KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

KOMPIUTERIŲ KATEDRA

Renatas Vencevičius

Saugaus verslo dokumentų perdavimo įmonių sąveikumo sistemose metodo sudarymas ir tyrimas

Magistro darbas

Darbo vadovas

doc. dr. N. Morkevičius

KAUNAS, 2011

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

KOMPIUTERIŲ KATEDRA

Renatas Vencevičius

Saugaus verslo dokumentų perdavimo įmonių sąveikumo sistemose metodo sudarymas ir tyrimas

Magistro darbas

Recenzentas

doc. dr. T. Adomkus

2011-05-

Vadovas

doc. dr. N. Morkevičius

2011-05-

Atliko

IFN-9/3 gr. stud.

Renatas Vencevičius

2011-05-24

KAUNAS, 2011

Turinys

Saugaus verslo dokumentų perdavimo įmonių sąveikumo sistemose metodo sudarymas ir tyrimas

Santrauka

Įmonių sąveikumo sistemose iškyla problemos apsaugant persiunčiamus verslo dokumentus. Sąveikumo sistemų specifika yra ta, kad jose naudojami serveriai ne tik persiunčia verslo dokumentus gautus iš siuntėjo gavėjui, bet ir atlieka nemažai kitų šių dokumentų transformavimo funkcijų. Dėl to tokiose sistemose galima išskirti keletą kylančių saugos problemų. Pirmiausia reikia užkrinti perduodamų dokumentų tarp serverio ir siuntėjo arba gavėjo konfidencialumą ir neišsiginamumą. Kita problema yra ta, kad serveris šioje architektūroje iš vienos pusės turi turėti galimybę perskaityti ir modifikuoti tam tikrus dokumentų laukus, o iš kitos pusės dviem verslo partneriams jis yra trečioji šalis, kuri neturėtų matyti tam tikros konfidencialios informacijos perduodamos verslo dokumentuose.

Apsaugoti duomenis reikalingas metodas, kuris užtikrina persiunčiamų dokumentų integralumą, konfidencialumą ir neišsiginamumą visą dokumento kelią. Spręsti šias problemas standartiniai apsaugos metodai nėra tinkami. Šio darbo tikslas yra sukurti metodą skirtą saugiai persiųsti verslo dokumentus nuo vieno kliento iki kito kliento per sąveikumo sistemos serverį ir įvertinti šio metodo greitaveiką. Darbe išanalizuotos verslo sąveikumo sistemos, standartiniai dokumentų apsaugos metodai, duomenų šifravimo metodai bei verslo dokumentų tipai. Suprojektuota ir realizuota programa, kurios pagalba atlikti greitaveikos tyrimai, pateiktas palyginimas su kitais apsaugos metodais. Išvadose apibendrinti eksperimentinio tyrimo rezultatai ir pateiktos rekomendacijos metodo naudojimui.

Secure business documents transfer method development and research in interoperability systems

Summary

Business documents protection while transferring in enterprise interoperability systems is one of major problems in this area. From interoperability systems nature interoperability servers are used not only to resend received business documents from sender to receiver, but also to perform a number of other document transformation functions. Therefore it is possible to distinguish several security problems in these systems. First of all it is a need to ensure documents integrity, confidentiality and non-repudiation on the way between server and sender or receiver. The problem is that server in this architecture has to have rights to read or modify some fields in the document, but on the other hand server itself is third party to two business partners, and some specified confidential information in business documents should not be revealed to it.

The approach that ensures business documents integrity, confidentiality and non-repudiation in all his way from document sender to ultimate receiver (another client) is required. Standard safe transfer methods are not suitable for solving these problems. The purpose of this research is to create a method, which securely transfers business documents from one client to another trough interoperability system server and evaluate method performance. Enterprise interoperability systems, standard safety methods for protecting documents, data encryption algorithms, and business document types are analyzed in this work. Three programs for collecting needed data for this research designed and implemented. Diagrams from collected results are presented at the end of the work. Research results are summarized and recommendations for further method use are presented in conclusions.

1. Įvadas

Naudodamiesi kompiuteriais kartais net nepastebime, kaip dažnai naudojamės žiniatinklio paslaugų (angl. Web Services) teikiamomis galimybėmis, norėdami gauti tam tikrą informaciją. Statistikos departamento atlikto tyrimo duomenimis Lietuvoje 2008 m. pradžioje 94,8 procento gamybos ir paslaugų įmonių, kuriose dirbo 10 ir daugiau darbuotojų, darbe naudojosi kompiuteriais, 92,7 procento – internetu, 2010 m. įmonių, besinaudojančių internetu, procentas padidėjo iki 96,4 [1]. Todėl sparčiai plečiantis interneto paklausai Lietuvoje ir visame pasaulyje vis dažniau minimi žiniatinklio paslaugomis grįsti sąveikumo sprendimai (angl. Web Services Interoperability) [2, 3, 4], skirti įmonės ar privatiems vartotojams. Šiomis dienomis įmonės stengiasi sąveikauti. Norėdamos gauti didesnį pelną pritraukia vartotoją, siūlydamos jam įvairius produktus ar paslaugas. Įmonėms reikia pagerinti verslo procesus, sumažinti jų ciklus ir naudojamus resursus. Tad palankiausia terpė įmonėms sąveikauti yra žiniatinklis ir jame realizuojami žiniatinklio paslaugomis grįsti sprendimai. Įmonių sąveikumo sprendimai, kurių pagalba yra keičiamasi dokumentais, turi būti perduodami saugiai, nepatekdami į trečiųjų asmenų rankas. Kadangi Interneto ir kompiuterinių tinklų naudojimas auga labai sparčiai, taigi auga ir poreikis apsaugoti duomenis, kuriuos programos siunčia tinklu. Mokslinėje literatūroje plačiai nagrinėjami būdai, kaip užkirsti kelią persiunčiamos informacijos atskleidimui. Norėdami pasiekti didesnį saugumo lygį sąveikumo programų persiunčiamiems duomenims apsaugoti jų kūrėjai naudoja duomenų šifravimą. Sąveikumo sistemos serveris koreguoja persiunčiamus dokumentus, todėl iškyla poreikis apsaugoti visus persiunčiamus ar tam tikrą dalį persiunčiamų dokumentų nuo jų kompromitavimo. Kadangi sąveikumo sistemos serveris privalo matyti tam tikrus duomenis ar juos pakeisti prieš persiunčiant galutiniam gavėjui įprastos dokumentų apsaugos priemonės šiuo atveju nėra tinkamos. Tad šiame darbe bus apžvelgiami galimi sprendimai ir sukurtas modelis saugiai persiųsti duomenis sąveikumo sistemoje iš vieno vartotojo kitam, kuris išspręstų persiunčiamų duomenų konfidencialumo, vientisumo ir neišsiginamumo problemas.

Analitinėje dalyje, apžvelgiant duomenų apsaugos sprendimus, labiausiai bus atsižvelgiama į sprendimo tinkamumą šiems kriterijams: dokumentų apsaugą kokybiniu požiūriu, realizaciją realiose sąveikumo sistemose, sprendimo įgyvendinimo kainą bei pritaikomumą mažoms ir vidutinėms įmonėms.

Šio tiriamojo darbo tikslas yra sukurti modelį, kuris leistų saugiai perduoti verslo dokumentus sąveikumo sistemose.

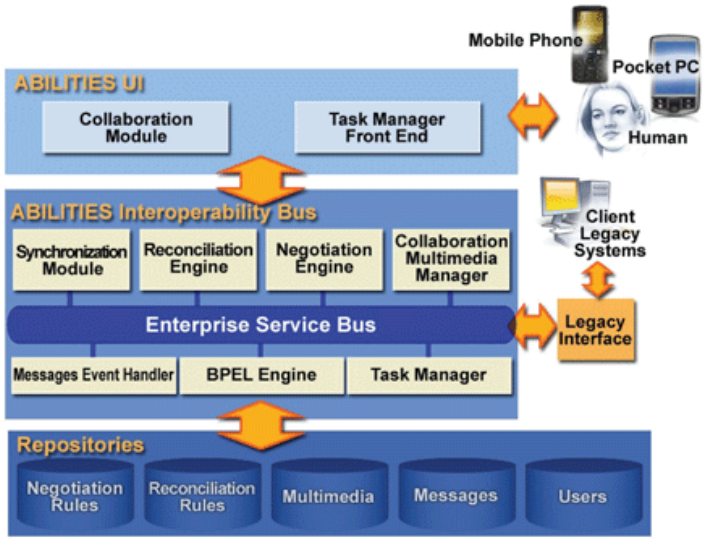
2. Analitinė dalis

2.1. Sąveikumo sprendimai

Kadangi pastaruoju metu vis daugiau įmonių persikelia į internetinę terpę daugėja ir sąveikumo sprendimų, skirtų žiniatinklio paslaugoms. Keletas sąveikumo sistemų: *Abilities* [4, 6, 14], *Athena* [7, 8, 14, 15], *Coin* [10], *Ecolead* [9, 14], *eNVISION*, *Fusion* [11, 14], *Genesis* [12, 13, 14, 15], *ImportNET* [5], *Interop* [15], *iSurf* [16], *Panda* [14], *Synergy*, *Visp* [17] ir t.t. Trumpai apžvelgiamos kelios sąveikumo sistemos.

2.1.1. Abilities

Abilities sprendimo pagrindas yra sąveikumo magistralė. Ši magistralė yra atsakinga už skirtingų modulių, sudarančių tinklą, integraciją ir sąveikumą. *Abilities* sprendimą ir jo paslaugas siūloma naudoti mažoms ir vidutinėms įmonėms, sujungiant liktines sistemas (*angl. legacy systems*) vieną su kita per *Abilities* įmonės paslaugų magistralę (*angl. Enterprise service bus, ESB*). Šio sprendimo architektūra pateikta 1 pav.



1 pav. Abilities projekto architektūra.[4]

Abilities per savo sąveikumo magistralę teikia tokias paslaugas, kaip bendradarbiavimo valdymo nustatymas (*angl. Collaboration configuration*), taisyklių derybų valdymas (*angl. negotiation rules handling*), procesų kūrimas (*angl. process design*), verslo dokumentų apibūdinimas ir koregavimas (*angl. business document definition and adjustment*), multimedijos valdymas (*angl. multimedia content management*) bei prieigos valdymas (*angl. roles management*) [6]. Kaip pavyzdys

pateikiamos toks scenarijus: pirkėjas turėtų pradėti atlikti transakciją savo vidinėje informacinėje sistemoje, kuri sugeneruoja UBL žinutę ir pasiunčia (naudojant liktinės sistemos sąsaja) per *Abilities* sąveikumo sprendimą pardavėjui. Pardavėjo vidinė informacinė sistema gauna UBL žinutę ir transformuoja ją pagal pardavėjo sistemos vartotojų ar pačios informacinės sistemos poreikius į tinkamą formatą. Šios problemos sprendimas gali būti automatizuotas arba reikalauti asmens įsikišimo. Sąveikumo sprendimą verslo dokumentų formatų lygyje *Abilities* teikia ontologijos pagrindu (*angl. Ontology-based*), pagal iš anksto automatiškai nustatyto sprendimo būdus bei numatytas taisykles (*angl. reconciliation rules*). Šios taisyklės leidžia apibrėžti ir nustatyti verslo dokumento tipus, specializuoti, kas bendra, bekuriant verslo dokumentus, kurie tuo tarpu yra pritaikomi konkrečiai rinkai ar net konkrečiai įmonei rinkoje. Šiais dokumentais taip pat gali būti keičiamasi tarp verslo partnerių pagal nustatytas ir patvirtintas bendradarbiavimo taisykles ar pagal bendrus verslo nuostatus.

Abilities pasiūlytas sprendimas susijęs su proceso ir bendravimo problemomis, įdiegiant pritaikytą įmonės paslaugų magistralę (*sut. ESB*), paremtą UBL dokumentais. Šis sprendimas gali būti įdiegtas specializuotai rinkai. Pagal architektūrą jis labiau tinkamas mažoms ir vidutinėms įmonėms nei tradiciniams integruotiems verslo sprendimams. Tradiciniai integruoti verslo modeliai reikalauja kiekvieną verslo partnerį naudoti ta patį duomenų tipą ir protokolus, tuo tarpu šis sprendimas nereikalauja vienodų verslo dokumentų tipų ar protokolų. Naudojantis juo dokumentai yra transformuojami į UBL formatą ir persiunčiami internetu kitam verslo partneriui.

Apibendrinant šį sprendimą galima teigti, kad *Abilities* teikiamas paslaugas galima pristatyti ir vertinti keliais požiūriais:

1. *Abilities* paslaugos tarporganizaciniam bendravimui. Šiuo požiūriu *Abilities* galima įsivaizduoti kaip magistralę, kuri jungia įmones ir per kurią galima siųsti dokumentus, pranešimus (klausimus-atsakymus), ir jais palaikyti prekybinius santykius: pradedant derybomis dėl užsakymų ir baigiant atsiskaitymais. Bendradarbiavimo ir derybų procesą palaiko *Abilities*.
2. *Abilities* paslaugos įmonės vidiniam vadybiniam darbui sutvarkyti pagal įmonės vidines tvarkos taisykles. Pageidaujamą tvarką, įskaitant dokumentų ir kitų pranešimų kelią įmonės viduje, bei asmenis, kurie gali prieiti prie iš magistralės gaunamų dokumentų ir pranešimų, palaiko *Abilities* magistralė.

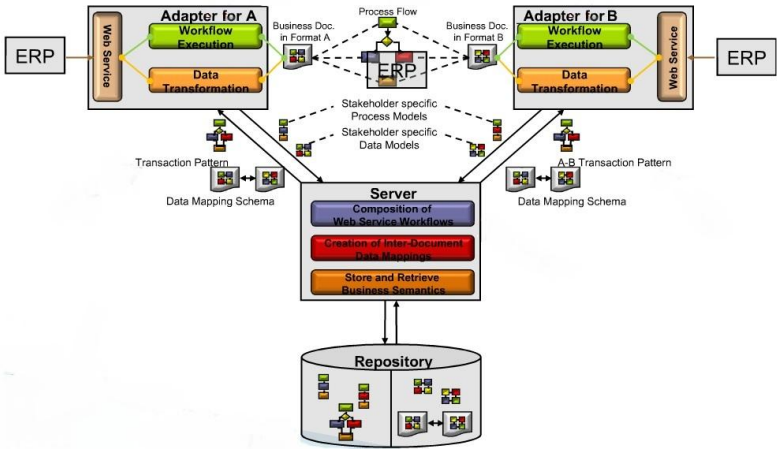
Tačiau nors ir šis sprendimas gali teikti didelę naudą smulkioms ir vidutinėms įmonės efektyvaus bendradarbiavimo tarp partnerių požiūriu, paskirstant resursus įmonėje, šiame

sprendime nėra realizuotas siunčiamų žinučių (perduodamų duomenų) saugos mechanizmas bei apsauga nuo neteisėto turinio paviešinimo.

2.1.2. Genesis

Projekto plėtotojai, atsižvelgę į tai, jog Europos sąjungos šalyse pradėję plisti žiniatinklio paslaugomis grįsti sprendimai tarp mažų ir itin mažų įmonių plėtoja būtent šioms įmonėms skirtą sprendimą, kurio pagalba šios įmonės galės sąveikauti tarpusavyje. Pagrindinis *Genesis* projekto tikslas yra reikiamų metodologijų, infrastruktūros ir programinių komponentų tyrimas, plėtojimas ir programų bandymas. Visa tai leis smulkioms ir vidutinėms Europos įmonėms keistis verslo dokumentais per internetą, sujungiant tarpusavyje transakcijas atliekančias programines įrangas ir sistemas su bendradarbiaujančiomis įmonėmis ar valstybės institucijomis.

Į standartinį *Genesis* sprendimą yra įtrauktos šios funkcijos: galimybė įmonėms keistis verslo dokumentais per egzistuojančią ERP[2] programą arba per internetinę naršyklę (jei nėra sukurtos tam specialiai skirtos programos). Dokumentai bus talpinami *Genesis* serveryje ir atsiradus galimybei persiųsti galutiniam gavėjui, užtikrinant saugą ir duomenų konfidencialumą. Gavėjui tapus pasiekiamam jam bus automatiškai nusiųstas dokumentas. Gautas dokumentas gavėjo sistemos pagalba bus integruotas į sistemą. *Genesis* sprendimo architektūra pateikta 2 pav. Šiame paveikslėlyje pavaizduoti pagrindiniai sistemos architektūriniai komponentai, adapteris padedantis integruotis į komercines ERP sistemas, ir žinučių siuntimo serveris, ne tik padedantis keistis verslo dokumentais tarp ERP adapterio ir kliento, bet ir atliekantis numatytus veiksmus su perduodamais dokumentais.



2 pav. Genesis sprendimo architektūra.[13]

Projekto *Genesis* tikslai yra:

- Legalus ir teisinis ES karkasas, naujos ES narės ir susijusių valstybių analizė.
- Pagrindinių verslo transakcijų mažoms ir vidutinėms įmonėms modeliavimas.
- Modeliuoti ir plėtoti protokolus ir duomenų formatus skirtus įmonių sąveikumo programoms
- Plėtoti centrinę infrastruktūrą ir paskirstytus susijungimo komponentus įmonių programoms.

Šis projektas yra koordinuojamas *Singular software, SA* ir plėtojamas 15 dalyvių konsorciumo iš Austrijos, Kipro, Čekijos, Vokietijos, Graikijos, Italijos, Lietuvos, Rumunijos, Lenkijos, Turkijos, Anglijos. Šiuo projektu naudojasi daugiau nei 20 000 vartotojų.

Nors ir projekto aprašyme yra pateikta, kad persiunčiamiems duomenims bus naudojamas šifravimas, tačiau nepaminėta, kokie apsaugos veiksmai bus naudojami (ar išvis bus naudojami) tuo metu, kai verslo dokumentai lauks sąveikumo sistemos serveryje kol galės būti persiųsti galutiniam gavėjui. Be to nėra paminėta apie duomenų apsaugą, siunčiant juos iki sąveikumo sistemos. Todėl galima daryti išvadą, kad ši sistema taip pat nėra pakankamai apsaugota nuo duomenų vagystės ar informacijos kompromitavimo.

2.1.2. Fusion

Mažos ir vidutinio dydžio įmonės bendradarbiavo su tarptautiniais partneriais iš visos Europos, kad sukurtų holistinius įmonių programų integravimo (*angl. Enterprise application integration, EAI*) sprendimus tam, kad pagerinti e-verslo procesus. Kuriant šią sąveikumo sistemą buvo susidurta su keletu tarptautinių barjerų. Vienas iš jų buvo tai, jog sąveikumo ir integravimo pastangos yra labiau orientuotos į persiunčiamus duomenis nei į atliekamus procesus. SAP sukūrė *FUSION* (Business Process Fusion Based on Semantically-enabled Service-oriented Business Applications) projektą, tam kad pagerintų verslo procesus bendradarbiaujant, išorinį susijungimą tarp įmonių. Taip pat plėtojant sistemą daug dėmesio buvo skirta naujoviškam technologijų orientavimui į verslo taikomąsias programas.

FUSION projekto mokslinių tyrimų veikla tikimasi nustatyti kelių Europos pramonės šakų bendrus verslo poreikius, tokius kaip el. bendradarbiavimą, internetinius sandorius, integracija tiekimo ar vertės grandinėse ir „front-end“ ir „back-office“ programų integracijose. Bet kurio iš aukščiau paminėtų reikalavimų patobulinimas gali drastiškai paveikti įmonių, organizacijų produktyvumą ar konkurencingumą. Ši galimybė atsiranda integruojant informacines sistemas,

pasinaudojant žiniatinklio paslaugomis. Bendradarbiavimas ypač bus juntamas B2B ir B2C sektoriuose, kur pagerės reagavimo laikas ir sumažės sąnaudos.

Pagrindiniai *Fusion* projekto aspektai:

- Bandomoji programa padengia sąveikumo aspektus, atitikdama daugelio Europoje veikiančių mažų ir vidutinių įmonių lūkesčius.
- Automatizuojama: pilnai, dalinai ar ranku darbu vykdomos operacijos
- Sudėtingumas: sudėtingi procesai, įtraukiantys keletą sistemų užuot dirbus su viena sistema.

Fusion panaudos scenarijai įtraukia tokias veiklas kaip išteklių (žaliavų, ar paslaugų) papildymo automatizavimas pagal poreikius, komponuojamumas su CRM ir HR[3] sistemomis, studentų mainų programų procesais.

2.1.3. COIN

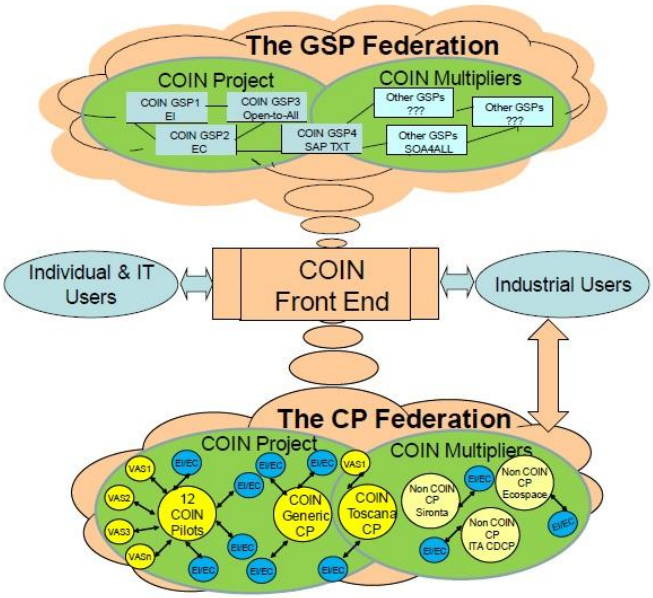
COIN yra dviejų angliškų žodžių darinys *COllaboration* (*bendradarbiavimas*) ir *INterporability* (*sąveikumas*). Santrumpa *COIN* pavadintas tarptautinis projektas, finansuojamas Europos Komisijos Bendrojoje 7-je programoje (sut. FP7).

COIN platformą sudaro e. paslaugos, kurias teikia įvairūs paslaugų tiekėjai, ir šių paslaugų kompozicijos, sukurtos pagal bendradarbiaujančių įmonių poreikius. E. paslauga gali padėti bendradarbiaujančioms įmonėms sukurti ir aptarti naują dokumentą (užsakymą, sutartį, specifikaciją), suteikti galimybę pravesti derybų seansą ir priimti bendrą sprendimą, automatizuoti dokumentų persiuntimą iš vienos sistemos į kitą, suderinant duomenų formatus bei atlikti daugelį kitų reikalingų funkcijų. E. paslaugas gali teikti įvairios kompanijos už tam tikrą mokestį, tačiau paslaugos gali būti ir nemokamos. E. paslaugos komponuojamos pagal pageidavimą, ir tokios kompozicijos tampa įmonių bendradarbiavimo ir sąveikavimo sistemomis, kurios toliau valdo veiklos procesus ir jiems „diriguoja“ pagal sumanytus scenarijus.[18]

Pagrindinis *COIN* projekto tikslas yra sukurti ir plėtoti galinčią plėstis ir prisitaikyti paslaugų sistemą. *COIN* sistema yra paremta federacinės platformos pagrindu ir yra prieinama vartotojams per internetinę sąsaja, leidžiančią EI/EC paslaugoms būti ieškomoms, atrastoms, integruotoms į įmonės informacinę sistemą, nustatytoms pagal poreikius ir vykdyti sąveikumo bei bendradarbiavimo uždavinius. *COIN* sistema yra sudaryta iš trijų pagrindinių architektūrinių komponentų (pavaizduota 3 pav.).

Pagrindiniai *COIN* sistemos komponentai:

- *COIN* Generic Service Platform (GSP) atviro kodo programa, pritaikyta įmonių sąveikumo ir bendradarbiavimo domenams. Šioje platformoje yra realizuotos papildomos galimybės: pasitikėjimas kita informacine sistema (*angl. trust*), apsauga *(angl. security*), pasiskirstymas (*angl. distribution*).
- Daugybė EI/EC paslaugų, kurios gali teigti ir kaupti informaciją, palaikyti verslo sąveikumą taip pat vartotojų tarpusavio bendravimą tokiais verslo klausimais, kaip: produkto plėtojimas, gamybos planavimas, projektų valdymas.
- *COIN* bendradarbiavimo platforma (*angl. collaboration platform*), kuri yra atviro kodo internetinis portalas. Bendradarbiavimo platforma savyje talpina sąveikaujančių servisų socialinį tinklą, žinių išteklius (*angl. business assets*) ir verslo procesų valdymą unikalioje, integruotoje, aplinkoje, pritaikytoje pagal specifinius poreikius, tarp daugelio tarpusavyje bendradarbiaujančių įmonių.



3 pav. COIN sprendimo koncepcinė schema.

COIN elementai gali būti apibūdinti taip pat, kaip debesų perspektyvos. Nagrinėjant pateiktąjį paveikslėlį aukščiausiai pateiktas debesis teikia išmaniųjų programų paslaugas su bendraisiais

EI/EC reikmenimis bei reikmenis, skirtais paslaugų plėtojimui, registracijai, viešinimui, paieškai ir atradimui, nustatymui ir vykdymui. Žemiausias debesis palaiko procesus tokius kaip: produkto gyvavimo ciklas, bendradarbiavimo fazės, liktinių sistemų integracija, tiekimo grandinės, bendradarbiavimo tinklai su sudėtingesniu EI/EC paslaugų pritaikymu. Tarp šių dviejų debesų yra programinė įranga ar interneto portalas, leidžiantis vartotojams bendrauti su *COIN* sistema.

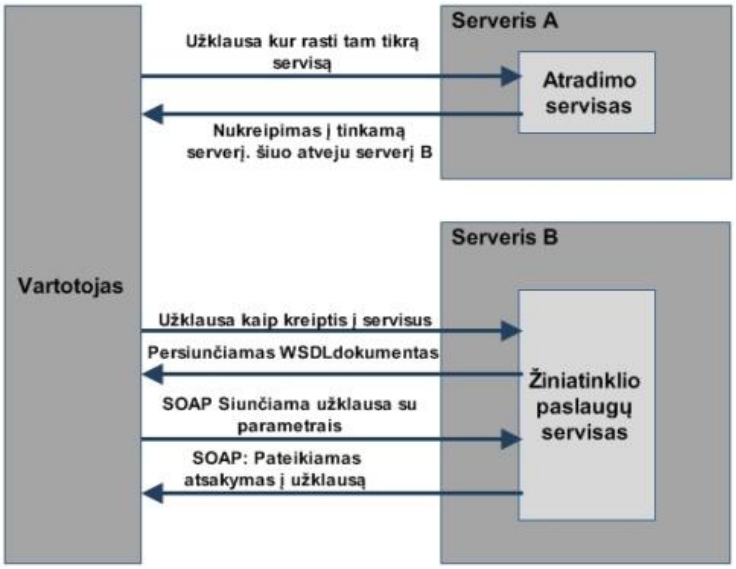
Apžvelgus keletą sąveikumo sistemų pavyzdžių pastebimos problemos kylančios įmonių sąveikumo sistemose:

- Sąveikumo sistemose sąveikaudamos mažos ir itin mažos įmonės tarpusavyje keičiasi konfidencialiais verslo dokumentais. Persiunčiami dokumentai turi būti apsaugoti nuo siuntėjo iki galutinio gavėjo.
- Pagal sąveikumo sistemos specifiškumą serveris keičia, modifikuoja persiunčiamą verslo dokumentą pagal numatytus verslo procesus. Tačiau koreguojamuose dokumentuose gali būti konfidencialių duomenų kurių, sąveikumo serveris neturėtų matyti.
- Dauguma apžvelgtų sąveikumo sistemų yra plėtojimo stadijose, todėl jose verslo dokumentų apsaugos problemos nėra sprendžiamos.
- Iškilusios dokumentų saugos problemos negali būti sprendžiamos tradiciniais apsaugos metodais, todėl reikalingas metodas užtikrinantis duomenų konfidencialumą ir neišsiginamumą.

2.2. Žiniatinklio paslaugos

Žiniatinklio paslaugos yra programiniai komponentai, kurie komunikacijai naudoja standartais grįstas žiniatinklio paslaugas: tai HTTP[3] ir XML. Žiniatinklio paslaugos yra pritaikytos įvairių programų naudojimui, kurios gali būti nuo kelias operacijas atliekančios žiniatinklio paslaugomis veikiančios programos iki daug sudėtingesnių, tokių kaip CRM[4] ar ERP sistemų. Nuo tada kai žiniatinklio paslaugos naudoja atvirus protokolus, tokius kaip HTTP, XML, o taip pat SOAP, WSDL žiniatinklio paslaugos tampa nepriklausančios nuo platformos ar programavimo kalbos [2, 19]. Žiniatinklio paslaugos labai priklauso nuo XML ir nuo dar trijų pagrindinių standartų: WSDL, SOAP, UDDI. Prieš naudojant žiniatinklio paslaugas, jų kūrėjai prieš tai apibrėžia teikiamas paslaugas WSDL dokumente, kuris nusako serviso adresą internete ir kokias paslaugas servisas gali atlikti. Informacija apie paslaugas gali būti įrašyta į UDDI registrą, kuris suteikia galimybę žiniatinklio paslaugų vartotojui ieškoti ir surasti paslaugas, kurios jiems reikalingos. Šis žingsnis

yra neprivalomas, bet yra naudingas tik tada, kai įmonė nori, kad jos žiniatinklio paslaugos galėtų būti pasiekiamos vidiniam ir/arba išoriniam vartotojui. Pagal informaciją UDDI registre, žiniatinklio paslaugų gamintojas naudoja instrukcijas WSDL faile konstruoti SOAP žinutėms tam, kad būtu įmanoma apsikeisti informacija su servisu per HTTP protokolą. Detaliau apie šias technologijas aprašyta žemiau. Prieš tiriant saugumo aspektus reikia išsiaiškinti, kaip veikia žiniatinklio paslaugomis grįsti sprendimai. Žiniatinklio paslaugų veikimo principas pavaizduotas 4 pav.



4 pav. Žiniatinklio paslaugų veikimo schema

1. Jeigu vartotojas nežino, į kokius viešus žiniatinklio paslaugų servisus kreiptis, kad gautų jį dominančią informaciją, tuomet siunčia užklausą viešai žinomiems atradimo servisams.
2. Atradimo servisai atsako, koks servisas galėtų suteikti tas paslaugas, kurios yra reikalingos vartotojui.
3. Nors programa žino, kur ieškoti žiniatinklio paslaugų serviso, tačiau nežino, kaip į jį kreiptis, todėl siunčia užklausą, kad žiniatinklio paslaugų servisas „apibūdintų“ save.
4. Žiniatinklio paslaugos persiunčia vartotojui WSDL failą, kuriame yra surašyti kokie gali būti kreipiniai į paslaugą, kokius duomenų formatus supranta servisas, kokiais žinučių tipais servisas bendraus su klientu ir t.t.
5. Vartotojas serveriui siunčia SOAP formato užklausą su parametrais, norėdamas gauti informaciją.

6. Žiniatinklio paslaugos atsako vartotojui pagal jo pateiktą užklausą arba žiniatinklio paslaugų servisas gali atsakyti su klaidos pranešimu (pasirinktinai).

Verta paminėti, kad WSDL dokumento gavimo procedūra gali būti nevykdoma, jeigu iš anksto yra aprašyta žiniatinklio paslauga vartotojo naudojamoje programoje. WSDL dokumentas programoje keičiamas per programos atnaujinimus. Programos, kurios norės pasiekti tam tikrus resursus, turės prisijungti prie žiniatinklio paslaugų serverio ir nusiųsti serviso užklausą, reikalaujančią tam tikros informacijos. Serveris tada gražins programai serviso atsakymą pateiktai užklausai.

Kaip bebūtų yra viena labai svarbi charakteristika, kuri išskiria žiniatinklio paslaugas. Nors ir tokios technologijos kaip *Corba*[7] ar EJB yra greitesnės ir priklausomos, t.y. klientas ir serveris yra labai priklausomi vienas nuo kito, žiniatinklio paslaugos yra labiau skirtos savarankiškoms sistemoms, kur klientas gali neturėti jokios išankstinės informacijos apie žiniatinklio paslaugas iki to laiko, kol jam jų prireikia. Sujungtos sistemos yra idealus atvejis vidiniam naudojimui įmonėse, bet labai retai naudojamas internete. Žiniatinklio paslaugas geriau pritaikyti veikti plačiu mastu. Be to kodo panaudojimas iš naujo suteikia žiniatinklio paslaugom sąveikumo ir lankstumo t.y. vienas servisas gali būti naudojamas keleto vartotojų, atliekant tam tikras verslo operacijas užuot kūrus kiekvienam klientui atskirus servisus. Sprendžiant iš visko, kas buvo paminėta, žiniatinklio paslaugų sprendimai yra labai palankūs įgyvendinant sąveikumo sprendimus tarp įmonių.

2.3. Žiniatinklio paslaugų dokumentų standartai

2.3.1. XML (angl. Extensible Markup Language)

XML yra standartas, sukurtas W3C (angl. WWW Consortium) [20]. Standartas yra lankstus, nepriklausantis nuo platformos, suprantamas žmogaus (galimybė pačiam autoriui keisti dokumento struktūrą). Šis paprastumas ir gebėjimas sąveikauti su programomis leido XML formatui paplisti plačiu mastu ir būti naudojamam kaip standartui, keičiantis informacija tarp skirtingų sistemų ir programų, įskaitant ir žiniatinklio paslaugas. XML dokumente duomenys yra apsupti tekstinių žymų (pvz.: <Vardas>Renatas Vencevičius</Vardas>), kurios teikia informaciją apie duomenis ir yra surūšiuotos hierarchijos tvarka (pvz.: <Informacija> <Miestas>23</Miestas> <Adresas>Krėves pr. 46 </Adresas> </Informacija>). XML formuoja pagrindus visoms naujoms žiniatinklio paslaugoms, kurios naudoja technologiją XML, pagrindu nusakančią jų sąsajas, siunčiamų žinučių šifravimą. WSDL, SOAP ir UDDI yra paremti XML formatu tam, kad visos šiuolaikinės mašinos galėtų jas interpretuoti. Daugelis XML pagrindu sudarytų standartų buvo tobulinami, jų tarpe

ebXML, EDI, RosettaNet, OBI, CBL, kurie yra e-verslo standartai, naudojami duomenų persiuntimui.

2.3.2. SOAP (angl. Simple Object Access Protocol)

SOAP protokolas naudoja XML kaip žinučių, nešančių informaciją, formatą ir programinio sluoksnio protokolus (dažniausiai RPC ir HTTP) žinučių persiuntimui ir deryboms. *SOAP* yra XML pagrindu paremtas protokolas, sukurtas W3C, skirtas keistis informacija per HTTP. Šis protokolas teikia paprastą standartizuotą metodą siųsti XML žinutes tarp programų. Žiniatinklio paslaugų servisai naudoja *SOAP* protokolą siųsti žinutes tarp serviso ir jo kliento. Dėl to, kad HTTP yra palaikomas visų žiniatinklio serverių ir naršyklių, *SOAP* žinutės gali būti siunčiamos nepriklausomai nuo veikiančios platformos ar programavimo kalbos. Ši savybė teikia žiniatinklio paslaugoms sąveikumo charakteristikas.

SOAP žinutės yra XML dokumentai, savyje turintys visus ar keletą iš šių elementų:

- Vokas (envelope) – nusakantis, kad XML dokumentas yra *SOAP* žinutė.
- Antraštė (neprivalomas) (header) – turi žinutės informaciją, tokią kaip žinutės išsiuntimo data, autentifikuota informacija ir t.t.
- Aprašas (body) – XML dokumento tekstas.
- Klaida (neprivalomas) (fault) – persiunčiama informacija apie vartotojo arba serverio klaidą, perduodant *SOAP* žinutę.

Duomenys yra siunčiami tarp vartotojo ir žiniatinklio paslaugos, naudojant „request“ ir „response“ *SOAP* žinutes, kurių formatas yra apibrėžtas WSDL dokumente. Kadangi vartotojas ir serveris laikosi WSDL susitarimo, kai kuria *SOAP* žinutes, todėl yra garantuojama, kad žinutės tarpusavyje bus suderinamos.

SOAP žinučių pavyzdžiai:

Užklausa

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
<soap:Body xmlns:m="http://www.example.org/stock">
  <m:GetStockPrice>
    <m:StockName>IBM</m:StockName>
  </m:GetStockPrice>
</soap:Body>
</soap:Envelope>
```

Atsakymas

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
<soap:Body xmlns:m="http://www.example.org/stock">
  <m:GetStockPriceResponse>
    <m:Price>34.5</m:Price>
  </m:GetStockPriceResponse>
</soap:Body>
</soap:Envelope>
```

Privalumai:

- *SOAP* per HTTP leidžia lengvesnį bendravimą per *proxy* serverius ir užkardas.
- *SOAP* protokolas yra ganėtinai įvairiapusiškas, nes leidžia naudoti skirtingus transporto protokolus. Nors standartiškai naudojamas HTTP, tačiau galima naudoti ir kitus, kaip pavyzdžiui SMTP.
- *SOAP* yra nepriklausomas nuo platformos.
- *SOAP* yra nepriklausomas nuo programavimo kalbos.

Trūkumai:

- Kadangi XML formatas yra daugiakalbis *SOAP* protokolas laikomas lėtesnis nei ankstesnės technologijos, tokios kaip *Corba, RMI, EJB*. Šis trūkumas išsisprendžia, kai yra siunčiamos nedidelės apimties žinutės, bet pagerinti efektyvumą didesniems dvejetainiams XML dokumentams buvo sukurtas žinučių persiuntimo optimizavimo mechanizmas „Message Transmission Optimization Mechanism“.

2.3.3. WSDL (angl. Web Service Definition Language)

WSDL yra XML pagrindu W3C sukurtas formatas, apibūdinantis žiniatinklio paslaugas. Vartotojas, norėdamas naudotis žiniatinklio paslaugomis, gali interpretuoti turimą *WSDL* failą ir taip sužinoti žiniatinklio paslaugų adresatą bei galimus metodus. Šiuo atveju *WSDL* failas suveikia kaip inicijuojama žiniatinklio paslaugos sąsaja, teikianti vartotojui visą informaciją, kurios jam reikia norint bendrauti su žiniatinklio serveriu standartiniais būdais. Per WSDL vartotojas gali sužinoti, koks yra žiniatinklio paslaugos adresas, kokias metodus paslauga teikia, protokolus, kuriuos supranta paslauga, ir formatus, kuriais galima siųsti žinutes žiniatinklio paslaugomis grįstam servisui.

WSDL failas yra XML dokumentas, apibūdinantis žiniatinklio paslaugų šiuos pagrindinius elementus:

- Jungties tipas (Port type) – sugrupuoja ir apibūdina operacijas, atliekamas serviso per nustatytą sąsają.
- Žinutė (Message) – nustatomi vardai ir žinutės formatai palaikomų žiniatinklio serviso.
- Tipai (Types) – nustatomi duomenų tipai, naudojami serviso, siųsti žinutes tarp vartotojo ir serverio.
- Surišimas (Binding) – nustatomi bendravimo protokolai, palaikomi serviso teikiamų operacijų.
- Servisas (Service) – nurodo URL adresą, kuriuo galima pasiekti servisą.

WSDL dokumento struktūra:

```
<definitions>
<types>
   Apibrėžiami tipai žinutės formatui........
</types>
<message>
   Apibrėžiama siunčiama žinutė....
</message>
<portType>
      <operation>
           Apibrėžiamos operacijos.......
      </operation>
</portType>
<binding>
   Apibrėžiamas žinučių surišimas....
</binding>
<service>
   Nusakomas kur pasiekiamas servisas....
</service>
</definitions>
```

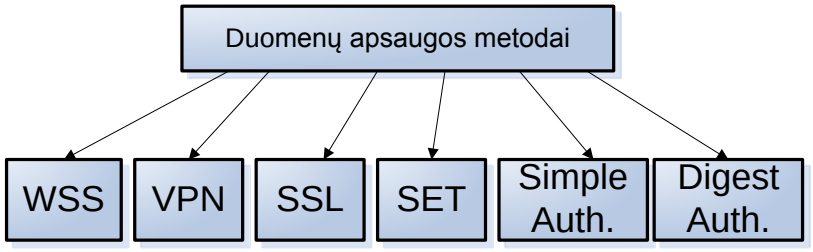
2.3.4. *UDDI* (angl. Universal Description, Discovery, and Integration)

Atradimo lygmenyje naudojama *UDDI* technologija. *UDDI* specifikacija aprašo registrą, skirtą kaupti informacijai apie servisus bei juos teikiančias organizacijas. Toks registras yra būtinas, norint pasinaudoti vienu iš SOA[6] privalumų, daugkartiniu komponentų panaudojimu. Programinės įrangos kūrėjams reikėjo vietos, kur būtų saugomos nuorodos į servisus bei informacija, kaip su jais elgtis. Tokia informacija suteikia galimybę dinamiškai surasti ir iškviesti reikiamą servisą tiek sistemos kūrimo, tiek jos veikimo metu. Pradžioje buvo planuojama, kad internete bus globalus viešas registras, kuriame visi galės talpinti informaciją apie save ir savo teikiamas paslaugas. Globalaus registro idėją palaikė tokios kompanijos kaip *IBM*, *Microsoft*, *SAP* ir kitos. Tačiau ji nepasiteisino. Vis dėlto, privatūs registrai, veikiantys kompanijų vidiniuose tinkluose, paplito gana plačiai. 511 *UDDI* registre gali būti saugoma informacija ne vien tik apie verslo organizacijas, bet ir apie bet ką. *UDDI* įrašo struktūrą sudaro trys dalys. Tai baltieji, geltonieji ir žalieji puslapiai. Baltieji puslapiai – tai bendro pobūdžio informacija apie organizaciją (organizacijos pavadinimas,

adresas, apibūdinimas). Geltonieji puslapiai skirti konkretesniam verslo aprašymui. Tai gali būti teikiamų paslaugų sąrašas. Taip pat čia saugomi verslo klasifikatoriai. Klasifikatorius identifikuoja verslą pagal tam tikrą klasifikavimo sistemą. Tai gali būti geografinis indeksas ar kita standartinė sistema. Klasifikavimo sistemą taip pat gali aprašyti ir pats vartotojas. Žaliuosiuose puslapiuose saugoma techninė informacija apie žiniatinklio paslaugas. Tai nuoroda į patį servisą, jo WSDL aprašą ir panašiai [38].

2.4. Žiniatinklio paslaugų naudojami apsaugos metodai

Norint tinkamai apsaugoti dokumentus, siunčiamus verslo sąveikumo sistemoje, būtina naudoti duomenų apsaugos metodus, užtikrinančius persiunčiamų dokumentų konfidencialumą bei integralumą ar neišsiginamumą. 5 pav. pateikiami dokumentų apsaugos metodai, kuriais galima apsaugoti persiunčiamus verslo dokumentus. Metodai aptariami šiame skyriuje.



5 pav. Keletas apsaugos metodų naudojamų duomenų apsaugai ir vartotojo autentifikacijai

Apžvelgus apsaugos metodus yra pabrėžiami jų naudojimo privalumai, trūkumai ir panaudojimas realiose sąveikumo sistemose.

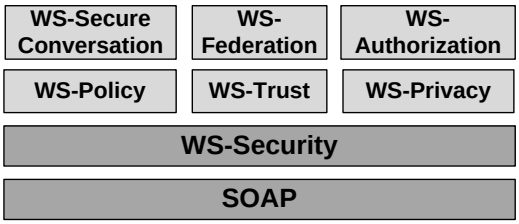
2.4.1. Žiniatinklio paslaugų saugumo architektūra

Žiniatinklio paslaugų saugos pagrindinis tikslas yra apsaugoti persiunčiamą *SOAP* žinutę suteikiant užtinkrinti:

- Autentifikaciją (angl. authentication)
- Autorizaciją – priėjimo kontrolę (angl. authorization)
- Integralumą (angl. integrity)
- Konfidencialumą (angl. Confidentiality)
- Neišsiginamumą (angl. Non-repudiation)
- Raktų / slaptažodžių apsikeitimą saugiose terpėse (angl. credential mediation)

- Paslaugų galimybių apibrėžimą (angl. Service capabilities)

Žiniatinklio paslaugų saugos standartai, padedantys pasiekti saugos tikslus, pateikti 6 pav.



6 pav. SOAP žinutes saugos protokolai

Standartų aprašai:

- **WS-Security** – tai *SOAP* žinučių apsauga, taikant integralumą, konfidencialumą, ir kiekvienos žinutės autentifikaciją. WS-Security yra išplečiamas ir lankstus standartas, aprėpiantis keletą apsaugos savybių: pasitikėjimą domenu, parašo formatus ir šifravimo technologijas. WS-Security apibrėžia *SOAP* antraštės elementus, susijusius su apsaugotais duomenimis. Jei yra naudojamas XML parašas, šioje antraštėje gali būti pateikiama informacija, nustatyta pagal XML parašą, kuri apibūdina, kaip buvo pasirašytas pranešimas, raktas, kuris buvo naudojamas pasirašant, ir parašo reikšmė. Žinutės antraštėje gali būti pateikta Informacija apie siuntėją, kaip žinutė buvo pasirašyta, ar kaip ji buvo užšifruota. WS-Security nenurodo parašo ar kodavimo formato, vietoj to, ji nurodo, kaip užšifruota informacija pagal specifikaciją yra išdėstyta *SOAP* žinutėje. WS-Security pateikia išbaigtą sprendimą žiniatinklio paslaugoms, laikant visą apsaugos informaciją vienoje *SOAP* žinutės dalyje.
- **WS-Trust**. Tai vienas iš WS-Security standarto plėtinių. WS-Trust padeda spręsti pasitikėjimo klausimus su kitu sąveikumo sistemos dalyviu. Net jei WS-Security apsaugotos *SOAP* žinutės saugumo aspektu formatas yra priimtinas gavėjo sąveikumo sintaksės lygmenyje, tai negarantuoja, kad gavėjas pasitikės šia žyma. WS-Trust specifikacija apibrėžia standartinį pasitikėjimo modelį, naudojant jungtis su jau egzistuojančiais pasitikėjimo modeliais tam, kad naudotojai apsikeistomis saugos žymomis galėtų pagrįstai pasitikėti. Taip pat WS-Trust dėl pasitikėjimo teikia komunikacinius procesus įtraukiant trečiąją šalį.
- **WS-Authorization** tai standartas, padedantis valdyti informaciją naudojamą autorizacijai ir prieigos politikai. Šioje politikoje yra aprašyta kaip identifikacijos tvirtinimai yra specifikuoti tarp apsaugos žymų ir kaip šie tvirtinimai bus priimti galutinio taško. Paprastai šis standartas apibūdina kaip valdyti autorizuotus duomenis ir autorizacijos politiką.
- **WS-SecureConversation –** taip pat vadinama Web Services Secure Conversation Language, yra specifikacija, kuri aprašo saugų bendravimą tarp žiniatinklio paslaugų, naudojant sesijos

raktus. WS-SecureConversation dirba nustatant ir kuriant šifravimo raktus, kurie bus naudojami tarp visų dalyvaujančių pusių. Pagrindinis protokolo tikslas yra nustatyti šifravimo algoritmą, ir sugeneruoti raktą, kuris bus pridėtas į *SOAP* žinutę (šis raktas taip pat gali būti panaudotas visos *SOAP* žinutės šifravimui). Kai gavėjas gauna žinutę, jis ją iššifruoja ir gauna sesijos raktą, kurį vėliau galima naudoti, siekiant palengvinti saugų ryšį likusiai sesijos daliai.

- **WS-Policy –** apibūdina bendro naudojimo XML pagrindu paremtą modelį ir jo sintaksę, kuri gali būti panaudota apibūdinti ir bendrauti su nustatyta saugumo politika, kurios priklauso, bet kuriam žiniatinklio paslaugomis paremtam servisui. Kitaip tariant, WS-Policy išreiškia galimybes ir apribojimus, kurie yra taikomi tam tikroms žiniatinklio paslaugoms, kurioms jie priklauso. Dar tiksliau, WS-Policy teikia lanksčią, išplečiamą gramatiką dėl teiginių apie galimybes, reikalavimus ir bendrąsias savybes konkrečioms XML žiniatinklio paslaugų sistemoms. WS-Policy leidžia tam tikras ypatybes suformuluoti kaip politiką, kuri gali būti paprastai aprašoma teiginiais, bet taip pat gali būti ir sudėtingesnės deklaracijos. Atsižvelgiant į tai, politika susideda iš vieno ar daugiau politikos teiginių ir gali būti įtraukta į tokius aspektus kaip autentifikacijos schemos, transporto protokolo pasirinkimas, privatumo politika, ir t.t.
- **WS-Privacy –** įmonėms, kuriančioms, valdančioms ir naudojančioms žiniatinklio paslaugas, dažnai reikia paskelbti savo privatumo politikas, ir reikalauti iš ateinančių užklausų pateikti tvirtinimus apie siuntėjo tapatybę, griežtai laikantis nustatytos politikos. Naudojant kartu su WS-Policy, WS-Security, WS-Trust įmonės gali paskelbti ir nustatyti atitikimą, nurodytai privatumo politikai. Ši specifikacija apibūdina modelį, kaip privatumas gali būti integruotas į WS-Policy aprašymą ir kaip WS-Security gali būti naudojamas, norint pasiekti privatumo patvirtinimą su siunčiama žinute.
- **WS-Federation –** tai tam tikrų papildymų rinkinys (saugumo domenų), kuris yra jau nustatęs santykius, leidžiančius saugiai dalintis resursais. Vienos srities resursų teikėjas gali suteikti autorizuotą prieigą prie savo valdomų resursų, remiantis gautomis žymomis (tokiomis kaip asmens tapatybė ar kiti skiriamieji požymiai), kad būtų galima įsitikinti tiekėjo tapatybe kitoje srityje.

Atsižvelgiant į žiniatinklio paslaugų saugos protokolus ir saugumo architektūrą yra pateikiama, kad šiuo būdu apsaugoma visa siunčiama *SOAP* žinutė. Naudojantis WS-Security standartais apsaugoma *SOAP* žinutė tarp siuntėjo ir galutinio gavėjo (angl. ultimate receiver). Apsauga *SAOP* žinutės lygyje yra užtikrinama serveriuose, kurie persiunčia žinutę iki sąveikumo sistemos serverio, kuris vėliau *SOAP* žinutę persiųs galutiniam klientui. Naudojant *SOAP* žinutes sąveikumo

sistemose gali iškilti poreikis neleisti tarpininkaujančiam serveriui skaityti tik dalies persiunčiamos žinutės, tokiu atveju šis sprendimas nėra tinkamas.

2.4.2. HTTP autentifikacija (angl. Simple authentication)

Paprasta autentifikacija yra autentifikavimo mechanizmas, kuris leidžia vartotojui pasiekti žiniatinklio teikiamas paslaugas tuomet, kai vartotojas suveda jam skirtą ID ir slaptažodį. Šiuo atveju:

1. Serveris turi vartotojų ir jų slaptažodžių sąrašus faile.
2. HTTP protokolas palaiko paprastą autentifikaciją, kai yra bandoma pasiekti resursus serveryje. Vartotojo kompiuteryje veikianti programa ar naršyklė siunčia autentifikavimo reikalaujančią žinutę serveriui.
3. Serveris gauna žinutę ir atsako vartotojui atsakymo žinute.
4. Kai vartotojas gauna atsakymo žinutę iš serverio jis siunčia vartotojo vardą (ID) ir slaptažodį serveriui, prašydamas prisijungimo leidimo.
5. Serveris patikrina vartoto atsiųstus duomenis ir jei jie sutampa, duodamas leidimas prisijungti ir pasiekti resursus.

Trūkumai:

- Kadangi HTTP protokolas neatnaujina sesijos (HTTP neišlaiko būsenos), jis negali laikyti vartotojo informacijos, todėl vartotojui reikia įvesti slaptažodį kiekvieną kartą sudarant sesiją su serveriu.
- Slaptažodis siunčiamas serveriui nėra šifruojamas, todėl gali būti pastebėtas šnipinėjančių asmenų.
- Žiniatinklio serverio pusėje vartotojo duomenys yra atvirai saugomi dokumentuose.
- Slaptažodžiai saugomi vienoje vietoje, juos kompromitavus visa sistema taptų nesaugi.

2.4.3. Santraukos autentifikacija (angl. digest authentication)

Šiek tiek saugesnis būdas autentifikuotis yra santraukos autentifikavimo mechanizmas. Šis būdas išsprendžia kelis paprastos autentifikacijos trūkumus. Nors šis algoritmas savo veikimu nedaug skiriasi nuo paprastos autentifikacijos, tačiau persiunčiamas slaptažodis yra šifruotas. Pasinaudojus MD5[5] kriptogramos maišos kodu slaptažodis tampa užšifruota skaičių ir raidžių eilute. Užkoduotas

slaptažodis yra saugomas serverio pusėje prie kitų vartotojo duomenų. Šiam jungiantis slaptažodis siunčiamas MD5 šifru, todėl žinutes perėmęs trečiasis asmuo negalės jų perskaityti.

Verta paminėti, kad šiuo metodu kuriamų slaptažodžių atspėjamumas labai priklauso ir nuo vartotojo. Slaptažodžius reikia sugalvoti kuo sunkesnius, nes sugalvojus dažnai vartojamus žodžius yra įmanoma rasti internete, tam tikruose tinklapiuose. Kaip pavyzdį galima pateikti MD5 santrauką:

Santrauka: 2b877b4b825b48a9a0950dd5bd1f264d
Puslapyje: http://www.md5decrypter.co.uk/ pateikta santraukos reikšmė
Reikšmė: jason
Trūkumai:

- Neužtikrina apsaugos prisijungimo slaptažodžių laikymui.
- Nėra galimybės apsaugoti jau persiųstą informaciją.
- Slaptažodžiai saugomi vienoje vietoje, juos kompromitavus visa sistema taptų nesaugi.
- Persiunčiamų duomenų saugumas labai priklauso nuo vartotojo sugalvoto slaptažodžio.
- Nors duomenys siunčiami neatviru tekstu tačiau juos taip pat gali atskleisti tretieji asmenys, turintys daugiau patirties.

2.4.4. SSL (angl. Secure Socket Layer)

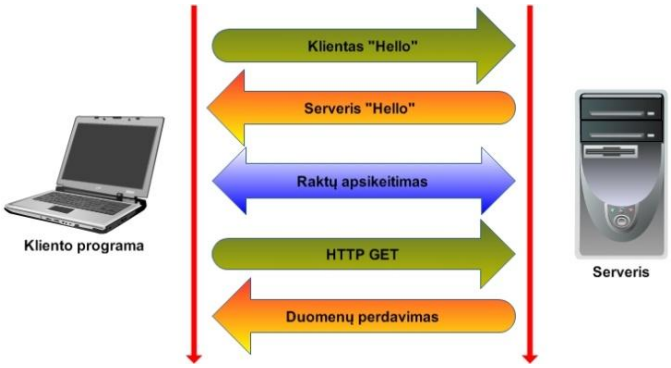
SSL yra OSI antro lygio protokolas sukurtas veikti tarp TCP ir programų lygmens. *SSL* plačiai naudojamas dėl to, kad šis protokolas sujungia visas senąsias apsaugos technologijas. Šis apsaugos metodas šifruoja duomenis, autentifikuoja ir užtikrina persiunčiamų duomenų integralumą. Jis naudojamas autentifikacijai atlikti, šifruoti, persiųsti duomenis, kliento/serverio komunikacijai palaikyti. Šis protokolas palaiko daugelį algoritmų, tokius kaip: DSA (angl. Digital signatare algorithm), KEA (angl. Key Exchange algorithm), MD5 (angl. message digest algorithm), RSA (angl. public key algorithm) naudojamų autentifikuoti vartotoją.

Autentifikavimui SSL naudoja skaitmeninius sertifikatus, kad sukurtų patikimą susijungimą tarp dalyvių. Autentifikavimo serveris, naudojant servisus, turi autentifikuoti klientą, o klientas gali pasirinkti ar autentifikuoti serverį. Taip pat vykdant „rankų paspaudimo“ (angl. handshaking) procesą klientas nurodo sesijos rakto ilgį, kuris bus naudojamas šifruoti duomenims, duomenų suspaudimo metodą ir serverį, su kuriuo jis norės susijungti. Serveris klientui pasiunčia sesijos raktą ir savo skaitmeninį viešąjį sertifikatą užšifruotą kliento atsiųstu slaptu šifravimo raktu. SSL

šifravimo dalis yra labai lanksti, sugebanti naudoti skirtingas šifravimo schemas skirtingose situacijose.

Užtikrinant persiunčiamos informacijos integralumą, SSL tikrina susijungimą, stebėdama ryšio nenutrūkstamumą, ir užtikrina sesijos atnaujinimą, naudojant „rankų paspaudimo“ procesą. Tik puse „rankų paspaudimo“ metodo yra užšifruojama. Pirma žinutė iš kliento serveriui yra siunčiama atviru tekstu, tačiau SSL protokolas turi daugelį metodų apsisaugoti nuo gręsiančių SSL protokolo atakų ar klaidų, aptiktų siunčiamuose duomenyse.

Kaip bendrauja serveris ir klientas, naudojant SSL, pavaizduota 7 pav.



7 pav. SSL metodas

Susijungus abi šalys apsikeičia ryšio sesijos raktais šifruoti pranešimus, siunčiamus tinklu vienas kitam. Abi šalys turi sesijos raktus, bet juos naudoja tik trumpą laiko tarpą. Ginčo metu nei viena šalis negali įrodyti, kad jos žinutė yra nepakeista (neišsiginamumas), be to, naudojant SSL/TLS nėra galimybės išsaugoti žinutės vėlesniam laikui patikrinti ar ji nebuvo modifikuota.

Nors SSL ir šifruoja susijungimą, tačiau informacija lieka neapsaugota, kai ji jau yra galutiniame taške. Kaip pavyzdį galima paimti mašinų vagystes: koks santykis yra tarp automobilių vagysčių, kai automobilis juda kelyje ir kai stovi pastatytas. Tas pats ir su duomenimis. Juos reikia saugoti ne tik persiunčiant internetu, bet ir laikant sąveikumo sistemoje, laukiant, kada bus galima juos persiųsti galutiniam vartotojui.

Persiunčiant *SOAP* žinutes SSL sukuria tunelį tarp dvejų taškų, todėl yra labai suku naudoti *SOAP*, kuris yra naudojamas įmonių sąveikume. SSL sukuria tunelį tarp dviejų taškų, o ne tarp galinių programų. Naudojant programas įmonių sąveikume būtina, kad informacija būtų persiųsta toliau, todėl neužtenka vieno prievado, norint sąveikauti su atskiromis įmonėmis. Naudojant SSL/TLS nėra galimybės išsaugoti žinutes vėlesniam laikui, patikrinti ar ji nebuvo modifikuota.

Privalumai:	**Trūkumai**

• Suteikia gerą duomenų apsaugą persiunčiant duomenis internetu. • Greitas apsaugos metodas, atliekant skaičiavimus.	• Kiekvienos sesijos pradžioje derybų pradžioje yra siunčiamos žinutės atviru tekstu. • Duomenys lieka nešifruoti juos persiuntus. • Naudojant SSL neįmanoma persiųsti informacijos per kelias sąveikumo sistemos programas. Vienu metu susijungimas tik tarp dviejų šalių.

2.4.5. Virtualus privatus tinklas (angl. virtual private network, VPN)

Vienas iš paprasčiausių sprendimų apsaugoti informaciją – naudoti privatų tinklą (toliau *VPN*). *VPT* sprendimai puikiai tinka šiuolaikiniam verslui: saugiai sujungia biurus skirtingose vietose, leidžia bendrauti su verslo partneriais, tiekėjais ir klientais, suteikia galimybę mobiliems darbuotojams per atstumą saugiai prisijungti ir naudotis visomis reikalingomis programinėmis funkcijomis. *VPN* gali būti sudaromi keliais būdais, nustatant *VPN* pritaikytus maršrutizatorius arba įrašant programas serveriuose ir nustatant joms funkcijas operacinėje sistemoje. Sukonfigūravus maršrutizatorių ar *VPN* serverį reikia sukonfigūruoti kliento pusėje susijungimo nustatymus arba įrašyti specialią programinę įrangą. Klientas jungdamasis į privatų tinklą turi suvesti savo prisijungimo duomenis: vartotojo vardą ir slaptažodį. Prisijungus klientui pritaikomos teisės pasiekti jam reikiamus resursus.

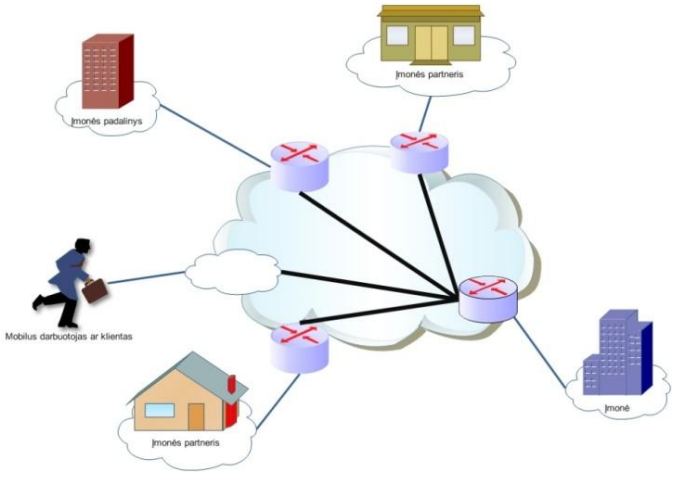
Kadangi atsižvelgiama į sąveikumo sistemų naudojamų duomenų apsaugą, vertėtų paminėti, IPSec *VPN* privalumą, leidžiantį bet kuriuo momentu prijungti klientą prie vidinio tinklo, ar būti tiesiogiai sujungus atskirus LAN segmentus į vieną tinklą.

Kitas *VPN* sprendimas, rinkoje atsiradęs ne taip seniai, yra standartinį IPSec VPN sprendimą mažoms ir vidutinės įmonėms keičiantis *SSL VPN,* skirtas įmonių darbuotojams pasiekti geresnių rezultatų. Verslui *SSL VPN* siūlo įvairiapusiškumą, lengvą naudojimą, saugumą, prisijungimą keliaujantiems darbuotojams. Naudotis *SSL VPN* yra daug lengviau, užtenka interneto naršyklės lange surinkti URL tinklo, prie kurio norima prisijungti, adresą ir suvesti savo prisijungimo duomenis. Suvedus duomenis vartotojas gali būti priskirtas vienai iš prisijungimo grupių. Prisijungusiam vartotojui gali būti suteikti leidimai prie įmonės vidinio tinklo, t.y naudoti tinklo resursus, prisijungti prie žiniatinklio technologijomis grįstų programų ar tinklalapių, tapti programų valdytojais (application manager).

SSL VPN privalumai prieš *IPSec VPN*:

- *SSL VPN* yra pigesnis sprendimas nei *IPSec VPN* dėl to, kad *SSL VPN* klientams nereikia įrašinėti *VPN* naudojimui skirtos programinės įrangos. Visa tai sumažina administravimo poreikį, įrašinėjant ir nustatinėjant programas, ar konsultuojant *VPN* vartotojus.
- *SSL VPN* naudoja TCP 433 jungtį, kuri dažniausiai yra atidaryta daugumos įmonių užkardose. Siunčiant duomenų paketus naudojantis SSL (naudoja 433 jungtį) apsaugos metodu paketų neblokuos užkardos ir nereikės nustatyti jokios papildomos konfigūracijos. IPSec protokolas naudoja UDP prievadus, kurie pagal pradinius nustatymus uždari užkardose, jeigu yra nenaudojami, todėl reikalinga nustatyti papildomus nustatymus užkardose.
- *SSL VPN* spendimai siūlo naršyklės duomenų apsaugą, kuri vartotojui atsijungus nuo *VPN*, ištrina svarbią informaciją, kuri galėjo būti naudojama būnant prisijungus. Į tai įeina ištrynimas „užlaikyti“ vartotojo prisijungimo duomenis ar laikinus užkrautus duomenis.

8 pav. pavaizduota virtualaus tinklo koncepcija. Verslo įmonė, teikianti paslaugą, kitas įmonės padalinys, galbūt esantis kitame mieste, įmonės verslo partneriai ir klientas, naudojantis įmonės teikiamas paslaugas. Pirmiausia klientas turi gauti programinę įrangą ir privilegijas prisijungti prie įmonės privataus tinklo, jeigu nori naudotis programa (ši problema gali būti išspręsta naudojant registravimo sistemas). Kai klientas turi programinę įrangą, kuri veikia tik privačiame tinkle, jam reikia prisijungti prie privataus tinklo, kad galėtų naudotis programos funkcijomis. Pastoviam susijungimui tarp įmonių yra gali būti naudojamas *VPN* tunelis.



8 pav. Virtualaus privataus tinklo schema

Apžvelgus šiuos *VPN* sprendimus galima išskirti jų privalumus ir trūkumus kuriamam įmonių sąveikumo sistemos persiunčiamų duomenų apsaugos sprendimui.

Privalumai:	Trūkumai
• VPN vartotojai yra viename tinkle. • Duomenys tarp įmonių tarpusavyje yra siunčiami saugiau.	• Esant dideliam įmonių skaičiui išnyksta duomenų saugumo privalumas. • Didėjant įmonių skaičiui laipsniškai didėja administravimo perteklius. • Ganėtinai brangūs sprendimai smulkioms ir vidutinėms įmonėms.

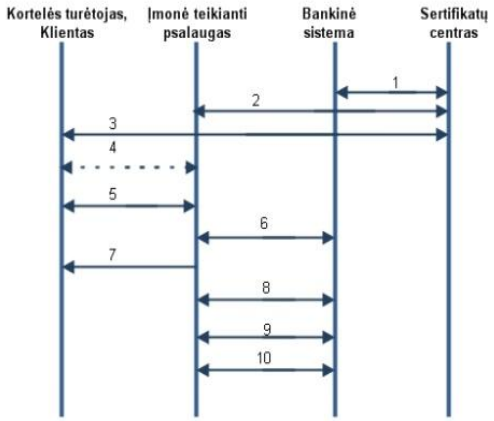
Šis sprendimas palankus esant nedideliam sąveikaujančių įmonių skaičiui, ne tik dėl to, kad didėjant įmonių skaičiui didėja administravimo perteklius, bet ir didėjančios rizikos saugumo aspektais, t.y. tarpusavyje sąveikaujančios įmonės turi pasitikėti viena kita ir įmonėse dirbančiais darbuotojais. Taip pat šio sprendimo pritaikymą labai ribotų tai, kad duomenų apsaugai taikomas saugumas yra persiunčiant duomenis. Duomenims pasiekus galinį persiuntimo tašką jie nėra apsaugoti, todėl tinklo įsibrovėlis ar kitos konkurencinės įmonės šnipinėjantis darbuotojas gali pavogti ir paviešinti informaciją.

2.4.6. SET (angl. Secure Electronic Transaction)

SET metodas šiuo metu yra bene saugiausias būdas keistis informacija tarp verslo partnerių. Kurie yra:

- Banko kortelės turėtojas (toliau klientas).
- Įmonė, teikianti paslaugas klientui.
- Banko sistema, kuri patikrintų, ar kliento pateikti duomenys (jei tai yra banko kortelės duomenys pateikti įmonei, kuri teikia paslaugas), yra teisingi. Banko sistema yra atsakinga už visas finansines operacijas ir turi užtikrinti, kad įmonei, teikiančiai paslaugas, bus sumokėta.
- Sertifikatų centras, kuris yra atsakingas už sertifikatų išdavimą vartotojams ir jų tvarkymą.

SET protokolo veikimas pavaizduotas 9 pav.



9 pav. SET protokolo veikimas

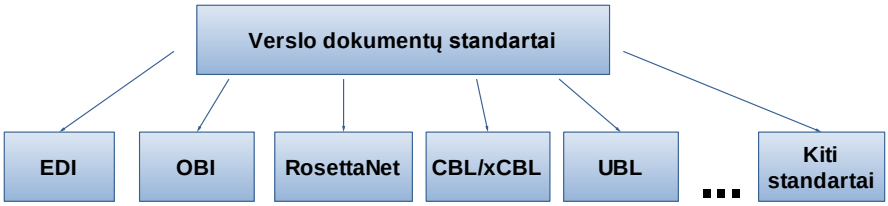
1. Banko sistema gauna sertifikatą iš sertifikatų centro.
2. Įmonė, teikianti paslaugas, gauna sertifikatą iš sertifikatų centro.
3. Vartotojas gauna sertifikatą iš sertifikatų centro.
4. Vartotojas, naudojantis programinę įrangą, išsirenka paslaugas, kurias nori gauti iš įmonės, teikiančios paslaugas, ir nusprendžia už jas sumokėti.
5. Įmonė nusiunčia vartotojui savo sertifikatą, reikalingą atlikti pinigų pervedimui.
6. Vartotojas pasiunčia prašymą įsigyti teikiamas paslaugas, kurias jis pasirinko. Šis prašymas turi užsakymo ir vartotojo kortelės informaciją. Įmonė gauna vartotojo užsakymą ir persiunčia banko sistemai vartotojo mokėjimo duomenis. Įmonė negali pamatyti vartotojo mokėjimo informacijos.
7. Įmonė ir bankinė sistema keičiasi autorizacijos informacija. Įmonė bankinei sistemai siunčia vartotojo korteles informaciją ir užsakymo kainą. Bankinė sistema gali atšaukti arba patvirtinti pavedimą, žinodama vartotojo kortelės duomenis.
8. Įmonė vartotojui nusiunčia pervedimo galimumo patvirtinimą.
9. Pasirenkamas punktas, leidžiantis įmonei pakeisti pinigų kiekį ar atmesti pinigus, autorizuotus 7 punkte.
10. Banko sistemos ir įmonės susitarimu apmokamas užsakymas.

Nors ir *SET* protokolas teikia privatumą, integralumą, autentifikaciją, tačiau taip pat turi ir didelių trūkumų dėl kurių, yra nepatogus naudoti persiunčiamų duomenų apsaugai sąveikumo sistemose. Apibendrinant *SET* protokolo veikimą galima įžvelgti tai, kad dalyvaujančioms šalims yra nustatyti tam tikri reikalavimai, pavyzdžiui, vienas vartotojas galės matyti tik tam tikrą verslo dokumento dalį, kas stipriai apribotų *SET* protokolo saugumo įgyvendinimą sąveikumo sistemose.

Privalumai:	Trūkumai
• Kriptografijos pagalba siunčiamos žinutės yra neperskaitomos trečiųjų asmenų. • Integralumas, maišos algoritmai ir skaitmeninis parašas užtikrina, kad siųsta žinutė yra nepakeista. • Autentifikavimas leidžia žinoti, kad siuntėjas yra tas, kuris ir sakosi esąs.	• Sunkiai įgyvendinamas *SET* programų sąveikumas. • Sunkus integravimas į sistemas. • Brangus ir lėtas apsaugos metodas.

2.5. Verslo dokumentų standartai

Tam, kad sąveikumo sistemoje įmonės galėtų sąveikauti, reikalingas bendras verslo dokumentų standartas, leidžiantis joms keistis užsakymais, sąskaitomis faktūromis, verslo ataskaitomis ir t.t. Pasirinkus vieną verslo dokumentų standartą įmonės galėtų jį naudoti persiunčiant duomenis į sąveikumo sistemą prieš tai šiam dokumentui pritaikius apsaugos galimybes. Šiame skyrelyje apžvelgiami, kokie verslo dokumentų standartai galėtų būti naudojami sąveikumo sistemose dokumentams persiųsti (10 pav.), t.y., jų tipas, dydis, turimi privalumai prieš kitus standartus. Pasirenkant verslo dokumentų standartą atsižvelgiama į jo populiarumą, panaudojimo sritį bei panaudot galimybes ateityje.



10 pav. Verslo dokumentų standartų įvairovė.

EDI (angl. Electronic Data Interchange). Prieš pasirodant XML verslo dokumentams perduoti tarp tiekėjų ir pirkėjų buvo naudojami *EDI* tinklai, kurie padarė vertingą įnašą į verslo produktyvumo ir našumo didinimą. *EDI* technologija yra naudojama nuo 1970-ųjų, todėl paplito daugelyje sričių. Ji buvo plačiai pripažinta kaip technologija, suteikianti galimybę keistis elektroniniais dokumentais, kurie buvo svarbūs įmonėms veikiančioms skirtinguose srityse. *EDI* standartinių verslo dokumentų, tokių kaip užsakymai, sąskaitos, važtaraščiai naudojimas įsitvirtino daugiausiai tose srityse, kur *EDI* buvo jau plačiai naudojama. Ši technologija buvo gerai pritaikyta persiųsti griežtai struktūralizuotiems duomenims, kurie dažnai nesikeisdavo. Kaip bebūtų, šiuolaikiniai verslas verslui (B2B) scenarijai reikalauja daugiau lankstumo, ypač kai įmonės taiko kitokius procesus, kurie gali paveikti visą gamybos ar tiekimo grandinę [21]. Pagrindiniai *EDI* trūkumai yra didelė kaina ir sudėtingas įdiegimas, verslo dokumentai nėra perduodami tiesiogiai internetu. Taigi nors *EDI* panaudojimas prasidėjo daug anksčiau, tačiau naujai sukurti XML paremti sprendimai, lengviau realizuojami sudėtingesnėse sistemose bei leidžia lengviau plėtoti jau esamus sprendimus.

OBI (angl. Open Buying on the Internet) – buvo sukurtas 1997-10. *OBI* yra lankstus karkasas *verslas verslui* (B2B), el. verslo sprendimams. Šio standarto paskirtis yra didelės apimties mažų

kainų transakcijos, kurios sudaro 80% daugumos organizacijų pirkimų [22, 23, 31, 32]. Taip pat šis standartas yra suderintas veikti su jau pripažinimo sulaukusiomis ir paplitusiomis technologijomis tokiomis kaip: *HTTP*, Skaitmeniniai sertifikatai (X509), *SSL*, ir *EDI* standartais. Dokumento struktūra padeda lengvai pašalinti pasikartojančius duomenis skirtingose dokumento vietose ir palengvina standarto koregavimą ateityje. *OBI* procese įtrauktas užsakymo pateikimas, jo patvirtinimas ir sprendimo procedūros. Įmonėsm naudojančios *OBI,* turi naudoti *EDI* sintaksę duomenų apsikeitimui. Standarto trūkumas yra tai, jog *OBI* apima tik nedidelę prekės įsigijimo proceso dalį, dėl kurios standartas nesugebėjo plačiai paplisti .

RosettaNET – sukurtas 1998m. RosettaNET – tai verslo dokumentų standartas, kuris taip pat remiasi XML verslo dokumentų formatais. RosettaNet standartai, leidžiantys automatizuoti verslo procesus tarp verslo partnerių ir sumažinti technines bei finansines kliūtis e-verslui. Per savo 13 gyvavimo metų RosettaNET standartai paplito daugelyje verslo sričių ir skirtingo dydžio įmonių. Tačiau RosettaNet yra naudojamas dažniau kaip protokolų rinkiniai, kurie labiau nusako būdus keisti informaciją negu pačius dokumentų standartus [24, 25].

CBL /xCBL (angl. Common Business Library) – buvo sukurtas 1977m. CBL yra XML dokumentų sudarymo karkasas, kuris leidžia aprašyti XML dokumentus prekybos srityje [26]. xCBL tarnauja kaip pagrindas el. rinkos dalyvių tarpusavio bendravimui. Šis standartas suteikia galimybę pereiti nuo EDI paremtų e-verslo sistemų, nes turi šio standarto semantinių pranašumų. xCBL yra tvirtos sandaros pakartotinai koreguojamas XML dokumentas, naudojamas prekybos srityje [30].

Pagrindinių minėtų standartų problemos yra tos, jog dauguma standartų yra orientuoti į siauras specifines sritis, be to šie standartai siūlo priemones e-verslo sistemoms kurti ar modifikuoti, tačiau bet koks veikiančios sistemos keitimas yra daug kaštų reikalaujantis procesas, o tai ypač aktualu mažoms organizacijoms.

UBL (angl. Universal Business Language). Kaip bendras standartas elektroninio verslo sistemoms, dėka OASIS ir UN/CEFACT pastangų, 1999 metų viduryje buvo sukurtas universalus XML dokumentų paketas, palengvinantis el. verslo plitimą tarp mažų ir vidutinių įmonių. OASIS rūpinasi technine UBL puse (pranešimais, registru, saugumu ir profiliais), tuo tarpu UN/CEFACT yra atsakinga už turinį ir verslo pusę (pagrindinius komponentus ir verslo procesų modelius). UBL standartas savyje apjungia geriausias EDI, OBI ir kitų XML standartų (CBL, xCBL, cXML, RosettaNet) savybes. UBL suteikia standartinių verslo dokumentų elektroniniame XML formate viešą biblioteką, skirtą perduoti verslo duomenis apie sąskaitas faktūras, užsakymus, važtaraščius ir

kitus duomenis tarp verslo partnerių. Ši biblioteka sudaryta iš 31 XML dokumento, paremto ebXML Core Components Technical Specification (ISO 15000-5) [27, 28, 29]. UBL standarto paplitimą sąlygojo šie veiksniai: sąveikumas su egzistuojančiomis EDI sistemomis, tinkamumas mažoms ir vidutinėms įmonėms, nes tai kartu ir pigus, ir paprastas įdiegimo sprendimas.

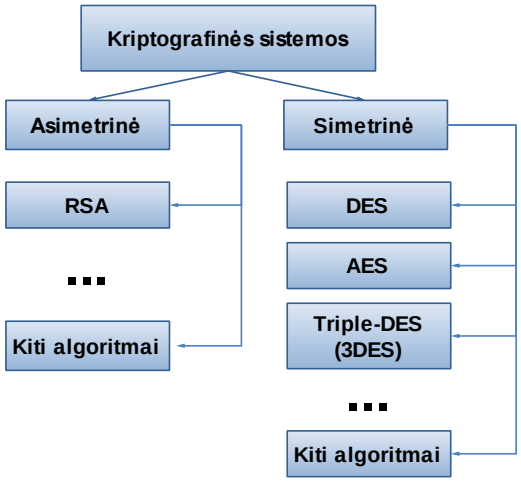
Dėl plataus UBL standarto paplitimo ir panaudojimo suderinamumu su žiniatinklio paslaugomis tyrimui atlikti tikslinga būtų pasirinkti tirti šio standarto šifravimo/ iššifravimo galimybes. Dokumentų dydžius galima suskirstyti į keletą tipų: dokumentai nuo 1Kb iki 50Kb, vidutinio dydžio dokumentai 50-100Kb ir dokumentai, skirti persiųsti didesniems dokumentams, nuo 100Kb iki 1Mb ir daugiau. Didelio dydžio dokumentai taip sąveikumo sistemos dalyvių yra persiunčiami rečiau, kadangi žiniatinklio paslaugų pagalba dažnai verslo dokumentai siunčiami realiu laiku, dokumentai nebūna itin dideli.

Verslo dokumentais siunčiama informacija gali būti konfidenciali, todėl būtina naudoti šifravimą. Šiame skyriuje nėra apžvelgiamos žiniatinklio paslaugomis grįstos sąveikumo sistemos ir nėra nustatyta kokiai sąveikumo sistemai bus taikomas šifravimas, todėl didesnis dėmesys bus skiriamas sąveikumo sistemoms, kurios tiesiogiai persiunčia gautus šifruotus/ pasirašytus dokumentus. Todėl galima teigti, jog tam tikri duomenys dokumente ir visas dokumentas yra koduojamas:

1. Šifruojama konfidenciali informacija: vardas, pavardė, adresas, sąskaitos numeris ir kt.
2. Šifruojamas visas dokumentas, kurį sudaro: užsakymo numeris, užsakomi produktai ar paslaugos, produktų ar paslaugų kiekis, užkoduota konfidenciali informacija ir kiti verslo dokumente reikalingi duomenys.

2.6. Duomenų šifravimo algoritmai

Konfidencialios informacijos apsaugai verslo dokumentuose būtina naudoti šifravimą, kad informacija nebūtų atskleista tretiesiems asmenims. Pasirinkus kelis populiariausius šifravimo algoritmus (11 pav.), apžvelgiamos jų stipriosios ir silpnosios savybės, algoritmai palyginami tarpusavyje.



11 pav. Tyrime naudojami kriptografijos algoritmai.

DES (angl. Data Encryption Standard) – DES yra simetrinis blokinis šifras, kuriame šifruojami 64-bitų atviro teksto blokai, naudojant 54-bitų raktą. Dešifravimas yra atvirkščias užšifravimui: atliekami užšifravimo veiksmai, vykdomi atvirkščia tvarka. Algoritmo saugumas priklauso nuo rakto. DES turi 64 silpnuosius raktus. DES algoritmas – tai dviejų pagrindinių šifravimo metodų išsklaidymo bei sumaišymo kombinacija. Viename algoritmo etape yra naudojamos vienetinės šių metodų kombinacijos (pakeitimas ir perstatymas), kurios priklauso nuo rakto. DES algoritmas susideda iš 16 etapų: atviram teksto blokui yra taikoma ta pati metodų kombinacija šešiolika kartų. Algoritmo pagrindas – *Feistelio struktūra* [32]. Pagrindiniai DES metodo trūkumai yra tai, jog metodo struktūra yra paprasta, lengvai vykdomos šifravimo operacijos. Nors šis metodas ir yra dažnai naudojamas šifruoti informacijai, tačiau dėl savo silpno 54 bitų rakto nėra apsaugotas nuo atakų.

3DES (angl. Triple Data Encryption Standard). Tobulėjant kompiuterinėms technologijoms, DES 56-bitų raktas tapo per trumpas. Kaip pakaitalas padėjęs išspręsti keletą DES algoritmo trūkumų vėliau pradėtas naudoti Triple DES (3DES). 3DES – tai DES algoritmo versija, kurioje baziniu DES algoritmu atviro teksto blokas šifruojamas tris kartus (duomenys yra užšifruojami pirmuoju raktu, dešifruojami antruoju raktu ir vėl užšifruojami trečiuoju raktu). Kadangi 3DES yra paremtas DES algoritmo pagrindu yra ganėtinai paprasta pakoreguoti programinę įrangą, kad ši naudotų 3DES šifravimą. 3DES Rakto ilgis – 168 bitai, bloko ilgis 64 bitai. 3DES paveldėjo visus DES privalumus, kartu padidino atsparumą daugiau nei du kartus [33]. Tačiau nors 3DES ir yra stipresnis už DES, tačiau yra nepakankamai saugus apsaugoti duomenis ilgą laiko tarpą. 1997 m. National Institute of Standards and Technology (NIST) įmonė paskelbė atvirą konkursą kurti AES algoritmą, o tuo tarpu 3DES buvo paskelbtas kaip laikinas standartas.

AES (angl. Advanced Encryption Standard). Oficialiai šis algoritmas yra apibūdintas FIPS PUB 197 standarte. Kaip ir anksčiau minėti standartai AES šiuo metu AES yra vienas iš plačiausiai naudojamų blokinio šifravimo algoritmų. AES duomenys šifruojami 128, 168,192 bitų ilgio blokais, t.y., AES algoritmo įėjimas ir išėjimas yra 128 bitų sekos. Rakto ilgis gali būti 128, 192 arba 256 bitai. Kiti įėjimo, išėjimo arba rakto ilgiai AES standarte neleidžiami [33]. Skirtingai nuo DES, kuris buvo daugiau optimizuotas aparatūrinei realizacijai, AES yra efektyvus skirtingose realizacijose. Kiekvienas algoritmo žingsnis susideda iš operacijų, kurios gali būti atliekamos vienu metu lygiagrečiai, o tai ir nulemia aukštą algoritmo greitį. Pasak tyrėjų, standartinės AES versijos kompromitacija galėtų būti įmanoma ne anksčiau kaip po 7 metų [34].

RSA (angl. Rivest, Shamir and Adleman) – yra viešo rakto asimetrinė kriptosistema [36], kuri palaiko duomenų šifravimą ir skaitmeninius parašus. Tai dažniausiai naudojamas ir populiariausias viešojo rakto algoritmas. RSA raktas gali būti sudarytas iš 512, 768, 1024, 2048 bitų. RSA algoritmas naudojamas tiek užšifravimui, tiek iššifravimui, panaudojant viešai žinomą ir privatų raktą. Tikimybė sugeneruoti du kartus tą pačią šifravimo raktų porą yra labai maža. Jei duomenys yra užšifruojami vienu raktu, tai juos iššifruoti įmanoma tik kitu tos poros raktu. Žinant tik vieną poros raktą neįmanoma atstatyti kito rakto [35]. Šiuo metu RSA algoritmu užšifruoti duomenys yra saugiausi.

2.7. Analitinės dalies apibendrinimas

Taigi atlikus analitinę dalį ir apžvelgus egzistuojančias sąveikumo sistemas pastebėta, kad daugumoje sąveikumo sistemų nėra skirta pakankamai dėmesio duomenų apsaugai. Atsižvelgus į GENESIS projektą galima susidaryti nuomonę, kad jeigu ir yra sąveikumo sistemoje realizuotas duomenų konfidencialumą užtikrinantis modelis, nėra patikimos apsaugos nuo trečiųjų asmenų ar net nuo sąveikumo sistemos. Kaip pavyzdį galima paimti mašinų vagystes: koks santykis yra tarp automobilių vagysčių, kai automobilis juda kelyje ir kai stovi pastatytas. Tas pats ir su duomenimis, duomenis reikia saugoti ne tik juos persiunčiant internetu, bet ir saugant juos sąveikumo serveryje laukiant, kada galima bus juos persiųsti.

Apsaugant verslo dokumentus reikia atsižvelgti į sąveikumo sistemos savybes. Dažnai sąveikumo sistemose yra būtinybė keisti persiunčiamo dokumento turinį pagal vykstančius verslo procesus, todėl reikia apsaugoti siunčiamus verslo dokumentus nuo siuntėjo iki gavėjo, kurie keliauja per sąveikumo sistemos serverį. Kaip rodo analizė, iš čia iškyla problema ir pagrindinis uždavinys: žiniatinklio paslaugomis grįstų įmonių sąveikumo sprendimuose naudojamų verslo dokumentų vientisumo, konfidencialumo ir neišsigynimo problemos sprendimas, skirtas smulkioms ir vidutinėms įmonėms.

Šioje dalyje taip pat buvo apžvelgti apsaugos metodai, kurie galėtų būti naudojami apsaugant siunčiamus verslo dokumentus, atsižvelgiant į tinkamumą iškilusiai problemai ir akcentuoti metodų privalumai ir trūkumai. Apžvelgti ir išanalizuoti dokumentų standartai: standartais, kuriais operuoja žiniatinklio paslaugos, bei keletas verslo dokumentų standartų tipų, kurių formatu galėtų būti siunčiami verslo dokumentai. Išnagrinėta žiniatinklio *SOAP* pranešimo architektūra saugos atžvilgiu bei populiariausieji šifravimo algoritmai, kurie galėtų būti naudojami, apsaugant konfidencialius duomenis.

Pagal vėliau išsikeltus reikalavimus bus pasiūlytas ir sukurtas pavyzdinis verslo dokumentų saugaus perdavimo sąveikumo sistemomis metodas, kuris bus realizuotas standartinėmis programavimo priemonėmis. Jis bus tiriamas naudojant supaprastintą įmonių sąveikumo sistemos prototipą. Bus sukurtos kelios pavyzdines programos: vartotojo (pirkėjo/ pardavėjo – sąveikumo sistemoje dalyvaujančių subjektų) ir kelias funkcijas turinti sąveikumo sistemos programa, kuri leis pastebėti sistemos saugos privalumus ir trūkumus. Metodo įgyvendinimui panaudojus viešojo rakto kriptografijos metodus bus užtikrinta, kad vieno vartotojo siųsti dokumentai pasieks kitą dalyvaujančią šalį be trečiųjų asmenų įsikišimo bei sąveikumo sistemos galimybes atskleisti tam

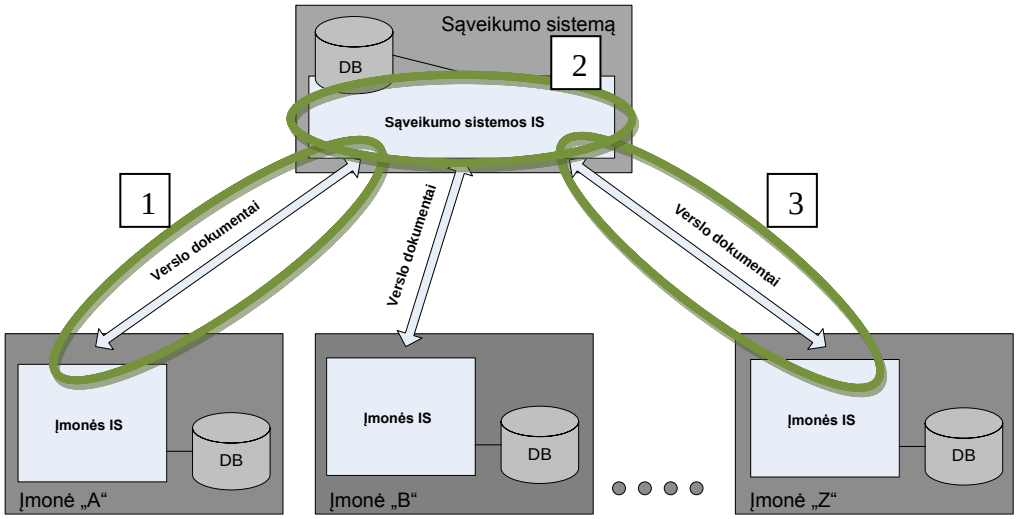
tikrą informaciją, tačiau leis sąveikumo sistemos serveriui kodifikuoti tam tikrus laukus verslo dokumente. Sukūrus pavyzdines programas ir jas išbandžius bus apžvelgta šio metodo trūkumai ir privalumai prieš *VPN* ar *SSL* metodus.

3. Projektavimas

3.1. Probleminė sritis

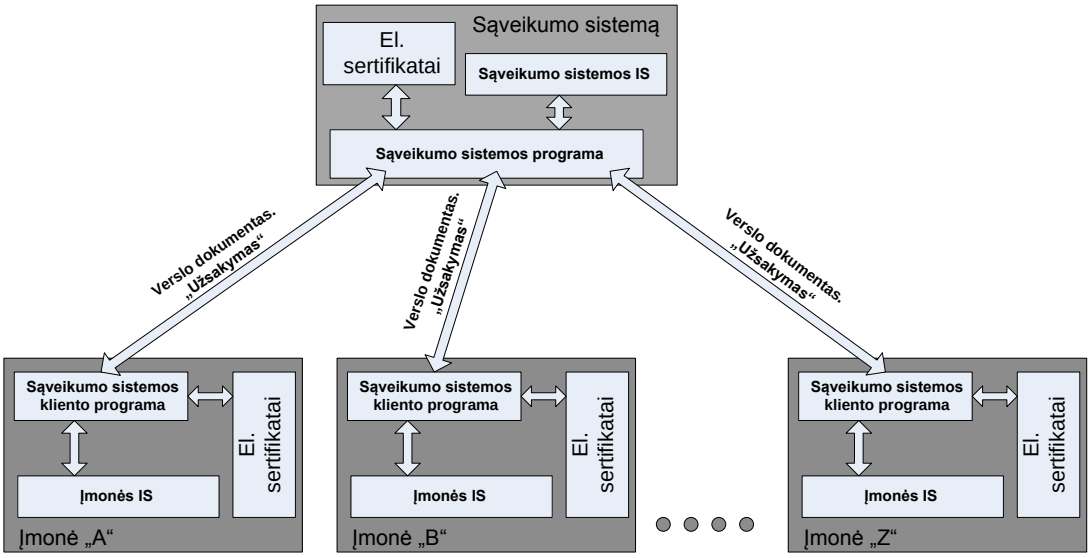
Vienas pagrindinių tikslų yra tikslus reikalavimų ir norimų pasiekti rezultatų užsibrėžimas. Šioje dalyje numatomi projektuojamo tyrimo tikslai, uždaviniai ir galinčios iškilti problemos, pristatomi resursai, kurie bus naudojami tyrimo eigoje bei apibrėžiami apribojimai tiriamajam metodui. Atlikus verslo įmonių sąveikumo sistemų ir jose kylančių saugos problemų analizę, išanalizavus verslo įmonių dokumentų standartus, naudojamus sąveikumo sistemose, bei apžvelgus esamus dokumentų vientisumo, konfidencialumo ir neišsigynimo užtikrinimo metodus, pateikiami darbo tikslai ir uždaviniai:

1. Suformuoti reikalavimus naujam dokumentų saugos metodui, kuris užtikrintų verslo dokumentų saugą persiunčiant dokumentus nuo siuntėjo iki gavėjo per sąveikumo sistemos serverį.
2. Nustatyti turimus išteklius, su kuriais bus atliekamas tyrimas.
3. Parengti projektuojamo metodo modelį saugiam dokumentų persiuntimui.
4. Realizuoti saugos metodo prototipą ir sukurti programinę įrangą metodo testavimui.
5. Programinės įrangos pagalba atlikti eksperimentą ir surinkus duomenis, išanalizuoti greitaveikos skirtumus tarp skirtingų apsaugos metodų.
6. Įvertinti sukurto metodo saugumą, paprastumą, greitaveiką ir palyginti su keliais kitais metodais.



12 pav. Sąveikumo sistemos apsaugos sritys

Projektuojamos sistemos modelis pavaizduotas 12 pav., kuriame matyti, kokias sistemos dalis norima apsaugoti: persiunčiamą informaciją nuo sąveikumo sistemos kliento programos (siuntėjo) iki serverio programos [1], verslo dokumento saugų išlaikymą sąveikumo sistemos kompiuteryje [2] ir nuo sąveikumo sistemos programos (serverio) iki sąveikumo sistemos kliento programos (gavėjo) [3].



13 pav. Detalesnis sistemos modelis.

Sąveikumo sistemoje pas klientą veikianti „Sąveikumo sistemos kliento programa“, naudodamasi kitų klientų bei sąveikumo sistemos sertifikatais, šifruoja persiunčiamą verslo dokumentą (pasirinktu atveju užsakymą) kitam sąveikumo sistemos klientui.

Atliekant tyrimą nėra gilinamasi, kokius veiksmus atlieka kliento ar sąveikumo informacinės sistemos. Modeliavimo aplinkoje nustatomos saugyklos, kuriose bus saugomi verslo dokumentai ir numatytieji sertifikatai.

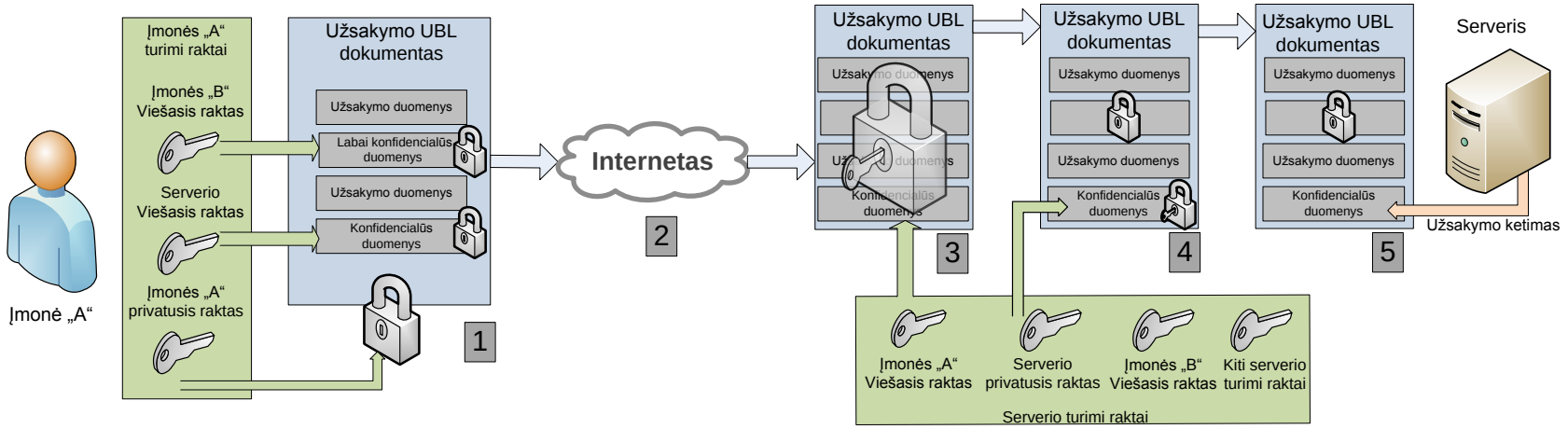
Įmonės „A“, „B“, ... , „Z“ programa paima verslo dokumentus iš nustatytos saugyklos ir juos užšifravusi pasiunčia sąveikumo sistemos programai (iš anksto nustatytu tinklo adresu).

Sąveikumo sistemos programa gavusi dokumentą ar dokumentus patikrina jų integralumą, taip pat išsišifruoja jai priskirtus koreguoti laukus ir, atlikusi reikiamus veiksmus, šifruoja savo pakeistą informaciją bei pasirašo visą dokumentą. Atlikusi visus veiksmus serverio programa persiunčia verslo dokumentą nustatytai klientinei programai (gavėjui).

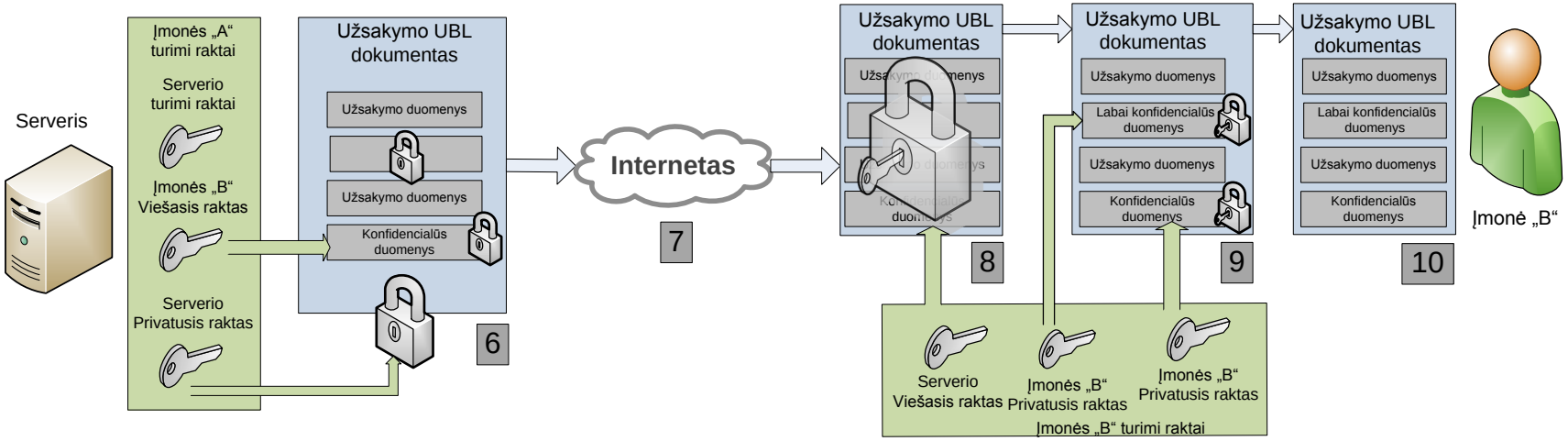
Gavusi verslo dokumentus sąveikumo sistemos kliento programa (gavėjas) patikrina jų integralumą ir išsišifruoja visus šifruotus laukus bei padeda apdorotą verslo dokumentą į saugyklą.

Modeliavimo sistemos schema pavaizduota 14-15 pav.

Simbolis	Reikšmė
	Raktas (gali reprezentuoti viešąjį arba privatųjį raktą).
	Užšifruoti/pasirašyti arba šifruojami/pasirašomi duomenys.
	Iššifravimas/pasirašytų dokumentų tikrinimas



14 pav. Verslo dokumento kelias nuo siuntėjo iki sąveikumo sistemos programos.



15 pav. Verslo dokumento kelias nuo sąveikumo sistemos programos iki gavėjo.

Siuntėjo programa gavusi verslo dokumentą iš siuntėjo informacinės sistemos, nuskaičiusi dokumentą iš numatytosios saugyklos, atlieka veiksmus, pradėdama saugų dokumento persiuntimą. Su verslo dokumentu atliekami veiksmai:

1. Šifruojami „labai konfidencialūs duomenys“ įmonės „B“ viešuoju raktu, šifruojami „konfidencialūs duomenys“ serverio viešuoju raktu, pasirašomas verslo dokumentas įmonės „A“ privačiuoju raktu.
2. Per internetą siunčiamas verslo dokumentas serveriui.
3. Patikrinamas gauto dokumento integralumas įmonės „A“ viešuoju raktu.
4. Iššifruojami „konfidencialūs duomenys“ serverio privačiuoju raktu.
5. Keičiami duomenys iššifruotoje dalyje.
6. Šifruojami „konfidencialūs duomenys“ įmonės „B“ viešuoju raktu. Pasirašomas verslo dokumentas serverio privačiuoju raktu.
7. Per internetą siunčiamas verslo dokumentas gavėjui.
8. Patikrinamas gauto dokumento integralumas įmonės serverio viešuoju raktu.
9. Iššifruojami „labai konfidencialūs duomenys“ ir „konfidencialūs duomenys“ įmonės „B“ privačiuoju raktu.
10. Persiųstas verslo dokumentas.

Persiuntus verslo dokumentą gavėjo informacinė sistema jį toliau naudoja savo nuožiūra, priklausomai, kokie veiksmai gavėjo sistemoje yra numatyti atlikti gavus persiųstą dokumentą. Verslo dokumentas gali būti įrašytas į dokumentų saugyklą ar tiesiog perduotas kitai programai veikiančiai toje pačioje informacinėje sistemoje.

3.2. Reikalavimai dokumentams

Iš daugelio verslo dokumentų standartų, aptartų analitinėje dalyje (EDI, OBI, RosettaNet, CBL, xCBL, UBL), sistemos modelio projektavimui dėl masiško standarto paplitimo, plataus panaudojimo, suderinamumu su žiniatinklio paslaugomis yra pasirinktas UBL verslo dokumento tipas.

Taip pat tyrimui atlikti iš daugybės verslo dokumentų tipų buvo pasirinktas “Užsakymo“ UBL dokumentas, kuriame yra itin konfidencialūs duomenys, konfidencialūs duomenys, kurie turi būti matomi sąveikumo sistemai, bei duomenys, kurių tretieji asmenys, išskyrus sąveikumo sistemos programą (serverį), negali keisti.

Ypač konfidencialiems duomenims priskiriami siuntėjo duomenys: banko sąskaitos numeris, įmonės pavadinimas ar kiti su identifikacija susiję laukai, gavėjo rekvizitinė informacija.

Konfidencialūs duomenys – tai tokie duomenys, kuriuos turi sąveikumo sistemos programa ir kuriuos ji gali keisti, tačiau tretieji asmenys šios informacijos neturi matyti. Tokio tipo duomenys gali būti gavėjo identifikacinis kodas sąveikumo sistemoje, pagal kurį sąveikumo sistema nustato gavėjo adresatą.

Nekonfidencialūs duomenys – tai tokie duomenys, kurie gali būti matomi tretiesiems asmenims ir negali nieko nurodyti apie gavėją ar siuntėją. Tai su užsakymu susiję duomenys, tokie kaip užsakomi produktai ar paslaugos, užsakomų produktų ar paslaugų kiekis (bet ne kaina), bendri produktų ar paslaugų aprašymai.

Reikia paminėti, kad laukai, kurie bus šifruojami yra nustatomi iš anksto, t.y. prieš atliekant tyrimą, jau yra žinomi kaip juos reikia šifruoti. Sąveikumo sistemos programa keičia iš anksto žinomą vietą persiunčiamame UBL verslo dokumente (užsakyme). Žemiau pateiktame pavyzdyje žemiau yra matomas dokumento pavyzdys ir išskirti laukai, kurie turėtų būti šifruojami ar pasirašomi:

```
<?xml version="1.0" ?>
<Order xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:cac="urn:oasis:names:draft:ubl:schema:xsd:CommonAggregateComponents-2"
xmlns:cbc="urn:oasis:names:draft:ubl:schema:xsd:CommonBasicComponents-2"
xmlns:udt="urn:un:unece:uncefact:data:
draft:UnqualifiedDataTypesSchemaModule:2"
xmlns:sdt="urn:oasis:names:draft:ubl:schema:xsd:SpecializedDatatypes-2"
xmlns:ccts="urn:oasis:names:draft:ubl:schema:xsd:CoreComponentParameters-2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="urn:oasis:names:draft:ubl:schema:xsd:Order-2"
xsi:schemaLocation="urn:oasis:names:draft:ubl:schema:xsd:Order-2 prd-UBL-
2.0/xsd/maindoc/UBL-Order-2.xsd">
  <cbc:ID>XYZ-4298</cbc:ID>
  <cbc:CopyIndicator>false</cbc:CopyIndicator>
  <cbc:IssueDate>2010-05-02</cbc:IssueDate>
  <cbc:IssueTime>18:14:36</cbc:IssueTime>
  <cbc:DocumentCurrencyCode>LTL</cbc:DocumentCurrencyCode>
<cac:BuyerCustomerParty>
      <cbc:CustomerAssignedAccountID>111222333</cbc:CustomerAssignedAccountID>
<cac:Party>
      <cac:PartyName>
            <cbc:Name>Imone „A"</cbc:Name>
      </cac:PartyName>
</cac:Party>
</cac:BuyerCustomerParty>
<cac:SellerSupplierParty>
    <cbc:CustomerAssignedAccountID>123456789</cbc:CustomerAssignedAccountID>
     <cac:Party>
       <cac:PartyName>
            <cbc:Name>Imone „B"</cbc:Name>
       </cac:PartyName>
     </cac:Party>
</cac:SellerSupplierParty>
<cac:LegalTotal>
```

```
  <cbc:TaxInclusiveAmount currencyID="LTL">250.00</cbc:TaxInclusiveAmount>
  <cbc:ToBePaidAmount currencyID="LTL">250.00</cbc:ToBePaidAmount>
</cac:LegalTotal>
<cac:OrderLine>
     <cac:LineItem>
     <cbc:ID>1</cbc:ID>
      <cbc:LineExtensionAmount currencyID="LTL">120</cbc:LineExtensionAmount>
       <cbc:MinimumQuantity unitCode="KG">10</cbc:MinimumQuantity>
      <cbc:MaximumQuantity unitCode="KG">20</cbc:MaximumQuantity>
     <cac:Item>
        <cbc:Description>Plikyti sausainiai "Laumė"</cbc:Description>
              <cbc:PackQuantity>6</cbc:PackQuantity>
        <cbc:PackSizeNumeric>1</cbc:PackSizeNumeric>
        <cbc:Name> Plikyti sausainiai "Laumė"</cbc:Name>
        <cbc:HazardousRiskIndicator>false</cbc:HazardousRiskIndicator>
            <cac:SellersItemIdentification>
               <cbc:ID>4770732193610</cbc:ID>
            </cac:SellersItemIdentification>
            <cac:ClassifiedTaxCategory>
              <cbc:ID>Standard Rate</cbc:ID>
              <cbc:Percent>18</cbc:Percent>
                   <cac:TaxScheme>
                     <cbc:ID>VAT</cbc:ID>
                     <cbc:Name>VAT</cbc:Name>
              </cac:TaxScheme>
       </cac:ClassifiedTaxCategory>
            <cac:ItemInstance>
       <cbc:ProductTraceID>1</cbc:ProductTraceID>
       <cbc:ManufactureDateTime>2010-05-12T12:00:00</cbc:ManufactureDateTime>
       </cac:ItemInstance>
    </cac:Item>
    </cac:LineItem>
    </cac:OrderLine>
  </Order>
```

Šifruojamos sąveikumo sistemos programos viešuoju raktu (konfidenciali informacija)

Šifruojamos gavėjo programos viešuoju raktu (labai konfidenciali informacija)

Pasirašomas verslo dokumentas (visa informacija)

Atliekant tyrimą gali būti nurodyta tik dalis persiunčiamo verslo dokumento, kuri yra pasirašoma, siekiant taupyti sistemos resursus.

3.3. Turimi ištekliai

Atliekant tyrimą priimta, kad yra turimi šie resursai ir programose jie nebus realizuojami. Šių išteklių paskirstymo ar integravimo į sistemą problemos nėra sprendžiamos šiame tyrime.

Sertifikatai:

- Atliekant šį tiriamąjį darbą priimta, kad vartotojai, besinaudojantys sąveikumo sistema, bei pati sąveikumo sistema, turi savo elektroninius sertifikatus, kuriais gali šifruoti ar iššifruoti duomenis.
- Priimta, kad elektroniniai sertifikatai gauti iš patikimos sertifikatų platinimo įstaigos.
- Priimta, kad elektroniniai sertifikatai įkelti į sistemą, ir juos gali pasiekti tyrime naudojamos programos.
- Visi viešieji sertifikatai yra talpinami patikimame serveryje, iš kurio programos galėtų pasiimti sertifikatus.

Autentifikacija:

- Kadangi pati viešojo rakto infrastruktūra teikia autentifikacijos paslaugą, kitokia apsauga tyrimo programose, ar dokumentų persiuntime nenumatyta.
- Modeliavimo programose didelis dėmesys yra skiriamas duomenų konfidencialumui ir apsaugai nuo neteisėto jų paviešinimo užtikrinti, bet ne autentifikacijos problemoms spręsti.

Vartotojai

- Programų vartotojai yra įmonės, priklausančios sąveikumo sistemai. Priimta, kad kiekvieną įmonę šiame tyrime simbolizuoja turimas įmonės (įmonės padalinio) elektroninis sertifikatas.
- Sąveikumo sistemoje dalyvaujančių vartotojų skaičius nėra apibrėžtas (tyrime naudojamos dvi įmonės ir sąveikumo sistemos programa).
- Visi vartotojai naudojasi viena programa, kurios pagalba siunčia ir gauna verslo dokumentus iš sąveikumo sistemos.

3.4. Reikalavimai programinei įrangai

3.4.1. Funkciniai reikalavimai

[Bendri]

- Serverio ir kliento programa turi vizualiai atvaizduoti savo būseną kompiuterio ekrane.
- Serverio ir kliento programa turi pateikti informaciją apie atliekamas operacijas vartotojo sąsajoje.
- Serverio ir kliento programa turi pateikti informaciją apie sugaištą laiką operacijoms atlikti.

- Serverio ir kliento programa turi operacijoms sugaištus laikus, sistemos būseną išsaugoti “log” bylose.
- Serverio ir kliento programa turi nuskaityti failą ar failus iš numatytosios saugyklos.
- Serverio ir kliento programa turi įrašyti persiuntimo informacija į „įrašų “ bylą.
- Serverio ir kliento programa turi turėti galimybę siųsti nešifruotus dokumentus. T.y. praleisti šifravimo atliekamas operacijas.

[Klientinės programos]

- Programoje turi būti galimybė nurodyti verslo dokumentų sąveikumo sistemos (serverio) kompiuterio, kuriame yra programinė įrangą tinklo adresą.
- Kliento programa (siuntėjas) pagal nustatytą logiką turi užšifruoti nuskaityto dokumento turinį.
    - Programa privalo šifruoti siuntėjo privačius duomenis gavėjo sistemos viešuoju raktu.
    - Programa privalo šifruoti kitą konfidencialią informaciją sąveikumo programos viešuoju raktu.
    - Programa privalo pasirašyti visą dokumentą.
- Kliento programa (siuntėjas) turi persiųsti šifruotą ir pasirašytą dokumentą sąveikumo sistemos programai.
- Kliento programa (gavėjas) pagal nustatytą logika turi iššifruoti gauto dokumento turinį.
    - Programa privalo patikrinti atsiųsto dokumento integralumą serverio viešuoju raktu.
    - Programa privalo iššifruoti kitą šifruotą konfidencialią informaciją gavėjo privačiuoju raktu.

[Sąveikumo sistemos programos]

- Serverio programa turi tikrinti, ar yra gauti verslo dokumentai iš klientinės programos (siuntėjo).
- Serverio programa, gavusi verslo dokumentą, papildo savo „įrašų“ bylą.
- Serverio programa gautus šifruotus verslo dokumentus po vieną iššifruoja.
- Serverio programa pagal nustatytą logiką turi iššifruoti nuskaityto verslo dokumento turinį.

    - Programa privalo patikrinti atsiųsto dokumento integralumą siuntėjo viešuoju raktu.
    - Programa privalo iššifruoti šifruotą konfidencialią informaciją serverio privačiuoju raktu.
- Serverio programa pagal nustatytą logika turi užšifruoti pakeisto verslo dokumento turinį.
    - Programa privalo šifruoti konfidencialią informaciją gavėjo viešuoju raktu.
    - Programa privalo pasirašyti visą dokumentą serverio privačiuoju raktu.
- Serverio programa pagal verslo dokumente pateiktą informaciją nusistato, kuriai klientinei programai siųsti apdorotą verslo dokumentą.
- Serverio programa persiunčia šifruotą ir pasirašytą verslo dokumentą gavėjui.
- Serverio programa privalo atlikus operaciją, jos būseną išsaugoti „įrašų“ byloje.

3.4.2. Nefunkciniai reikalavimai

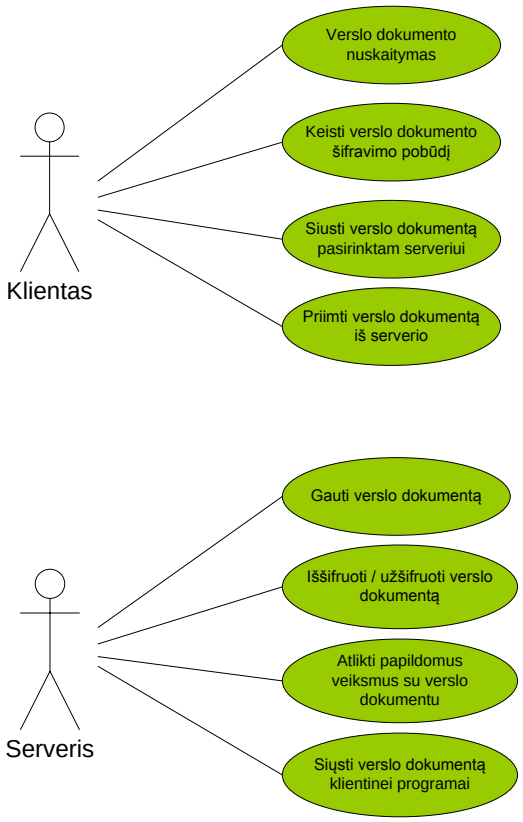
[Bendri]

- Sistemos programos negali leisti atlikti veiksmų, jeigu nėra naudojamas viešasis ir privatusis raktai.
- Sistemos programose turi būti galimybė pasirinkti nustatymus, pritaikytus konkrečiai aplinkai, t.y. nustatyti verslo dokumentų paėmimo saugyklą, sąveikumo sistemos tinklo adresą.
- Sistemos programos turi turėti galimybę išsaugoti nustatymus, kad nereiktų jų pakartotinai nustatinėti kiekvieną kartą.
- Projektuojamose programose turi būti kuo daugiau funkcijų automatizuota.

3.5. Informacinės posistemės projektas

Šiame skirsnyje pateikiamos panaudos ir sekų diagramos, padėsiančios suprasti, kaip veiks projektuojama programa. 16 pav. pateikta projektuojamo modelio panaudos diagrama.

3.5.1. Panaudos diagrama



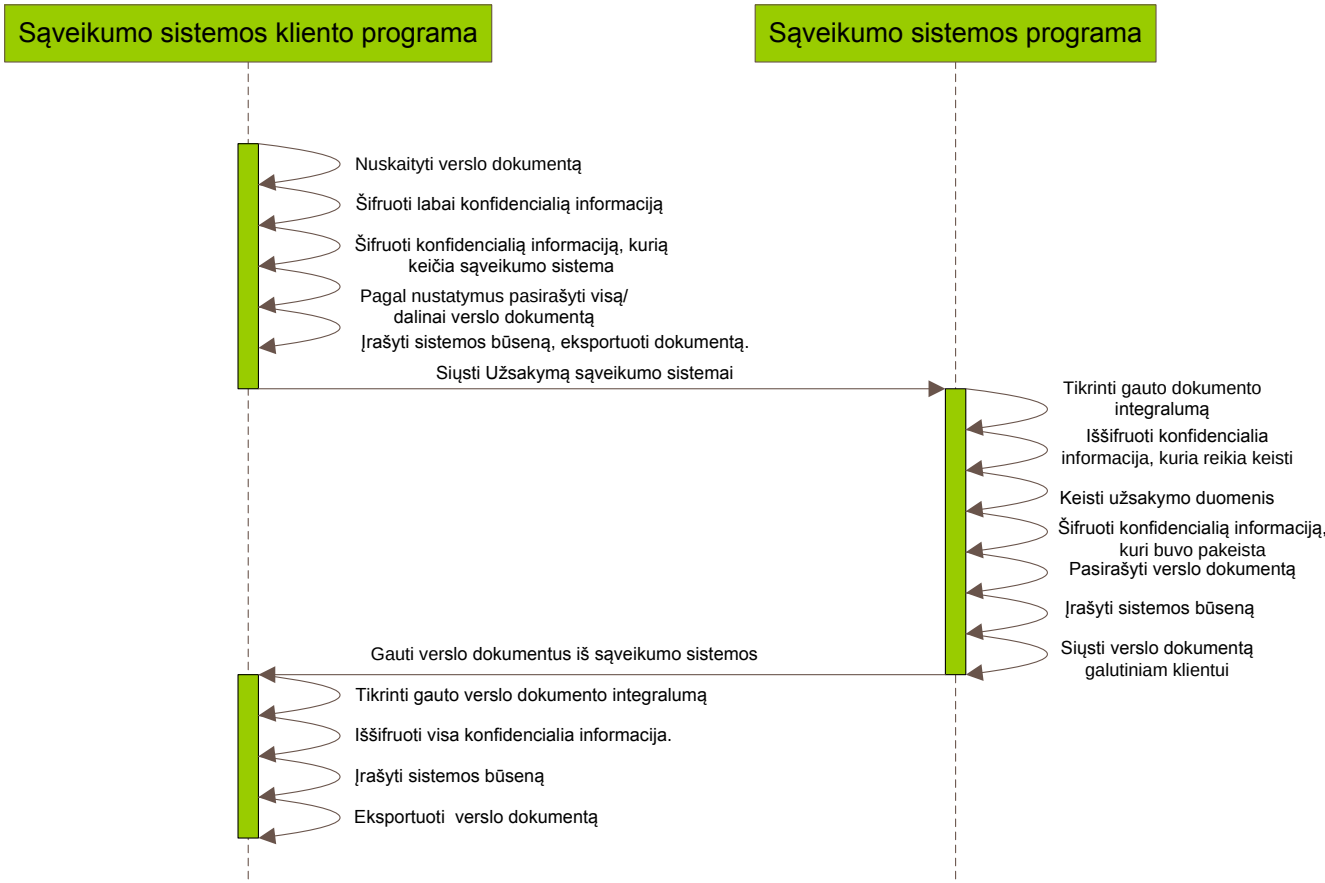
16 pav. Panaudos diagramos.

Kaip aprašyta reikalavimuose ir koncepcijoje, sąveikumo sistemos klientinės programos valdytojas galės: „nustatyti programos parametrus“, „tikrinti ar galima siųsti dokumentus“, t.y. ar sąveikumo sistema yra pasiekiama, „siųsti dokumentus sąveikumo sistemai“ bei „gauti dokumentus iš sąveikumo sistemos programos“. Į šiuos paskutinius dvejus atvejus įtraukiama ir tai, kad dokumentai programoje yra apdorojami.

Sąveikumo sistemos programa sugeba „gauti dokumentus iš savo klientinių programų“, „tikrinti, ar klientai yra prisijungę“ ir siųsti verslo dokumentus klientinei programai. Kadangi dauguma programos funkcijų yra automatizuota, šių panaudojimo atvejų pakanka atlikti tyrimui.

3.5.2. Sekų diagrama

17 pav. pateikta projektuojamo modelio sekų diagrama.



17 pav. Modeliavimo programų sekų diagrama

17 pav. yra pavaizduota sistemos sekų diagrama, kurios pagalba lengviau suprantama, kaip turėtų veikti programa. Sąveikumo sistemos klientinė programa šifruoja/pasirašo dokumentą, papildo „Įrašų“ būseną įrašais apie sistemos būseną ir, jeigu yra pasiekiama sąveikumo sistemos programa (serveris), išsiunčia pasirašytą dokumentą. Serverio programa priima verslo dokumentą, patikrina jo integralumą, pakeičia (arba nekeičia) reikiamą informaciją, užšifruoja dokumento nustatytus laukus ir pasirašo visą dokumentą, jeigu yra pasiekiama klientinė programa (gavėjas), pasiunčia jai pasirašytą verslo dokumentą bei įrašo savo būseną į „Įrašų“ bylą. Klientinė programa (gavėjas), gavusi verslo dokumentą iš sąveikumo sistemos programos, patikrina jo integralumą, iššifruoja visus šifruotus laukus ir taiko gautą verslo dokumentą savo reikmėm bei įrašo sistemos būsena į savąją „log“ bylą.

3.6. Dokumentų apsaugos kokybinė analizė

WS-Security apsaugos metodas apsaugo visą siunčiamą *SOAP* tipo žinutę. Nors apsaugos būdas saugus ir naudojamas realiose sistemose, tačiau apsaugant persiunčiamus dokumentus sąveikumo sistemose, kuriose sąveikumo serveris modifikuoja dokumento turinį, nėra tinkamas. WS-Security apsaugo siunčiamą verslo dokumentą nuo siuntėjo iki galutinio gavėjo, šiuo atveju tai būtų sąveikumo sistemos serveris, dėl ko dokumentas niekaip nepasaugomas tuo metu kai jis laikinai saugomas serveryje.

SET apsaugos metodas taip pat nėra naudojamas greitaveikos tyrimo scenarijuose dėl labai sudėtingo sistemos įgyvendinimo ir lėtos greitaveikos.

Naudojant SSL duomenų apsaugos metodą persiunčiami duomenys yra apsaugomi nuo siuntėjo iki gavėjo, tačiau negarantuojama apsauga sąveikumo sistemos serveryje.

VPN saugos metodu paremtame sąveikumo sistemos sprendime iškyla neišsiginamumo problemos. Persiunčiami duomenys nuo trečiųjų asmenų yra apsaugomi transporto lygmenyje. Saugant duomenis sąveikumo sistemos serveryje jie nėra papildomai apsaugoti nuo kylančių vidinių grėsmių.

Nors šie metodai pilnai neužtikrina saugos reikalavimų sąveikumo sistemoje, tačiau, jų pagalba, galima įvertinti naujai kuriamo saugos metodo greitaveikos rezultatus.

Kokybinės analizės rezultatai pateikti 1-oje lentelėje.

Apsaugos metodas	Viso dokumento apsauga persiunčiant	Dalies dokumento apsauga persiunčiant	Duomenų apsauga sąveikumo sistemos serveryje	Autentifikacija	Šifravimas
WS-Security	TAIP	NE	NE	TAIP	TAIP
SSL	TAIP	NE	NE	TAIP	TAIP
VPN	TAIP	NE	NE	TAIP	TAIP
SET	TAIP	NE	NE	TAIP	TAIP
Siūlomas metodas	TAIP	TAIP	TAIP	TAIP	TAIP

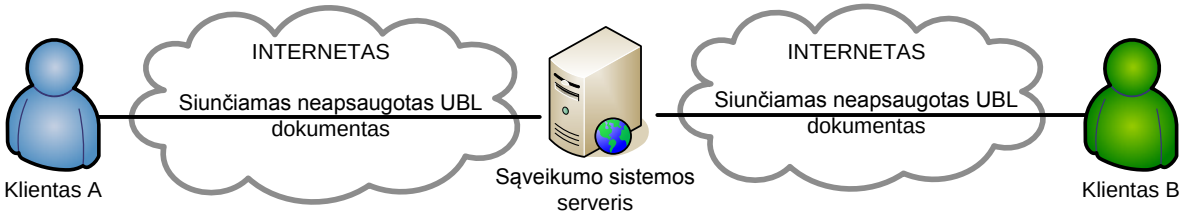
1 lentelė. Kokybinės analizės rezultatai.

3.7. Tyrimo scenarijai

Tyrimo scenarijai yra atliekami pritaikant skirtingus apsaugos metodus (pritaikytą metodą, bei SSL ir VPN technologijas) ir gaunami rezultatai:

- Siunčiant užsakymą nešifruojant

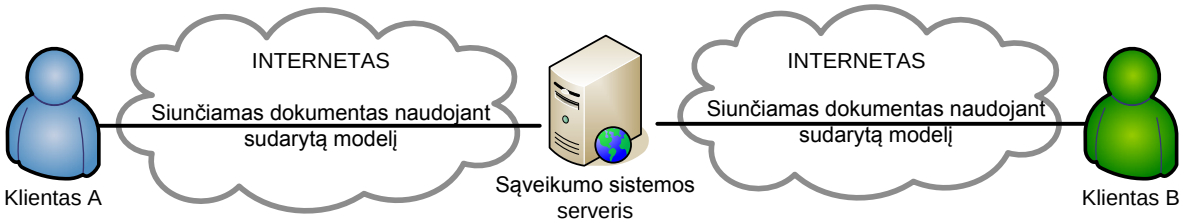
Šio scenarijaus pagalba gaunami užsakymo dokumento apdorojimo laikai sistemoje, matuojami laikai, kiek yra užtrunkama, kol neapsaugotas dokumentas pasiekia galutinį vartotoją. Pagrindinis šio scenarijaus tikslas – gauti atskaitos tašką, su kuriuo būtų galima lyginti duomenis, gautus taikant numatytus apsaugos metodus. Šio scenarijaus koncepcija pateikta 18 pav.



18 pav. Siunčiamas neapsaugotas dokumentas

- Siunčiant užsakymą, pritaikant sukurtą metodą

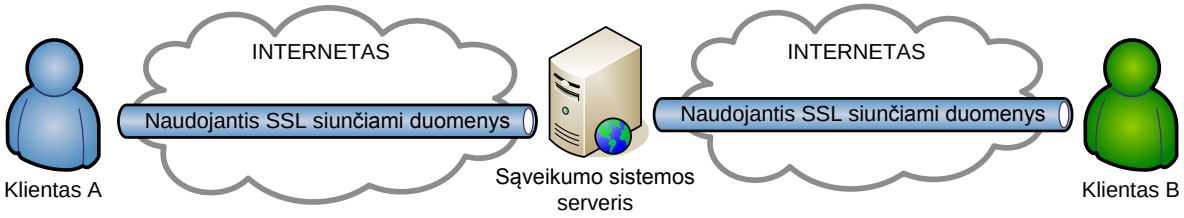
Simuliuojant šį scenarijų yra parenkami keli būdai, tačiau viso metodo principas išlieka tas pats. Pirmuoju atveju siunčiamas dalinai šifruotas verslo dokumentas, kuriame šifruojami tik nurodyti laukai, o dokumentas pasirašomas visas. Antruoju atveju galutiniam vartotojui per sąveikumo sistemą yra siunčiamas visas šifruotas ir pasirašytas dokumentas. Šių scenarijų koncepcija pateikta 19 pav.



19 pav. Siunčiamas dokumentas apsaugotas sudarytąja metodika

- Siunčiant užsakymą, naudojant SSL protokolą

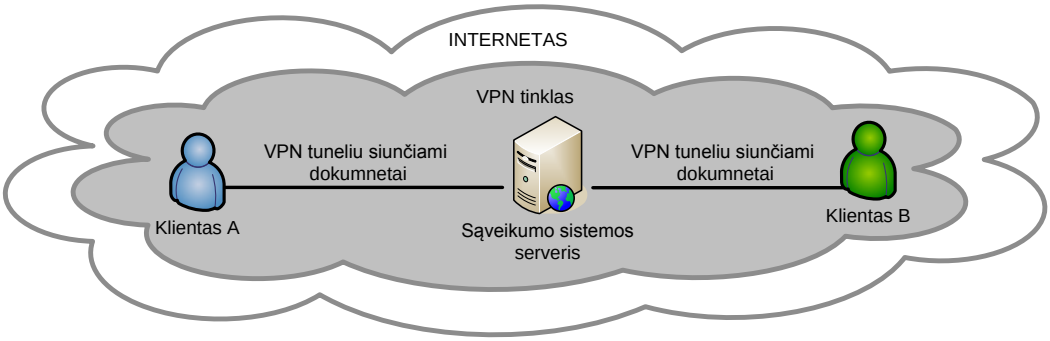
Naudojantis SSL protokolu pasirinktas užsakymo dokumentas siunčiamas iki serverio, kuriame atliekami atitinkami veiksmai su šiuo dokumentu ir jis toliau persiunčiamas galutiniam gavėjui. Šio scenarijaus koncepcija pateikta 20 pav.



20 pav. Siunčiamas užsakymo dokumentas naudojant SSL protokolą

- Siunčiant nešifruotą užsakymą, pasijungus VPN

Šiame bandyme gaunami rezultatai sąveikumo sistemos dalyvius sujungus į privatų tinklą, ir siunčiant neapsaugotą verslo dokumentą tarp jų per sąveikumo sistemos serverį. Taip pat kaip ir kituose scenarijuose fiksuojami laikai reikalingi duomenų analizei. Šio scenarijaus koncepcija pateikta 21 pav.



21 pav. Siunčiamas dokumentas pasinaudojant VPN paslauga.

Eksperimentu fiksuojami duomenys yra surašomi į lenteles ir atvaizduojami diagramomis (pateikta sekančiame skyrelyje).

3.8. Tyrimo eiga

Atliekamo eksperimento bendroji veiksmų seka sukurtiems verslo dokumentų perdavimo scenarijams:

1) Paleidžiamos programos, dvejuose kompiuteriuose, kurie yra ne viename tinklo segmente (prieš naudojant turi būti galimybė nustatyti nutolusio kompiuterio tinklo adresą).
2) Įkeliamas verslo dokumentas (užsakymas) į saugyklą, iš kurios klientinė programa paima jį (saugykla yra nustatoma kliento programos nustatymuose).
3) Rankiniu būdu paleidžiama sąveikumo sistemos klientinė programos (siuntėjo) funkcija, kuri atlieka dokumento šifravimą ir siunčia jį serverio programai (kitas kompiuteris).
4) Serverio programa gauna dokumentus iš klientinės programos (siuntėjo) automatiškai.
5) Serverio programa atlieka reikiamus veiksmus ir persiunčia dokumentą klientinei programai (gavėjui) automatiškai, kai tik gavėjas yra pasiekiamas.

6) Sąveikumo sistemos klientinė programa (gavėjas) gauna dokumentus iš sąveikumo sistemos.
7) Klientinė programa (gavėjas) juos apdoroja ir padeda iš anksto numatytoje ir nustatytoje saugykloje.
8) Tyrimai kartojami su nešifruotu dokumentu ir kaupiami rezultatai.
9) Technologijų greitaveikai analizuoti ir palyginimui paimami sistemų būsenų įrašų dokumentai.

4. Eksperimentas

4.1. Tyrimo aplinka

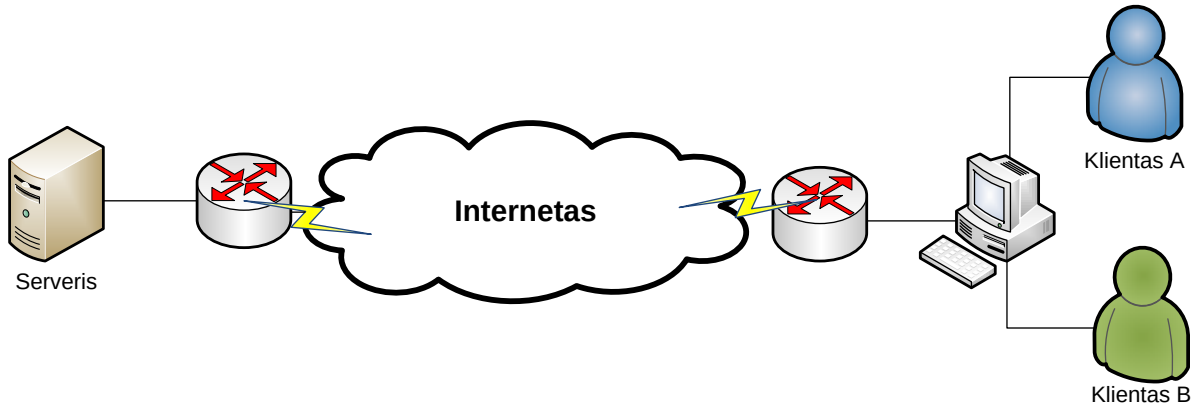
Atliekant tyrimą buvo naudotasi keliais kompiuteriais, esančiais skirtingose vietose. Šiuose kompiuteriuose buvo įrašyta programinė įranga, kurios pagalba sumodeliuoti užsakymo dokumento siuntimo scenarijai.

Naudota techninė įranga:

	Serveris	**Klientas A ir Klientas B**
Operacinė sistema	Windows Server 2008 x64	Windows 7 x64
Procesorius	Intel® Xeon® CPU E5420 2.5GHz	Pentium® Dual-Core T4300 2.1Ghz
Operatyvioji atmintis:	1.5Gb	4.0Gb

2 lentelė. techninė įranga.

Tyrimui naudojama tinklo schema:



22 pav. Tinklo schema naudojama tyrime.

4.2. Programinė įranga

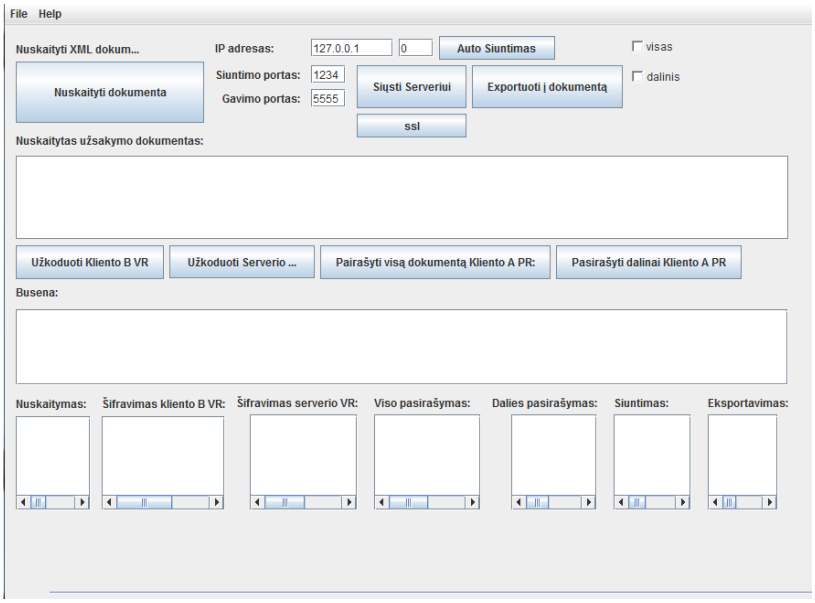
Suprojektuota ir parengta programinė įranga, skirta tirti verslo dokumentų persiuntimui. Šios programos naudojamos gauti tyrimui reikalingus duomenis. Programinė įranga sudaryta iš trijų programų: *Klientas A, Serveris, Klientas B*. Sumodeliuotos programos realizuotos *Java* programavimo kalba, naudojant NetBeans IDE 7.0 redaktorių.

4.2.1. „Klientas A"

Parengta sudarytojo metodo programa „Klientas A“ (*vartotojo sąsajos atvaizdavimas 23 pav.*) sugebanti:

- Nuskaityti iš dokumentų saugyklos UBL užsakymo dokumentą.

- Nustatyti prievado numerius ir sąveikumo sistemos serverį, kuriam bus siunčiamas dokumentas.
- Užšifruoti numatytąsias dokumento dalis gavėjo sistemos viešuoju raktu.
- Užšifruoti numatytąsias dokumento dalis serverio sistemos viešuoju raktu.
- Užšifruoti visą persiunčiamą dokumentą serverio viešuoju raktu.
- Pasirašyti visą užsakymo dokumentą privačiuoju raktu.
- Numatyti automatinį užsakymo dokumentų siuntimą serveriui, n-kartų (siuntimo kartai nurodomi programos vartotojo sąsajoje).
- Išsaugoti numatytoje saugykloje verslo dokumentą.
- Siųsti serveriui UBL užsakymo dokumentą.
- Atvaizduojanti visų atliekamų veiksmų būsenas ir operacijoms sugaištus laikus.
- Siųsti verslo dokumentą serveriui naudojant SSL.



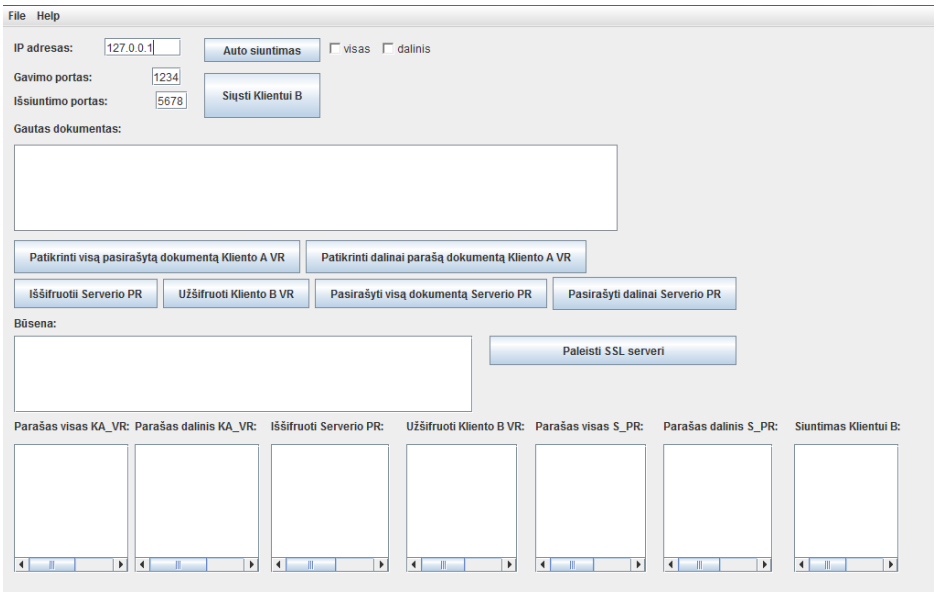
23 pav. Kliento A vartotojo sąsaja

4.2.2. „Serveris“

Parengta sudarytojo metodo serveryje veikianti programa (*vartotojo sąsajos atvaizdavimas 24 pav.*). Programoje numatomi duomenų įėjimo, išėjimo prievadų numeriai, kurias serveris gauna ar išsiunčia duomenis. Ši programa gali atlikti šias operacijas:

- Gautą užsakymo dokumentą atvaizduoti vartotojo sąsajoje.
- Patikrinti gauto užsakymo dokumento parašą
    - Viso dokumento parašą, jei buvo pasirašytas visas dokumentas
    - Dalinai pasirašyto dokumento parašą, jei dokumentas buvo pasirašytas dalinai.
- Išsišifruoti konfidencialią informaciją, skirtą serveriui.

- Užšifruoti konfidencialią dokumento informaciją gavėjo viešuoju raktu.
- Pasirašyti visą užsakymo dokumentą.
- Dalinai pasirašyti verslo dokumentą (tik numatytuosius laukus).
- Numatyti automatinį užsakymo dokumentų siuntimą gavėjui, n-kartų (siuntimo kartai nurodomi programos siuntėjo-programos vartotojo sąsajoje).
- Siųsti gavėjui UBL užsakymo dokumentą.
- Atvaizduoti visų atliekamų veiksmų būsenas ir operacijoms sugaištus laikus.
- Paleisti SSL serverį (laukti SSL metodu siunčiamų verslo dokumentų).



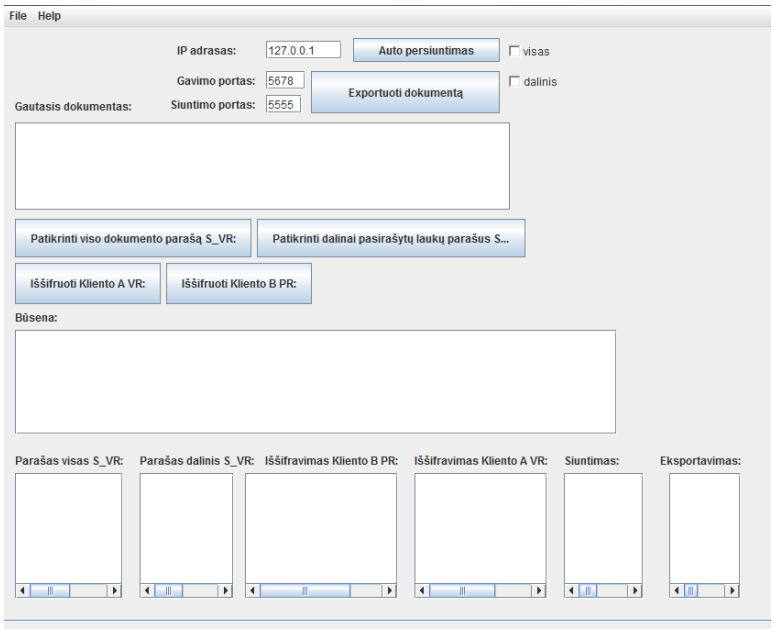
24 pav. Serverio vartotojo sąsaja

4.2.3. „Klientas B“

Parengta sudaryto metodo gavėjo pusėje veikianti programa (*vartotojo sąsajos atvaizdavimas 25 pav.*). Ši programa yra siuntėjo programos modifikacija, pritaikyta veikti kaip gavėjas („*Klientas B*“). Programoje numatomi duomenų įėjimo, išėjimo prievadų numeriai, kurias gavėjo pusėje veikianti programa gauna duomenis ir, atlikus visus veiksmus, išsiunčia siuntėjo programai patvirtinimą, reikalingą apskaičiuoti visą užsakymo dokumento sugaištą laiką, keliaujant nuo vienos informacinės sistemos iki kitos. Gavėjo pusėje veikiančios programos atliekami veiksmai:

- Gautą užsakymo dokumentą atvaizduoti vartotojo sąsajoje.
- Patikrinti gauto užsakymo dokumento parašą
    - Viso dokumento parašą, jei buvo pasirašytas visas dokumentas
    - Dalinai pasirašyto dokumento parašą, jei dokumentas buvo pasirašytas dalinai.
- Patikrinti parašą siuntėjo viešuoju raktu, jei dokumentas buvo pasirašytas siuntėjo raktu.

- Išsišifruoti pasirašytus duomenis privačiuoju raktu.
- Numatyti automatinį užsakymo dokumentų patvirtinimo siuntimą n-kartų (siuntimo kartai nurodomi programos siuntėjo-programos vartotojo sąsajoje).
- Išsaugoti saugykloje gautą verslo dokumentą.
- Atvaizduoti visų atliekamų veiksmų būsenas ir operacijoms sugaištus laikus.



25 pav. Kliento B vartotojo sąsaja

4.3. Eksperimento eiga

Atliekama eksperimento eiga, kurios rezultatas – gauti duomenys iš programinės įrangos:

1. Paleidžiamos programos siuntėjo/gavėjo sistemoje ir serveryje.
2. Nustatomi IP adresai ir prievadų numeriai, naudojami programų bendravimui.
3. Nustatomas dokumentų kiekis, kurį norima siųsti.
4. Vykdomas programos ciklas, šifruojami ir pasirašomi užsakymo duomenys, automatiškai persiunčiami kitai programai.
5. Surenkami rezultatai iš programų.

4.4. Eksperimento tikslai

Programų dėka yra surenkami rezultatai, užsibrėžtiems parametrams matuoti:

Matavimo aspektai:

1. Užsakymo dokumento siuntimo laikas (nuo siuntėjo iki gavėjo informacinės sistemos).
2. Skirtingo dydžio užsakymo dokumentų siuntimas, naudojantis metodu.
3. Pagal numatytus eksperimento scenarijus projektavimo dalyje parenkami dokumentų dydžiai:

- Užsakymo dokumento siuntimas per SSL
    - Siunčiamų dokumentų kiekis (10 ir 50). (Tyrimo programoje nurodomas siuntimų ciklų skaičius, siekiant gauti tikslesnius rezultatus).
    - Siunčiamų dokumentų dydis (3Kb, 10Kb, 100Kb). (Tyrimo programoje naudojami dokumentai, pagal kuriuos bus gaunami tyrimo rezultatai).

4. Kiekvienoje programos dalyje surenkami rezultatai: nuskaitymo, siuntimo, šifravimo, pasirašymo, iššifravimo, parašo patikrinimo laikai.
5. Tiriant šifravimo, pasirašymo laikus naudojami (3Kb, 10Kb, 20Kb, 30Kb, …, 100Kb) dydžio dokumentai.

4.5. Eksperimentu gauti duomenys

Pagal numatytus eksperimento tikslus 3-5 lentelėse pateikiami eksperimentų duomenys. Kiekvienas įrašas lentelėje yra penkiasdešimties siuntimų vidurkis. Gavėjo programos duomenų lentelėje taip pat pateikiamas laikas, sugaištas persiųsti vieną UBL užsakymo dokumentą nuo siuntėjo sistemos iki gavėjo.

Visose lentelėse duomenys yra pateikiami sekundėmis. Lentelėse pirmiausia yra nurodomas užsakymo dokumento siuntimo būdas (pvz., „Taikant metodą siunčiamas visas pasirašytas užsakymo dokumentas“) ir siunčiamas užsakymo dokumento dydis (pvz., „10Kb“). Kliento „A“ programa atlieka šiuos veiksmus: nuskaito užsakymo dokumentą (0.09949s), Šifruoja duomenis kliento “B“ viešuoju raktu (0.027047s), šifruoja duomenis sąveikumo sistemos serverio viešuoju raktu (0.008858s), pasirašo visą siunčiamą dokumentą (0.22050s), siunčia jį sąveikumo sistemos serveriui (0.01398s), patalpina dokumentą archyvavimui (0.010128s). Laukai pažymėti (*) simboliu reiškia, kad šie veiksmai nėra galimi atlikti numatytuoju apsaugos metoduose.

2-oje lentelėje yra pateikiami sąveikumo sistemos serveryje sugaišti laikai. Pasirenkamas siuntimo būdas ir dokumento dydis kaip ir pirmoje lentelėje. Serverio programa, gavusi dokumentą, patikrina viso dokumento parašą (*0.09949s)*, išsišifruoja norimus keisti duomenis *(0.02832s)*, užšifruoja pakeistus duomenis Kliento „B“ viešuoju raktu *(0.02231s*), vėl pasirašo visą užsakymo dokumentą *(0.0322s*) ir pasiunčia jį Klientui „B“ (*0.01452s*).

3-oje lentelėje pateikiami Kliento „B“ programos atliekami veiksmai ir jiems sugaišti laikai. Pasirenkamas siuntimo būdas ir dokumento dydis kaip ir pirmoje lentelėje. Klientas „B“, gavęs verslo dokumentą, patikrina viso dokumento parašą (*0.09731s*), išsišifruoja užšifruotus laukus (*0.03314s*), išsaugo dokumentą į saugyklą (*0.24875s*), nusiunčia patvirtinimo signalą pirmajam klientui, kad siuntimas baigtas (*0.0069s*). Paskutinis stulpelis šioje lentelėje nurodo, kiek laiko buvo sugaišta nuo siuntėjo sistemos iki gavėjo sistemos (*0.85133s*).

Dokumento dydis	Dokumento nuskaitymas	Duomenų šifravimas Kliento B viešuoju raktu	Duomenų šifravimas Serverio viešuoju raktu	Viso dokumento pasirašymas	Dokumento dalies pasirašymas	Dokumento siuntimas	Dokumento eksportavimas
Taikant metodą siunčiamas visas pasirašytas užsakymo dokumentas							
3Kb	0.09224	0.027943	0.010437	0.08576	*	0.00464	0.007179
10Kb	0.09949	0.027047	0.008858	0.22050	*	0.01398	0.010128
30Kb	0.13365	0.023621	0.007908	0.55877	*	0.04094	0.017308
100Kb	0.09114	0.035834	0.008746	1.92913	*	0.08593	0.029803
Taikant metodą siunčiamas dalinai pasirašytas užsakymo dokumentas							
3Kb	0.094343	0.024088	0.009656	*	0.007969	0.00599	0.00733
10Kb	0.104904	0.049891	0.008764	*	0.007749	0.0148	0.00889
30Kb	0.115217	0.090044	0.015014	*	0.009334	0.04719	0.00962
100Kb	0.092875	0.093106	0.009622	*	0.007894	0.09827	0.01240
Siunčiamas užsakymo dokumentas nėra pasirašomas ir šifruojamas							
3Kb	0.097355	*	*	*	*	0.00558	0.00931
10Kb	0.105352	*	*	*	*	0.01937	0.01237
30Kb	0.136650	*	*	*	*	0.02665	0.01306
100Kb	0.101643	*	*	*	*	0.06157	0.01578
Siunčiamas užsakymo dokumentas naudojant VPN							
3Kb	0.093614	*	*	*	*	0.00595	0.00592
10Kb	0.109774	*	*	*	*	0.01227	0.01263
30Kb	0.295309	*	*	*	*	0.03136	0.01410
100Kb	0.100923	*	*	*	*	0.06296	0.01734
Siunčiamas užsakymo dokumentas naudojant SSL							
3Kb	0.0163	*	*	*	*	0.03224	*
10Kb	0.0631	*	*	*	*	0.10863	*
30Kb	0.1821	*	*	*	*	0.20049	*
100Kb	1.7310	*	*	*	*	0.35165	*

3 lentelė. Klientas A apdoroja ir perduoda užsakymo dokumentą Serverio programai

Dokumento dydis	Viso gauto dokumento parašo patikrinimas	Dalinai pasirašyto dokumento parašo patikrinimas	Užšifruotų duomenų iššifravimas skirtų keisti	Duomenų šifravimas Kliento B viešuoju raktu	Viso užsakymo dokumento pasirašymas	Pasirašymas tik numatytų laukų	Dokumento siuntimas Klientui B
Taikant metodą siunčiamas visas pasirašytas užsakymo dokumentas							
3Kb	0.06034	*	0.01732	0.01271	0.0164	*	0.00582
10Kb	0.07765	*	0.02832	0.02231	0.0322	*	0.01452
30Kb	0.14322	*	0.04421	0.05624	0.0792	*	0.02209
100Kb	0.29064	*	0.11533	0.1735	0.2651	*	0.05886
Taikant metodą siunčiamas dalinai pasirašytas užsakymo dokumentas							
3Kb	*	0.047196	0.02528	0.0114	*	0.0049	0.00488
10Kb	*	0.052007	0.02775	0.02115	*	0.005	0.01321
30Kb	*	0.072529	0.05855	0.05518	*	0.0062	0.0219
100Kb	*	0.048641	0.11076	0.17768	*	0.0108	0.05031
Siunčiamas užsakymo dokumentas nėra pasirašomas ir šifruojamas							
3Kb	*	*	*	*	*	*	0.00691
10Kb	*	*	*	*	*	*	0.01537
30Kb	*	*	*	*	*	*	0.02066
100Kb	*	*	*	*	*	*	0.05967
Siunčiamas užsakymo dokumentas naudojant VPN							
3Kb	*	*	*	*	*	*	0.00809
10Kb	*	*	*	*	*	*	0.02455
30Kb	*	*	*	*	*	*	0.04357
100Kb	*	*	*	*	*	*	0.09466

4 lentelė. Serveris apdoroja ir perduoda užsakymo dokumentą Kliento B programa

Dokumento dydis	Viso dokumento parašo patikrinimas	Dalinai pasirašyto dokumento iššifravimas	Užšifruotų duomenų iššifravimas Kliento B privačiuoju raktu	Užsakymo dokumento eksportavimas	Siuntimo patvirtinimas	Užtruktas laikas
Taikant metodą siunčiamas visas pasirašytas užsakymo dokumentas						
3Kb	0.11889	*	0.03298	0.01782	0.0095	0.49503
10Kb	0.09731	*	0.03314	0.24875	0.0069	0.85133
30Kb	0.20019	*	0.05356	0.76369	0.0069	2.13156
100Kb	1.93695	*	0.16433	2.54771	0.0073	7.70578
Taikant metodą siunčiamas dalinai pasirašytas užsakymo dokumentas						
3Kb	*	0.04711	0.03354	0.02122	0.00806	0.353
10Kb	*	0.05372	0.03777	0.16028	0.00742	0.5733
30Kb	*	0.08379	0.10338	0.71726	0.0123	1.4539
100Kb	*	0.05003	0.14596	2.72871	0.00788	3.6095
Siunčiamas užsakymo dokumentas nėra pasirašomas ir šifruojamas						
3Kb	*	*	*	0.01652	0.00192	0.13027
10Kb	*	*	*	0.16438	0.00308	0.68702
30Kb	*	*	*	0.55832	0.00362	0.96436
100Kb	*	*	*	2.2824	0.00182	2.47788
Siunčiamas užsakymo dokumentas naudojant VPN						
3Kb	*	*	*	0.0144	0.00221	0.15762
10Kb	*	*	*	0.1936	0.00259	0.42993
30Kb	*	*	*	0.5975	0.00244	0.75598
100Kb	*	*	*	2.5999	0.00204	3.1587

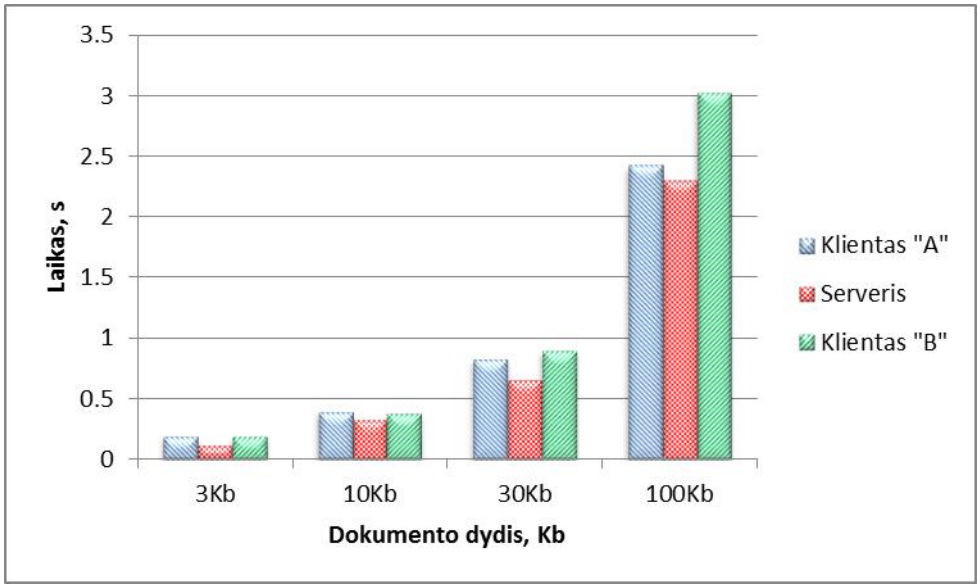
5 lentelė. Klientas B apdoroja užsakymo dokumentą.

4.6. Eksperimentinių tyrimų rezultatų analizė

4.6.1. Metodo etapų efektyvumo palyginamasis tyrimas

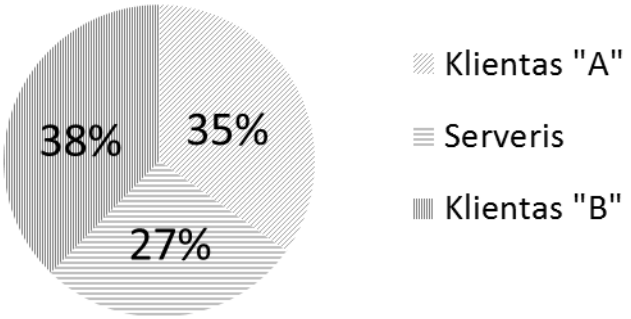
Greitaveikos tyrimui buvo naudota programa, aprašyta ankstesniuose skyriuose. Su šia programa buvo išmatuota greitaveika penkiuose dokumentų persiuntimo metoduose, naudojant skirtingų dydžių užsakymo dokumentus. Šie dokumentai tarp visų procese dalyvaujančių programų buvo siunčiami 100 kartų ir jų vidurkis patalpintas 1-3 lentelėse. Pagal gautus rezultatus galima palyginti šiuos kelis metodus greitaveikos požiūriu.

26 pav. Diagramoje galima matyti laiko skirtumus sugaištus klientinėje ir serverio dalyse.



26 pav. Dokumentų apdorojimo laikas sąveikumo sistemoje

Kaip matyti siunčiant dokumentus laiko sąnaudos kiekvienoje programoje yra proporcingos dokumento dydžiui. Atsižvelgiant į tai, kad serverio dalyje dokumentas nebuvo saugomas, o tiesiai persiunčiamas kitam klientui, galima teigti, kad resursų sąnaudos yra mažesnės negu būtų realiai veikiančioje sistemoje. Atsižvelgus į laiko suvartojimą (pateikta skritulinėje diagramoje 27 pav.) nustatyta, kad naudojant šį metodą reikalinga įranga serverio dalyje dalyse turi turėti pakankamai resursų atlikti reikiamus veiksmus, kai sąveikumo sistemoje sąveikauja daugelis įmonių.

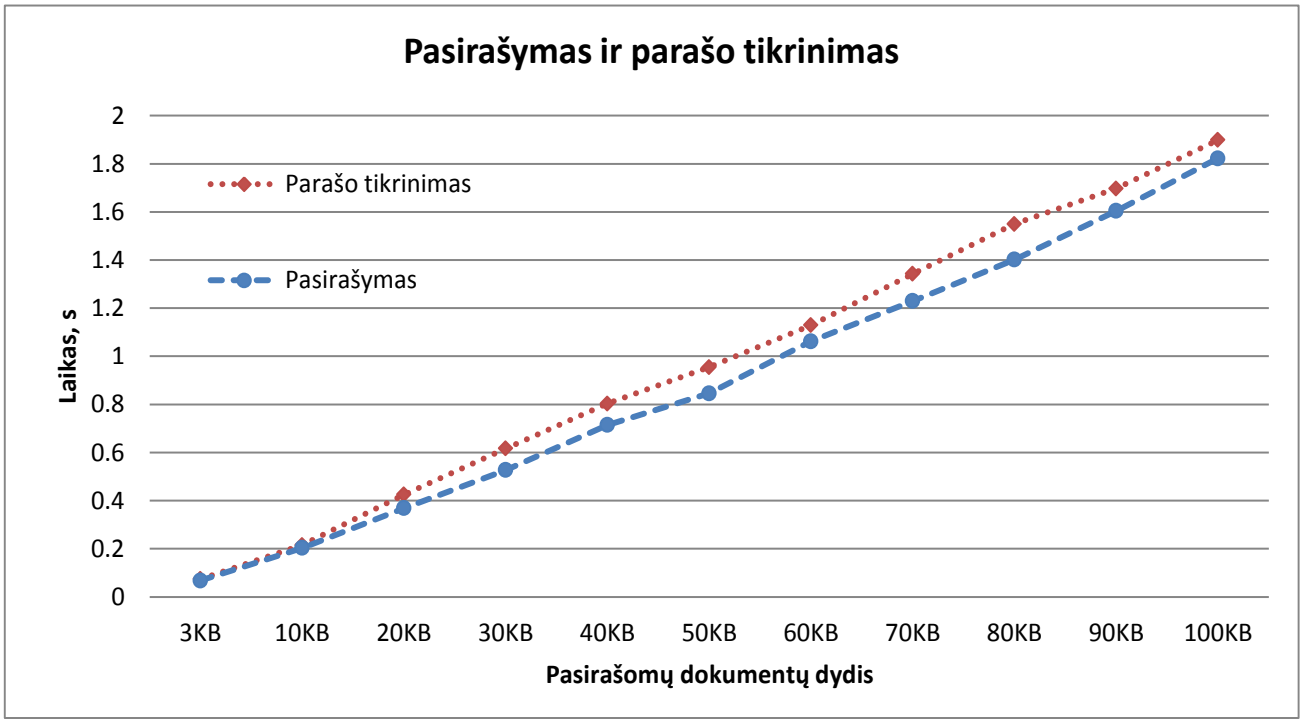


27 pav. Laiko požiūriu apdorojimo laikas sistemos programose 30Kb dokumentams.

Serverio pusėje visi skaičiavimai taip pat, kaip ir klientinėje, priklauso nuo serverio skaičiavimo pajėgumų. Atsižvelgiant į tai, kad sąveikumo sistemoje yra daugybė klientų, galima daryti išvadą, jog reikalingas ir pakankamai galingas serveris.

4.6.2. Dokumentų pasirašymo efektyvumo tyrimas

Tiriant sumodeliuotąjį metodą taip pat atsižvelgiama į pasirašymo / parašo tikrinimo laiką, skirtą apsaugoti siunčiamam užsakymo dokumentui.



28 pav. Pasirašymo greitis „Klientas A“ programoj. Parašo tikrinimas Klientas B programoje.

Dokumento dydis, Kb	Dokumento pasirašymas, s	Parašo patikrinimas, s
3Kb	0.0683	0.0745
10Kb	0.2035	0.2156
20Kb	0.3689	0.4256
30Kb	0.5271	0.6168
40Kb	0.7148	0.8034
50Kb	0.8458	0.9549
60Kb	1.0626	1.1293
70Kb	1.2297	1.3425
80Kb	1.4015	1.5499
90Kb	1.6041	1.6970
100Kb	1.8223	1.8986

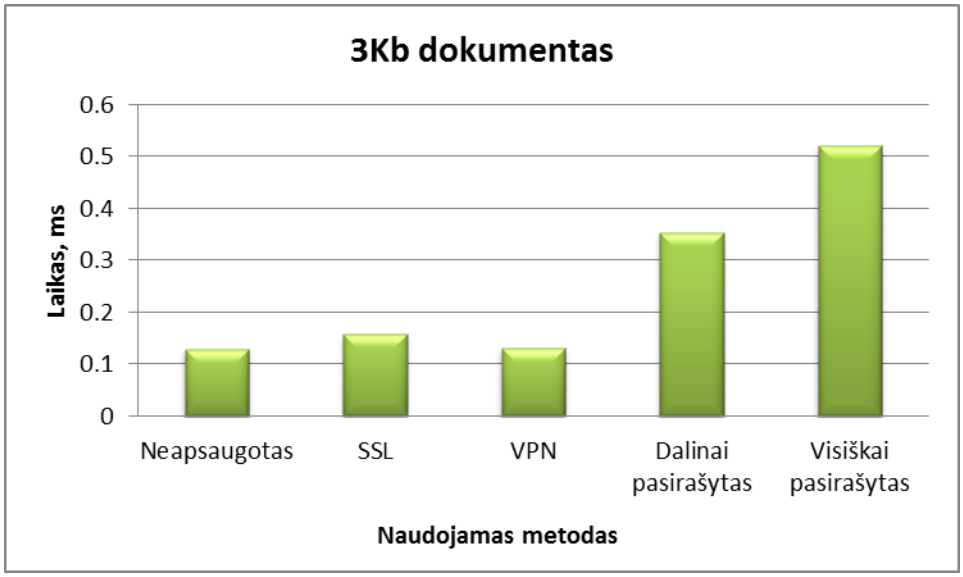
6 lentelė. Pasirašymo ir parašo tikrinimo laikai

Atsižvelgiant į rezultatus, pateiktus šioje diagramoje, matoma, kad pasirašymo laikas, pasirašant visą dokumentą, yra tiesiškai proporcingas dokumento dydžiui. Pagal šį parametrą galime pasakyti, kad pasirašinėti visą dokumentą yra neefektyvu, siunčiant didesnius dokumentus. Patartina naudoti dalinį parašą tik numatytiesiems laukams, kurie negali būti pakeisti. Pasirašinėjant didelių dokumentų būtinus laukus sutaupoma laiko ir efektyviau išnaudojamas metodas.

4.6.3. Metodo greitaveikos tyrimas

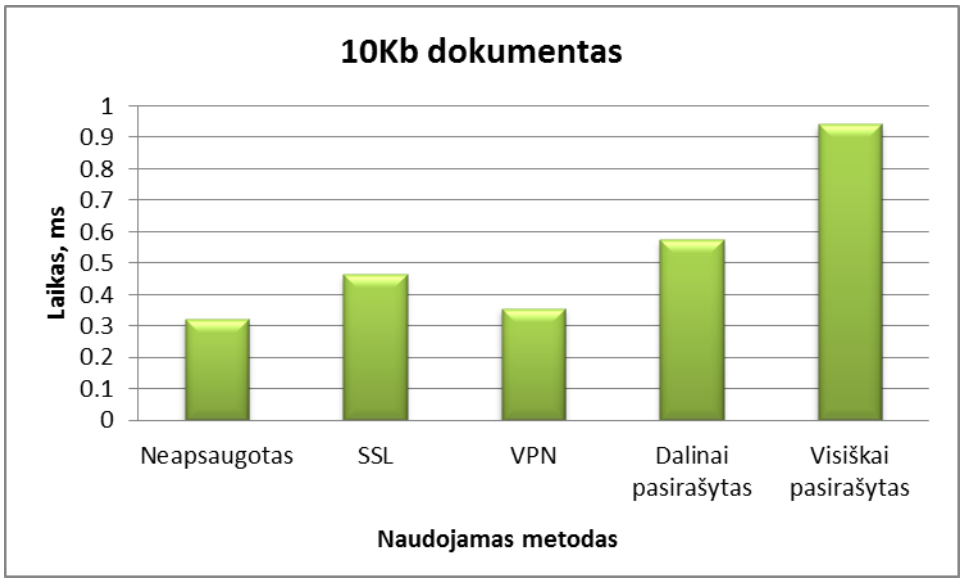
Vienas svarbiausių faktorių yra tai, kaip šių trijų metodų greitaveika atsispindi visoje persiuntimo grandinėje. Šio eksperimento diagramos pateiktos 28, 29, 30, 31 paveikslėliuose. Šiose diagramose matyti ryškus skirtumas tarp sukurtojo metodo lankstumo, kuris leidžia pasirašyti visą dokumentą, kas būtų neefektyvu, naudojant didelius užsakymo dokumentus sąveikumo sistemose, tačiau metodas numato ir dalinį siunčiamų dokumentų pasirašymą, kurio pagalba sutaupoma laiko persiunčiant dokumentus.

Naudojant 3KB dokumentą, pastebimas greitaveikos skirtumas tarp tiriamųjų metodų laiko sąnaudose. Kaip ir tikėtasi, neapsaugotas dokumentas nukeliauja visa kelią greičiausiai. Iš atliktų bandymų taip pat matyti, kad VPN paslauga siunčiamo dokumento laikas yra labai panašus kaip ir siunčiant neapsaugotą dokumentą SSL metodu. Standartiniais metodais siunčiamas verslo dokumentas sunaudoja dvigubai mažiau laiko nei sumodeliuotu metodu siunčiamas dalinai ar visai pasirašomas užsakymo dokumentas. 3KB dokumento siuntimo diagrama pavaizduota 29 pav.



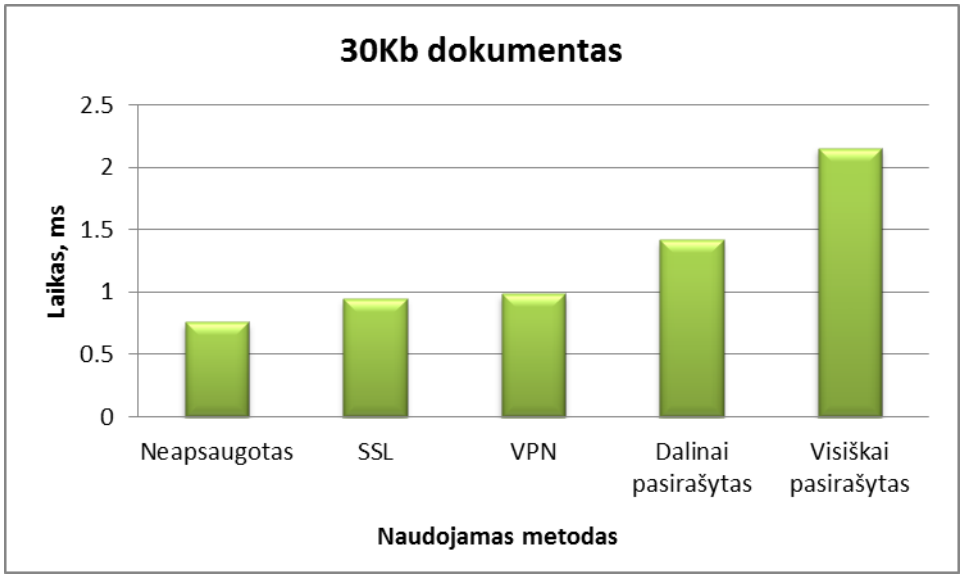
29 pav. dokumento kelias sugaištas nuo siuntėjo programos iki gavėjo naudojant 3KB dokumentą

Tiriant 10KB užsakymo dokumento siuntimo scenarijus pastebėta, kad nuo siuntimo laikas pasirašant visą dokumentą išaugo proporcingai dokumento dydžiui. Dalinai pasirašyto dokumento siuntimo laikas labiau susilygina su standartinių metodų siuntimo laikais. Diagrama pateikta 30 pav.



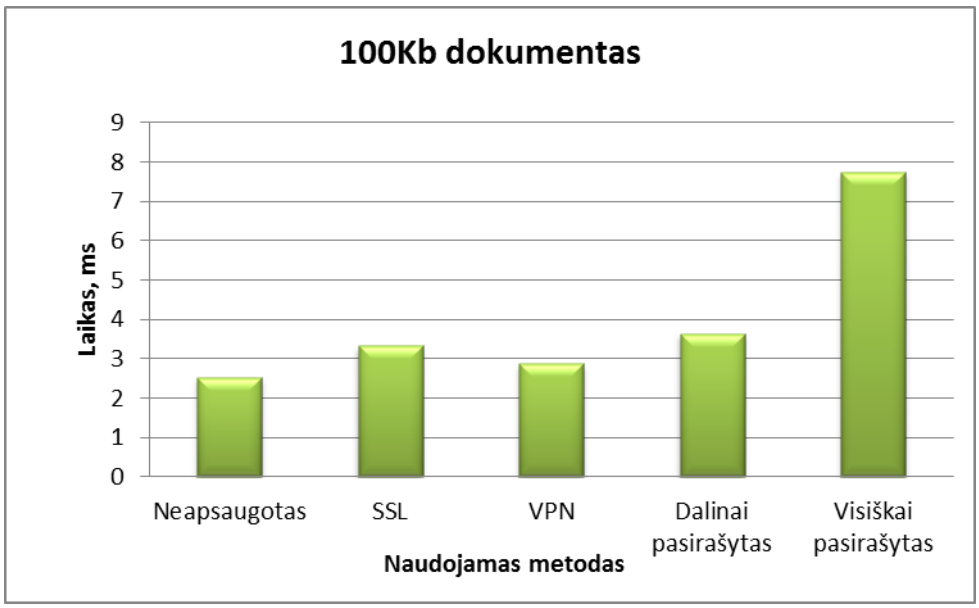
30 pav. dokumento kelias sugaištas nuo siuntėjo programos iki gavėjo naudojant 10KB dokumentą

Stebint 30Kb dokumento siuntimo laiko sąnaudų diagramą matyti, kad visi metodai sunaudoja laiko proporcingai ankstesniems tyrimams. Čia pastebima, kad VPN siunčiami duomenys pasiekia galutinį klientą lėčiau negu SSL. Naudojant metodą dalinai pasirašytam dokumentui matyti, kad jis yra <50% lėtesnis negu standartinės apsaugos priemonės.



31 pav. dokumento kelias sugaištas nuo siuntėjo programos iki gavėjo naudojant 30KB dokumentą

Stebint 100KB užsakymo dokumento siuntimo galutiniam vartotojui diagramą matoma, kad siūlomasis metodas, pasirašant visą dokumentą, suvartoja akivaizdžiai daugiau laiko greitaveikos požiūriu nei visi kiti metodai. Tačiau siunčiant dokumentą, panaudojant dalinį parašą, laiko sąnaudos susilygina su standartinėmis apsaugos priemonėmis ir yra lėtesnės <10% standartinių apsaugos priemonių suvartojimo laiko. 100KB dokumento siuntimo diagrama pavaizduota 31 pav.



32 *pav. dokumento kelias sugaištas nuo siuntėjo programos iki gavėjo naudojant 100KB dokumentą*

5. Išvados

1. Persiunčiant verslo dokumentus sąveikumo sistemose, kuriose sąveikumo sistemos serveris modifikuoja siunčiamus duomenis, kyla problema apsaugoti šių dokumentų konfidencialią informaciją ne tik perduodant dokumentą nesaugiomis ryšio linijomis, bet ir saugant jį sąveikumo sistemos serveryje.

2. Išanalizavus sąveikumo sprendimus, skirtus sąveikumo sistemoms, nustatyta, kad dauguma šių sprendimų yra vystymo ir plėtojimo stadijose ir šiuose sprendimuose nėra įdiegta pakankamai priemonių skirtų verslų dokumentų saugai užtikrinti.

3. Apžvelgus galimus dokumentų saugos užtikrinimo būdus (SSL, VPN, WS-Security, SET) pastebėta, kad norint apsaugoti siunčiamą dokumentą visą jo kelią standartiniai būdai nėra visiškai tinkami problemos sprendimui. Norint saugiai persiųsti dokumentą reikia naudoti apjungtas kelias standartines apsaugos priemones.

4. Darbe pasiūlytas saugaus verslo dokumentų persiuntimo metodas, skirtas saugiai persiųsti verslo dokumentus tarp sąveikumo sistemos dalyvių per sąveikumo sistemos serverį, kuris gali keisti informaciją. Metodas naudoja viešojo rakto kriptografiją, skirtas *UBL* verslo dokumentams ir užtikrina dokumentų konfidencialumo ir neišsiginamumo reikalavimus sąveikumo sistemose.

5. Pasiūlytas metodas realizuotas programiškai ir ištirtas greitaveikos požymiu. Greitaveikos rezultatai palyginti su standartinių apsaugos priemonių rezultatais.

6. Išanalizavus resursų sunaudojimą skirtinguose metodo etapuose nustatyta, kad serveryje reikalaujama 27% bendro sistemos našumo, o klientuose 35-38%. Todėl įgyvendinant metodą realiose sistemose, serverio techninė įranga turėtų būti galingesnė, kadangi sąveikumo sistemoje dažniausiai sąveikauja didelis kiekis klientų.

7. Ištyrus metodo efektyvumo priklausomybę nuo dokumento dydžio nustatyta, kad pasirašymo laikas, pasirašant visą dokumentą yra tiesiškai proporcingas dokumento dydžiui. Dokumento parašo tikrinimo laikas yra didesnis 9-11% negu pasirašymo laikas. Remiantis šiais rezultatais galime teigti, kad pasirašinėti visą dokumentą yra neefektyvu. Siunčiant didesnius dokumentus patartina naudoti dalinį parašą tik numatytiesiems laukams, kuriems būtina užtikrinti neišsiginamumą.

8. Ištyrus viso metodo greitaveikos priklausomybę nuo dokumento dydžio galima teigti, kad siunčiant nedidelius verslo dokumentus (iki ~30Kb) tikslinga juos pilnai apsaugoti (pasirašyti ir šifruoti). Esant didesniems dokumentams geriau naudoti dalinį pasirašymą, pasirašant tik numatytuosius laukus, o ne visą dokumentą.

9. Atsižvelgiant į kokybinius dokumentų saugos metodų įvertinimus galima teigti, kad siūlomas metodas nors ir būdamas truputį lėtesnis negu standartinės apsaugos priemonės bendrai užtikrina gerą saugos ir greitaveikos santykį.

10. Tolimesniems tyrimas vertėtų realizuoti ir išbandyti metodo modifikaciją, naudojančią simetrinį šifravimą kartu su asimetriniu raktu. Nors šis būdas būtų mažiau saugus perduodamiems verslo dokumentams apsaugoti, tačiau tikėtina geresnė dokumentų apsaugos greitaveika.

6. Literatūros sąrašas

[1] **Statistikos departamentas**. Įmonės, kurios naudoja informacines technologijas. Prieiga per internetą: www.stat.gov.lt. 2008-08-08 ir 2010-10-15 duomenys.

[2] **K Mackford.** Web services Architecture. BT Technology Journal, Vol 22 No.1, 2004-01.

[3] **A. Godac.** B2B interoperability: web services approach. Prieiga per internetą: http://www.srdc.metu.edu.tr/webpage/courses/ceng520/lecture_notes/_Chapter6_B2B_Interoperability.pdf

[4] Interoperability for SMEs Abilities. Interoperability in e-Business: Background, Kaunas University of Technology. 2007-12-27, p71.

[5] Intelligent modular open source platform for intercultural and cross-domain SME Networks. Prieiga per internetą: http://imipc33.imi.unikarlsruhe.de:8080/typo3/index.php?id=2

[6] **Claudia Guglielmina, Audrone Janavičiute, Mindaugas Kiauleikis, Valentinas Kiauleikis, Nerijus Morkevičius.** Performance modeling of interoperability system for SME's. TXT e-solutions, Information technology and control, 2006, Vol.35, No.4.

[7] **Rainer Ruggaber.** Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Applications, FP6-2002-IST-1, Integrated Project – Annex I, 2004.

[8] ATHENA - Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Applications, FP6-2002-IST-1, Integrated Project – Annex I, 2004.

[9] **Ricardo J.Rabelo, M.Mar Rodrigo Castro, Alex Conconi, Michele Sesana**. The Echolead plug & play coloborative business infrastructure. IFIP International Federation for Information Processing, 2005, Volume 186/2005, 3-16, DOI: 10.1007/0-387-29360-4_1.

[10] COIN Enterprise Collaboration & Interoperability EU FP7 IST IP Project 216256, Prieiga per internetą: http://www.coin-ip.eu/.

[11] Business process fusion based on Semantically-enabled Service-oriented Business Applications. Prieiga per internetą: http://www.fusionweb.org/FUSION/home.asp.

[12] **Eva Bucherer, Volker Hoyer**. Business Models for Enterprise Interoperability Platforms. SAP Research Switzerland, Blumenbergplatz 9, 9000 St.Gallen, Switzerland. 2008

[13] **Sotirios Koussouris, George Gionis, Aikaterini Maria Sourouni, Dimitrios Askounis and Kostas Kalaboukas.** Heterogeneous Domains' e-Business Transactions

Interoperability with the use of Generic Process Models. Presentation Transcript. Prieiga per internetą: http://www.slideshare.net/skous/heterogeneous-domains-ebusiness-transactions-interoperability-with-the-use-of-generic-process-models.

[14] **Annexes**. Unleashing the Potential of the European Knowledge Economy. Value Proposition for Enterprise Interoperability. Prieiga per internetą: ftp://ftp.cordis.europa.eu /pub/ist/docs/ict-ent-net/unleashing-potential-eke-annexe_en.pdf. 2008-01-21 p. 39.

[15] Prieiga per internetą: http://www.veforum.org/apps/pubs.asp?Q=1&T=Clusters%20 and%20Projects. Enterprise Interoperability Cluster: Developing open and secure technologies to connect systems and enterprises.

[16] **Asuman DOGAC, Gokce B. LALECI, Alper OKCAN, Mehmet OLDUZ, Michele SESANA, Alessandro CANEPA**. iSURF -An Interoperability Service Utility for Collaborative Supply Chain Planning across Multiple Domains: Textile Supply Chain Pilot. Prieiga per internetą: http://www.srdc.metu.edu.tr/webpage/publications/2008/12.pdf .2008

[17] **Klaus-Peter Eckert, Jane Hall, Eric Mannie-Corbisier, Henri-Jean Pollet**. The Business Context for Enterprise Interoperability between small ISPs. ISP: IST-FP6-027178. 2008

[18] COIN enterprise coloboration and Interoperability. Prieiga per internetą: http://www.oksl.ktu.lt/lankstukas.pdf

[19] Web Services Tutorial. Prieiga per internetą: http://www.w3schools.com /webservices/default.asp

[20] **Liam Quin**, **XML Activity Lead**. Extensible Markup Language (XML). Prieiga per internetą: http://www.w3.org/XML/ 2009-04-16

[21] **Ken Vollmer**. B2B Integration Trends: Message formats. Forrest research. 2007-06-06

[22] **Stephen Luster, Theresa Yee, Mark Crawford, Robert Parker, Christo Andonyadis, Daniel J. Drake**. Open Buying on the Internet and Extensible Markup Language Recommendations on Adoption. LOGISTICS MANAGEMENT INSTITUTE. 2001-01. Prieiga per internetą: http://www.idmanagement.gov/smartcard/information/ OpenBuyingXML.pdf

[23] IBM Institute of Advanced Commerce, Business-to-Business e-Commerce with Open Buying on the Internet. Prieiga per internetą: http://www.research.ibm.com/iac/papers/obi-paper/index.html

[24] **Robin Cover**. RosettaNET. Prieiga per internetą: http://xml.coverpages.org/rosettaNet.html. 2009-01-24 .

[25] RosettaNET Standards . Prieiga per internetą: http://www.rosettanet.org/ . 2009

[26] **OASIS**, XML Common Business Library (xCBL). Prieiga per internetą: http://xml.coverpages.org/cbl.html 2001-01-26.

[27] **OASIS**, Universal Business Language v2.0, Prieiga per internetą: http://docs.oasis-open.org/ubl/cs-UBL-2.0/ 2007-12-09.

[28] **Jon Bosak**. The Universal Business Language. U.S. Goverment XML Working group Washington, D.C 2002-02-20.

[29] **Robin Cover**. OASIS Universal Business Language. Prieiga per internetą: http://www.oasis-open.org. 2001.

[30] XML Common Business Library (xCBL). Prieiga per internetą: http://xml.coverpages.org/cbl.html

[31] **The OBI concortium**, OBI technical specification, Prieiga per internetą: http://lib.store.yahoo.net/lib/vw/OBIv210.pdf 1999.

[32] **Zhong Tian , Leo Y Liu, Jing Li, Jen-Yao Chung, Vibby Guttemukkala,** Business-to-Business e-Commerce with Open Buying on the Internet, IBM China Research Lab http://www.research.ibm.com/iac /papers/obi-paper/index.html.

[33] **Olga Zmejevskaja**. Blokinių šifrų algoritmų analizė. Vilniaus pedagoginis universitetas, magistro darbas. 2007.

[34] **Courtous, 2007. Paimta iš: Eligijus Sakalauskas, Narimantas Listopadskis, Gediminas Simonas Dosinas, Kęstutis Lukšys, Artūras Katvickis**. Kriptografinės sistemos, 5.1.2.6 Saugumo analizė. 2008.

[35] **Rimantas Plėšys, Dangis Rimkus, Rimantas Kavaliūnas, Ingrida Langzdinytė, Nijolė Sarafinienė**. Tinklų sauga, 1.4.2.3 Asimetriniai metodai. 2008-01-14.

[36] **Challa Narasimham, Jayaram Pradhan**. Evatuation of performance characteristics, of cryptosystem using text files. Journal of Theoretical and Applied Information Technology. 2008.

[37] **Erl T**. Service-Oriented Architecture: Concepts, Technology, and Design. New Jersey: Prentice Hall PTR, 2005, p. 792.

[38] Web service technologija. Karolis Kairaitis, Mantas Tamonis **išvertė WEERAWARANA, S.; CURBERA, F.; LEYMANN, F.** Web Services Platform

Architecture: SOAP, WSDL, WS-Policy, WS-Addressing, WS-BPEL, WS-Reliable Messaging, and More. Prentice Hall PTR, New Jersey, 2005, p. 27–113.

[39] Apibrėžimas. Prieiga per internetą: http://ec.europa.eu/research/fp7/pdf/fp7-brochure_lt.pdf.

7. Terminai ir sutrumpinimai

BP7 – tai pagrindinė Europos Sąjungos mokslinių tyrimų Europoje finansavimo priemonė. BP7, besitęsianti 2007–2013 metais, natūraliai pakeičia Šeštąją bendrąją programą (BP6) ir yra daugelį metų trukusių konsultacijų su akademine bendruomene, mokslinių tyrimų ir politikos formavimo įstaigomis ir kitomis suinteresuotomis šalimis rezultatas.[39]

[2]**ERP - Enterprise resource planning** – "įmonės resursų planavimas" tai nauja įmonės resursų planavimo strategija, plačiai plintanti daugelyje Europos valstybių. Tai strategija, kurios pagrindą sudaro visų kompanijos departamentų ir veiklos grupių integravimas į vieną kompiuterinę sistemą, kuri aptarnauja visus departamentus.

[2]**HR – Human Resorces** - žmogiškieji ištekliai

[3] **Hypertext Transfer Protocol** - Hipertekstų persiuntimo protokolas žiniatinklio duomenims (ištekliams) persiųsti. Taip pat HTTP apibrėžia serverio ir kliento programos (dažniausiai naršyklės) sąveiką.

[4] **CRM - Customer Relationship Management** - „santykių su klientais valdymas" tai būdas analizuoti ir panaudoti rinkodaros duomenų bankų duomenis, bei įdarbinant komunikacines technologijas sukurti bendrą įmonės praktiką ir metodus, kurie maksimaliai padidintų kiekvieno individualaus kliento ilgalaikę vertę (angl. life time value) įmonei. Taip pat tai kompleksinė sistema, leidžianti grupuoti, skirstyti klientus, taip pat taikyti veiksmingus vertingiausių klientų lojalumo skatinimo metodus.

[5] **MD5 - Message-Digest algorithm 5** - šiuo metu yra vienas iš populiariausių maišos algoritmų, apskaičiuojantis 128 bitų ilgio parašą. Nors MD5 algoritmas (kaip ir kiti algoritmai) gali apskaičiuoti parašą nuo begalinio skaičiaus įeinamų duomenų, galimų sugeneruotų kodų (parašų) skaičius yra baigtinis – 2 128.

[6] **SOA - Services Oriented Architecture** - į paslaugas orientuota architektūra yra vienas iš programinės įrangos architektūros stilių, kurio pagrindinis tikslas – užtikrinti laisvą ryšį tarp dviejų bendraujančių programinės įrangos agentų [37].

[7] **CORBA - Common Object Request Broker Architectur**e - tarptautinis standartas, kurį kūrė daugiau nei 500 kompanijų: Sun, Hewlett-Packart, IBM. CORBA - tai objekto prašymų tarpininkas (tarpinė programinė priemonė), skirta valdymui ir bendravimui tarp paskirstytų objektų. Gali veikti tiek su Unix, tiek su Microsoft operacinėmis sistemomis.

8. Priedai

Priedas 1: Užsakymo dokumentas, naudotas modeliavimo scenarijuose. Dokumentas paimtas iš realiai veikiančios sistemos, taip pat atitinkantis visus UBL standarto reikalavimus. Duomenys šiame dokumente yra pakeisti konfidencialumo sumetimais.

```
<?xml version="1.0" ?>
<Order xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:cac="urn:oasis:names:draft:ubl:schema:xsd:CommonAggregateComponents-2"
xmlns:cbc="urn:oasis:names:draft:ubl:schema:xsd:CommonBasicComponents-2" xmlns:udt="urn:un:unece:uncefact:data:
draft:UnqualifiedDataTypesSchemaModule:2" xmlns:sdt="urn:oasis:names:draft:ubl:schema:xsd:SpecializedDatatypes-2"
xmlns:ccts="urn:oasis:names:draft:ubl:schema:xsd:CoreComponentParameters-2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns="urn:oasis:names:draft:ubl:schema:xsd:Order-2" xsi:schemaLocation="urn:oasis:names:draft:ubl:schema:xsd:Order-2
prd-UBL-2.0/xsd/maindoc/UBL-Order-2.xsd">
  <cbc:ID>XYZ-4298</cbc:ID>
  <cbc:CopyIndicator>false</cbc:CopyIndicator>
  <cbc:IssueDate>2010-05-02</cbc:IssueDate>
  <cbc:IssueTime>18:14:36</cbc:IssueTime>
  <cbc:DocumentCurrencyCode>LTL</cbc:DocumentCurrencyCode>
  <cac:BuyerCustomerParty>
    <cbc:CustomerAssignedAccountID>111222333</cbc:CustomerAssignedAccountID>
    <cac:Party>
      <cac:PartyName>
        <cbc:Name>Imone „A"</cbc:Name>
      </cac:PartyName>
    </cac:Party>
  </cac:BuyerCustomerParty>
  <cac:SellerSupplierParty>
    <cbc:CustomerAssignedAccountID>123456789</cbc:CustomerAssignedAccountID>
    <cac:Party>
      <cac:PartyName><cbc:Name>Imone „B"</cbc:Name>
      </cac:PartyName>
    </cac:Party>
  </cac:SellerSupplierParty>
  <cac:LegalTotal>
    <cbc:TaxInclusiveAmount currencyID="LTL">250.00</cbc:TaxInclusiveAmount>
    <cbc:ToBePaidAmount currencyID="LTL">250.00</cbc:ToBePaidAmount>
  </cac:LegalTotal>
  <cac:OrderLine>
    <cac:LineItem>
      <cbc:ID>1</cbc:ID>
      <cbc:LineExtensionAmount currencyID="LTL">120</cbc:LineExtensionAmount>
      <cbc:MinimumQuantity unitCode="KG">10</cbc:MinimumQuantity>
      <cbc:MaximumQuantity unitCode="KG">20</cbc:MaximumQuantity>
      <cac:Item>
        <cbc:Description>Plikyti sausainiai "Laumė"</cbc:Description>
        <cbc:PackQuantity>6</cbc:PackQuantity>
        <cbc:PackSizeNumeric>1</cbc:PackSizeNumeric>
        <cbc:Name> Plikyti sausainiai "Laumė"</cbc:Name>
        <cbc:HazardousRiskIndicator>false</cbc:HazardousRiskIndicator>
        <cac:SellersItemIdentification>
          <cbc:ID>4770732193610</cbc:ID>
        </cac:SellersItemIdentification>
        <cac:ClassifiedTaxCategory>
          <cbc:ID>Standard Rate</cbc:ID>
          <cbc:Percent>18</cbc:Percent>
          <cac:TaxScheme><cbc:ID>VAT</cbc:ID>
          <cbc:Name>VAT</cbc:Name></cac:TaxScheme>
        </cac:ClassifiedTaxCategory>
      <cac:ItemInstance>
      <cbc:ProductTraceID>1</cbc:ProductTraceID>
      <cbc:ManufactureDateTime>2010-05-12T12:00:00</cbc:ManufactureDateTime>
      </cac:ItemInstance></cac:Item>
    </cac:LineItem>
  </cac:OrderLine>
</Order>
```