uždaviniai

Remiantis atlikta el. tranzakcijų dokumentų apsaugos poreikio ir problemų analize, atsižvelgiant į apžvelgtų duomenų saugos technologijų, esamų saugos metodų privalumus ir trūkumus, iškeltas darbo tikslas ir uždaviniai.

Tikslas - remiantis esamų saugos technologijų ir metodų privalumais sukurti el. tranzakcijų dokumentų pasirašymo metodą, sudarantį ir gebantį šifruoti ir XMLDSig bei XAdES parašo standartus atitinkančius el. tranzakcijų dokumentus. Atlikti metodo kiekybinius ir kokybinius tyrimus ir, remiantis jų rezultatais, padaryti išvadas dėl metodo tinkamumo naudoti „debesies“ architektūros aplinkoje bei spręsti joje kylančias, pirmoje darbo dalyje įvardintas dokumentų apsaugos problemas.

Uždaviniai:

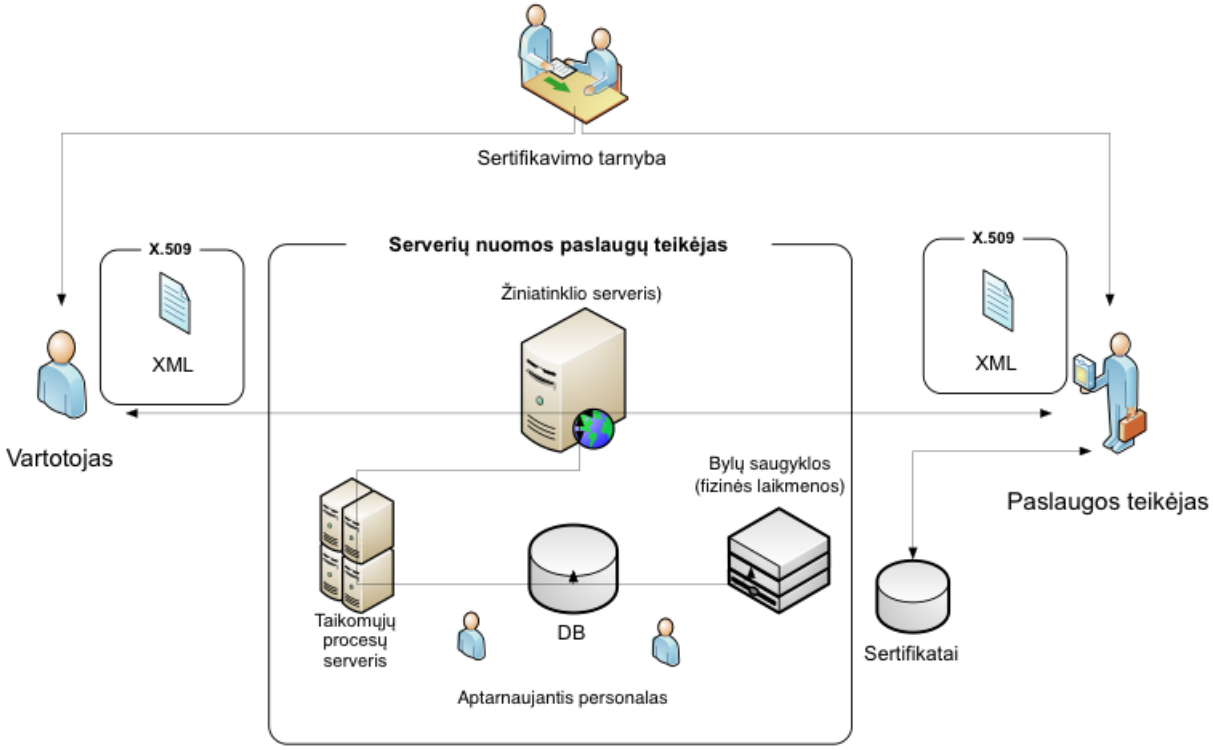
- Parengti metodo veikimo žiniatinklio aplinkoje koncepciją (veikimo principą, reikalavimus);
- Sudaryti metodo veikimo schemas, parodančias, kokia tvarka, būdais ir priemonėmis suformuojami taisyklingi pasirašyti ir užšifruoti XML dokumentai;
- Pagal sudarytas veikimo schemas parengti žiniatinklio aplinkoje veikiantį ir tinkantį tyrimams metodą realizuojančio sprendimo prototipą. Prototipas privalo turėti galimybę naudoti įvairias raktų ir sertifikatų saugyklas, išorines bibliotekas su šifravimo ir parašo formavimo procedūromis.
- Naudojant parengtą prototipą, išmatuoti ir įvertinti raktų ir sertifikatų nuskaitymo iš skirtingų tipų saugyklų greitaveiką. Taip pat išmatuoti bazinio XMLDSig ir XAdES standartų parašo formavimo bei tikrinimo, užšifravimo bei iššifravimo algoritmais procedūrų greitaveiką, matavimus atliekant su skirtingo dydžio XML dokumentais. Remiantis matavimų rezultatais, įvertinti sukurto metodo įvairius panaudojimo atvejus (didelių dokumentų pasirašymas ir šifravimas, dalies dokumento pasirašymas ir šifravimas, prioritetų teikimas tam tikro tipo raktų saugykloms).
- Atlikti metodo kokybinę analizę (įvardinti ir įvertinti galimybes, privalumus bei trūkumus), įvardinti bendrus kokybinius kriterijus ir pagal juos palyginti metodą su kitais, pirmoje darbo dalyje aprašytais metodais.

Uždavinių sprendimas aprašytas specifikacijos, realizacijos bei tyrimo dalyse.

2. METODO SPECIFIKACIJA

2.1 Metodo esmė ir ypatybės

Norimu kurti el. tranzakcijų dokumentų pasirašymo metodu pagrįsta bendroji elektroninių paslaugų teikimo schema pavaizduota žemiau.



13 pav. Bendroji paslaugų, paremtų el. parašu, teikimo schema

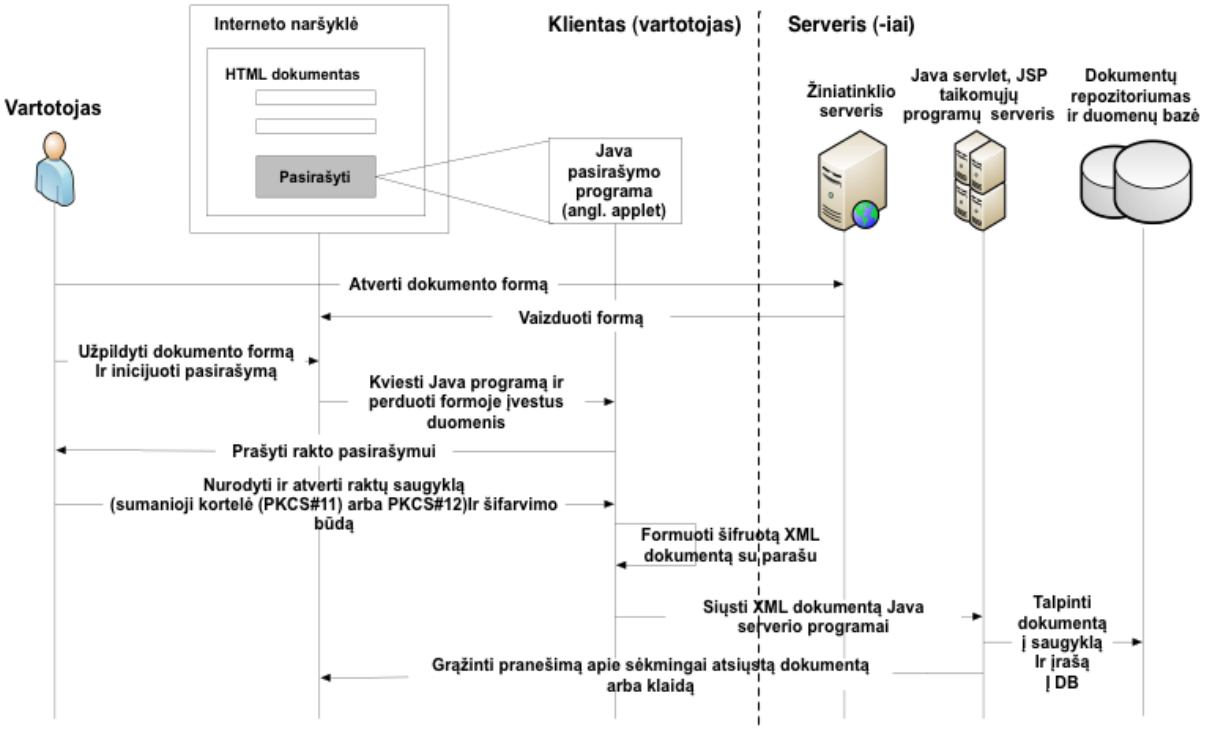
Pagal šią schemą, vartotojas (klientas, partneris) el. tranzakcijos dokumentą pasirašo savo privačiu raktu, dokumentą šifruoja (visą dokumentą arba dalį dokumento, bendru slaptu raktu arba paslaugų teikėjo viešuoju raktu (asimetrinis šifravimas)) ir grąžina į el. paslaugų teikimo sistemą, veikiančią pas interneto ryšio, serverių nuomos paslaugų teikėją (bendrai – „debesyje“). Tačiau tik el. paslaugos teikėjos, turintis šifravimo/iššifravimo raktus ir sertifikatus, gali peržiūrėti tranzakcijų dokumentus ir patikrinti klientų (ar partnerių) parašus.

Šiuo atveju sertifikavimo tarnyba gali būti atskira organizacija, užsiimanti tik raktų ir sertifikatų išdavimu, arba jos funkciją gali atlikti paslaugų teikėjas, savo klientams arba partneriams pats išduodantis raktus ir sertifikatus.

El. tranzakcijų dokumentų pasirašymo metodas kurtas remiantis pirmoje darbo dalyje apžvelgtų jau sukurtų sprendimų privalumais ir vengiant jų trūkumų: dokumento apdorojimas ir pasirašymas vykdomas kliento pusėje, vengiant atskiro dokumento ar informacijos apie jį siuntimo į serverį; dokumentas šifruojamas kliento pusėje ir serveryje laikomas šifruotu pavidalu – tokiu būdu vengiama saugos technologijų naudojimo dokumento siuntimo metu.

Saugos priemonių, naudojamų metode, efektyvumas vertintas atliekant tyrimą (metodika, eiga ir rezultatai aprašyta tolimesnėse darbo ataskaitos dalyse).

El. tranzakcijų dokumentų pasirašymo metodo schemos dalys (komunikacija tarp serverio ir el. paslaugos vartotojo bei komunikacija tarp serverio ir el. paslaugos teikėjo) pavaizduotos ir aprašytos žemiau.



14 pav. Metodo schemos dalis (komunikacija tarp el. paslaugos gavėjo ir serverio)

Užmegzdamas ryšį su serveriu, vartotojas siunčia nešifruotą užklausą žiniatinklio serveriui, kad šis paruoštų ir atsiųstų tam tikrą dokumentą ar formą, ir pateiktų vartotojui interneto naršyklėje.

Serveris sugeneruoja atitinkamą *HTML* puslapį, „prisega“ prie jo paslaugų teikėjo sertifikatu pasirašytą Java programą (*angl. Java applet*) ir siunčia klientui.

Klientas savo naršyklėje patikrina atsiųstos Java programos autentiškumą (kad ši priklauso tikrai tam paslaugų teikėjui, į kurį buvo kreiptasi), palygindamas savo saugykloje laikomą sertifikatą su atsiųstuoju (jei nesutampa, kreipiasi dėl patvirtinimo į patikimą Sertifikavimo Tarnybą (*angl. Certificationa Authority*)).

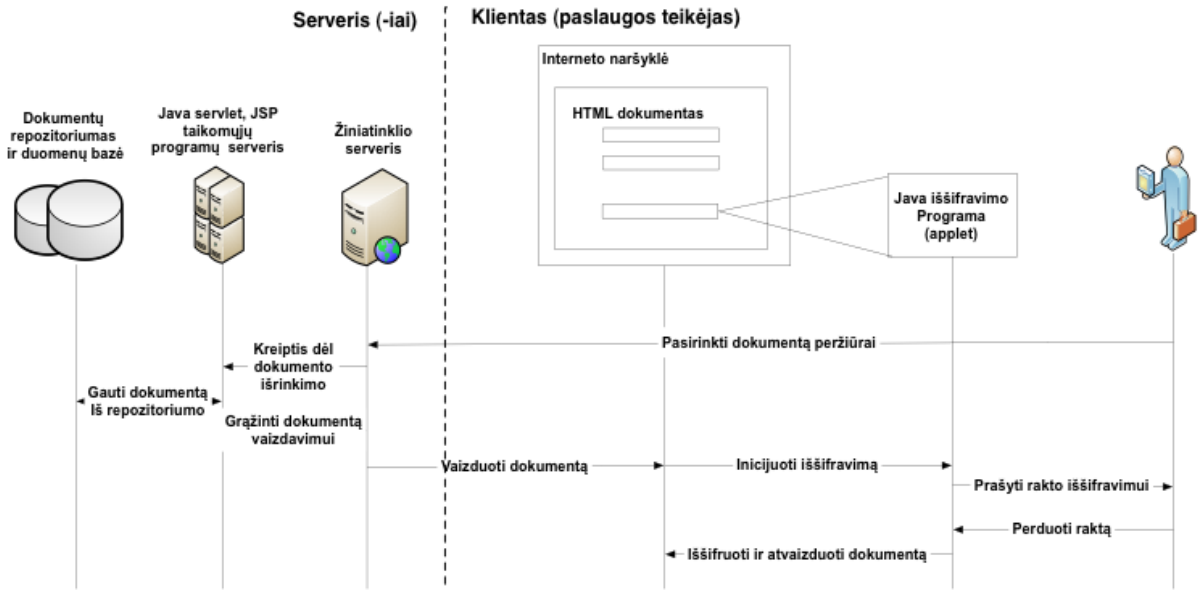
Jeigu autentiškumą pavyksta patvirtinti, vartotojas užpildo, redaguoja ar kitaip modifikuoja atsiųstą dokumentą bei inicijuoja dokumento pasirašymą.

Sertifikuota Java programa turi išskirtines teises kreiptis į vartotojo failų sistemą. Šiuo atveju ji turėtų būti sukurta taip, kad nuskaitytų duomenis iš prijungtos vartotojo asmeninės USB atmintinės ar sumaniosios kortelės, kurioje saugomas el. parašas. Java programa

kreipiasi į saugyklą su parašu, paprašo vartotojo įvesti PIN kodą arba slaptažodį, nuskaito el. parašo failą, užšifruoja dokumentą (arba dalį jo), prideda suformuotą, privačiu raktu koduotą dokumento santraukos funkcijos reikšmę (*angl. hash function*). Po skaitmeninio parašo sudarymo, ryšys su laikmena ar saugykla iš karto nutraukiamas.

Vartotojo užpildytas bei pasirašytas šifruotas (ar dalinai šifruotas) dokumentas siunčiamas serveriui.

Serveris identifikuoja ir autentifikuoja vartotoją pagal savo duomenų bazėje saugomą vartotojo sertifikatą (arba kreipiasi į patikimą sertifikavimo tarnybą). Jei autentifikacija sėkminga, patikrinama santraukos f-ja, ar pranešimas nebuvo modifikuotas. Jei pranešimas buvo modifikuotas, santraukos f-ja bus pakitusi, el. parašas bus pripažintas kaip negaliojantis, nebus įmanoma perskaityti pranešimo. Bet kuriuo atveju pranešama apie įvykusią arba neįvykusią tranzakciją.



15 pav. Metodo schemos dalis (komunikacija tarp serverio ir el. paslaugos teikėjo)

Norėdamas peržiūrėti pasirašytą ir šifruotą dokumentą, paslaugos teikėjas prisijungia prie el. paslaugų valdymo sistemos, išsirenka dokumentą, taikomųjų procesų serveris pagal registracijos įrašą duomenų bazėje kreipiasi į repozitoriumą ir grąžina kliento pusei šifruotą ir pasirašytą XML dokumentą kartu su Java iššifravimo programa.

Vaizdavimo metu, pirmiausiai paleidžiama paslaugų teikėjo sertifikatu apsaugota Java programa, pareikalaujanti rakto dokumento iššifravimui (jei dokumentas šifruotas paslaugos teikėjo viešuoju raktu arba bendru slaptu raktu, arba tiesiogiai vykdo iškodavimą, jei dokumentas koduotas Base64).

Iššifruotą dokumento informaciją ir informaciją apie parašą (galioja, ar negalioja, kas išdavė sertifikatą parašui ir pan.), Java programa grąžina interneto naršyklei atvaizduoti vartotojui suprantamu formatu.

Pastebėti metodo privalumai:

- Iš esmės sprendžia dokumentų apsaugos problemą, kai el. paslaugų sistema veikia „debesies“ architektūros aplinkoje;
- Tiesioginė vartotojo identifikacija ir autentifikacija;
- Nebūtina naudoti transporto ar tinklo lygmens duomenų saugos protokolų;
- Formuojami dokumentai atitinka plačiai naudojamą XML standartą, naudojami pažangūs XAdES standarto parašai.
- Neturi kai kurių trūkumų, kuriuos turi pirmoje dalyje apžvelgti metodai.

Metodo trūkumai:

- Siaura taikymų sritis;
- Specialios techninės ir programinės įrangos poreikis;

Dėl aukščiau įvardintų trūkumų pasiūlytas metodas negali ir iš esmės nėra skirtas visiškai pakeisti kitus saugos protokolus, tačiau iš principo gali būti tam tikrais atvejais naudojamas atskirai arba kartu su kitais saugos metodais. Pastaruoju atveju gali būti ne tik panaikinami kai kurie kitų metodų trūkumai, bet užtikrinama ir ypatinga siunčiamos informacijos apsauga. Tačiau tokiu atveju smarkiai gali nukentėti duomenų perdavimo sparta, taip pat ir metodo efektyvumas. Duomenų apsaugos metodo Java programa efektyvumą galima įvertinti pamatavus metodo greitaveiką, kai atliekamas viso ar dalies dokumento šifravimas simetriniu ir asimetriniu raktu.

2.2 Reikalavimai metodui

2.2.1 Funkciniai reikalavimai

Metodas turi atlikti žemiau išvardintas funkcijas:

- Pateikti (perduoti ir korektiškai atvaizduoti) tranzakcijos dokumentą vartotojui.
- Pasirašymo ir šifravimo procedūroms naudoti pasirašytą (sertifikuotą) programą (*angl. applet*).
- Inicijuoti prieigą prie vartotojo privataus rakto dokumento pasirašymui.
- Formuoti pasirašomo dokumento turinio santrauką naudojant SHA-1 algoritmą.
- Veikti pagal X.509 standarto el. parašo schemą. Naudoti parašo formavimo ir šifravimo algoritmą RSA.
- Formuoti XMLDSig ir XAdES standartus atitinkančius parašus su sertifikatų grandinės hierarchija, laiko žyma.

- Įterpti į dokumentą skaitmeninį parašą ir užšifruoti dokumento turinį (atlikti atitinkamas XML dokumento transformacijas).
- Organizuoti pasirašyto dokumento siuntimą į serverį.
- Tikrinti atsiųsto dokumento autentiškumą (santraukos reikšmę ir parašą) kliento (gavėjo) pusėje.
- Inicijuoti dokumento iššifravimą ir atvaizdavimą gavėjo naršyklėje.

Metodo funkcionavimo aplinkybes nusako nefunkciniai reikalavimai.

2.2.2 Nefunkciniai reikalavimai

Metodo funkcionavimo aplinkybės ir nefunkciniai reikalavimai yra tokie:

- Formuoti ir keisti XML dokumentus, atlikti transformacijas su jais griežtai laikantis XML standarto ir su dokumentu susietos XML schemos.
- Turėti galimybę pasirašyti ir šifruoti XML dokumentų fragmentus.

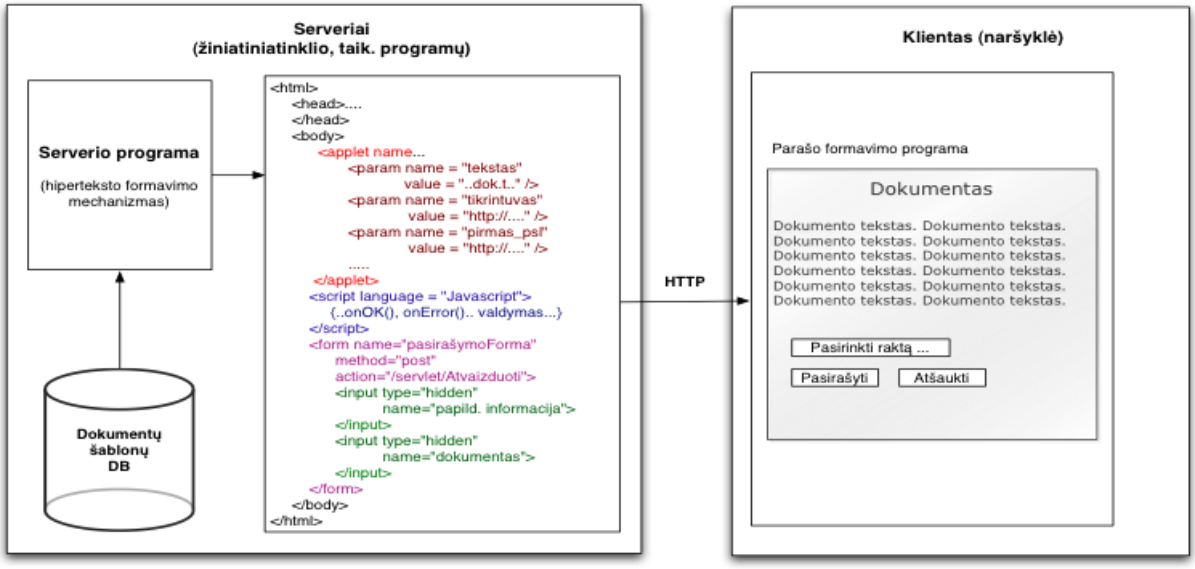
Funkciniai ir nefunkciniai reikalavimai lemia sprendimo metodų ir priemonių pasirinkimą.

2.3 Metodo veikimo schemos

2.3.1 Dokumento šablono pateikimas vartotojui

Dokumento pateikimo vartotojui schema pavaizduota 13 paveiksle.

Dokumentų šablonai su atitinkamu turiniu, kurį turės patvirtinti vartotojas, gali būti sukurti iš anksto ir saugomi kokioje nors duomenų saugykloje. Serverio taikomosios programos funkcija yra formuoti žiniatinklio puslapį su visa informacija ir programine įranga, reikalinga dokumento pasirašymui.

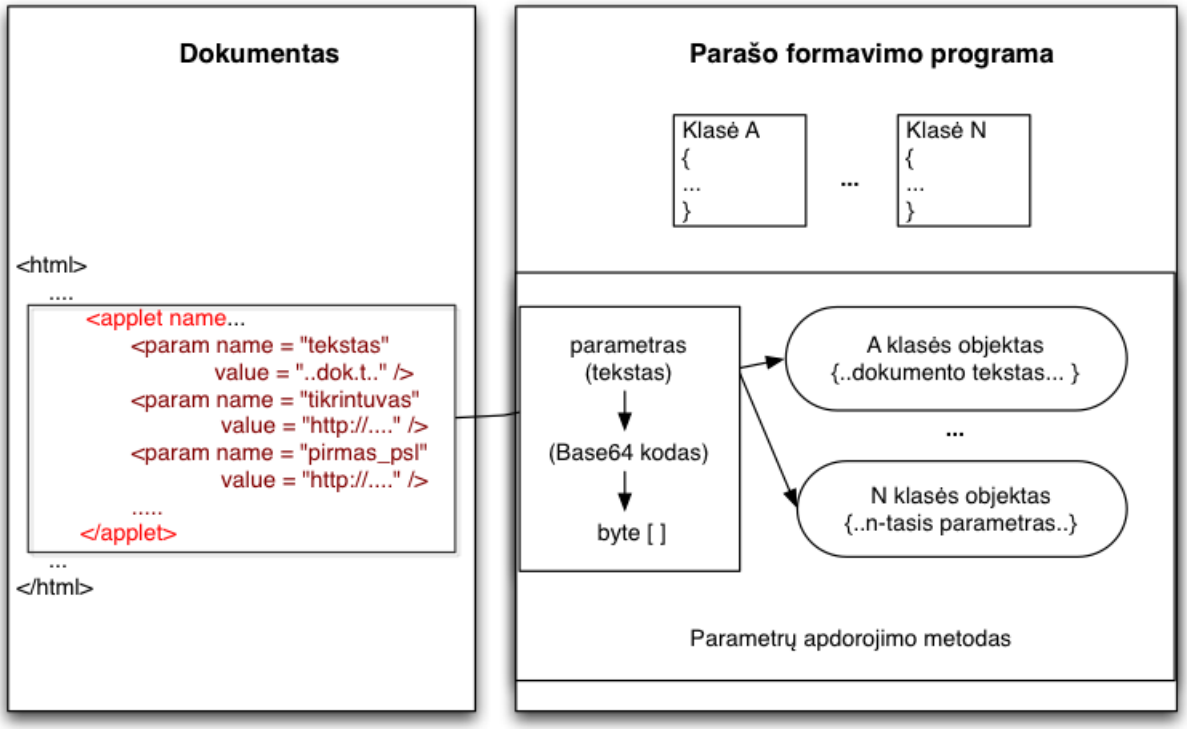


16 pav. Pasirašytino dokumento pateikimo vartotojui schema

Tam, kad žiniatinklio puslapyje pateikto dokumento turinio nebūtų galima redaguoti, jis iš anksto pateikiamas kaip parametras pasirašymo programai. Kaip matyti 16 paveiksle, dokumento tekstas atvaizduojamas jau nebe žiniatinklio puslapyje, o pasirašymo programos duomenų atvaizdavimo komponente, kurio vaizduojamo turinio redaguoti negalima. Kaip parametrus žiniatinklio programai galima pateikti ir kitokius duomenis. Tokiu būdu galima valdyti pasirašymo programos veikimą. Ši galimybė išnaudojama bandymuose, kuomet nurodoma, kiek ciklų vykdyti dokumento pasirašymą matuojant greitaveiką, kokį naudoti raktą ir sertifikatą.

2.3.2 Dokumento apdorojimas ir parengimas pasirašymui

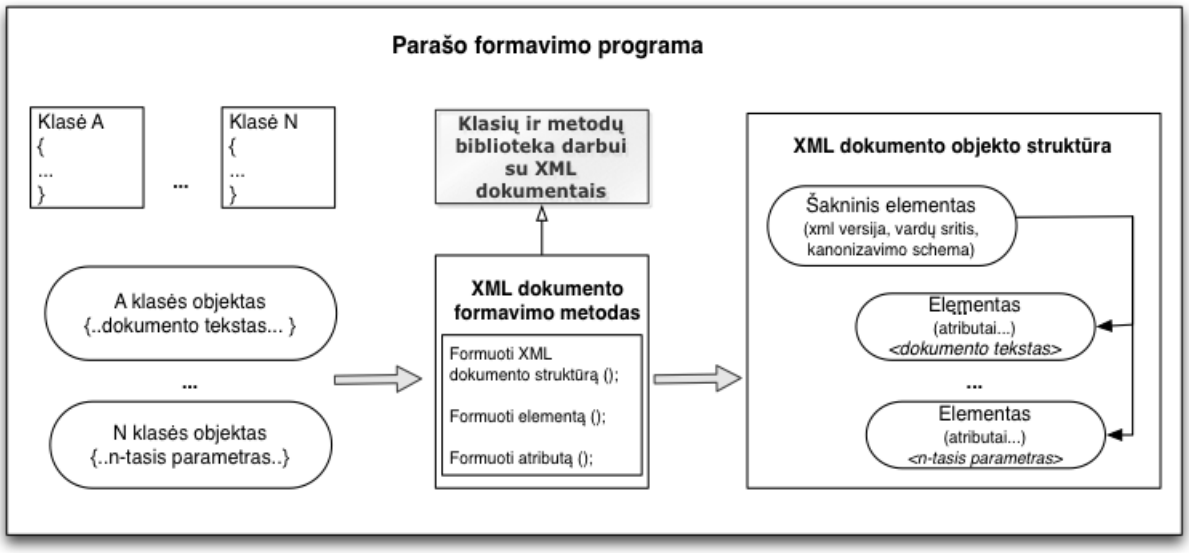
Prieš vykdant dokumento turinio pasirašymą, šis turi būti pakeistas taip, kad su juo būtų galima atlikti įvairias skaičiavimo operacijas. Dėl paprastumo metode dokumento tekstas koduojamas Base64 kodu ir talpinamas į baitų masyvą, su kuriuo jau galima dirbti objektinėje aplinkoje, vykdyti įvairias skaičiavimo operacijas nepaisant duomenų prigimties.



17 pav. Dokumento turinio pavertimas į objektinę struktūrą

Pasirašomoje programoje dėl lankstumo gali būti numatyta aibė įvesties duomenų tipų (skirtingi parametrai programos viduje gali būti skirtingai interpretuojami). Pavyzdžiui dokumento tekstas gali būti programai paduodamas jau XML pavidalo, o papildomi parametrai gali būti skaitinio ar specialaus pavidalo (pvz. data).

Skirtingų tipų duomenys toliau pagal kanonizavimo schemą apdorojami ir pažingsniui, prijungiant kiekvieną programai perduotą parametrą, kuriamas XML dokumento objektas (kaip parodyta 18 paveiksle).



18 pav. Perduotų duomenų pavertimas į XML objektą

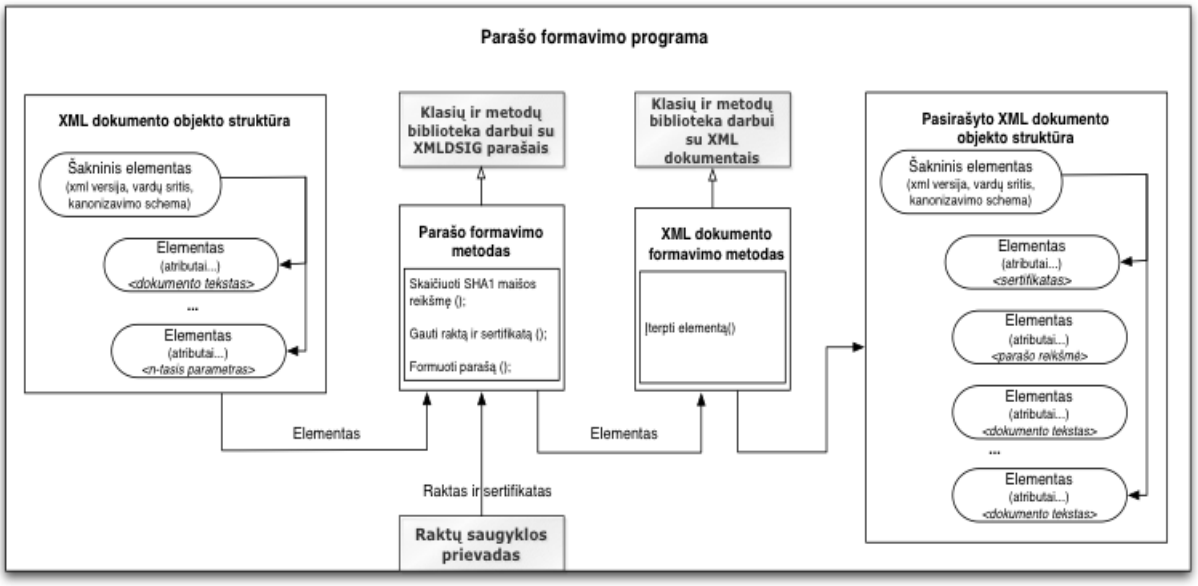
Kiekvienas XML dokumento objektas turi pagrindinius XML dokumentui būdingus parametrus: elemento vardą, atributų ir jų reikšmių aibes ir vaikinių elementų (kurie hierarchiškai yra žemesnio lygmens) aibę. Elemento vardas, atributai, vaikinių elementų vardai yra eilutės tipo ir gali vienareikšmiškai identifikuoti objektą, o šių reikšmės yra Base64 koduota (anksčiau perduota per parametrus) informacija baitų masyvų pavidalu. Tokias struktūras galima kaip parametrus perduoti objektiškai realizuotiems parašo formavimo metodams.

2.3.3 Dokumento pasirašymas ir šifravimas

Dokumento pasirašymas vykdomas perduodant XML dokumento objektą ir papildomus duomenis (maišos algoritmo pavadinimas, parašo schemos pavadinimas, raktas) objektiškai realizuotam parašo formavimo metodui. Maišos algoritmo pavadinimas, parašo algoritmas, raktų saugyklos pavadinimas (modulio ar sąsajos pavadinimas), parašo tipas (gaubiamasis ar apgaubtasis), pasirašytini duomenys programai perduodami kaip parametrai. Tai padidina programos lankstumą: kai kuriuos duomenis automatiškai perduos serverio programa, kai kuriuos gali ar privalo (pavyzdžiui raktą) nurodyti vartotojas.

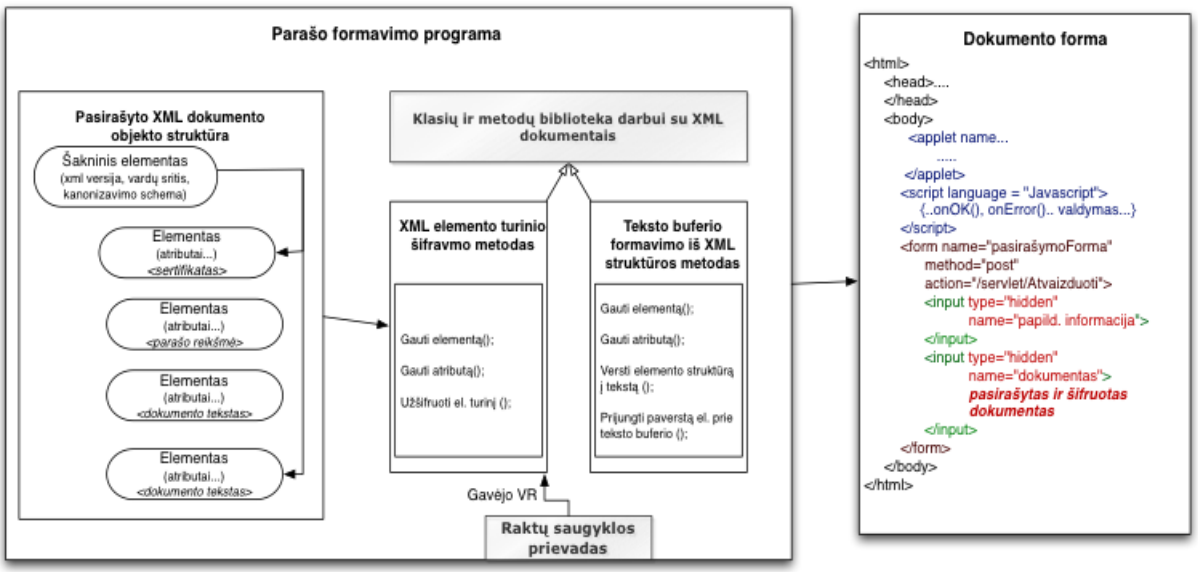
Parašo formavimo metodas realizuojamas panaudojant jau sukurtus ir realizuotus, ištestuotus ir plačiai naudojamus parašo formavimo algoritmus. Tai leidžia daryti prielaidą, kad gauti rezultatai yra patikimi.

Kadangi pasirašomas gali būti ne visas dokumentas, o tik tam tikri elementai, ir po pasirašymo būtina į patį XML dokumentą įterpti papildomas žymas, naudojamas dar vienas metodas. Būtent jis atlieka naujai suformuotų laukų (parašo, santraukos, sertifikatų grandinės reikšmės) įterpimą į XML dokumento objektinę struktūrą. Aukščiau išvardintos procedūros pavaizduotos 16 paveiksle. Jų rezultatas – pasirašyto XML dokumento objektas.



19 pav. XML dokumento pasirašymo procedūros

Toliau pasirašytas dokumentas gali būti šifruojamas (vėlgi, vykdyti šią procedūrą, ar ne, galima nurodyti pasirašymo programai per parametrus). Šifravimas iš esmės gali būti atliekamas su bet kokiu raktu, tačiau atsižvelgiant į raktų valdymo problemą, paprasčiausia naudoti jau turimą, su programa gautą ir sertifikavimo tarnybos patvirtintą, gavėjo viešąjį raktą. Tokiu būdu gali būti užtikrinama, jog dokumentą galės perskaityti tik turintis susietą slaptą raktą subjektas. Šifravimo procedūros ir rezultatas pavaizduoti 20 paveiksle.

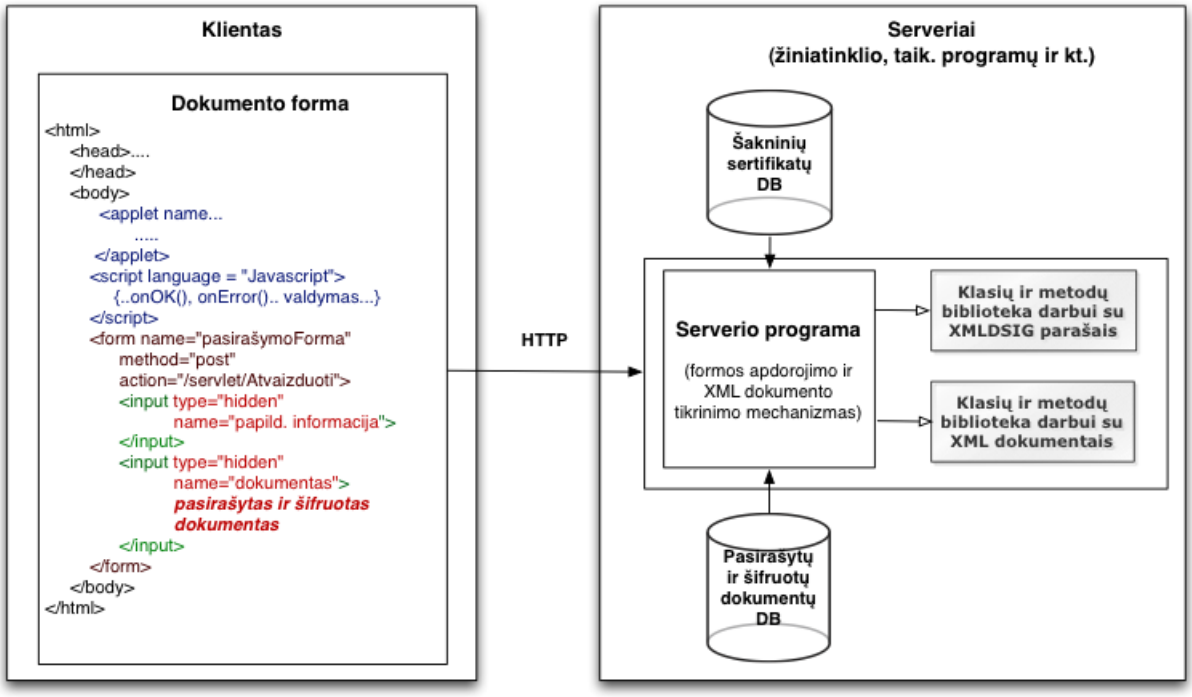


20 pav. Šifravimo procedūros ir rezultatas

Kaip matyti iš schemos, šifravimas yra paskutinė programos procedūra. Po jos visas XML dokumento objekto turinys paelemenčiui verčiamas į tekstinę formą (kaupiamas specialiame buferyje) ir per programos išvesties parametrus, panaudojant scenarijų, grąžinamas į žiniatinklio dokumento paslėptus formos laukus. Šiuo atveju apibrėžti tik du laukai: pasirašymo procedūros sėkmę nusakanti žinutė ir pats pasirašytas dokumentas. Jei dokumento dėl kokios nors klaidos pasirašyti nepavyksta, antrasis laukas lieka tuščias, pirmojo turinys – klaidos pranešimas. Tokių laukų pagal poreikį gali būti ir daugiau, jie pasirašymo rezultatui jokios įtakos nedaro.

2.3.4 Pasirašyto ir šifruoto dokumento siuntimas į serverį

Programa pasirašymo ir šifravimo rezultatus turi persiųsti į serverį. Tam, kad nebūtų sudaroma papildoma jungtis programai tiesiogiai jungiantis su serverio programa (panaudojant *socket* jungtis, dėl ko komunikacijos valdymas taptų tik sudėtingesnis), pasirašymo rezultatai (pasirašytas ir šifruotas dokumentas ar klaidos pranešimas) yra grąžinami į žiniatinklio dokumento paslėptus formos laukus. Rezultatai į serverį išsiunčiami automatiškai *HTTP post* metodu (automatinį išsiuntimą gali vykdyti *Javascript* scenarijus). Siuntimo procedūros schema pavaizduota 21 paveiksle.



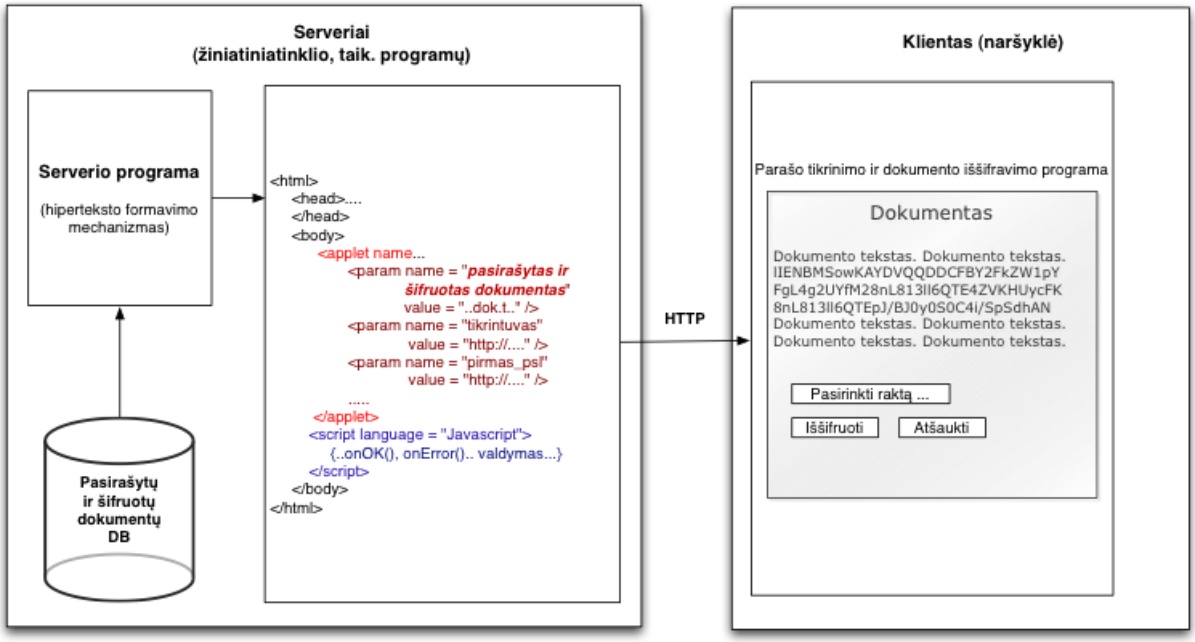
21 pav. Pasirašyto ir šifruoto XML dokumento siuntimas į serverį

Serveryje dokumentai, vien pagal žymių reikšmes (jei šios nėra šifruotos) gali būti įvairiai apdorojami (tikrinami jų parašai) ir klasifikuojami. Tai atlieka taikomosios serverio programos, naudodamos XMLDSIG ir apskritai darbo su XML dokumentais metodus turinčias bibliotekas. Parašai ir šifruotas elementų turinys užtikrina dokumento vientisumą ir

konfidencialumą, nes iššifravimui reikalingą raktą turi tik gavėjas, kuris, kaip minėta darbo pradžioje, nėra serverių savininkas.

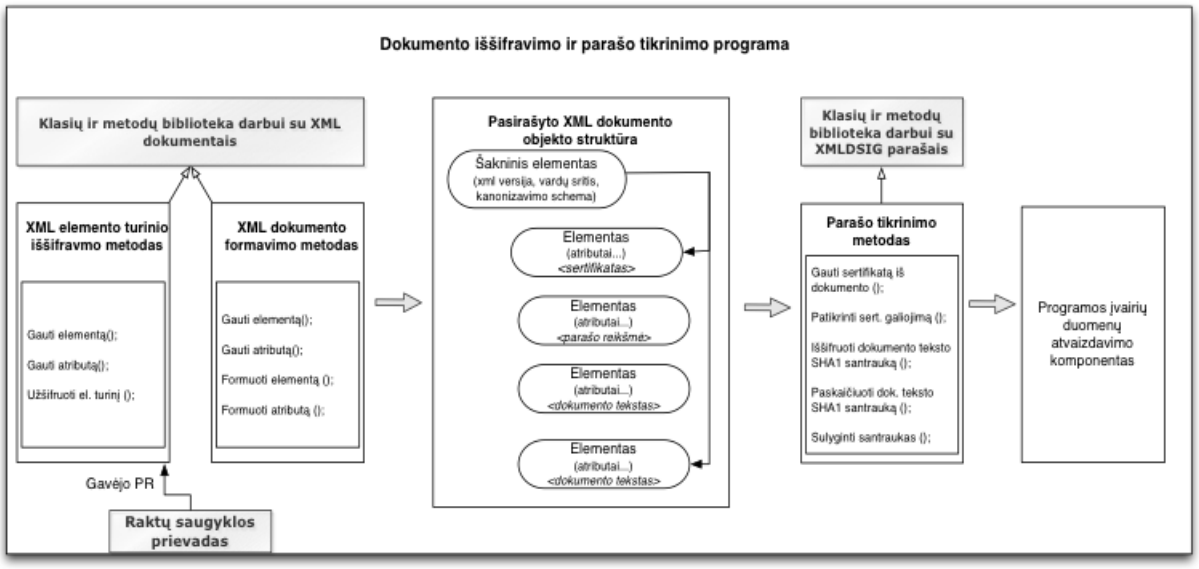
2.3.5 Dokumento pateikimo gavėjui procedūros

Pasirašytas ir šifruotas dokumentas gavėjui pateikiamas panašiai kaip ir dokumento šablonas buvo pateikiamas siuntėjui. Patį dokumentą serverio programa įterpia kaip peržiūrėjimo programos parametrą. Sertifikuota peržiūros programa dokumentą atvaizduoja naudodama savo priemones, t.y. dokumentas vaizduojamas ne tiesiogiai žiniatinklio dokumente, nepatenka už programos „ribų“. Taip užtikrinama, kad joks pašalinis veiksnys (pvz. *javascript* scenarijus) teksto nepakeis iš karto po iššifravimo ar parašo patikros. Tai matyti ir 22 paveiksle.



22 pav. Šifruoto ir pasirašyto dokumento atvaizdavimas

Dokumento parašo tikrinimo ir iššifravimo procedūros pavaizduotos 23 paveiksle. Šiuo atveju įvesties parametras – XML dokumentas, tad reikia atlikti atvirkščias procedūras, nei užšifruojant ir pasirašant, bei rezultatą perduoti į duomenų vaizdavimo komponentą.



23 pav. Dokumento iššifravimas, parašo patikrinimas ir atvaizdavimas

2.4 Metodo specifikacijos išvados

Remiantis suformuota metodo koncepcija buvo suformuluoti funkciniai ir nefunkciniai reikalavimai, bendrai apibrėžiantys, ką konkrečiai realizuotas pagal kuriamą metodą sprendimas turėtų atlikti ir kaip tai padaryti. Laikantis reikalavimų buvo detalizuotas ir pateiktas schemų pavidalu metodo veikimo principas. Remiantis veikimo schemomis galima projektuoti ir realizuoti sprendimo prototipą, kuris vėliau galėtų būti panaudotas metodo tyrimams.

3. METODO REALIZACIJA IR TYRIMAS

3.1 Metodo realizacijos ypatybės

3.1.1 Realizacijos būdas ir priemonės

Elektroninių tranzakcijų dokumentais keičiamasi paslaugų valdymo sistemose. Šios dažniausiai veikia kliento – serverio architektūros pagrindu, tad dokumentų pasirašymo metodas taip pat veikia kliento – serverio aplinkoje.

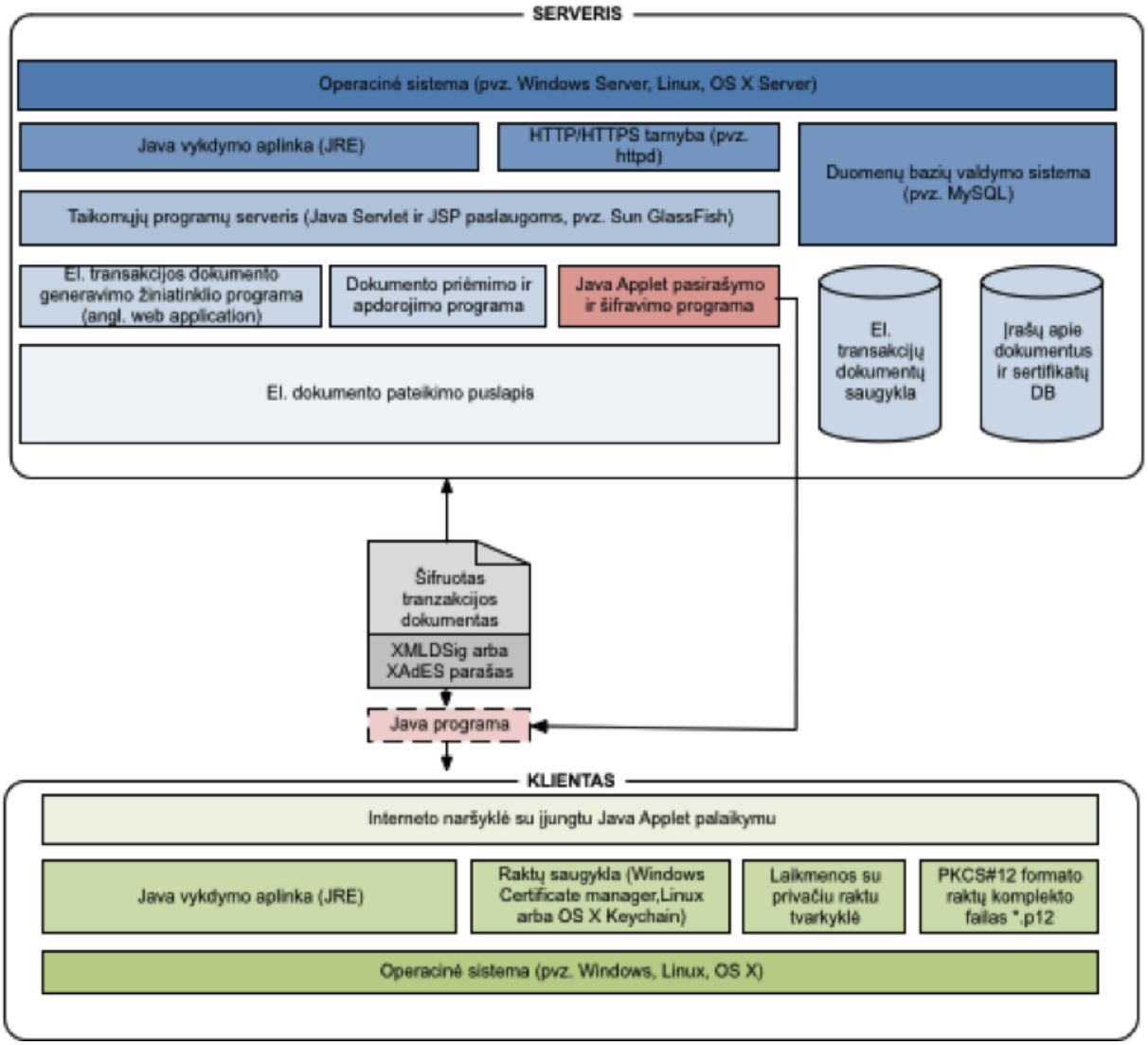
Bendrai serverio funkcijas gali atlikti šiuolaikinis asmeninis kompiuteris su įdiegtomis serverio tarnybomis, leidžiančiomis jungtis prie jų nuotoliniu būdu per interneto naršyklę ir interaktyviai gauti tam tikras paslaugas. Šiuo atveju sprendimui realizuoti ir tirti pasirinkta *Oracle Java* platforma dėl savo universalumo:

- Aplinka nepriklauso nuo operacinės sistemos. Dėl to sukurtą sprendimą bus galima tirti naudojant skirtingas operacines sistemas;

- *Java* platformos pagrindu yra sukurtos ir plačiai naudojamos žiniatinklio aplinkoje veikiančios tarnybos, leidžiančios generuoti žiniatinklio puslapius su kintamu turiniu, interaktyviai sąveikauti su vartotoju ar kitomis (nutolusiomis arba veikiančiomis tame pačiame kompiuteryje) tarnybomis, duomenų bazėmis, naudojant *Java Applet*, *Java Servlet, JSP* technologijas ir žiniatinklio programas (*angl. web applications*) [7, 8, 13].
- Yra sukurta nemažai galingų ir laisvai prieinamų naudoti įrankių, kurių pakanka el. tranzakcijų dokumentų valdymo posistemei, į kurią bus integruotas dokumentų pasirašymo metodas, realizuoti. Tai integruotos programavimo aplinkos (*angl. IDE – Integrated Developement Environment*), kompiliatoriai, derintuvai (*angl. debuggers*), taip pat Java technologijos pagrindu veikiantys tyrimų įrankiai algoritmų, metodų spartai, apkrovoms matuoti ir panašiai.
- Didelės pakartotinio panaudojimo galimybės realizuojant el. tranzakcijų dokumentų pasirašymo metodą: jau yra sukurti, ištestuoti ir laisvai prieinami komponentai darbui su XML dokumentais, bibliotekos su duomenų struktūromis ir realizuotais įvairiais santraukos, el. parašo formavimo algoritmais.

Kuriamo metodo pagrindu veikiantis komponentas skirtas integruoti į didesnę sistemą, tad reikia žinoti sistemos architektūrą, kad būtų galima apibrėžti metodo vietą sistemoje, sąsajas su kitais sistemos komponentais, išsiaiškinti galimus realizacijos variantus. Po šių žingsnių galimas ir paties pasirašymo komponento sudėtinių dalių specifikavimas ir t.t. Taigi pasirašymo mechanizmas projektuojamas „iš viršaus į apačią“ (*angl. top-down*) principu.

Galima sistemos su integruotu el. tranzakcijų dokumentų pasirašymo moduliu (paryškintas raudonu fonu) architektūra pavaizduota 24 paveiksle.



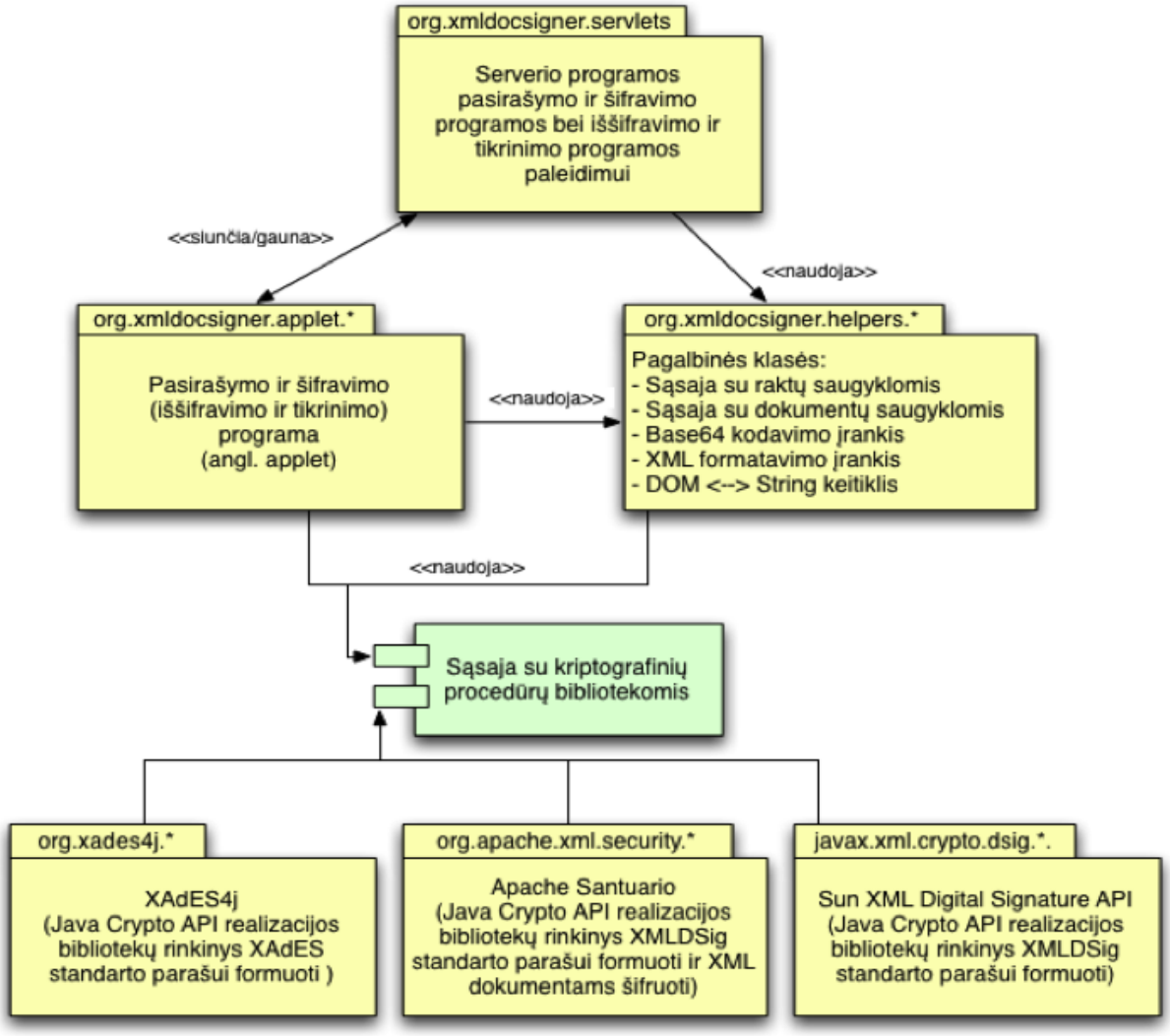
24 pav. Galima sistemos su integruotu el. tranzakcijų dokumentų pasirašymo moduliu architektūra

Tokios architektūros sistema turi visus būtinus komponentus, kad galėtų veikti el. tranzakcijų dokumentų pasirašymo mechanizmas: sertifikatų ir el. tranzakcijų dokumentų bazes, dokumentų formavimo, pasirašymo ir apdorojimo modulius. Pastarieji veikia *Java* vykdymo aplinkoje ir yra kontroliuojami *Apache Tomcat* serverio. Šio paskirtis - generuoti interaktyvų puslapį su dokumento turiniu naudojant sertifikatus ir dokumento tekstą iš duomenų bazių bei vykdant generavimo, apdorojimo komponentų kodą. Vykdant dokumento generavimo kodą, sukuriamas tranzakcijos dokumentas su *Java applet* programa pasirašymui.

3.1.2 Realizuoto sprendimo komponentai

Sistemos komponentai

Visos sistemos komponentinė schema pavaizduota 25 paveiksle.



25 pav. Sistemos komponentai

Pasirašymo ir šifravimo (taip pat iššifravimo ir tikrinimo) komponentas yra vienas klasių paketas, realizuotas pagal MVC (*angl. Model View Controller*) principą.

Serverio dalies programos realizuotos atskirai. Jos, kartu su nuorodomis į pasirašymo ir šifravimo programą bei konfigūracijos bylomis sudaro žiniatinklio programą, vykdomą Java EE palaikančiame taikomųjų programų serveryje.

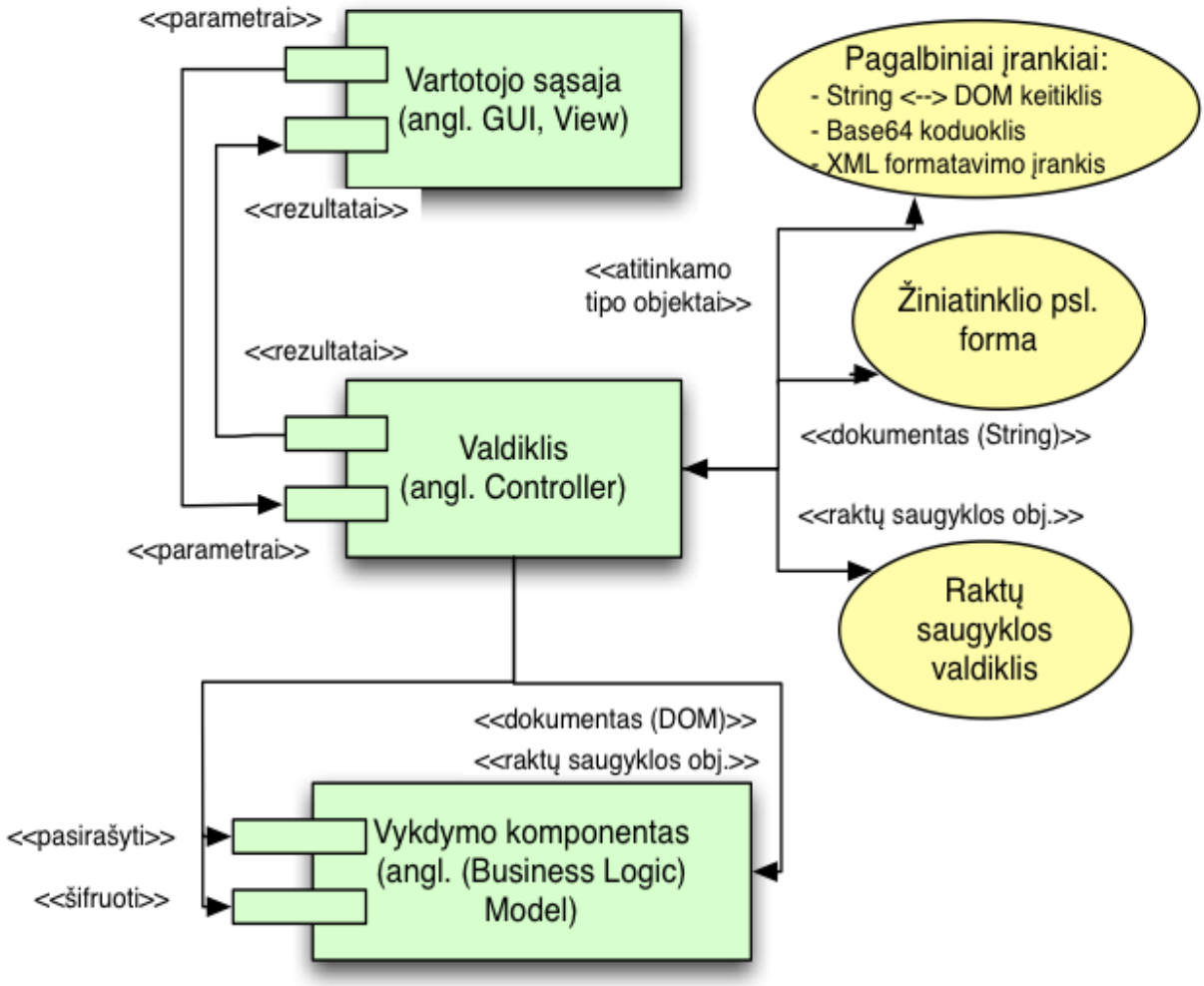
Pagalbinių klasių paketas skirtas sąsajoms su raktų ir dokumentų saugyklomis organizuoti, atlikti pagalbines funkcijas, tokias kaip dokumento keitimas iš *DOM* (*angl. Document Object Model*) struktūros į *String* eilutę, koduoti ir iškoduoti Base64 srautą, formatuoti tvarkingą XML dokumentą (su atitraukimais, kad matytųsi hierarchinė struktūra).

Kadangi *Oracle Java SE/EE* pateikta tik su saugos problemoms spręsti skirtomis programavimo sąsajomis (angl. API), tad tenka naudoti trečiųjų šalių realizacijas arba programuoti savo. Šiuo atveju skaitmeninio parašo ir šifravimo funkcijoms vykdyti panaudota *Apache Santuario* biblioteka (klasių ir metodų iš kelių *Apache* projektų, tokių, kaip *Xalan* ar

Xerces samplaika *Java Crypto API* realizacijai išpildyti). Ši biblioteka yra laisvai prieinama, nekomercinė, šiuo metu vystoma, palaikoma ir gana plačiai naudojama.

Pasirašymo ir šifravimo bei iššifravimo ir tikrinimo programa

Programos komponentinė struktūra pavaizduota 26 paveiksle.



26 pav. Pasirašymo programos komponentai

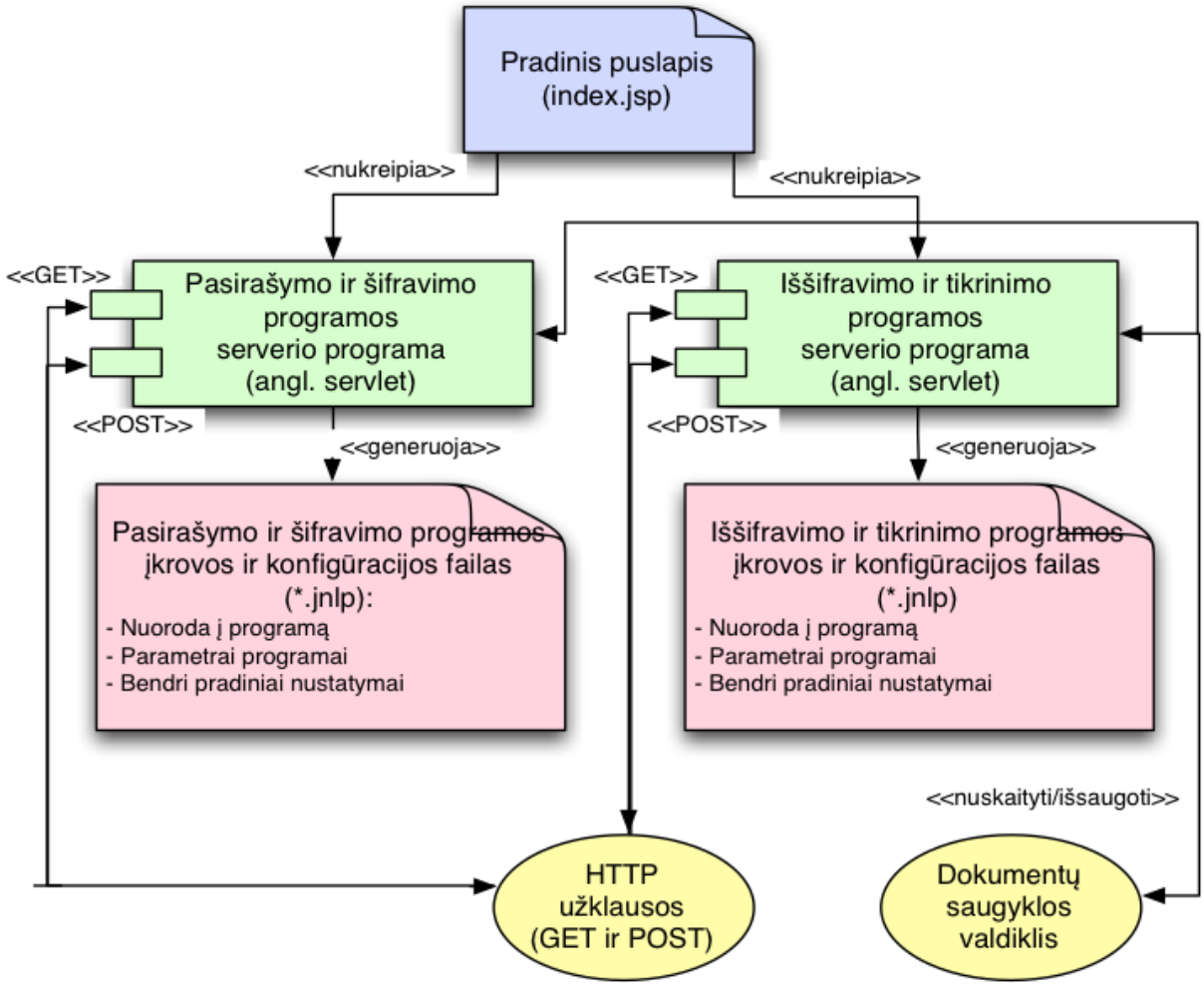
Programa realizuota MVC principu. Valdiklis yra tarpinė grandis, kontroliuojanti ir paskirstanti duomenų srautus programos komponentams (organizuoja duomenų paėmimą, inicijuoja skaičiavimus, valdo rezultatų grąžinimą). Kadangi ta pati programa atlieka visas 4 funkcijas (pasirašo, tikrina parašą, užšifruoja ir iššifruoja), naudojami du valdikliai. Kiekvienas valdo savo vartotojo sąsają, tačiau naudojasi bendrais skaičiavimo mechanizmais (modeliu). Valdiklis taip pat palaiko ryšius su kitais valdikliais (kaip matyti 17 paveiksle – kreipiasi į raktų saugyklos valdiklį, kad šis parinktų tinkamą sąsają ir suorganizuotų ryšį su tam tikra nurodyta raktų saugykla. Būtent valdikio klasė paveldi *JApplet* klasę ir inicijuoja vartotojo sąsajos sukūrimą bei atvaizdavimą.

Vartotojo sąsajos klasė šiuo atveju yra standartinių *Java AWT* ir *Swing* komponentų bei sąsajos metodų rinkinys. Visa vartotojo sąsaja sutalpinta į vieną *JPanel* tipo objektą, kuris grąžinamas valdikliui, o šis sukuria konteinerį ir gautą *JPanel* objektą į jį patalpina.

Vykdymo komponentas arba modelis – klasių rinkinys kriptografinėms operacijoms organizuoti ir atlikti. Modelį sudaro 3 klasės: šifratoriaus, pasirašymo mechanizmo ir dokumentų valdymo mechanizmo. Pastarasis iš *String* tipo eilutės sukuria DOM (*angl. Document Object Model*) struktūrą ir perduoda šią struktūrą šifratoriaus bei pasirašymo mechanizmo objektams. Dokumentų valdymo mechanizmas pats pirmas ir gauna rezultatus. Toliau jau valdiklis kreipiasi į jį, kad šis rezultatus grąžintų. Šifratorius bei pasirašymo mechanizmai veikia pagal 2 skyriuje pateiktus modelius: jie, naudodami išorines bibliotekas, patys papildo XML dokumentų objektus reikiamomis struktūromis.

Serverio programos

Serverio programų komponentinė struktūra pavaizduota 27 paveiksle.



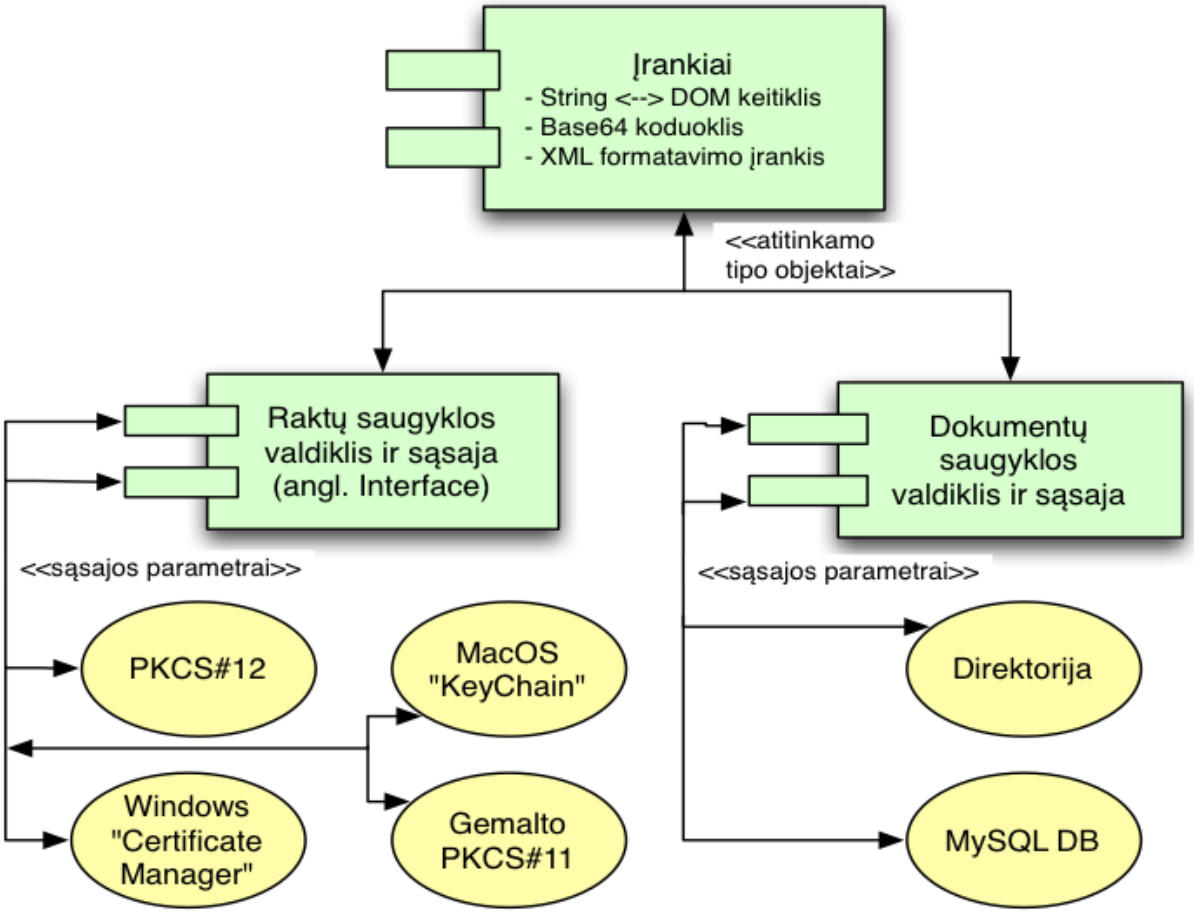
27 pav. Serverio programų komponentai

Serverio programų paskirtis – po žiniatinklio protokolu gautos užklausos sugeneruoti pasirašymo ir šifravimo (tikrinimo ir iššifravimo) programos konfigūracinį užkrovimo failą (*angl. Java Network Launching Protocol file, *.jnlp*), bei pradėti jį vykdyti. Būtent tokiu būdu vartotojo naršyklė užkraunama programa kriptografinėms procedūroms atlikti.

Be žiniatinklio protokolo užklausų apdorojimo, programos turi savo valdiklį, kuris jungiasi su dokumentų saugyklos valdikliu, tam, kad gautų universalią prieigą per norimą sąsają prie dokumentų (pvz. nutolusios direktorijos, kurioje dokumentai laikomi *.xml failų pavidalu arba duomenų bazės per DBVS sąsają).

Pagalbiniai komponentai

Serverio programų komponentinė struktūra pavaizduota 28 paveiksle.



28 pav. Pasirašymo programos komponentai

Pagalbinių komponentų paketą sudaro dar 3 komponentai:

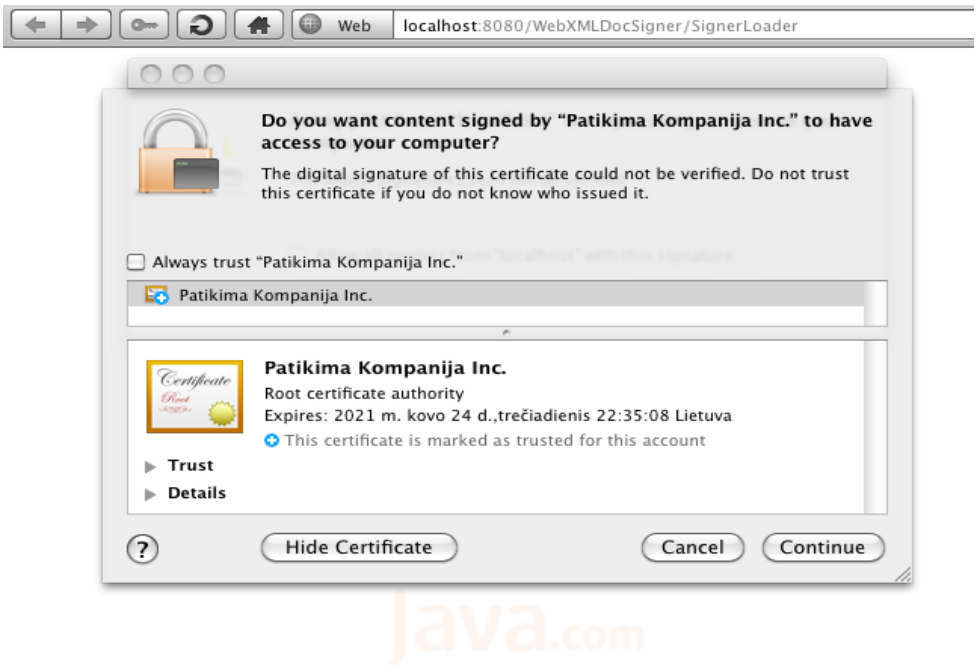
- Įrankių paketas su klasėmis, kurios padeda pakeisti dokumento struktūrą, užkoduoti ar iškoduoti, paversti iš vieno tipo į kitą;
- Raktų saugyklos paketas su raktų saugyklos valdikliu, sąsają bei įvairiomis sąsajos realizacijomis (tam, kad naudojant tuos pačius metodus ir tik nurodant sąsajos tipą, būtų galima pasiekti saugyklų turinį);

- Dokumentų saugyklos paketas su dokumentų saugyklos valdikliu, sąsaja bei jos realizacijomis (taip pat, kad būtų galima pasiekti skirtingų tipų saugyklas tuo pačiu būdu).

3.1.3 Prototipas tyrimui

Tyrimams skirtas prototipas sukurtas remiantis metodo veikimo schemomis bei komponentų diagramomis. Toliau pateikta keletas prototipo veikimo ekrano nuotraukų.

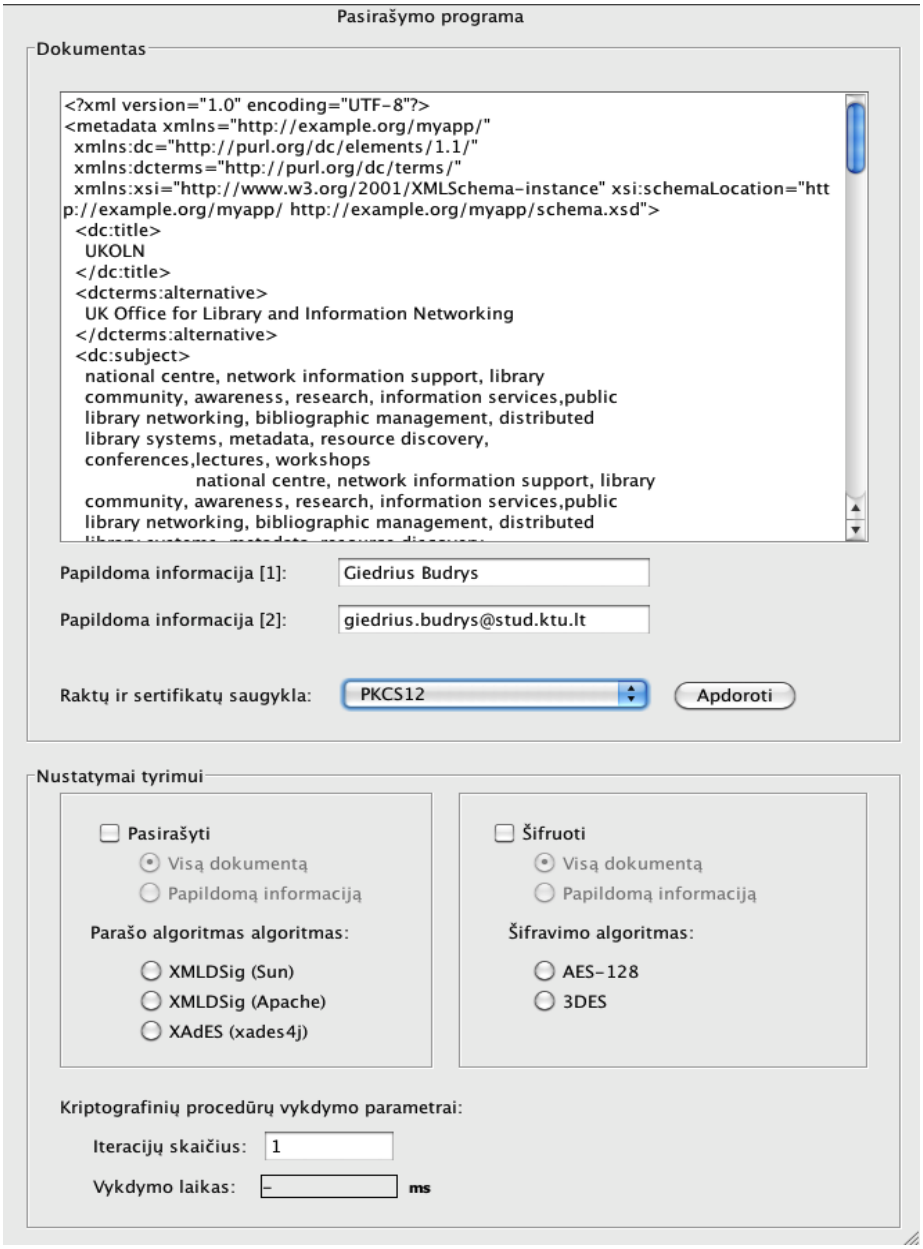
Atveriant puslapį su pasirašymo programa, pastaroji prašo suteikti prieigą prie failų sistemos (raktui nuskaityti) ir pateikia autentiškumą patvirtinantį liudijimą. Šiuo atveju buvo panaudotas laikinas nemokamas raktas ir sertifikatas pagal pabandymui skirtą licenciją. Naršyklės langas su atveriama programa pavaizduotas 29 paveiksle.



29 pav. Atveriama parašo formavimo programa

Atvėrus programą, matomas jos langas su programos teksto komponente vaizduojamu pasirašytinu dokumentu. Dokumentas programai perduodamas per parametrus, o papildomą informaciją (ši dokumente talpinama, kaip atskiri elementai) įveda vartotojas. Dar vienas vartotojo nurodomas parametras – raktų saugyklos tipas. Parašo formavimo programos langas su dokumento šablonu pavaizduotas 30 paveiksle.

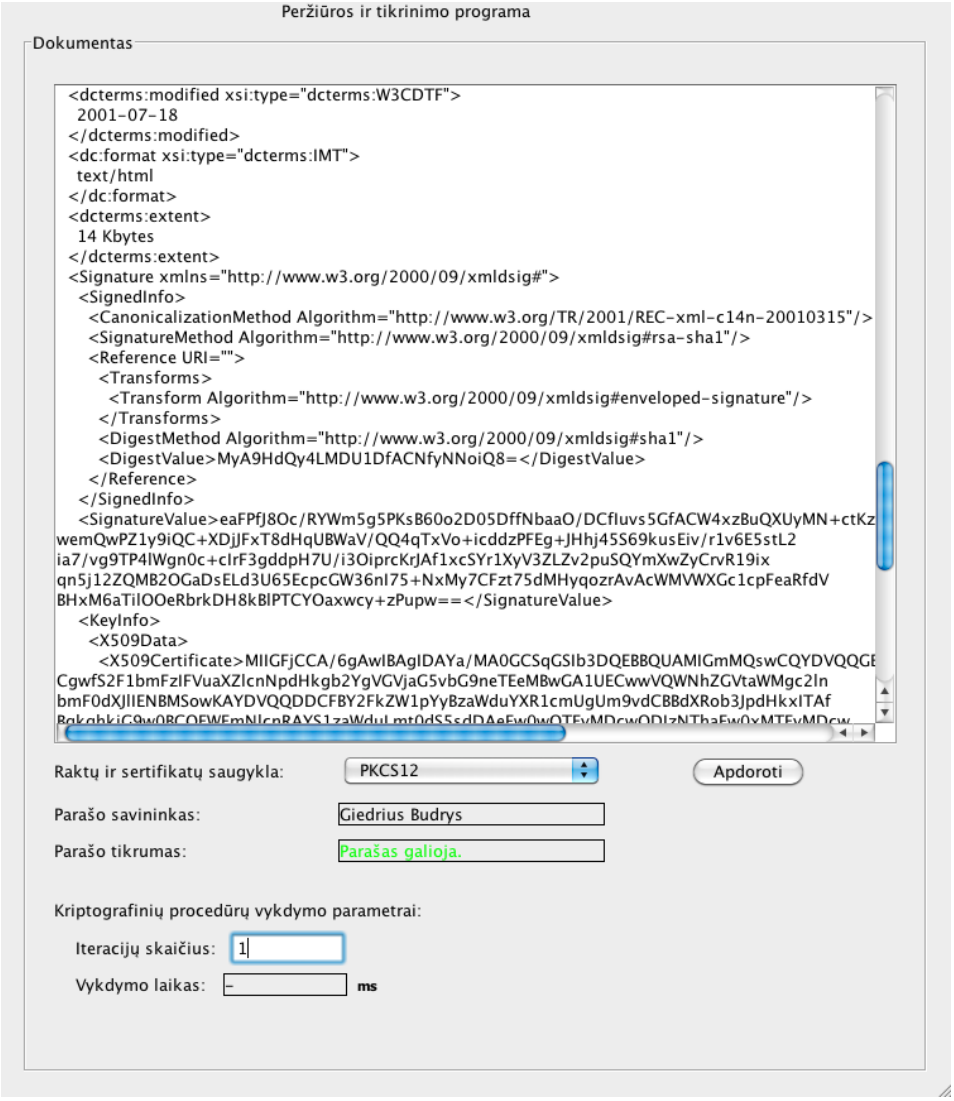
Pastaba. Prototipe dėl patogumo testuoti nerealizuotas XML dokumento vertimas ir atvaizdavimas HTML. Kaip matyti 30 paveiksle, prototipo lange atvaizduojamas grynas XML dokumento tekstas. Realiame sprendime, lange turėtų būti rodoma atitinkama HTML forma. Tačiau šis atitikimas jokios įtakos tyrimo eigai ir rezultatams nedaro.



30 pav. Parašo formavimo programos langas su dokumento šablonu

Tik pasirinkus raktų saugyklą (šiuo atveju .p12 formato failas) ir suvedus slaptažodį, leidžiama pasirašyti dokumentą (pasirašymo metodo kvietimo mygtukas tampa aktyvus).

Bandomoji programa pasirašo dokumentą ir per paslėptus formos laukus siunčia į serverį. Dokumento iššifravimui ir parašo tikrinimui naudojama ta pati programa (turimas omenyje tai, kad vartotojui atsiunčiamas tas pats programos vykdomasis failas, tik jam nurodomi kiti parametrai, būtent, vartotojo sąsajos valdiklis (jo pavadinimas), kviečiantis atitinkamus metodus. Šis atvejis pavaizduotas 31 paveiksle.



31 pav. Pasirašyto, nešifruoto dokumento atvaizdavimas

Šio dokumento XML atitikmuo su *Sun XML Digital Signature API* suformuotu XMLDSig parašu pavaizduotas 32 paveiksle (visi dokumentai, su kuriais buvo atlikti bandymai, pateikti prieduose) .

Pastaba. Prototipo programos lange ir 32 paveiksle pavaizduotas dokumentas yra taisyklingas, „Dublin Core“ schemą atitinkantis XML dokumentas [1]. Tokiais dokumentais disponuojama elektroninių dokumentų valdymo sistemose. Šis dokumentas buvo naudotas tyrime (taip pat pateiktas prieduose). 32 paveiksle raudonai apvestas parašo elementas bei įterptas papildomas elementas *dc:creator*. Plačiau apie tyrimui naudotus dokumentus rašoma 3.2.1 skyriuje „Tyrimų metodika“.

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://example.org/myapp/" xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://purl.org/dc/terms/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://example.org/myapp/ http://example.org/myapp/schema.xsd">
    <dc:title>
    UKOLN
  </dc:title>
    <dcterms:alternative>
    UK Office for Library and Information Networking
  </dcterms:alternative>
    <dc:subject>
    national centre, network information support, library
                        ...
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
    national centre, network information support, libra
  </dc:subject>
    <dc:subject xsi:type="dcterms:DDC">
    062
  </dc:subject>
    <dc:subject xsi:type="dcterms:UDC">
    061(410)
  </dc:subject>

    <dc:description>
    UKOLN is a national focus of expertise in digital information
    management. It provides policy, research and awareness services
    to the UK library, information and cultural heritage communities.
    UKOLN is based at the University of Bath.
                        ...
    UKOLN is based at the University of Bath.
  </dc:description>
    <dc:description xml:lang="fr">
    UKOLN est un centre national d'expertise dans la gestion de l'information
    digitale.
  </dc:description>
    <dc:creator>Giedrius Budrys</dc:creator>
    <dc:publisher>
    UKOLN, University of Bath
  </dc:publisher>
    <dcterms:isPartOf xsi:type="dcterms:URI">
    http://www.bath.ac.uk/
  </dcterms:isPartOf>
    <dc:identifier xsi:type="dcterms:URI">
    http://www.ukoln.ac.uk/
  </dc:identifier>
    <dcterms:modified xsi:type="dcterms:W3CDTF">
    2001-07-18
  </dcterms:modified>
    <dc:format xsi:type="dcterms:IMT">
    text/html
  </dc:format>
    <dcterms:extent>
    48 Kbytes
  </dcterms:extent>
```

32 pav. Pasirašytas XML dokumentas

Matyti, kad XML dokumentas suformuotas griežtai laikantis „Dublin Core“ standarto schemos (nurodytos vardų srities žymos).

Dalinai šifruoto dokumento pavyzdys pateiktas 33 paveiksle (užšifruotas vienas dokumento elementas *dc:creator*).

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://example.org/myapp/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://example.org/myapp/
http://example.org/myapp/schema.xsd">
  <dc:title>
    UKOLN
  </dc:title>
  <dcterms:alternative>
    UK Office for Library and Information Networking
  </dcterms:alternative>
  <dc:subject>
    national centre, network information support, library
    community, awareness, research, information services,public
                              ...
    national centre, network information support, libra
  </dc:subject>
  <dc:subject xsi:type="dcterms:DDC">
    062
  </dc:subject>
  <dc:subject xsi:type="dcterms:UDC">
    061(410)
  </dc:subject>
  <dc:description>
    UKOLN is a national focus of expertise in digital information
    management. It provides policy, research and awareness services
    to the UK library, information and cultural heritage communities.
                            ...
    UKOLN is based at the University of Bath.
  </dc:description>
  <dc:description xml:lang="fr">
    UKOLN est un centre national d'expertise dans la gestion de l'information
    digitale.
  </dc:description>

   <dc:creator>
     <xenc:EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Content"
xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
       <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
       <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
         <xenc:EncryptedKey xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
           <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p"/>
           <xenc:CipherData>

<xenc:CipherValue>IKIG6WyJUWhVWeJPDw/o4JLqVai/tKB5hItNkWoEGM+jt6oAxA1sZMTHtzXqGmlB45l8YJzqL5yy
mbYwRgxm1sBFZGVRk8R+Pzt7gtc5vbVk0mM+RjHGt+AZVcMOwlvXhWW1BLfvBOg/Z8Z08jkceKXL
VkXEquf20OdbsObKWzw=</xenc:CipherValue>
           </xenc:CipherData>
         </xenc:EncryptedKey>
       </ds:KeyInfo>
       <xenc:CipherData>
         <xenc:CipherValue>OdykRAA/yXnfONjhomK4N9GOu4QuoQQ+2dHgzpqY3eY=</xenc:CipherValue>
       </xenc:CipherData>
     </xenc:EncryptedData>
   </dc:creator>

  <dc:publisher>
    UKOLN, University of Bath
  </dc:publisher>
  <dcterms:isPartOf xsi:type="dcterms:URI">
    http://www.bath.ac.uk/
  </dcterms:isPartOf>
  <dc:identifier xsi:type="dcterms:URI">
    http://www.ukoln.ac.uk/
  </dc:identifier>
  <dcterms:modified xsi:type="dcterms:W3CDTF">
    2001-07-18
  </dcterms:modified>
  <dc:format xsi:type="dcterms:IMT">
    text/html
  </dc:format>
  <dcterms:extent>
    14 Kbytes
  </dcterms:extent>
</metadata>
```

33 pav. Dokumentas su užšifruotu elementu

Realizacijos išvados

Sprendimo (metodo realizacijos) kūrimo priemonės parinktos atsižvelgiant į šiandieną prieinamų įrankių ir technologijų galimybes, stengiantis jas maksimaliai išnaudoti. Sprendimo architektūra suprojektuota remiantis pasirinktomis technologijomis. Parengti sprendimo komponentų modeliai nedetalizuoja sprendimo veikimo klasių lygiu (tai bus pateikta prieduose).

Pagal parengtą specifikaciją realizuotas sprendimas iš principo veikia, suformuoja pasirašytą ir (ar) šifruotą XML dokumentą, turi parametrų įvesties sąsajas, tad su juo galima atlikti kriptografinių procedūrų greitaveikos matavimus.

3.2 Realizuoto sprendimo greitaveikos tyrimai

3.2.1 Tyrimų metodika

Bandymams naudojamas išbaigtas prototipas (turintis ir pasirašymo, ir šifravimo funkcijas. Kadangi kuriant darbe aprašomu metodu paremtą sprendimą viena svarbiausių jo savybių yra veikimo efektyvumas (kuo didesnė veikimo sparta užtikrinant priimtiną saugos lygį), svarbu ištirti, kaip sprendimo greitaveika priklauso nuo pasirašomų ir šifruojamų duomenų kiekio, pasirinktų pasirašymo ir šifravimo metodų bei algoritmų. Tyrimams naudojami skirtingų dydžių dokumentai, tyrimai atliekami dokumentams taikant 4 skirtingas saugos procedūras:

- Apgaubtojo parašo formavimas (visam dokumentui ir tik pasirinktam elementui);
- Viso dokumento šifravimas gavėjo viešuoju raktu;
- Tik pasirašyto elemento šifravimas gavėjo viešuoju raktu.

Papildomai atliekamas rakto nuskaitymo iš skirtingų tipų raktų ir sertifikatų saugyklų tyrimas. Juo siekiama išsiaiškinti, kokio tipo raktų ir sertifikatų saugykla veikia greičiausiai ir kokie nuskaitymo spartos skirtumai tarp saugyklų. Iš rezultatų galima spręsti, ar reikia ir ar verta optimizuoti metodo veikimą, siekiant, kad metodo veikimo sparta, naudojant kiekvieną nagrinėtą saugyklą, būtų panaši.

Tam, kad būtų galima objektyviai įvertinti visų procedūrų greitaveikos skirtumus, matavimai atliekami daug kartų ciklišk ai taikant procedūrą vienam dokumentui. Rezultatas – laikas, sugaištas 1 kartą atliekant nurodytą procedūrą (skaičiuojamas n atliktų matavimų rezultatų vidurkis).

Tyrimams naudota įranga

Tyrimams naudota įranga pateikta 4 lentelėje.

4 lentelė. Tyrimams naudota įranga

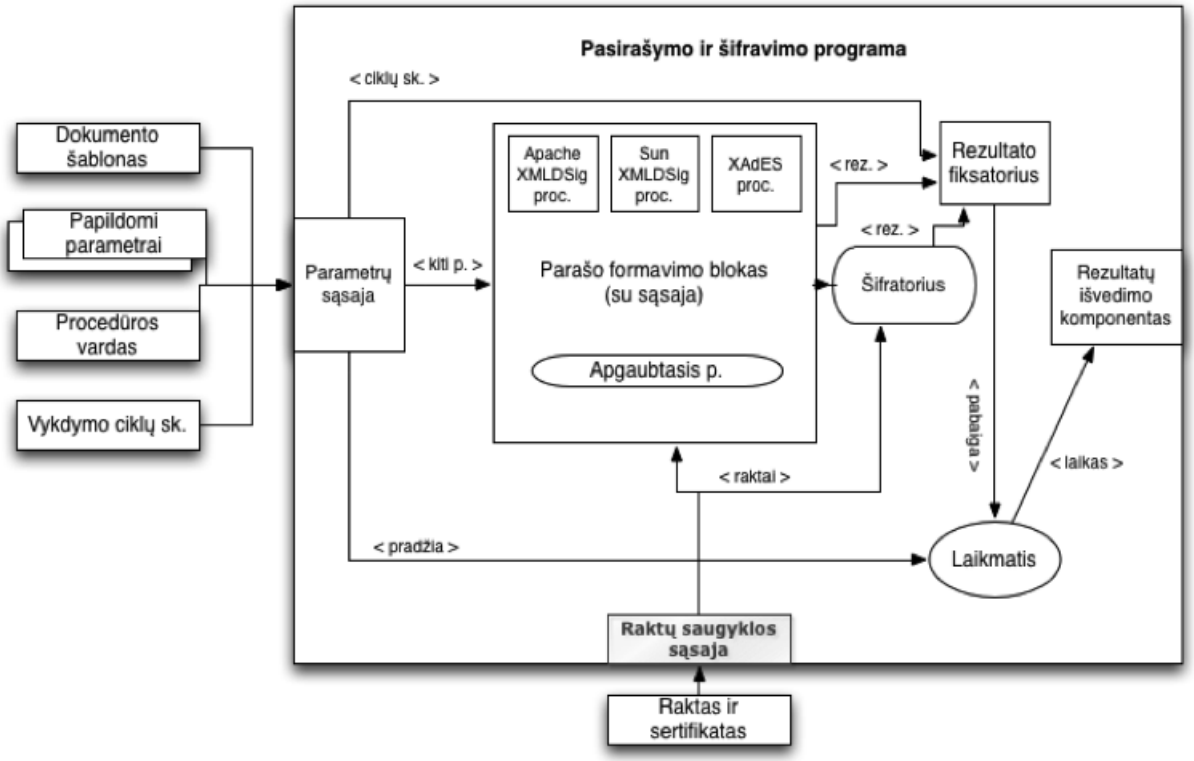
Pagrindiniai kompiuterio komponentai	
Procesorius	Intel Core 2 Duo, 2,53GHz.
Darbinė atmintis	DDR3, 1067MHz, 4GB.
Standusis diskas	250GB, 5400 sūkių/min.
Papildoma techninė įranga	
Lustinių kortelių skaitytuvas	HID OmniKey 3021.
Lustinė kortelė	LR asmens tapatybės kortelė (GemXpresso Pro R3 JavaCard su GemSAFE programine įranga (*angl. applet*)).
Programinė įranga	
Operacinės sistemos	Windows XP Professional SP3; Mac OS X 10.6.7 (Snow Leopard).
Papildoma programinė įranga	Java JRE 1.6; Gemalto ClassicClient 6.

Bendroji tyrimų schema

Įvesties duomenys:

- Dokumento šablonai (kiekvienas matavimas atliekamas su 5000 B, 50000 B ir 500000 B dokumentais);
- Papildomi parametrai (pasirašiusio asmens vardas ir pavardė);
- Siuntėjo slaptasis raktas ir sertifikatas – automatiškai nuskaitomi iš raktų ir sertifikatų saugyklos, kurios inicializavimo parametrai (pvz. vieta failų sistemoje, slaptažodis) nurodomi programos kode (kad nereikėtų parinkinėti rankiniu būdu);
- Saugos procedūros pavadinimas;
- Procedūros taikymo dokumentui ciklų skaičius;

Bendroji tyrimų schema pavaizduota 34 paveiksle.



34 pav. Bendroji tyrimų schema

Schemoje pavaizduoti įvesties duomenys ir pagrindiniai pasirašymo programos komponentai tiesiogiai lemiantys matavimo rezultatus. Parašą formuojantys komponentai dar gali būti detalizuoti įvardijant konkrečius parašo algoritmus. Kadangi įvairių parašo formavimo algoritmų greitaveika yra plačiai ištirta, šiame darbe jiems skiriama nedaug dėmesio.

Bandymas atliekamas pasirinkus SHA1 ir RSA santraukos ir parašo algoritmus, nes jie plačiai naudojami ne žiniatinklio taikomosiose programose dokumentams pasirašinėti. Bandymai atlikti panaudojant dvi skirtingai realizuotas XMLDSig standarto ir vieną XAdES standarto parašą formuojančias atvirojo kodo bibliotekas, gebančias suformuoti SHA1 santrauką ir ją užšifruoti privačiu raktu pagal RSA algoritmą. Kiekvieno matavimo bendras algoritmas pateiktas 3.2.2 skyriuje.

Šifravimo procedūrų greitaveikos tyrimas atliktas naudojant AES-128 (atvirojo kodo *Apache Santuario* biblioteka nepalaiko ilgesnių, nei 128 bitai raktų) bei 3DES algoritmus. Algoritmai pasirikti atsižvelgiant į naudojimo populiarumą (dažnai naudojami kartu su PGP, S/MIME [20, p.71]) Dokumentai buvo šifruojami simetriniu raktu (kiekvienu minėtų algoritmų), pastarasis buvo užšifruojamas gavėjo viešuoju raktu ir įterpiamas į dokumentą. Tokiu būdu išnaudojama simetrinio šifro sparta ir asimetrinio saugos ypatybės.

Tyrimams naudoti dokumentai

Dokumentų dydžiai pasirinkti atsižvelgiant į tai, kad dokumento, kurį sudaro vienas A4 formato lapas gryno teksto be paveikslėlių, dydis yra maždaug 5000 B. Tad didelis dokumentas (10-ies lapų gryno teksto be paveikslėlių) užims apie 50000 B. Dokumentų dydis darbe nurodomas baitais dėl tikslumo: visi tyrimuose naudoti dokumentai koduoti UTF-8 (vienam simboliui koduoti skirtas 1B), tad dokumento dydis atspindi, kiek tiksliai simbolių yra kiekviename dokumente. Tokiu atveju galima tiksliai apskaičiuoti, kiek kartų ilgiau ar trumpiau vykdomos dokumento apdorojimo procedūros dokumento dydžiui pakitus lygiai 10 kartų.

Rengiant dokumentus tyrimams, stengtasi, kad jie savo struktūra ir keliamais reikalavimais kuo labiau atitiktų dokumentus, kuriais yra realiai disponuojama dokumentų valdymo ir paslaugų sistemose. Kiekvienas dokumentas sudarytas taip, kad atitiktų „Dublin Core“ standarto XML schemą ir būtų taisyklingas tiek pritaikius parašo formavimo, tiek šifravimo procedūras. Kartu su parašo ir šifro elementais įterpiamos nuorodos į vardų sritis pastarųjų XML sintaksės teisingumui patikrinti. Dokumentų pavyzdžiai pateikti prieduose.

„Dublin Core“ standartas pasirinktas laisvai, dokumentų turinys parengtas pagal šį standartą plėtojančios ir prižiūrinčios organizacijos „The Dublin Core Metadata Initiative“ (*trump. DCMI*) interneto svetainėje pateiktus pavyzdinius dokumentus ir rekomendacijas [1].

3.2.2 Tyrimų eiga ir rezultatai

Parašo formavimo ir tikrinimo procedūrų greitaveikos tyrimas

Ši tyrimo tikslas ir esmė – imituoti aibės dokumentų, esančių, pavyzdžiui, duomenų bazėje ar specialiame repozitoriume, pasirašymą konvejerio principu (panašaus pobūdžio procedūros gali būti atliekamos dokumentų valdymo sistemose), išmatuoti skirtingai realizuotų XMLDSig bei XAdES standarto parašus formuojančių ir tikrinančių procedūrų greitaveiką, ištirti procedūrų greitaveikos priklausomybę nuo dokumento dydžio (pasirašomų duomenų kiekio) bei pasiūlyti tinkamiausias procedūras metodo realizacijai įvairiems atvejams.

Tyrimas vykdytas pagal 35 paveiksle pavaizduotą algoritmą.

35 pav. Kriptografinių procedūrų greitaveikos matavimo algoritmas

Matuojant pasirašymo ir tikrinimo procedūrų greitaveiką, kiekviena procedūra įvykdyta po 10 kartų ir apskaičiuojant šių 10 matavimų rezultatų vidurkį, nustatytas vidutinis procedūros vykdymo laikas ((1) formulė).

Vidutinis n kriptografinių procedūrų vykdymo laikas:

$$t_{vid\_n} = \frac{1}{n}\sum_{i=1}^{n} t_i \qquad (1)$$

Čia n – kriptografinės procedūros vykdymo iteracijų skaičius,

i – einamosios iteracijos numeris,

t – einamosios iteracijos vykdymo laikas,

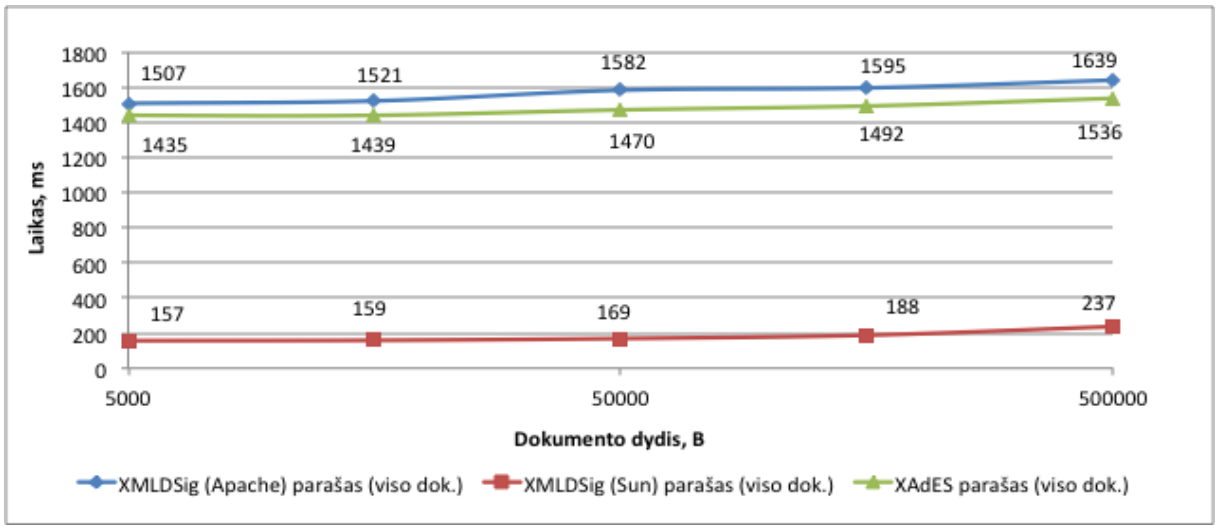
t_{vid_n} *– vidutinis n kriptografinių procedūrų vykdymo laikas.*

Tyrimo metu procedūrų greitaveika vertinta keliais būdais:

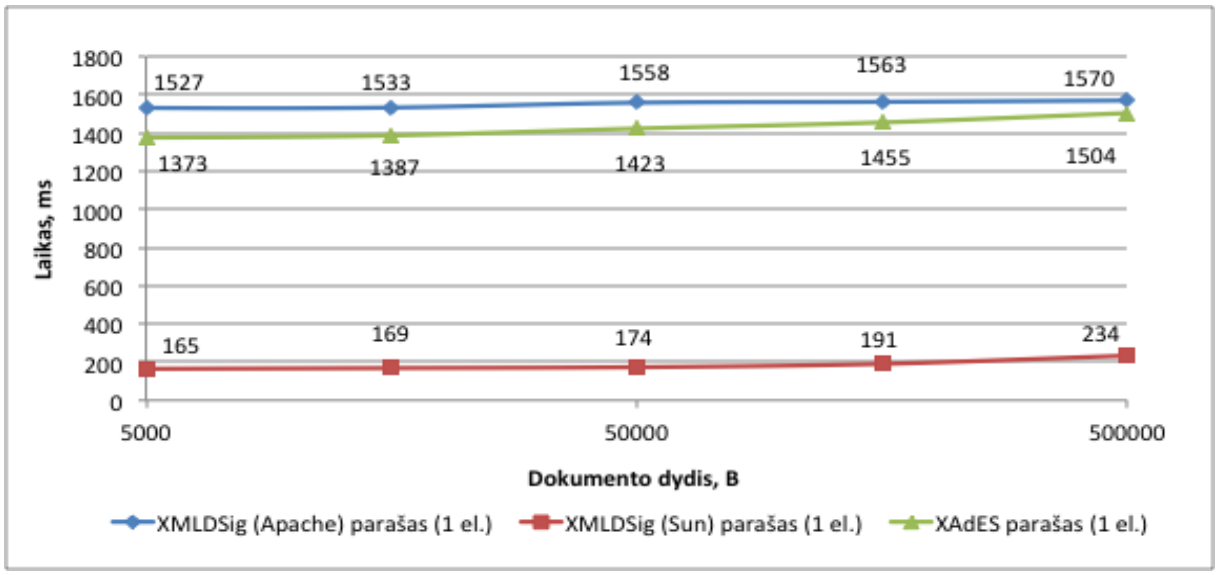
- Parašo sudarymas ir įterpimas bei tikrinimas visam dokumentui;
- Parašo sudarymas ir įterpimas bei tikrinimas 1 dokumento elementui.

Antruoju atveju pasirašomas elementas įterpiamas prieš vykdant pasirašymo procedūrą, o jo turinys gaunamas per pasirašymo programos parametrus ir taip pat įterpiamas prieš vykdant pasirašymo procedūrą. Įterpiamas ir pasirašomas vienas elementas yra 50-ies simbolių (taigi jo dydis 50 B). Tokiu būdu galima įvertinti SHA-1 santraukos formavimo procedūros greitaveiką, kai paduodamas apdoroti skirtingas kiekis duomenų (50B (1 elementas) , 5000 B, 50000 B ir 500000 B (dokumentai)).

Tyrime naudotos *Apache Santuario* (XMLDSig parašui formuoti), *Sun Java XML Digital Signature API* (XMLDSig parašui formuoti) bei *XAdES4j* (XAdES standarto parašui formuoti) bibliotekos. Pasirašymo procedūrų vykdymo laiko grafikai pavaizduoti 36 paveiksle.



a.



b.

36 pav. Viso dokumento (a) ir dokumento vieno elemento (b) pasirašymo procedūrų vykdymo laikas

Iš matavimų rezultatų matyti, jog greičiausiai dokumentą apdoroja *Sun* realizuotos XML dokumentų apdorojimo ir XMLDSig parašo formavimo procedūros. Tačiau *Sun* biblioteka suformuotas parašas yra paprasčiausias – nėra laiko žymų, neįterpiama papildoma informacija apie pasirašiusįjį asmenį, neįterpiami viešojo rakto kriptografiniai parametrai p ir q (modulis ir eksponentė). Tokio dokumento pavyzdys pateiktas 1 priede.

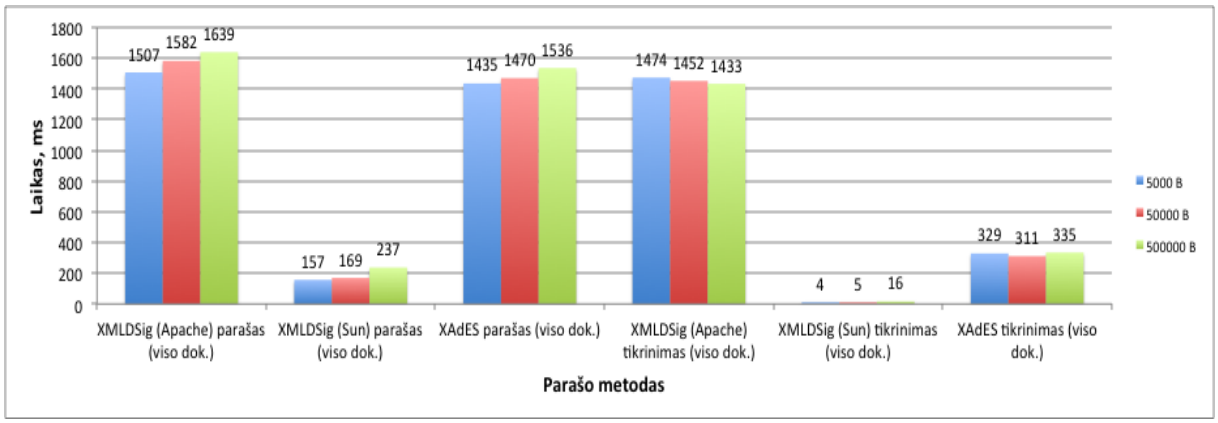
Dokumento pasirašymas *Apache Santuario* bibliotekos procedūromis truko apytiksliai 7 – 9 kartus ilgiau. Galima daryti dvi prielaidas: parašo formavimo algoritmas realizuotas neoptimaliai (vykdoma pernelyg daug ciklų, tikrinama daug sąlygų), arba papildomos informacijos „ištraukimas“ iš sertifikatų bei įterpimas į performuoto dokumento struktūrą trunka žymiai ilgiau nei pagrindinės kriptografinės procedūros XMLDSig parašui suformuoti ir įterpti. Čia „pagrindinėmis“ vadinamos procedūros yra santraukos sudarymas, jos užšifravimas privačiu raktu, parašo reikšmės ir sertifikatų įterpimas į dokumento struktūrą. Iš esmės *Sun* bibliotekoje esančios procedūros tik tai ir atlieka. *Apache Santuario* suformuotame, XMLDSig parašu pasirašytame dokumente, palyginti su *Sun Java XML Digital Signature API* pasirašytu dokumentu, papildomai įterpti viešojo rakto parametrai p ir q (modulis ir eksponentė). *Apache Santuario* pasirašyto dokumento pavyzdys pateiktas 1 priede.

Trečiuoju atveju buvo pasirinkta formuoti XAdES-T standarto parašą (raidė T reiškia, kad papildomai į parašą įterpiama kvalifikuota laiko žyma, nurodanti datą ir laiką, nuo kurio parašas įsigaliojo). Griežtesnės ir sudėtingesnės XAdES parašo standarto atmainos, pavyzdžiui XAdES-C (su pilna sertifikatų grandinėle), XAdES-A (archyvavimui skirta papildoma informacija) daugiau naudojamos tais atvejais, kai parašai saugomi atskirai nuo pasirašytų dokumentų (atskirtojo parašo, *angl. detached signature atvejis*). Darbe nagrinėjamu atveju dokumento pilnaverčiam autentiškumui užtikrinti pakanka XAdES-T standarto parašo.

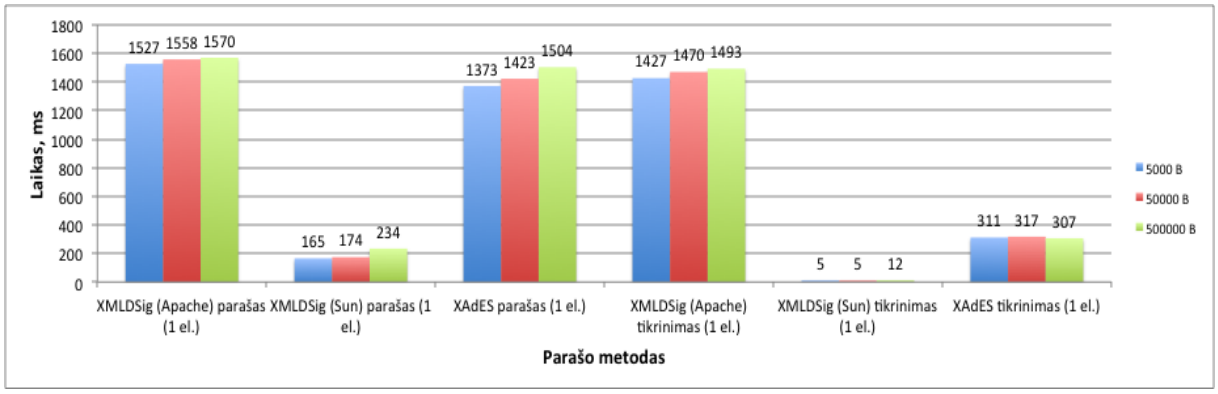
Dokumento pasirašymas XAdES-T standarto parašu visais atvejais truko 100-150 ms trumpiau, nei XMLDSig su *Apache Santuario* biblioteka, nors pati procedūra yra kur kas sudėtingesnė. Palyginti su *Apache Santuario* XMLDSig, papildomai įterpiami papildomi saugos parametrai: laiko žyma, patvirtinta serverio, kuris ją sudarė, parašu ir šakninės sertifikatų išdavimo tarnybos (kuri dokumentą pasirašiusiam asmeniui išdavė raktus ir sertifikatus) sertifikato duomenys (maišos algoritmo pavadinimas, skaitmeninis parašas, pavadinimas, sertifikato serijinis numeris). Maždaug pusę sugaišto laiko (apie 500-700ms, priklausomai nuo interneto ryšio spartos) truko laiko žymos gavimo iš specializuoto serverio procedūra (HTTP užklausos suformavimas ir atsakymo priėmimas ir apdorojimas). *XAdES4j* pasirašyto dokumento pavyzdys pateiktas 1 priede.

Pasirašymo ir tikrinimo procedūrų vykdymo trukmės asimetriškumo tyrimas.

Siekiant išsiaiškinti, ar skiriasi ir kiek skiriasi atvirkštinės operacijos – parašo tikrinimo procedūros greitaveika, buvo atlikti atitinkami matavimai. Lyginamieji parašo suformavimo bei tikrinimo procedūrų vykdymo trukmės matavimų rezultatai pateikti 37 paveiksle.



a.



b.

37 pav. Dokumento pasirašymo ir parašo tikrinimo procedūrų vykdymo trukmės asimetriškumas (a – visam dokumentui, b – vienam elementui)

Iš aukščiau pateiktų rezultatų matyti, jog greičiausiai tikrinimo procedūrą įvykdo *Sun Java XML Digital Signature API* procedūra. Parašo tikrinimo procedūros vykdymo trukmė 20-30 kartų mažesnė nei parašo suformavimo, vidutiniškai apie 230 kartų mažesnė už *Apache Santuario* tikrinimo procedūros trukmę bei vidutiniškai apie 40 kartų mažesnė už *XAdES4j* tikrinimo procedūros trukmę.

Apskritai palyginus parašo formavimo ir tikrinimo procedūrų trukmę skirtingų dydžių dokumentams, matyti, kad trukmė, dokumento dydžiui pakitus 10 kartų, padidėja nuo vidutiniškai 3,8% (5000 B ir 50000 B dydžio dokumentų atveju) iki vidutiniškai 11.5% (50000 B ir 500000 B dydžio dokumentų atveju). Iš 37 pav. a. dalies grafikų matyti, jog kreivės, priklausomai nuo dokumento dydžio, kinta panašiai, tad skaičiuojant vidutinį visų trukmių pokytį procentais negausime didelių paklaidų.

Nagrinėjant absoliučias laiko reikšmes, matyti, kad dokumento dydžiui pakitus 10 kartų, jo apdorojimo trukmė pasikeičia iki 100ms. Apdorojant vieną dokumentą, toks laiko pokytis vartotojui praktiškai nėra pastebimas, o apdorojant šimtus dokumentų, kelios papildomos sekundės taip pat nėra reikšmingos, tad galima teigti, kad nagrinėtos parašo formavimo ir tikrinimo procedūros apskritai nėra jautrios dokumento dydžiui (smarkiai kintant dokumento dydžiui, procedūros vykdymo laikas pakinta palyginti mažai).

Taigi, jei sistemoje, kurioje realizuotas darbe aprašomas el. tranzakcijų dokumentų pasirašymo metodas, būtina naudoti XMLDSig standarto parašus, verčiau naudoti *Sun Java XML Digital Signature API* biblioteką ir trūkstamas funkcijas (palyginti su *Apache* Santuario – tai viešojo rakto parametrų įterpimas į dokumentą) realizuoti patiems sistemos kūrėjams. Tikėtina, kad tokiu atveju kriptografinės procedūros bus vis tiek įvykdytos greičiau nei naudojant *Apache Santuario* biblioteką, tačiau šiai prielaidai įrodyti reikia papildomų greitaveikos tyrimų.

XAdES standarto parašo naudojimas suteikia papildomų dokumento apsaugos priemonių (kvalifikuota laiko žyma, sertifikavimo tarnybos sertifikato duomenys), tad siekiant geriau apsaugoti dokumentą, nepaisant mažesnės kriptografinių ir dokumento apdorojimo procedūrų spartos, verčiau naudoti šio standarto parašus.

Šifravimo procedūrų greitaveikos tyrimas

Šifravimo procedūrų greitaveikos tyrimo tikslas ir esmė – imituoti aibės dokumentų užšifravimą ir iššifravimą, siekiant išmatuoti šifravimo procedūrų greitaveiką priklausomai nuo naudojamo šifravimo algoritmo, dokumento dydžio (šifruojamų duomenų kiekio).

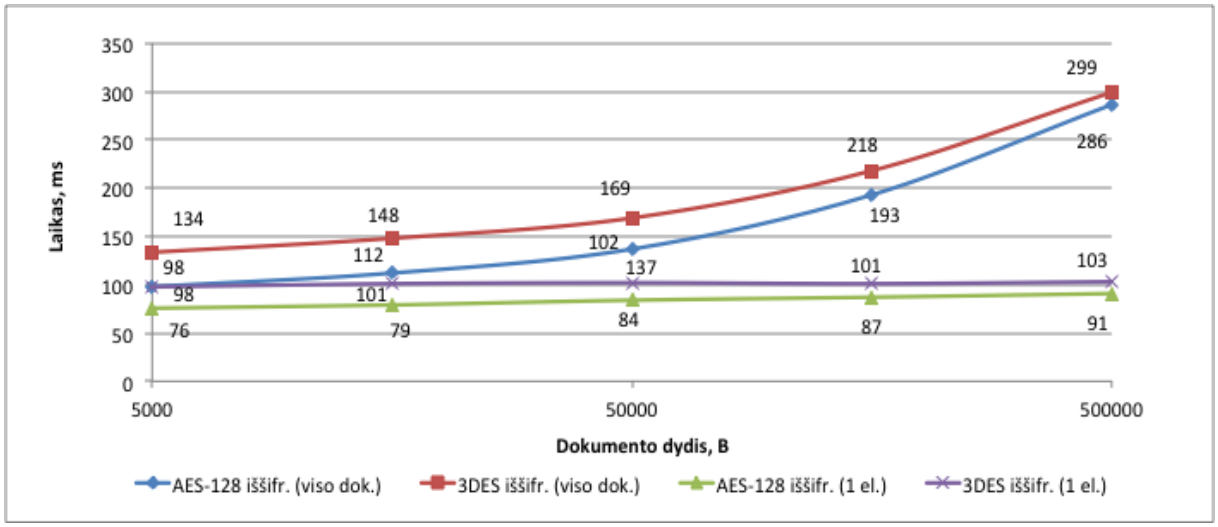
Matavimo algoritmas iš esmės atitinka pavaizduotąjį 35 paveiksle (blokas „Kviesti kriptografinę procedūrą“ šiuo atveju reiškia, kad kviečiama dokumento užšifravimo arba iššifravimo procedūra). Kaip ir pirmojo tyrimo atveju, matuojant šifravimo procedūrų greitaveiką, kiekviena procedūra įvykdyta po 10 kartų, ir, apskaičiuojant šių 10 matavimų rezultatų vidurkį, nustatytas vidutinis vienos procedūros vykdymo laikas ((1) formulė).

Tyrimo metu kriptografinių procedūrų greitaveika vertinta taip pat dviem būdais:

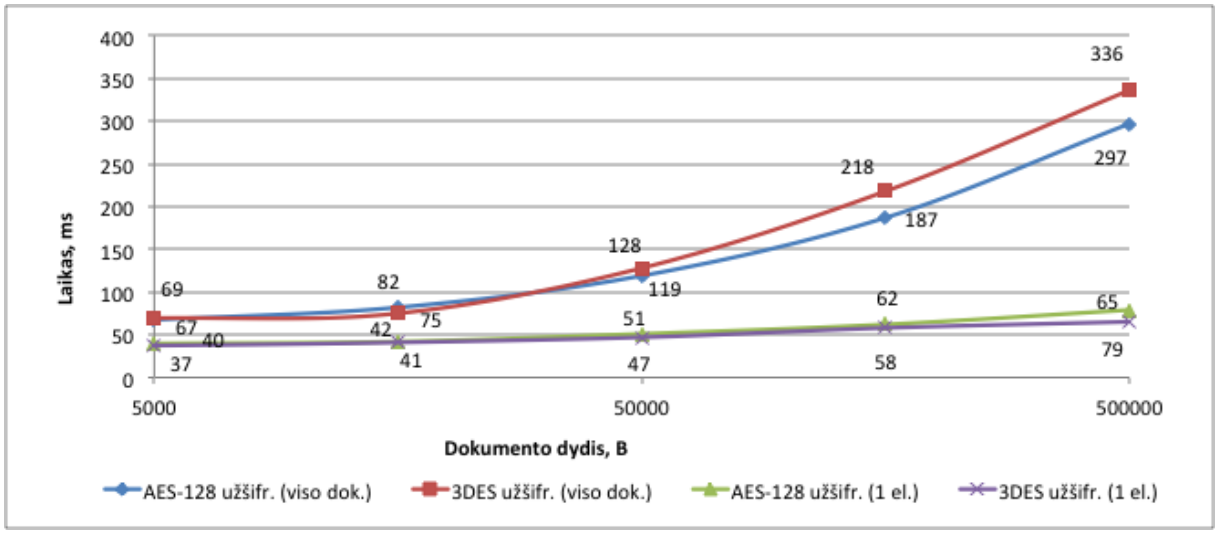
- Vykdant viso dokumento užšifravimą ir iššifravimą.
- Vykdant vieno pasirinkto elemento užšifravimą ir iššifravimą.

Antruoju atveju, buvo įterptas ir pasirinktas šifruoti 50 B dydžio (50 simbolių elementas), prieš tai šifravimo programai paduotas per parametrus. Tokiu būdu vertinta greitaveikos priklausomybė nuo šifruojamų duomenų kiekio (dokumento dydžio ir parinkto elemento dydžio).

Žemiau pateikti viso dokumento ir pasirinkto elemento užšifravimo bei iššifravimo procedūrų greitaveikos rezultatai.



a.



b.

38 pav. Dokumento užšifravimo (a) ir iššifravimo (b) procedūrų trukmė

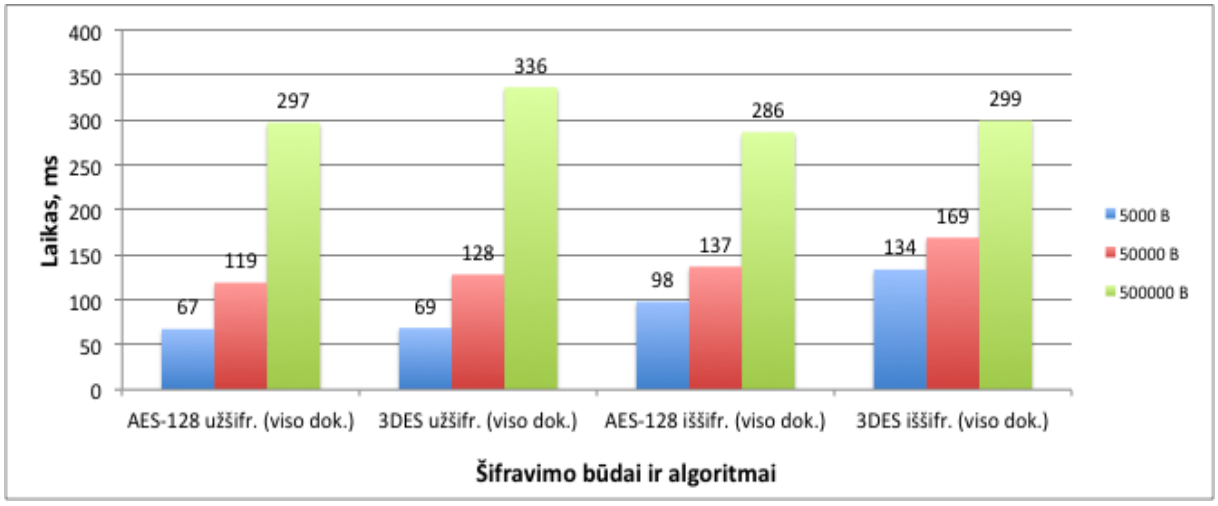
Iš matavimų rezultatų matyti, jog viso dokumento užšifravimo procedūros vykdymo trukmė sparčiai kinta priklausomai nuo šifruojamo dokumento dydžio. Dokumento dydžiui pakitus 10 kartų, viso dokumento užšifravimo trukmė pakito nuo vidutiniškai 45% (5000 B ir 50000 B dydžio dokumentų atveju) iki vidutiniškai 61% (50000 B ir 500000 B dydžio dokumentų atveju).

Analogiškai gali būti apskaičiuoti ir vieno elemento užšifravimo, viso dokumento ir vieno elemento iššifravimo procedūros vykdymo trukmės pokyčiai priklausomai nuo dokumento dydžio. Tačiau svarbiau pastebėti, kad vieno elemento užšifravimo procedūros trunka vidutiniškai 43.5% (5000 B atveju) – 77,2% (500000 B) trumpiau, negu viso dokumento. Panašaus santykio pokyčius galima įžvelgti ir iššifravimo grafikuose. Taigi

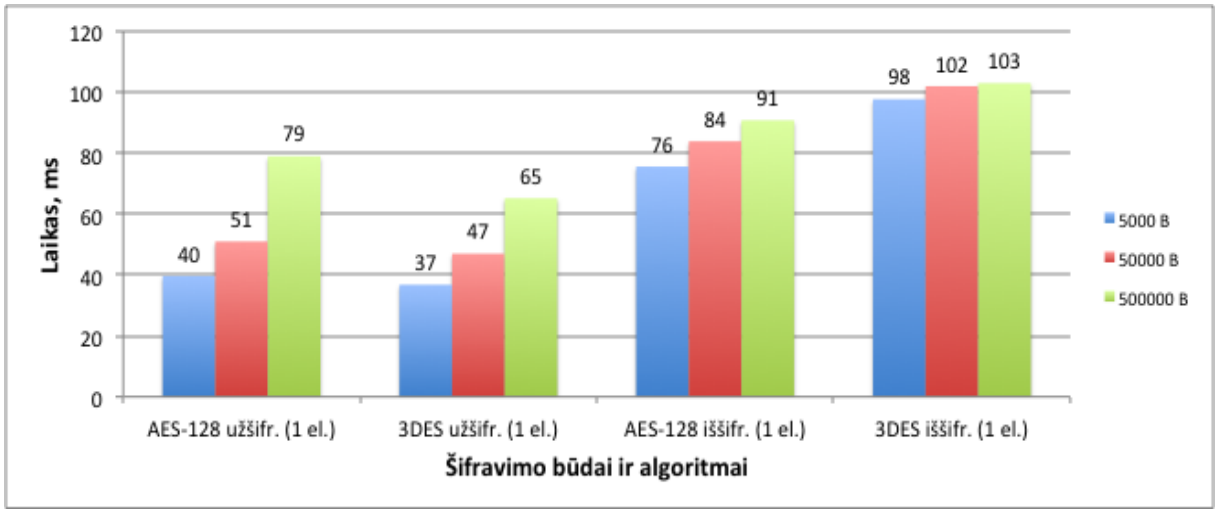
kintant dokumento dydžiui, ženkliai keičiasi ir procedūrų vykdymo trukmė, todėl galima teigti, jog šifravimo ir iššifravimo procedūros yra jautrios dokumento dydžiui (bendrai – šifruojamų duomenų kiekiui).

Šifravimo procedūrų vykdymo trukmės asimetriškumo tyrimas.

Siekiant išsiaiškinti, ar skiriasi ir kiek skiriasi atvirkštinės operacijos – dokumento iššifravimo procedūros greitaveika nuo užšifravimo greitaveikos, buvo atlikti atitinkami matavimai. Lyginamieji užšifravimo ir iššifravimo procedūrų vykdymo trukmės matavimų rezultatai pateikti 39 paveiksle.



a.



b.

39 pav. Viso dokumento ir vieno pasirinkto elemento užšifravimo ir iššifravimo procedūrų trukmės asimetriškumas

Iš diagramų matyti, jog užšifravimo ir iššifravimo procedūrų vykdymo trukmė šifruojant tą patį duomenų kiekį skiriasi santykinai mažai, palyginti su parašo formavimo procedūrų vykdymo trukme. Šiek tiek didesnę trukmę užšifravimo atveju galima paaiškinti

tuo, kad be šifravimo operacijos, dar atliekamos rakto generavimo bei šifravimo ir dokumento struktūros pakeitimo operacijos, o iššifravimo procedūros metu vykdomos tik rakto iššifravimo ir dokumento struktūros pakeitimo operacijos.

Taigi jei ne visas dokumento turinys yra konfidencialus, verta šifruoti tik tam tikrus pasirinktus elementus ir sutaupyti dalį laiko, kuris būtų sugaištamas užšifruoti visą dokumentą.

Raktų ir sertifikatų nuskaitymo iš saugyklų greitaveikos tyrimas

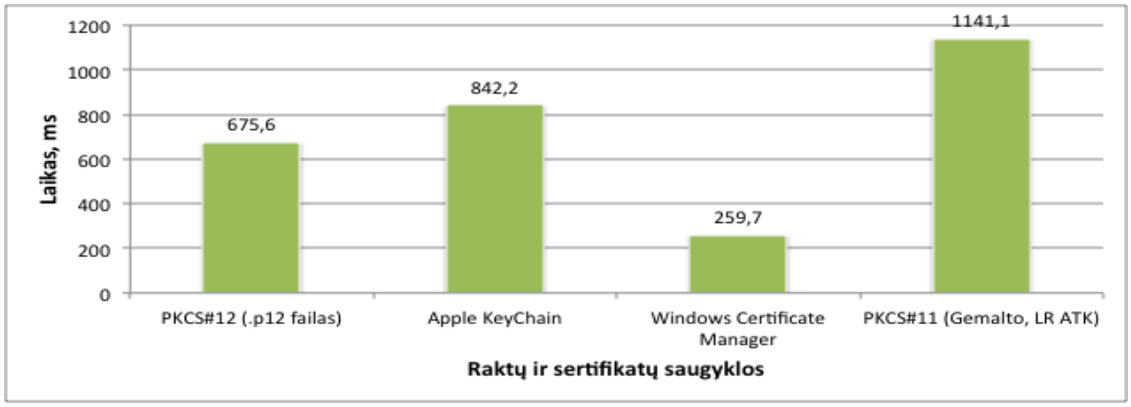
Šio tyrimo tikslas ir esmė – pamatuoti raktų ir sertifikatų nuskaitymo iš kelių populiarių tipų saugyklų spartą, įvertinti greitaveikos skirtumus bei pagal juos priskirti prioritetus, kokia tvarka galėtų būti pagal šiame darbe aprašomą metodą realizuotame sprendime naudojamos (ar siūlomos naudoti) saugyklos pagal prieinamumą ir veikimo spartą.

Matavimų algoritmas yra praktiškai toks pat, kaip pavaizduotasis 35 paveiksle, tik bloko „Kviesti kriptografinę procedūrą“ vykdymo metu, per standartizuotą *Java Crypto API* ir paties realizuotą sąsają, yra „užkraunama“ (*angl. load*) raktų saugykla bei standartizuotais kreipiniais, per parametrus paduodant autentifikacijos duomenis nuskaitomi raktai bei sertifikatai.

Autentifikacijos duomenys kiekvienai raktų saugyklai yra skirtingi:

- PKCS#12 (*.p12 byla) – slaptažodis;
- *Apple KeyChain* – Mac OS X vartotojo slaptažodis;
- *Windows Certificate Manager* – *null* reikšmė prisijungimui prie raktų saugyklos ir slaptažodis privačiam raktui (pasiekiama tik aktyvaus vartotojo saugykla, registre identifikuojama kaip „Windows-MY“, „Windows-ROOT“ saugykla pasiekiama tik administratoriaus teises turintiems vartotojams;
- PKCS#11 (Lietuvos Respublikos asmens tapatybės kortelė) - PIN kodas.

Matavimų, atliktų raktus nuskaitant iš aukščiau minėtų 4 populiarių saugyklų, rezultatai pateikti 40 paveiksle.



40 pav. Raktų ir sertifikatų nuskaitymo iš įvairių saugyklų trukmė

Iš paveikslo matyti, kad raktai ir sertifikatai greičiausiai buvo nuskaityti iš *Windows Certificate Manager* saugyklos. Tokį rezultatą galima paaiškinti tuo, kad *Windows Certificate Manager* yra *Microsoft Management Console* tarnybos įskiepis, o ši tarnyba paleidžiama kartu su OS. Taigi programa jau budi darbinėje atmintyje, užtrunkama tik nuskaityti raktus ir sertifikatus iš duomenų bazės.

Antra pagal spartą – PKCS#12 saugykla (dažniausiai, tai šifruota byla su .p12 ar .pfx plėtiniu). Iš jos užtrunkama raktus ir sertifikatus nuskaityti tiek, kiek trunka bylos nuskaitymas iš standžiojo disko, vartotojo autentifikacija (slaptažodžio apdorojimas) bei raktų bei sertifikatų eksportavimas.

Nuskaitymas iš *Apple KeyChain* užtruko ilgiau nei iš .p12 bylos, nes prieš nuskaitant raktus ir sertifikatus iš specialios šifruotos bylos, *MacOS X KeyChain* tarnyboje atliekama papildoma vartotojo autentifikacija ir leidimų patvirtinimo operacijos.

Bandymų rezultatai taip pat parodė, kad raktų ir sertifikatų nuskaitymas iš PKCS#11 saugyklos (išorinės laikmenos, o šiuo atveju – lustinės kortelės – LR asmens tapatybės kortelės) trunka ilgiausiai. Laikas gaištamas kreipiantis į specialią biblioteką (sąsajos klasių ir metodų rinkinys), tuomet į kortelių skaitytuvo tvarkyklę, kortelės tvarkyklę, autentifikuojantis kortelės programoje (*angl. smart-card applet*).

Taigi iš tyrimo rezultatų matyti, jog raktų nuskaitymo trukmė iš skirtingų tipų saugyklų gali skirtis kelis kartus. Vis dėlto OS esančios saugyklos (*Windows Certificate Manager* ir *Apple KeyChain*) taiko daugiau apsaugos priemonių, negu kitos (ribojama prieiga ne tik prie raktų bet ir prie pačių saugyklų ar jų skyrių). Tyrimui naudota lustinė kortelė turi papildomą apsaugos priemonę – ribotą kiekį neteisingų PIN įvedimo kartų, po kurių kortelės programa užsiblokuoja. Įvertinus visose nagrinėtos saugyklose taikomas apsaugos priemones, galima teigti, jog bylos pavidalo saugykla yra nesaugiausia iš nagrinėtų. Į tai verta atsižvelgti ir realizuojant sprendimą, paremtą darbe aprašomu el. tranzakcijų dokumentų pasirašymo metodu.

3.3 Kokybinė metodo analizė

Individualus metodo vertinimas

Kaip buvo minėta pirmoje darbo dalyje, metodo tikslas yra išlaikyti dokumentą nepakeistą ir konfidencialų visame kelyje nuo siuntėjo iki gavėjo. Šiuo atveju darbe aprašytu metodu šifruotas ar dalinai šifruotas dokumentas gali būti saugos požiūriu pranašesnis už tik pasirašytą dokumentą, perduodamą transportu, apsaugotu SSL/TLS protokolu, nes pastarasis dokumentas tarpiniuose taškuose (taikomųjų programų serveriuose ir duomenų saugyklose) niekaip nesaugomas.

Darbe aprašytu metodu realizuoti sprendimai gali būti dar vienas žingsnis link vartotojui draugiškos ir patogios (paremtos principu „ką matau, tą pasirašau“), žiniatinklio aplinkoje veikiančios tranzakcijų dokumentų valdymo terpės sukūrimo. Tokia terpė skatina naujų, tiesiogine vartotojo identifikacija ir autentifikacija pagrįstų paslaugų kūrimą tiek valstybiniame tiek verslo sektoriuje. Viena iš tokių paslaugų galėtų būti sutarčių pasirašymas ir pateikimas tiesiogiai internetu: vartotojas naršyklės lange gauna sutarties dokumentą, jį pasirašo ir siunčia paslaugos teikėjui. Jei paslaugų teikėjo IT infrastruktūros dalis yra nuomojama (naudojasi duomenų centrų bei taikomųjų programų serverių paslaugomis), naudojant tik transporto lygmens apsaugą, gali kilti pavojus dokumentų konfidencialumui ir vientisumui. Šių grėsmių įtaka iš principo sumažėja taikant darbe aprašytu metodu paremtus sprendimus.

Darbe aprašytas metodas taip pat turi ir trūkumų. Vienas iš jų – pasirašymo programos dydis. Kuo daugiau raktų saugyklų tipų, pasirašymo metodų bus palaikoma, tuo didesnė bus programos apimtis ir gali siekti keletą MB. Tokiu atveju reiktų galimų raktų saugyklų bibliotekas atskirti nuo pasirašymo programos ir inicijuoti jų parsiuntimą į vartotojo kompiuterį atskirai, po panaudojimo – pašalinti. Tad optimalaus sprendimo paieška šiai problemai spręsti galėtų būti viena iš tolimesnių tyrimų krypčių.

Palyginimas su kitais metodais

Darbe buvo siekiama sukurti metodą, turintį kuo daugiau gerųjų pirmoje darbo dalyje apžvelgtų metodų savybių ir kuo mažiau šių metodų trūkumų. Kadangi metodai tarpusavyje skiriasi tiek savo galimybėmis, tiek veikimo principu (netgi paskirtimi), negalima atlikti bendrais kriterijais paremtų kiekybinių tyrimų (pavyzdžiui, greitaveikos matavimų su vienodais parametrais ir tarpusavyje lyginamais rezultatais).

Tačiau metodus galima palyginti kokybiškai pagal kriterijus, sudarytus remiantis darbo tikslu ir iškeltais uždaviniais:

- Tinkamumas naudoti žiniatinklio aplinkoje. Tai reiškia, jog metodo realizacija turi veikti interaktyviai per žiniatinklio naršyklę.
- Kvalifikuotų XAdES standarto parašų formavimas. Pirmojoje darbo dalyje (1.4.1 skyriuje) teigta, jog baziniai XMLDSig parašai nepakankamai užtikrina dokumento saugą įvairiais panaudojimo atvejais, todėl ir buvo sukurtas XAdES standartas. Be to, tik šio standarto kvalifikuoti parašai pripažįstami ir naudojami Lietuvos e-valdžios infrastruktūroje.
- Dokumento šifravimas. Skaitmeninis parašas gali užtikrinti dokumento autentiškumą, tačiau dažnai kartu naudojamos priemonės ir konfidencialumui užtikrinti. Dažniausiai, tai transporto lygmens saugos protokolai. Kadangi šie protokolai nesprendžia darbo pirmojoje dalyje įvardintų problemų (dokumento perdavimo per tarpininką), tenka dokumentus šifruoti individualiai.
- Sąsajos su įvairiomis raktų saugyklomis. Numatyta galimybė nuskaityti raktus ir sertifikatus iš įvairių raktų saugyklų per vienodą sąsają – lankstumo ir kokybės požymis.
- Taisyklingo XML dokumento (su parašu ir šifruoto) formavimas. Darbe siekiama, kad aprašytu metodu suformuoti dokumentai atitiktų iš anksto apibrėžtą schemą. Taip pat siekiama, kad pasirašytus (ar net šifruotus) dokumentus būtų galima apdoroti, įvairiai jais disponuoti elektroninėje erdvėje naudojant prieinamus (neužšifruotus) metaduomenis.

Metodų vertinimas pagal įvardintus kriterijus pateiktas 5 lentelėje.

5 lentelė. Kokybinis metodų vertinimas

Metodai / Kriterijai	Nakov Document Signer	WebSign Project	OpenSign	Darbe aprašytasis
Tinkamas naudoti žiniatinklio aplinkoje	T	T	T	T
Formuoja kvalifikuotus XAdES standarto parašus	N	T	N	T
Gali šifruoti dokumentą	N	N	N	T
Sąsajos su įvairiomis raktų saugyklomis	• PKCS#12 • PKCS#11[2]	• PKCS#12 • PKCS#11[1]	• PKCS#12 • PKCS#11[1]	• PKCS#12 • PKCS#11[2] • Apple KeyChain • Windows Certificate Manager
Formuoja taisyklingą XML dokumentą	N	T	T	T

Pirmąjį kriterijų tenkina visi nagrinėti metodai – visi pritaikyti interaktyviai veikti per interneto naršyklę.

XAdES standarto parašus formuoja tik „WebSign Project“ ir darbe aprašytasis metodas. Tiesa, „WebSign Project“ kuria PDF dokumentus ir jiems formuoja atskirtuosius (*angl. detached*) parašus, kuriuos į serverį taip pat siunčia atskirai. Šiuo atveju darbe aprašytasis metodas yra pranašesnis už „WebSign Project“, nes kuria XML dokumentus su įterptu parašu, ir tokiais dokumentais lengviau disponuoti (be papildomų priemonių galima pasiekti dokumento turinį bei apdoroti pačius dokumentus).

Šifravimo galimybė nėra esminė, nes pagrindinė visų metodų funkcija – pasirašyti dokumentus. Tačiau darbe keliamos problemos, kaip apsaugoti dokumentus, kai tarp siuntėjo ir gavėjo veikia trečioji (galimai nepatikima) šalis. Tokiu atveju metode numatytas šifravimas, kaip papildoma apsaugos priemonė, yra privalumas.

Visi nagrinėti metodai palaiko vienokias ar kitokias raktų saugyklas. „Websign Project“ ir „OpenSign“ daugiau orientuoti į Microsoft technologijas, tad kitokių raktų saugyklų palaikymą ir pačią sąsają tektų realizuoti pačiam sistemos integruotojui, kuris realizuoja metodą. Kuriant darbe aprašytą metodą buvo galvojama apie tai, kad jo pagrindu realizuotas sprendimas gali būti naudojamas įvairioje aplinkoje (gali veikti skirtingose OS su

[1] Pritaikyta tik Micrososft Crypto API.

[2] Pritaikyta veikti su bet kokia Java Crytpo API palaikančia biblioteka.

skirtingomis raktų saugyklomis), tad numatyta, apibrėžta bei išbandyta bendra sąsaja įvairioms raktų saugykloms.

Visi nagrinėti metodai, išskyrus „Nakov Document Signer“, suformuoja XML dokumentus su XMLDSig arba XAdES standarto parašais. Dokumentas laikomas taisyklingu, jei jis po visų transformacijų atitinka nurodytą XML schemą. Šiame darbe su XML schemomis susiję dokumentų saugos aspektai nenagrinėjami. Kadangi „Nakov Document Signer“ neformuoja XML dokumento, o parašą ir sertifikatų grandinėlę grąžina tiesiog užkoduotą Base64, metodas negali būti tiesiogiai taikomas el. paslaugų sistemoms, kuriose keičiamasi XML dokumentais, tačiau gali būti pritaikytas dokumentų valdymo sistemose, kur dokumentams formuojami specialūs (galbūt nestandartinio formato) konteineriai su parašais.

IŠVADOS

Remiantis atlikta problemos analize, tyrimų rezultatais ir kokybės analize galima daryti tokias išvadas:

1. Išanalizavus el. paslaugų teikimo schemą „debesies“ architektūros IT infrastruktūroje, įsitikinta, jog transporto lygmens apsauga pagrįsti metodai netinkami perduodamų dokumentų apsaugai. Jais negalima užtikrinti dokumento autentiškumo ir konfidencialumo, kol jis yra saugomas „debesyje“.
2. Išnagrinėjus kelis egzistuojančius sprendimus el. tranzakcijų dokumentų apsaugai žiniatinklio aplinkoje, paaiškėjo, jog nė viename iš jų taikomos apsaugos priemonės nėra pakankamos el. tranzakcijų dokumentų apsaugai perduodant, apdorojant ir saugant juos „debesyje“.
3. Remiantis išnagrinėtų el. dokumentų apsaugos metodų privalumais, sukurtas elektroninių tranzakcijų pasirašymo žiniatinklio aplinkoje metodas. Pagrindiniai funkciniai privalumai (kas nenumatyta kituose metoduose): pažangaus XAdES standarto parašo formavimas XML dokumentams, dokumento dalies pasirašymo galimybė, viso XML dokumento ar jo dalies šifravimas, kelių tipų raktų saugyklų palaikymas.
4. Atlikus greitaveikos matavimus, nustatyta, kad parašo formavimo ir tikrinimo procedūros mažai jautrios dokumento dydžiui, todėl, siekiant užtikrinti viso dokumento autentiškumą, parašą galima formuoti visam dokumento turiniui – dėl to procedūros vykdymo sparta pastebimai nesumažėja. Kur galima, rekomenduojama formuoti XAdES standarto parašus. Jie užtikrina geresnę dokumento apsaugą, o formavimo procedūrų greitaveika ne mažesnė nei procedūrų, formuojančių pilnaverčius XMLDSig parašus.
5. Dokumento turinio šifravimo procedūros yra jautrios dokumento dydžiui, todėl, jei galima, verčiau šifruoti tik dalį didelio dokumento (elementus su aukšto konfidencialumo lygio informacija). Tyrimas parodė, kad tokiu būdu iki 70% galima sumažinti šifravimo procedūros vykdymo trukmę.
6. Greitaveikos skirtumai tarp skirtingų tipų raktų ir sertifikatų saugyklų yra pagrįsti ir priklauso nuo pačios raktų saugyklos realizacijos ir naudojimo pobūdžio. Vis dėlto, įvertinus greitaveiką bei apsaugos priemones, siūloma prioritetą teikti lustinėms kortelėms, nes jose naudojama ne tik loginė, bet ir fizinė raktų bei sertifikatų sauga, o raktų ir sertifikatų nuskaitymo sparta vartotojui nėra pastebimai mažesnė.

7. Remiantis atlikta kokybės analize, galima teigti, jog metodas kai kuriomis savo savybėmis ir galimybėmis yra pranašesnis už kitus, darbe apžvelgtus metodus, tačiau, siekiant jį pritaikyti realioje sistemoje, reikia atlikti papildomus tyrimus ir patobulinimus, susijusius su išorinių resursų (bibliotekų su kriptografinėmis procedūromis, raktų saugyklų, dokumentų repozitoriumų) panaudojimu.

Atsižvelgiant į pateiktas išvadas, galima teigti, jog metodas tinkamas el. paslaugoms teikti „debesies“ architektūra paremtuose sprendimuose ar dokumentų apsaugai organizuoti žiniatinklio aplinkoje veikiančiose dokumentų valdymo sistemose.

LITERATŪRA

1. DCMI. Expressing Dublin Core Description Sets using XML (DC-DS-XML). Prieiga per internetą: http://dublincore.org/documents/dc-ds-xml/ Žiūrėta 2011-03-16.
2. Defective D. Sign & Encrypt in S/MIME, PKCS#7, MOSS, PEM, PGP, and XML Prieiga per internetą: http://world.std.com/~dtd/sign_encrypt/sign_encrypt7.html, Žiūrėta 2010-01-27.
3. Elektroniniai valdžios vartai. Prieiga per internetą: http://www.epaslaugos.lt. Žiūrėta 2010-01-27.
4. ETSI. XML Advanced Electronic Signatures (XAdES). June 2009. TS 101 903 v1.4.1.
5. European Parliament. „Directive 1999/93/ec on a community framewrok for electronic Signatures“. Official Journal of the European Communities, January 2000.
6. Fitzgerald M. Building B2B Applications with XML : A Resource Guide. John Wiley & Sons, 2001, p. 137-154.
7. Gudas A. Elektroninio parašo standartai Lietuvoje. „Mokslas – Lietuvos ateitis“ - 10-osios Lietuvos jaunųjų mokslininkų konferencijos medžiaga [Vilnius, 2007 balandžio 17d.] , Informatika, Vilnius 2007, p. 490 – 497.
8. Hassler V., Then O. Controlling Applets' Behavior in a Browser. Computer Security Applications Conference, Proceedings, 14th Annual 7-11 Dec. 1998, p. 120-125.
9. Kazanavičius E, Venčkauskas A., Liutkevičius A., Vrubliauskas A. Informacijos saugos vadyba. Technologija, Kaunas 2008, 168 p.
10. Kuzmowycz G., E-mail security with S/MIME. SANS GIAC Security Essentials, SANS Institute, 2001. Prieiga per internetą: www.sans.org/reading_room/whitepapers/vpns/email_security_with_s/mime_739 Žiūrėta 2010-01-27.
11. Loshin P. J. Electronic Commerce, 4th Edition. Charles River Media, 2002, p. 15-23.
12. Lucas M. PGP and GPG. No Starch Press, Incorporated, 2006, p. 216.
13. Masud E., Mahbubur Rahman Md., Mehedi Masud Md.. Design of Extensible Security Architecture for Java Applets.

International Journal of The Computer, The Internet and Management, Vol. 11, No.2, 2003, p. 15-23.

14. Miller H.G., Veiga J. Cloud Computing: Will Commodity Services Benefit Users Long Term? IT Professional, vol. 11, Nov.-Dec. 2009, p. 57 – 59.
15. Nakov S. Java Applet for Signing with a Smart Card.
Prieiga per internetą: www.developer.com//article.php/3587361. Žiūrėta 2009-10-21.
16. Nakov S. Nakov Document Signer: A System for Digitally Signing Documents in Web Applications.
Prieiga per internetą: www.developer.com//article.php/3298051. Žiūrėta 2009-10-21.
17. OpenCES Project.
Prieiga per internetą: http://www.openoces.org/opensign/index.html
Žiūrėta 2010-04-20.
18. Oppliger R. Security Technologies for the World Wide Web, 2nd Edition..
Artech House, Inc, 2002.
19. Ramsdell B. Internet RFC 2633 „S/MIME Version 3 Message Specification“,
June 1999.
Prieiga per internetą: ftp://ftp.isi.edu/in-notes/rfc2633.txt Žiūrėta 2010-01-27.
20. Rhee, M.Y. Internet security: cryptographic principles, algorithms, and protocols.
John Wiley & Sons Ltd, ISBN 0-470-85285-2, 2003. p.70 – 75, 107 – 122.
21. Sakalauskas E., Blažauskas T., Lukšys K. Elektroninių dokumentų ir duomenų sauga.
Vitae Litera, Kaunas 2008, 47 p.
22. Statistikos departamentas prie Lietuvos Respublikos Vyriausybės. Informacinės ir žinių visuomenės pagrindiniai rodikliai.
Prieiga per internetą:
http://www.stat.gov.lt/uploads/docs/Inf_zin_vis_pletr_statist_rodikliai_091130.doc .
Žiūrėta 2010-01-27.
23. W3C. XMLSignature Syntax and Processing.
Prieiga per internetą: http://www.w3.org/TR/xmldsig-core/
Žiūrėta 2011-04-22.
24. W3C. XML Encryption Syntax and Processing.
Prieiga per internetą: http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/
Žiūrėta 2011-04-22.
25. WebSign Project.
Prieiga per internetą: http://rcardon.free.fr/websign/wakka.php?wiki=MainPage
Žiūrėta 2010-04-20.

PRIEDAI

1 priedas. Tyrime naudoti pradiniai po apdorojimo gauti XML dokumentai

Pradinis XML dokumentas

```
<?xml version="1.0"?>
<metadata
  xmlns="http://example.org/myapp/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://example.org/myapp/ http://example.org/myapp/schema.xsd"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/">
  <dc:title>
    UKOLN
  </dc:title>
  <dcterms:alternative>
    UK Office for Library and Information Networking
  </dcterms:alternative>
  <dc:subject>
    national centre, network information support, library
    community, awareness, research, information services,public
    library networking, bibliographic management, distributed
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
        …
    national centre, network information support, library
    community, awareness, research, information services,public
    library networking, bibliographic management, distributed
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
  </dc:subject>
  <dc:subject xsi:type="dcterms:DDC">
    062
  </dc:subject>
  <dc:subject xsi:type="dcterms:UDC">
    061(410)
  </dc:subject>
  <dc:description>
    UKOLN is a national focus of expertise in digital information
    management. It provides policy, research and awareness services
    to the UK library, information and cultural heritage communities.
    UKOLN is based at the University of Bath.
  </dc:description>
  <dc:description xml:lang="fr">
    UKOLN est un centre national d'expertise dans la gestion de l'information
    digitale.
  </dc:description>
  <dc:publisher>
    UKOLN, University of Bath
  </dc:publisher>
  <dcterms:isPartOf xsi:type="dcterms:URI">
    http://www.bath.ac.uk/
  </dcterms:isPartOf>
  <dc:identifier xsi:type="dcterms:URI">
    http://www.ukoln.ac.uk/
  </dc:identifier>
  <dcterms:modified xsi:type="dcterms:W3CDTF">
    2001-07-18
  </dcterms:modified>
  <dc:format xsi:type="dcterms:IMT">
    text/html
  </dc:format>
  <dcterms:extent>
    14 Kbytes
  </dcterms:extent>
</metadata>
```

Dokumentas, pasirašytas naudojant Sun XML Digital Signature API

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://example.org/myapp/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://example.org/myapp/ http://example.org/myapp/schema.xsd">
  <dc:title>
    UKOLN
  </dc:title>
  <dcterms:alternative>
    UK Office for Library and Information Networking
  </dcterms:alternative>
  <dc:subject>
    national centre, network information support, library
    community, awareness, research, information services,public
    library networking, bibliographic management, distributed
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
       …
    national centre, network information support, library
    community, awareness, research, information services,public
    library networking, bibliographic management, distributed
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
  </dc:subject>
  <dc:subject xsi:type="dcterms:DDC">
    062
  </dc:subject>
  <dc:subject xsi:type="dcterms:UDC">
    061(410)
  </dc:subject>
  <dc:description>
    UKOLN is a national focus of expertise in digital information
    management. It provides policy, research and awareness services
    to the UK library, information and cultural heritage communities.
    UKOLN is based at the University of Bath.
  </dc:description>
  <dc:description xml:lang="fr">
    UKOLN est un centre national d'expertise dans la gestion de l'information
    digitale.
  </dc:description>
  <dc:creator>Giedrius Budrys</dc:creator>
  <dc:publisher>
    UKOLN, University of Bath
  </dc:publisher>
  <dcterms:isPartOf xsi:type="dcterms:URI">
    http://www.bath.ac.uk/
  </dcterms:isPartOf>
  <dc:identifier xsi:type="dcterms:URI">
    http://www.ukoln.ac.uk/
  </dc:identifier>
  <dcterms:modified xsi:type="dcterms:W3CDTF">
    2001-07-18
  </dcterms:modified>
  <dc:format xsi:type="dcterms:IMT">
    text/html
  </dc:format>
  <dcterms:extent>
    14 Kbytes
  </dcterms:extent>
  <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
    <SignedInfo>
      <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <Reference URI="">
        <Transforms>
          <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
        </Transforms>
        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <DigestValue>MyA9HdQy4LMDU1DfACNfyNNoiQ8=</DigestValue>
      </Reference>
    </SignedInfo>
    <SignatureValue>
eaFPfJ8Oc/RYWm5g5PKsB60o2D05DffNbaaO/DCfIuvs5GfACW4xzBuQXUyMN+ctKzx6ES8BjBHM
qn5j12ZQMB2OGaDsELd3U65EcpcGW36nI75+NxMy7CFzt75dMHyqozrAvAcWMVWXGc1cpFeaRfdV
BHxM6aTilOOeRbrkDH8kBlPTCYOaxwcy+zPupw==
    </SignatureValue>
```

```
    <KeyInfo>
      <X509Data>
       <X509Certificate>
MIIGFjCCA/6gAwIBAgIDAYa/MA0GCSqGSIb3DQEBBQUAMIGmMQswCQYDVQQGEwJMVDEoMCYGA1UE
        …
xSWQbOAUM9YZ6KhIBf96DlXA+fVrAfkYsVYF0/Zvrd7euvM6llnJKxz3mV/xSB0L1vcZ0V1Fn7Lt
rmYQfNI7ij16Gd2ajv++Bd9zldSOwn0=</X509Certificate>
      </X509Data>
    </KeyInfo>
  </Signature>
</metadata>
```

Dokumentas, pasirašytas naudojant Apache Santuario

```
 <?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://example.org/myapp/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://example.org/myapp/ http://example.org/myapp/schema.xsd">
  <dc:title>
    UKOLN
  </dc:title>
  <dcterms:alternative>
    UK Office for Library and Information Networking
  </dcterms:alternative>
  <dc:subject>
    national centre, network information support, library
    community, awareness, research, information services,public
    library networking, bibliographic management, distributed
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
       …
    national centre, network information support, library
    community, awareness, research, information services,public
    library networking, bibliographic management, distributed
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
</dc:subject>
  <dc:subject xsi:type="dcterms:DDC">
    062
  </dc:subject>
  <dc:subject xsi:type="dcterms:UDC">
    061(410)
  </dc:subject>
  <dc:description>
    UKOLN is a national focus of expertise in digital information
    management. It provides policy, research and awareness services
    to the UK library, information and cultural heritage communities.
    UKOLN is based at the University of Bath.
  </dc:description>
  <dc:description xml:lang="fr">
    UKOLN est un centre national d'expertise dans la gestion de l'information
    digitale.
  </dc:description>
  <dc:creator Id="1">Giedrius Budrys
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"/>
        <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
        <ds:Reference URI="">
          <ds:Transforms>
            <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
            <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315#WithComments"/>
          </ds:Transforms>
          <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
          <ds:DigestValue>KocWq/gc/Siq5aW2wdHCgsFgn9I=</ds:DigestValue>
        </ds:Reference>
        <ds:Reference URI="http://www.w3.org/TR/xml-stylesheet">
          <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
          <ds:DigestValue>pABdjWJGeIB7U4NdYaZwMfhGCfQ=</ds:DigestValue>
        </ds:Reference>
      </ds:SignedInfo>
      <ds:SignatureValue>
vhyZo22cV27Wjwj+hzED+denQRQQdqb9aRh3zrlzQAVnzBQCTA4WPGZe/pNtZS12rbuvcttt7Nyy
```

```
Rs0mTE2Fe+6iTD3HpbncXCxoyL/92iNGK3v4GFxGE0vAm54kwU//S8yjQ1FlAUEqYAeD0VBHDv9Q
f+0RxmJqtR6GPUolbO0ej4csiBvcZXwc2wh5TQ==
      </ds:SignatureValue>
      <ds:KeyInfo>
        <ds:X509Data>
          <ds:X509Certificate>
MIIGFjCCA/6gAwIBAgIDAYa/MA0GCSqGSIb3DQEBBQUAMIGmMQswCQYDVQQGEwJMVDEoMCYGA1UE
        ...
xSWQbOAUM9YZ6KhIBf96DlXA+fVrAfkYsVYF0/Zvrd7euvM6llnJKxz3mV/xSB0L1vcZ0V1Fn7Lt
rmYQfNI7ij16Gd2ajv++Bd9zldSOwn0=
          </ds:X509Certificate>
        </ds:X509Data>
        <ds:KeyValue>
          <ds:RSAKeyValue>
            <ds:Modulus>
ywAFRXFr7u5Kp2J2OCqwgAZBLVLRMCm8dcWAviDZRh8zbycvzXeWXpBMSkn8EnTLRLQLiL9KlJ2E
        ...
8uIMjIrdwaO5xB17nPwTWLlP1EOQn0NNngT+SlJUNr6/5I1exfwODQjZjtXfdHXRg4rOYg7EbHuw
M+KyBOUeyA7mzIwqJIgGG+PuE/PaCxzEZjavSQ==
            </ds:Modulus>
            <ds:Exponent>AQAB</ds:Exponent>
          </ds:RSAKeyValue>
        </ds:KeyValue>
      </ds:KeyInfo>
    </ds:Signature>
  </dc:creator>
  <dc:publisher>
    UKOLN, University of Bath
  </dc:publisher>
  <dcterms:isPartOf xsi:type="dcterms:URI">
    http://www.bath.ac.uk/
  </dcterms:isPartOf>
  <dc:identifier xsi:type="dcterms:URI">
    http://www.ukoln.ac.uk/
  </dc:identifier>
  <dcterms:modified xsi:type="dcterms:W3CDTF">
    2001-07-18
  </dcterms:modified>
  <dc:format xsi:type="dcterms:IMT">
    text/html
  </dc:format>
  <dcterms:extent>
    14 Kbytes
  </dcterms:extent>
</metadata>
```

Dokumentas, pasirašytas naudojant XAdES4j

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://example.org/myapp/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://example.org/myapp/ http://example.org/myapp/schema.xsd">
  <dc:title>
    UKOLN
  </dc:title>
  <dcterms:alternative>
    UK Office for Library and Information Networking
  </dcterms:alternative>
  <dc:subject>
    national centre, network information support, library
    community, awareness, research, information services,public
    library networking, bibliographic management, distributed
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
  </dc:subject>
  <dc:subject xsi:type="dcterms:DDC">
    062
  </dc:subject>
  <dc:subject xsi:type="dcterms:UDC">
    061(410)
  </dc:subject>
  <dc:description>
    UKOLN is a national focus of expertise in digital information
    management. It provides policy, research and awareness services
    to the UK library, information and cultural heritage communities.
```

```
    UKOLN is based at the University of Bath.
       …
    UKOLN is a national focus of expertise in digital information
    management. It provides policy, research and awareness services
    to the UK library, information and cultural heritage communities.
    UKOLN is based at the University of Bath.
  </dc:description>
  <dc:description xml:lang="fr">
    UKOLN est un centre national d'expertise dans la gestion de l'information
    digitale.
  </dc:description>
  <dc:creator>Giedrius Budrys</dc:creator>
  <dc:publisher>
    UKOLN, University of Bath
  </dc:publisher>
  <dcterms:isPartOf xsi:type="dcterms:URI">
    http://www.bath.ac.uk/
  </dcterms:isPartOf>
  <dc:identifier xsi:type="dcterms:URI">
    http://www.ukoln.ac.uk/
  </dc:identifier>
  <dcterms:modified xsi:type="dcterms:W3CDTF">
    2001-07-18
  </dcterms:modified>
  <dc:format xsi:type="dcterms:IMT">
    text/html
  </dc:format>
  <dcterms:extent>
    14 Kbytes
  </dcterms:extent>
  <ds:Signature Id="xmldsig-3cb0b656-4314-4d21-a716-c190a2c73225"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <ds:Reference
        Id="xmldsig-3cb0b656-4314-4d21-a716-c190a2c73225-ref0" URI="">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
        </ds:Transforms>
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
        <ds:DigestValue>1gTBVgi9zFjs7bpDRzBe/BPEhJpHaZnTc1Rk46B/9OE=</ds:DigestValue>
      </ds:Reference>
      <ds:Reference Type="http://uri.etsi.org/01903#SignedProperties" URI="#xmldsig-3cb0b656-
4314-4d21-a716-c190a2c73225-signedprops">
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
        <ds:DigestValue>vhtyl5HIwTSYE8N3U3EXNMmUpdokHL8j92ZJ+Ise/Qo=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue Id="xmldsig-3cb0b656-4314-4d21-a716-c190a2c73225-sigvalue">
j9c1jjTyf4916nP9nZd+/m+1ReDt/qRQdTLAycavBkVeXn97ia6cIBFTQanWYWYgHyM/AKaNNOpr
cCnBiJtYQUIPMZNbY48E0RrWmNZiexgUeAw7O5ptohk65ckkFJvHfL5P3X4swuTBNHGLcybLa9DD
yB3TGpEWjBsdldPcvhZXxxvCoNG1a3ouOIimkFx9EGgwdK0dxjbJqea5+zQ2eZOnqFd/iZpRAvuQ
yscdIuHXgTW+eVnwYoTvC5lWz9YEw+B6nQbLHVw/sKPGPfcM8SCAH/prDQ+LkaEcjOrYO2ngIznv
v/VXk1qZITw5vpttTuCjCBIGfnPF7ZxlEzvvEA==
</ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
       <ds:X509SubjectName>
1.2.840.113549.1.9.1=#161b67696564726975732e627564727973407374756425642e6b74752e6c74,CN=Giedrius
Budrys,2.5.4.42=#0c084769656472697573,2.5.4.4=#0c06427564727973,C=LT</ds:X509SubjectName>
        <ds:X509IssuerSerial>
          <ds:X509IssuerName>
1.2.840.113549.1.9.1=#16126365727440612d7369676e2e6b74752e6c74,CN=Academic signature Root
Authority,OU=Academic signature CA,O=Kaunas University of Technology,C=LT
          </ds:X509IssuerName>
          <ds:X509SerialNumber>100031</ds:X509SerialNumber>
        </ds:X509IssuerSerial>
        <ds:X509Certificate>
MIIGFjCCA/6gAwIBAgIDAYa/MA0GCSqGSIb3DQEBBQUAMIGmMQswCQYDVQQGEwJMVDEoMCYGA1UE
        …
xSWQbOAUM9YZ6KhIBf96DlXA+fVrAfkYsVYF0/Zvrd7euvM6llnJKxz3mV/xSB0L1vcZ0V1Fn7Lt
rmYQfNI7ij16Gd2ajv++Bd9zldSOwn0=
</ds:X509Certificate>
      </ds:X509Data>
      <ds:KeyValue>
        <ds:RSAKeyValue>
          <ds:Modulus>
ywAFRXFr7u5Kp2J2OCqwgAZBLVLRMCm8dcWAviDZRh8zbycvzXeWXpBMSkn8EnTLRLQLiL9KlJ2E
A3FuJzQ1SZ+E0U4PCO5jKO/rjgY05SWeAAeINbjYwJTchM8W8Gfvvdhd3dFOR7hdb0r2CMrFXb0l
```

```
2CoCM2XFLosAMAHv2c4pEqdutCTGdkN/0Q8o1K3gem4cpdyJbZYeiMDZZT9uRNL+bOnl4dTw0iKY
8uIMjIrdwaO5xB17nPwTWLlP1EOQn0NNngT+SlJUNr6/5I1exfwODQjZjtXfdHXRg4rOYg7EbHuw
M+KyBOUeyA7mzIwqJIgGG+PuE/PaCxzEZjavSQ==
</ds:Modulus>
          <ds:Exponent>AQAB</ds:Exponent>
        </ds:RSAKeyValue>
      </ds:KeyValue>
    </ds:KeyInfo>
    <ds:Object>
      <XAdES:QualifyingProperties
        Target="#xmldsig-3cb0b656-4314-4d21-a716-c190a2c73225"
        xmlns:XAdES="http://uri.etsi.org/01903/v1.3.2#"
xmlns:XAdES141="http://uri.etsi.org/01903/v1.4.1#">
        <XAdES:SignedProperties Id="xmldsig-3cb0b656-4314-4d21-a716-c190a2c73225-signedprops">
          <XAdES:SignedSignatureProperties>
            <XAdES:SigningTime>2011-05-12T21:57:28.039+03:00</XAdES:SigningTime>
            <XAdES:SigningCertificate>
              <XAdES:Cert>
                <XAdES:CertDigest>
                  <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                  <ds:DigestValue>
                       6H/vbqTR6ZNahpEWlz7fN5TzcPU8cmk96LNt6/FSQMk=
                  </ds:DigestValue>
                </XAdES:CertDigest>
                <XAdES:IssuerSerial>
                  <ds:X509IssuerName>
1.2.840.113549.1.9.1=#1612636572744061 2d7369676e2e6b74752e6c74,CN=Academic signature Root
Authority,OU=Academic signature CA,O=Kaunas University of Technology,C=LT
                  </ds:X509IssuerName>
                  <ds:X509SerialNumber>100031</ds:X509SerialNumber>
                </XAdES:IssuerSerial>
              </XAdES:Cert>
            </XAdES:SigningCertificate>
          </XAdES:SignedSignatureProperties>
        </XAdES:SignedProperties>
        <XAdES:UnsignedProperties>
          <XAdES:UnsignedSignatureProperties>
            <XAdES:SignatureTimeStamp>
              <ds:CanonicalizationMethod
                    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
       `       <XAdES:EncapsulatedTimeStamp>
MIIPPwYJKoZIhvcNAQcCoIIPMDCCDywCAQMxCzAJBgUrDgMCGgUAMIG7BgsqhkiG9w0BCRABBKCBqwSBqDCBpQIBAQYMKw
YBBAG/VQMCAQEAMB8wBwYFKw4DAhoEFKCq0mnJsUDsLRriIeaN1Ce6YKMlAgQA5VsOGA8yMDExMDUxMjE4NTcyOFoBAf+g
       …
+rJWWBUjDijuY+h6mouIDQ2Pvk80Cng+h3Fnzx8rYgz9rMwWI48NitOQxb3X9H+hrxazN2oAWJJwBsd5NZs822Znn5nPQ6
KV6DSeWq0SHKiuI0ZBhN72NuHsrz0xVqEhg4/gzNR7FLOoS7AYkxXyj0Y3IJuijd9lTJ6Wk7dK+Ru0l6ozkTN2AloETop8
gl52XMn8LByGrb0I5voGn1qA==
               </XAdES:EncapsulatedTimeStamp>
            </XAdES:SignatureTimeStamp>
          </XAdES:UnsignedSignatureProperties>
        </XAdES:UnsignedProperties>
      </XAdES:QualifyingProperties>
    </ds:Object>
  </ds:Signature>
</metadata>
```