



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

Ing. Béla GENGE

TEZA DE DOCTORAT

**CONTRIBUȚII LA COMPUNEREA ȘI IMPLEMENTAREA
PROTOCOALELOR DE SECURITATE**

- REZUMAT -

**CONDUCĂTOR ȘTIINȚIFIC:
Prof. dr. ing. Iosif IGŊAT**

2009

Susținerea publică a tezei de doctorat va avea loc în data de 22 mai 2009, la Universitatea Tehnică din Cluj Napoca, str. Constantin Daicoviciu nr. 15.

Componența comisiei de doctorat:

PREȘEDINTE:

Prof. Dr. Ing. Sergiu **NEDEVSCI**, decan,
Universitatea Tehnică din Cluj-Napoca

MEMBRII:

Prof. Dr. Ing. Iosif **IGNAT**, conducător științific,
Universitatea Tehnică din Cluj-Napoca

Prof. Dr. Ing. Victor Valeriu **PATRICIU**, referent,
Academia Tehnică Militară din București

Prof. Dr. Ing. Valentin **CRISTEA**, referent,
Universitatea „Politehnica” din București

Prof. Dr. Ing. Dorian **GORGAN**, referent,
Universitatea Tehnică din Cluj-Napoca

PREFAȚĂ

Teza de doctorat cu titlul „*Contribuții la compunerea și implementarea protocoalelor de securitate*” reprezintă rodul activității de cercetare desfășurată de autor pe parcursul studiilor de doctorat începute în oct. 2005 în cadrul Facultății de Automatică și Calculatoare a Universității Tehnice din Cluj-Napoca. În lucrare sunt prezentate rezultatele cercetărilor fundamentale și aplicative ale autorului în domeniul compunerii și implementării protocoalelor de securitate.

Lucrarea își propune ca obiectiv principal elaborarea unor metode care să asigure compunerea și implementarea automată a protocoalelor de securitate în domeniul serviciilor Web. Pentru atingerea acestui obiectiv s-au identificat următoarele sub-obiective:

- Compunerea secvențială automată a protocoalelor de securitate;
- Compunerea secvențială automată a protocoalelor cu proprietăți de securitate egale;
- Implementarea automată a protocoalelor de securitate compuse.

Pornind de la un model al protocoalelor de securitate propus de autor, s-a elaborat o nouă metodă de compunere ce asigură menținerea proprietăților de securitate ale protocoalelor originale. Avantajul acestei metode față de metodele existente o reprezintă posibilitatea automatizării întregului proces de compunere, ce nu necesită intervenția operatorului uman.

În contextul compunerii protocoalelor, s-a identificat o nouă direcție de cercetare: compunerea protocoalelor cu proprietăți de securitate egale. În această direcție, în lucrare se prezintă o metodă comparativă de evaluare a performanțelor protocoalelor ce nu necesită cunoștințe legate de detaliile de implementare ale acestora.

Pentru implementarea automată a protocoalelor rezultate din procesul de compunere, s-a dezvoltat un nou model de specificație. Modelul specificație propus conține o descriere a precondițiilor, efectelor, participanților, termenilor și mesajelor protocoalelor. În plus, acesta conține și o descriere a detaliilor legate de implementarea mesajelor, precum și a detaliilor legate de procesarea și construirea mesajelor. Această specificație este utilizată în contextul unei platforme ce asigură atât compunerea cât și implementarea automată a protocoalelor de securitate. Platforma propusă în lucrare pune în evidență două nivele soft: nivelul mesagerie și nivelul serviciu. Ambele nivele sunt implementate prin intermediul tehnologiilor din domeniul serviciilor Web. Utilizarea unei asemenea implementări asigură o deschidere și o flexibilitate protocoalelor, specifică serviciilor Web.

Lucrarea de față este structurată pe 7 capitole, dintre care primul capitol are un caracter introductiv, identifică direcțiile de cercetare considerate deschise și stabilește obiectivele tezei iar ultimul conține concluziile, contribuțiile și dezvoltările viitoare propuse. În afară de primul și ultimul, toate capitolele încep cu o introducere și se încheie cu concluzii iar în fiecare din capitolele 3, 4, 5 și 6 sunt prezentate rezultate experimentale care asigură validarea metodelor teoretice elaborate. În cadrul lucrării s-au sintetizat 238 referințe bibliografice dintre care 11 lucrări indexate în baze de date internaționale și 8 lucrări neindexate publicate de autor.

În **Capitolul 2**, intitulat „**Stadiul actual al cercetării**”, sunt trecute în revistă principalele lucrări din direcția de cercetare corespunzătoare fiecărui obiectiv propus. Într-o primă fază sunt prezentate lucrările din domeniul compunerii protocoalelor de securitate. Se identifică două metode de compunere în funcție de sursa protocoalelor compuse: compunerea protocoalelor predefinite și compunerea protocoalelor existente. În continuare, se identifică direcția de cercetare legată de evaluarea performanțelor protocoalelor de securitate. Principalele lucrări din această direcție se împart în două categorii: evaluarea performanței algoritmilor criptografici și evaluarea

performanței protocoalelor de securitate. Ultima direcție majoră de cercetare, implementarea protocoalelor de securitate împarte lucrările din domeniu în două categorii: implementarea manuală a protocoalelor de securitate și implementarea automată a protocoalelor de securitate. Prima categorie pune în evidență lucrări ce propun protocoale predefinite pentru implementarea anumitor proprietăți, iar a doua categorie propune metode prin care procesul de implementare poate fi automatizat.

Capitolul 3, intitulat „**Compunerea protocoalelor de securitate**”, cuprinde, în principal construirea modelului protocoalelor ce pune în evidență informațiile necesare procesului de compunere: condiții, efecte, termeni generați și secvențe de termeni. Pornind de la acest model se definește prima fază, de compunerea a condițiilor și efectelor și a doua fază, de compunere a termenilor. Prima fază presupune verificarea satisfacerii condițiilor și verificarea proprietății de non-distruktivitate prin examinarea termenilor din mulțimea condițiilor și efectelor. A doua fază presupune verificarea independenței termenilor prin utilizarea unui model canonic construit pe baza modelului inițial. Metoda de compunere propusă este validată printr-o propoziție prin care se demonstrează că dacă sunt îndeplinite toate condițiile enunțate, intrusul nu-și poate folosi puterile pentru construirea unor atacuri noi asupra protocoalelor implicate. Validitatea demonstrației este susținută și de mulțimea protocoalelor compuse asupra cărora s-a aplicat metoda propusă, protocoale ale căror proprietăți au fost ulterior verificate prin utilitare existente și consacrate în domeniu.

În **Capitolul 4**, intitulat „**Evaluarea performanțelor protocoalelor de securitate**”, se elaborează o metodă comparativă de evaluare a performanțelor protocoalelor prin modelarea operațiilor criptografice utilizate de participanți în construirea și procesarea mesajelor precum și atașarea unor costuri pentru fiecare operație în parte. În acest caz s-a utilizat modelul canonic propus în capitolul precedent întrucât forma redusă a termenilor este ideală pentru o asemenea evaluare ce nu necesită informații legate de implementarea protocoalelor. Pentru determinarea costului fiecărei operații criptografice s-a construit un model polinomial pentru aproximarea performanței tipurilor de algoritmi utilizați. Coeficienții polinomului sunt determinați pentru fiecare tip de algoritm în parte prin evaluarea exhaustivă a performanței algoritmilor simetrici, asimetrici și hash pentru trei biblioteci bine-cunoscute: Cryptlib, OpenSSL și Crypto++. Modelele construite sunt validate prin evaluarea comparativă a performanței a 1000 de protocoale generate automat. Metoda de evaluare propusă este validată prin evaluarea comparativă a unui set de protocoale definite în biblioteca SPORE (en. „Security Protocol Open Repository”) și în biblioteca menținută de John Clark.

Capitolul 5, intitulat „**Construirea specificațiilor pentru implementarea protocoalelor de securitate**”, cuprinde construirea unei specificații formale, pornind de la modelul protocol definit în capitolele precedente, ce conține atât informațiile necesare compunerii cât și cele necesare execuției. Specificația construită este formată din două componente: componenta secvențială, care identifică secvențele de mesaje, condițiile și efectele, și componenta semantică ce oferă detalii legate de construirea și procesarea mesajelor. Pentru adnotarea mesajelor din prima componentă, se construiește o structură ontologică de bază. Aceasta definește o serie de sub-ontologii specifice tuturor protocoalelor de securitate (e.g. proprietăți criptografice, tipuri de termeni) și sub-ontologii ce trebuie extinse pentru fiecare protocol în parte. Ontologiile de bază

sunt construite prin consultarea specificațiilor protocoalelor de securitate din domeniu, însă acestea pot fi extinse cu concepte noi pentru îndeplinirea cerințelor unor protocoale noi. Pornind de la modelul protocol inițial, se definește un set de reguli și algoritmi pentru generarea specificațiilor. Prin utilizarea acestora, se asigură menținerea proprietăților de securitate ale protocoalelor modelate. Corectitudinea specificației propuse în acest capitol și ale regulilor sunt validate prin identificarea corespondențelor dintre modelul inițial și cel propus aici. Totodată, demonstrarea posibilității execuției specificațiilor astfel generate se realizează prin generarea și execuția mai multor specificații ale unor protocoale cunoscute precum ISO9798 sau Kerberos.

Capitolul 6, intitulat „**Platformă intermediară pentru compunerea și implementarea protocoalelor de securitate**”, cuprinde proiectarea și implementarea unei platforme intermediare bazate pe servicii Web. Se definește arhitectura orientată pe servicii și arhitectura soft a platformei propuse. În cadrul primei arhitecturi se identifică tipurile de servicii implementate: nume, specificații, compunere și resursă. Arhitectura soft identifică un nivel transport bazat pe protocoale de securitate, peste care se definește un nivel mesagerie serviciu pentru transportul mesajelor specifice fiecărui serviciu. Pentru fiecare tip de serviciu se prezintă arhitectura acestuia și se elaborează nivelul mesagerie corespunzător. Pornind de la platforma definită, în cadrul rezultatelor experimentale se construiește un sistem de monitorizare prin servicii video. Prin rezultatele experimentale obținute, s-au pus în evidență performanțele platformei și posibilitatea utilizării acesteia pentru transferul datelor multimedia.

Capitolul 7, intitulat „**Concluzii finale**”, realizează o trecere în revistă a conținutului tezei, a contribuțiilor aduse precum și a direcțiilor de cercetare rămase deschise. Pentru fiecare direcție de cercetare propusă, sunt identificate contribuțiile aduse și sunt motivate aspectele care nu au fost acoperite. Între contribuțiile originale se numără o metodă de compunere sintactică a protocoalelor de securitate, o metodă de evaluare comparativă a performanței, specificații formale precum și o platformă pentru automatizarea procesului de compunere și implementare. Printre aspectele rămase neacoperite se numără compunerea capabilităților serviciilor, controlul accesului, automatizarea procesului de generare a specificațiilor. Aspectele care nu au fost tratate în cadrul tezei sunt fie acoperite de alte direcții de cercetare, fie reprezintă direcții de cercetare noi care trebuie aprofundate.

Această lucrare de doctorat a fost realizată sub îndrumarea științifică a domnului profesor universitar dr. ing. Iosif Ignat, căruia autorul dorește să-i aducă cele mai sincere mulțumiri și recunoștință pentru sprijinul acordat.

De asemenea, mulțumiri sunt adresate catedrei de Calculatoare și colectivului Facultății de Automatică și Calculatoare din cadrul Universității Tehnice din Cluj-Napoca, colegilor din Catedra de Inginerie Electrică a Universității „Petru Maior” din Târgu Mureș, în special doamnei Conf. dr. ing. Haller Piroaska, pentru ajutorul acordat în perioada de pregătire și realizare a lucrării de doctorat, precum și familiei și prietenilor pentru susținere.

Cuprins

1. Introducere	5
1.1 Protocole de securitate	5
1.2 Direcții de cercetare	9
1.3 Obiectivele lucrării și contribuții	12
1.4 Structura lucrării	16
1.5 Concluzii	18
2. Stadiul actual al cercetării	20
2.1 Introducere	20
2.2 Compunerea protocoalelor de securitate	20
2.3 Evaluarea performanțelor protocoalelor de securitate	25
2.4 Implementarea protocoalelor de securitate	28
2.5 Concluzii	32
3. Compunerea protocoalelor de securitate	34
3.1 Introducere	34
3.2 Modele protocol	37
3.2.1 Construirea modelelor protocol	37
3.2.2 Construirea modelelor protocol intrus	41
3.3 Compunerea precondițiilor și efectelor	42
3.3.1 Compunerea modelelor participant	42
3.3.2 Compunerea modelelor protocol	44
3.4 Compunerea termenilor	44
3.4.1 Construirea modelului canonic	45
3.4.2 Construirea modelului intrus canonic	47
3.4.3 Funcții de mapare	48
3.4.4 Independența modelelor protocol	50
3.4.5 Compunerea termenilor a două modele protocol	52
3.5 Compunerea secvențelor de modele protocol	52
3.6 Rezultate experimentale	55
3.6.1 Compunerea protocoalelor Yahalom-Lowe și Kao-Chow	55
3.6.2 Compunerea protocoalelor Lowe-Needham-Schroeder și ISO9798	62
3.6.3 Compunerea protocoalelor din biblioteci existente	68
3.7 Concluzii	70
4. Evaluarea performanțelor protocoalelor de securitate	71
4.1 Introducere	71
4.2 Extinderea modelului canonic	72
4.3 Evaluarea performanțelor algoritmilor criptografici	75
4.3.1 Evaluarea performanțelor algoritmilor simetrici	76
4.3.2 Evaluarea performanțelor algoritmilor hash	77
4.3.3 Evaluarea performanțelor algoritmilor asimetrici	78
4.4 Modelarea performanțelor algoritmilor criptografici	79
4.4.1 Construirea modelului	79
4.4.2 Validarea modelului	81
4.5 Rezultate experimentale	85
4.5.1 Evaluarea comparativă a performanțelor protocoalelor CCITT X.509 v1 și v1c ...	85
4.5.2 Evaluarea comparativă a performanțelor protocoalelor din biblioteci existente ...	88
4.6 Concluzii	90

5. Construirea specificațiilor pentru implementarea protocoalelor de securitate	91
5.1 Introducere	91
5.2 Formularea cerințelor	91
5.2.1 Cerințe pentru modelarea precondițiilor și efectelor	92
5.2.2 Cerințe pentru modelarea termenilor transmiși și recepționați	92
5.2.3 Cerințe pentru modelarea cunoștințelor	93
5.3 Alegerea tehnologiilor pentru descrierea specificațiilor	93
5.4 Specificația secvențelor de mesaje (S-SEC)	94
5.4.1 Structura S-SEC	95
5.4.2 Modelul S-SEC	96
5.5 Specificația semantică a protocoalelor de securitate (S-SEM)	97
5.5.1 Structura S-SEM	97
5.5.2 Modelul S-SEM	105
5.6 Generarea specificațiilor SEC-SEM (S-SEC-SEM)	107
5.6.1 Modelarea precondițiilor, efectelor și a secvențelor de mesaje	107
5.6.2 Modelarea cunoștințelor	109
5.6.3 Modelarea legăturilor dintre secvențele de mesaje și cunoștințe	115
5.6.4 Algoritmi pentru generarea S-SEC-SEM	118
5.6.5 Menținerea proprietăților de securitate în S-SEC-SEM	124
5.7 Rezultate experimentale	126
5.7.1 Construirea S-SEC-SEM al protocolului Lowe-BAN	127
5.7.2 Execuția specificațiilor generate	135
5.8 Concluzii	137
6. Platformă intermediară pentru compunerea și implementarea protocoalelor de securitate	139
6.1 Introducere	139
6.2 Formularea cerințelor	140
6.3 Arhitectura orientată pe servicii	140
6.3.1 Arhitectura serviciului de nume	142
6.3.2 Arhitectura serviciului de specificații	143
6.3.3 Arhitectura serviciului de autorizare și compunere	144
6.3.4 Arhitectura serviciului resursă	147
6.4 Arhitectura soft	150
6.4.1 Nivelul comunicații și mesagerie XML	151
6.4.2 Nivelul protocoalelor de securitate	151
6.4.3 Nivelul protocoalelor serviciu	153
6.5 Accesarea resurselor	157
6.6 Rezultate experimentale	158
6.6.1 Transferul datelor	160
6.6.2 Accesarea serviciilor resursă simple	161
6.6.3 Accesarea serviciilor resursă compuse	163
6.6.4 Transferul datelor compuse	165
6.7 Concluzii	167
7. Concluzii finale	169
Bibliografie	175
Anexe	187

REZUMAT

O dată cu cerințele de securizare a comunicațiilor pe Internet, s-a pus problema corectitudinii protocoalelor deja existente. În urma analizelor și verificărilor efectuate, s-au descoperit o serie de atacuri [Cla96, Hol05] asupra unor protocoale publicate chiar cu un deceniu în urmă. Un exemplu ilustrător în acest sens este cel al protocolului Needham-Schroeder [NS78], asupra căruia Gavin Lowe a descoperit un atac cu aproape douăzeci de ani după publicare [Low96b]. În ultimul deceniu însă, atenția cercetătorilor s-a orientat nu numai spre analiza și verificarea protocoalelor de securitate ci și către dezvoltarea unor metode formale de proiectare a acestora. Una dintre cele mai răspândite metode o reprezintă proiectarea modulară, bazată pe procesul de *compunere* [CR03, Cho06, Cre06b, DDMR07, ACG+08], ce presupune combinarea a două sau mai multor protocoale cu scopul acumulării proprietăților de securitate.

În această lucrare, autorul propune într-o primă fază o metodă de compunere care nu necesită intervenția operatorului uman [GI07b, GI08, GHIO08]. Compunerea protocoalelor de securitate se va realiza în timpul execuției și este completată cu implementarea automată a protocoalelor rezultate. Metodele de compunere existente se bazează pe identificarea manuală a proprietăților de securitate, compunerea manuală a acestora și verificarea menținerii proprietăților rezultate prin utilitare consacrate în domeniul verificării corectitudinii protocoalelor de securitate [Cre06a]. Față de acestea, autorul propune o metodă sintactică bazată pe proprietatea de independență a protocoalelor formulată de Guttman și Fabrega [GF00] prin care se asigură menținerea proprietăților de securitate pentru protocoalele compuse. Conform acestora, două protocoale sunt independente dacă utilizează termeni criptați cu structuri sau chei diferite. Pornind de la această proprietate, autorul propune un model al protocoalelor [GI06, GI07a] ce pune în evidență structurile termenilor și asigură o verificare sintactică a acestei proprietăți. Contribuția lucrării la verificarea proprietății de independență constă într-o metodă sintactică de identificare a atacurilor de tip redare și de tip confuzie a tipurilor [HLS00, Mea03, GI06, GI07a]. Pe lângă verificarea independenței, metoda de compunere asigură compunerea condițiilor și efectelor pentru propagarea cunoștințelor generate de protocoale și verificarea asigurării cunoștințelor necesare execuției secvențelor de protocoale rezultate.

Prin eliminarea operatorului uman din procesul de compunere, secvența de protocoale rezultată poate conține protocoale cu aceleași proprietăți (e.g. două sau mai multe protocoale schimb de cheie). Pentru a elimina un asemenea rezultat, autorul propune o nouă direcție de cercetare: compunerea protocoalelor cu proprietăți de securitate egale. O asemenea compunere asigură alegerea unui protocol pe baza criteriilor de performanță. Metodele existente de evaluare a performanțelor protocoalelor se bazează pe cunoașterea mediului de execuție a protocoalelor și asigură modele parametrizate [HM02, BBC+05] sau asigură o evaluare prin implementare [APS99, Das00, Gut03, SRW05, KH05, CDW06]. În lucrarea de față se propune o metodă comparativă de evaluare a performanțelor ce nu necesită informații legate de contextul de execuție a protocoalelor [GHIO08]. Metoda se bazează pe atașarea unor funcții cost pentru fiecare operație criptografică și construirea unor funcții polinomiale pentru fiecare asemenea operație. Funcțiile polinomiale asigură modelarea performanței claselor algoritmilor criptografici de criptare/decriptare (e.g. criptare simetrică, decriptare simetrică, criptare hash) identificate în modelul protocoalelor. Aceste funcții sunt construite prin evaluarea exhaustivă a performanțelor algoritmilor criptografici implementați în 3 biblioteci criptografice: OpenSSL [OSS08], Cryptlib [Gut08] și Crypt++ [Cry08].

În continuare, lucrarea propune o metodă nouă pentru implementarea automată a protocoalelor de securitate. Implementarea automată presupune fie generarea automată a codului corespunzător unei specificații date [SPP01, AM03, TH05], fie utilizarea unei implementări ce permite localizarea și execuția automată a specificațiilor date, fără necesitatea modificării codului [MBJ+02, DKF+03, KLM05, BLG+08]. Metoda de implementare propusă de autor intră în a doua categorie și utilizează o specificație ce conține o descriere a operațiilor de construire și procesare a mesajelor [GH08a, GH08, GH09a] precum și o platformă intermediară prin care se asigură localizarea și execuția automată a specificațiilor [GH08b, GH09b, GH09c]. Metodele existente în această direcție fie utilizează o descriere proprie a protocoalelor [MBJ+02], fie asigură un set de descrieri a protocoalelor din domeniul serviciilor Web [DKF+03, KLM05, BLG+08] ce asigură o selecție automată a protocoalelor pe baza unui set de criterii. Pentru a asigura o flexibilitate și deschidere a protocoalelor specifică serviciilor Web, specificația propusă precum și componentele platformei au fost implementate prin tehnologii specifice serviciilor Web. Utilizarea acestor tehnologii asigură specificației propuse o deschidere ce permite extinderea specificației fără modificări majore cu ontologii de securitate existente în acest domeniu [BLG+08]. O asemenea abordare reprezintă o îmbunătățire clară față de [MBJ+02] și o completare a descrierilor [DKF+03, KLM05, BLG+08] cu operații de construire și procesare a termenilor mesajelor. Pornind de la specificația și platforma propusă, s-a dezvoltat un sistem de supraveghere video care asigură compunerea serviciilor ce utilizează protocoale de securitate și execuția automată a protocoalelor compuse.

Capitolul 1 este dedicat realizării unei introduceri în domeniul protocoalelor de securitate, în domeniul compunerii și implementării acestor protocoale. În același capitol s-au identificat direcțiile de cercetare încă deschise și s-au stabilit principalele obiective ale tezei.

Capitolul 2 este dedicat realizării unui studiu asupra principalelor lucrări din cele trei direcții de cercetare identificate:

- Compunerea protocoalelor de securitate;
- Evaluarea performanțelor protocoalelor de securitate;
- Implementarea protocoalelor de securitate.

În direcția compunerii protocoalelor de securitate s-au identificat două sub-direcții de cercetare: compunerea protocoalelor predefinite și compunerea protocoalelor existente. Prima direcție presupune utilizarea unor primitive predefinite, reprezentând protocoale de dimensiuni reduse, pentru construirea protocoalelor complexe [Cho06, Gut02, Gut01, GF02]. A doua direcție presupune utilizarea protocoalelor existente și verificarea proprietăților non-distructive ale acestora [CR03, DDMR07, Cre06b, DDMP03, CDPW07, ACG+08]. În direcția compunerii protocoalelor de securitate se constată existența unei literaturi bogate care tratează atât compunerea protocoalelor predefinite cât și compunerea protocoalelor existente. Cu toate acestea, se constată existența unei singure metode care asigură o compunere semi-automatizată [ACG+08] prin compunerea manuală a precondițiilor și efectelor și utilizarea unui utilitar existent pentru verificarea automată a menținerii proprietăților de securitate ale protocolului rezultat. Celelalte metode necesită asistența operatorului uman pentru modelarea protocoalelor într-un format corespunzător sau pentru luarea deciziilor în procesul de verificare, acestea prezentând doar rezultate experimentale obținute manual, fără menționarea posibilităților de automatizare a metodelor prezentate.

Pentru asigurarea compunerii protocoalelor cu proprietăți de securitate egale, ce presupune alegerea unui protocol după anumite criterii de performanță, în continuare sunt prezentate principalele lucrări din cea de-a doua direcție de cercetare. În acest context se prezintă într-o primă fază principalele lucrări din domeniul evaluării performanțelor algoritmilor criptografici [VW01, Hir03, Kir05], urmată de prezentarea principalelor lucrări din domeniul evaluării performanțelor protocoalelor de securitate [Das00, HM02, Gut03, BBC+05, SRW05, Zha05, CDW06]. Întrucât prin execuția unui protocol de securitate participanții schimbă mesaje criptate între ei, evaluarea performanțelor acestor protocoale se reduce la evaluarea resurselor necesare construirii și procesării mesajelor precum și evaluarea resurselor necesare transmiterii și recepționării acestora.

Din studiul metodelor existente se constată inexistența unor lucrări care să abordeze construirea unor modele ale performanțelor tipurilor de algoritmi criptografici: simetric, asimetric și hash. Metodele existente bazate pe utilizarea modelelor [Kir05, BBC+05] necesită cunoașterea parametrilor fizici de implementare a algoritmilor sau a protocoalelor sau cunoștințe exacte legate de structura mesajelor, informații care în faza de compunere nu sunt disponibile.

În direcția implementării protocoalelor de securitate, s-a constatat o abundență a lucrărilor atât în cazul implementării manuale [DA99, Ylo06, OAS05a, OAS05b, WWWC06, OAS07a, OAS07b, IBM07] cât și în cazul implementării automate [SPP01, MBJ+02, AM03, DKF+03, KLM05, TH05, BLG+08]. Cu toate acestea, lucrările care fac referire la o implementare automată a protocoalelor de securitate în domeniul serviciilor Web sunt reduse la număr [DKF+03, KLM05, BLG+08] și oferă o semi-automatizare al întregului proces, în sensul că asigură doar selectarea automată a protocoalelor, o clasificare a acestora și nu o metodă pentru implementarea automată a protocoalelor noi.

În **Capitolul 3** autorul propune o metodă sintactică pentru compunerea secvențială a protocoalelor de securitate. Datorită asemănărilor existente între compunerea secvențială și cea paralelă, aplicarea metodei propuse în procesul de compunere paralelă presupune utilizarea unei submulțimi a operațiilor definite pentru compunerea secvențială.

Metoda de compunere prezentată în cadrul acestui capitol abordează într-o primă fază compunerea condițiilor și efectelor (i.e. *compunerea PE*). Această compunere realizează verificarea asigurării condițiilor de execuție a protocoalelor și identificarea secvenței corecte de protocoale. Datorită propagării termenilor generați de la un protocol la celălalt, compunerea PE trebuie să asigure și proprietatea de non-distructivitate a proprietăților termenilor preluați din alte protocoale.

A doua fază presupune compunerea termenilor (i.e. *compunerea T*) prin verificarea proprietăților de non-distructivitate pentru termenii protocoalelor compuse. În această fază, pe baza cunoștințelor participanților, se identifică termenii cu structuri criptografice ce permit redarea și acceptarea lor în alte protocoale. O asemenea acceptare duce la construirea atacurilor asupra altor protocoale, chiar dacă protocoalele în cauză sunt corecte. Astfel, pe lângă atacurile clasice, ce pot fi construite asupra unui singur protocol, dacă două sau mai multe protocoale sunt executate în cadrul aceluiași mediu, apare o nouă clasă de atacuri, denumită clasa atacurilor *multi-protocol*. Una din primele lucrări în care aceste atacuri au fost semnalate este [KSW97]. Aici, autorii demonstrează cum se poate construi un atac pentru un protocol dat prin utilizarea unui alt protocol ales. Ulterior, tema, a fost abordată de o serie de alte lucrări [Cre05, GH07c, Maf05].

Metoda de compunere propusă utilizează un model al protocoalelor ce pune în evidență condițiile, efectele, termenii generați precum și termenii transmiși și recepționați corespunzători

fiecărui participant. Totodată, întrucât analiza termenilor se realizează în contextul unui intrus, în cadrul acestui capitol se construiește și modelul intrus Dolev-Yao [DY83, Cer01, Cer02].

Modelul protocol propus [GHO108, GH07a, GH07b, GI07b] este similar modelului *spațiilor strand* [FHG99b, FHG98]. Un *strand* reprezintă o secvență de transmisii și recepționări de termeni, iar un *spațiu strand* reprezintă o colecție de strand-uri. Modelul construit este asemănător cu cel al spațiilor strand în sensul că oferă o descriere a participanților în forma secvențelor de transmisii și recepționări. Însă diferențele introduse sunt multiple, astfel încât modelul propus în cadrul acestui sub-capitol diferă prin:

- construirea explicită a termenilor, prin utilizarea construcțiilor gramaticale;
- modelarea denumirii participanților atașată unei secvențe de termeni;
- introducerea claselor de secvențe de mesaje corespunzătoare operațiilor interne efectuate de participanți pentru transmiterea și recepționarea termenilor;
- modelarea cunoștințelor participanților;
- modelarea precondițiilor și a efectelor protocolului.

Participanții protoalelor de securitate comunică prin schimbarea unor termeni ce aparțin elementelor mulțimilor: P , reprezentând mulțimea denumirilor participanților; N , reprezentând mulțimea numerelor aleatoare; K , reprezentând mulțimea cheilor utilizate în cadrul unui protocol (e.g. chei de scurtă durată, chei de lungă durată, chei private și chei publice); C , reprezentând mulțimea certificatelor; M , reprezentând mulțimea componentelor definite de aplicațiile utilizator (e.g. octeți imagini video, octeți streaming audio). Pentru mulțimile date se definesc și mulțimile P^* , N^* , K^* și M^* reprezentând, mulțimi ale căror elemente sunt mulțimi din, respectiv, P , N , K și M .

Pentru reprezentarea tipurilor de funcții criptografice utilizate, se definește *NumeFunc*, ca fiind o colecție de denumiri:

$NumeFunc ::= sk$	(funcție de criptare simetrică)
$ pk$	(funcție de criptare asimetrică)
$ h$	(criptare cu funcție hash)
$ hmac$	(criptare cu funcție hash și cheie criptografică - hmac)

Criptarea unui termen se realizează printr-o funcție criptografică și o cheie. Decriptarea este posibilă doar dacă participantul deține cheia de decriptare. În cazul criptografiei simetrice, cheia de decriptare este aceeași cu cheia de criptare, însă, în cazul criptografiei asimetrice, pentru decriptarea unui termen criptat cu o cheie publică, este nevoie de cheia privată și vice-versa. Pentru determinarea cheii inverse, se definește funcția $_^{-1} : K \rightarrow K$.

Pornind de la mulțimile date, noțiunea de *termen* este definit astfel:

$$T ::= . \mid P \mid N \mid K \mid C \mid M \mid (T, T) \mid \{T\}_{NumeFunc(T)},$$

unde ‘.’ reprezintă un termen gol, ‘(,)’ reprezintă concatenare, iar ‘{ }’ reprezintă criptare. Pentru reprezentarea operațiilor criptografice ce nu necesită o cheie, cum ar fi de exemplu aplicarea unei funcții hash, se utilizează ‘.’ pentru reprezentarea cheii. Definim mulțimea T^* ale cărei elemente sunt mulțimi cu elemente din T .

Operația de bază realizată de participanți reprezintă transmiterea și recepționarea termenilor. Prin atașarea unor *semne* termenilor definiți anterior, vor rezulta *termeni cu semn*, pe care le vom denumi *noduri*. Nodurile sunt utilizate la modelarea operațiilor de transmitere și recepționare corespunzătoare participanților.

Definiția 1. Un model participant (M-PART) reprezintă un tuplu $\langle prec, eff, tip, gen, part, lan\tau \rangle$.

Definiția 2. Un model protocol (M-PROT) reprezintă o colecție de modele participant în cadrul cărora pentru fiecare nod pozitiv (i.e. termen transmis) există un nod negativ (i.e. termen recepționat).

Principiile de compunere a condițiilor și efectelor corespunzătoare unei secvențe de modele protocol sunt aplicate într-o primă fază modelelor participant și sunt extinse ulterior asupra modelelor protocol.

Pentru două modele participant definim predicatul $PART_PREC$: $T^* \times PR_CC^* \times PR_CC^*$, care pe baza unui set de cunoștințe inițiale și a unei mulțimi de predicate efect stabilește dacă un set de predicate condiție este activ sau nu. Pentru două modele participant, ς_1 și ς_2 , componentele $\varsigma_1 = \langle prec_1, eff_1, tip_1, gen_1, part_1, lan\tau_1 \rangle$ și $\varsigma_2 = \langle prec_2, eff_2, tip_2, gen_2, part_2, lan\tau_2 \rangle$, și un model context având cunoștințele inițiale ci , predicatul $PART_PREC$ este definit astfel:

$$PART_PREC(ci, eff_1, prec_2) = \begin{cases} \text{Adevărat,} & \text{dacă } eff_1 \subseteq prec_2 \cup \{\cup\{CON_TERM(t) \mid t \in ci\}\}, \\ \text{Fals,} & \text{în caz contrar.} \end{cases}$$

Cea de-a doua proprietate ce trebuie asigurată protocolului compus o reprezintă proprietatea de non-distructivitate. Pentru modelarea acestei proprietăți definim predicatul $PART_NONDISTR$: $PR_CC^* \times PR_CC^* \times PR_CC^*$ activat numai în cazul în care dacă un termen este confidențial într-un model participant atunci acesta este confidențial și în celelalte modele. Pentru cele două modele participant considerate anterior, ς_1 și ς_2 , predicatul $PART_NONDISTR$ este definit astfel:

$$PART_NONDISTR(eff_1, prec_2, eff_2) = \begin{cases} \text{Adevărat,} & \begin{aligned} & \text{dacă pentru } \forall EF_1(t_1) \in eff_1 \wedge \forall PR_2(t_2) \in prec_2 \\ & EF_1 \neq CON_CONF \vee \text{dacă } EF_1 = CON_CONF \wedge t_1 = t_2 \text{ atunci} \\ & \exists EF_2(t_2) \in eff_2 : EF_2 = CON_CONF, \end{aligned} \\ \text{Fals,} & \text{în caz contrar.} \end{cases}$$

În cazul în care, pentru două modele participant, predicatele $PART_PREC$ și $PART_NONDISTR$ sunt activate, rezultă că condițiile și efectele modelelor participant pot fi compuse.

Definiția 3. Două modele participant $\varsigma_1, \varsigma_2 \in MPART$, $\varsigma_1 = \langle prec_1, eff_1, tip_1, gen_1, part_1, lan\tau_1 \rangle$, $\varsigma_2 = \langle prec_2, eff_2, tip_2, gen_2, part_2, lan\tau_2 \rangle$ pot fi compuse PE într-un context $prci = \cup\{CON_TERM(t) \mid t \in ci\}$ cu mulțimea termenilor inițiali $ci \in T^*$, dacă $part_1 = part_2$ și:

$$\begin{aligned} & prec_1 \subseteq prci \wedge PART_PREC(ci, eff_1, prec_2) \wedge PART_NONDISTR(eff_1, prec_2, eff_2) \text{ sau} \\ & prec_2 \subseteq prci \wedge PART_PREC(ci, eff_2, prec_1) \wedge PART_NONDISTR(eff_2, prec_1, eff_1). \end{aligned}$$

Compunerea PE este doar primul pas în procesul de compunere secvențială a protocoalelor de securitate. Al doilea pas îl constituie compunerea T, prin care se verifică proprietatea de non-destructivitate a termenilor criptați. Acest proces utilizează un model canonic construit pe baza modelului protocol prezentat anterior.

Model canonic este construit pornind de la o colecție de tipuri de bază. Scopul acestui model este de a reprezenta termenii ce pot fi verificați de participanți și de a pune în evidență termenii ce nu pot fi verificați nici măcar pe baza structurii acestora. Posibilitatea verificării structurilor termenilor recepționați depinde într-o mare măsură de implementarea utilizată. Astfel, dacă implementarea atașează fiecărui termen informații legate de tipul acestuia sau dacă structura termenilor permite extragerea unor informații de tip, atunci verificările pot fi efectuate. În caz contrar, verificările nu pot fi efectuate, iar participanții protocolului pot fi duși în eroare prin injectarea unor pachete malițioase din alte protocoale de către un intrus.

Această confuzie a unui termen cu altul poate duce, de cele mai multe ori, la atacuri, unde intrusul, fără a cunoaște cheile deținute de participanți reușește să convingă participanții asupra utilizării numelui sau altor componente cunoscute în criptarea mesajelor. Asemenea atacuri poartă denumirea de atacuri de *confuzie a tipurilor* [Mea03, HLS00].

Pe de altă parte, dacă verificarea tipurilor este posibilă pentru toți termenii, atacatorul poate în continuare să construiască atacuri de *redare* [Sv94] a mesajelor, unde mecanismul de construire al atacurilor este același ca și în cazul atacurilor de confuzie a tipurilor. Diferența constă în faptul că participanții pot verifica tipurile mesajelor, dar nu pot verifica instanțele acestora.

O metodă pentru identificarea acestor atacuri a fost propusă de autor în [GI06, GI07a]. Utilizând această metodă, atacurile de redare a mesajelor dintr-un protocol în celălalt precum și atacurile de confuzie a tipurilor pot fi identificate nu numai pentru mai multe protocoale ci și în cazul protocoalelor izolate. În continuare se va construi modelul canonic utilizat la identificarea acestor tipuri de atacuri și implicit în procesul de verificare a independenței protocoalelor.

În definiția unui *TipDeBază*, elementele sunt definite pornind de la mulțimile utilizate în construirea termenilor, date în secțiunile precedente. Pentru modelarea termenilor ce nu pot fi verificați datorită cunoștințelor limitate a participanților (e.g. termeni recepționați în timpul execuției) sau datorită inexistenței unui tip atașat în implementare, se utilizează tipul de bază u .

Funcțiile de criptare utilizate la construirea termenilor criptografici sunt aceleași definite pentru modelul precedent, *FuncName*. Termenii canonici sunt definiți astfel:

$$\mathcal{T} ::= . | \text{TipDeBază} \mid (\mathcal{T}, \mathcal{T}) \mid \{\mathcal{T}\}_{\text{NumeFunc}(\mathcal{T})}.$$

Definiția 4. Un model participant canonic (M-PART-C) este o pereche $\langle \text{part}, ml_{cc} \rangle$, unde $\text{part} \in \mathbb{R}$ reprezintă denumirea participantului iar $ml_{cc} \in (\text{Clasificator} \times (\pm \mathcal{T})^*)^*$ reprezintă o mulțime LCC corespunzătoare participantului dat.

Definiția 5. Un model protocol canonic (M-PROT-C) reprezintă o colecție de modele participant canonice în cadrul cărora pentru fiecare nod canonic pozitiv există un nod canonic negativ.

În continuare s-au formulat condițiile pentru independența modelelor protocol, sub forma independenței-istanță și independență-canonică. Independența-istanță presupune că termenii corespunzători nodurilor unui model participant își păstrează proprietățile de securitate în prezența termenilor unui alt model participant. Satisfacerea acestei cerințe este asigurată prin verificarea proprietății de non-destructivitate a precondițiilor și efectelor.

Definiția 6. Două modele participant $\zeta_1 = \langle prec_1, eff_1, tip_1, gen_1, part_1, lan_{t_1} \rangle$ și $\zeta_2 = \langle prec_2, eff_2, tip_2, gen_2, part_2, lan_{t_2} \rangle$ sunt independente-instanță dacă proprietățile acestora sunt non-destructive, adică $PART_NONDISTR(eff_1, prec_2, eff_2)$ și $PART_NONDISTR(eff_2, prec_1, eff_1)$.

Prin extindere, această cerință poate fi aplicată asupra tuturor perechilor de participanți, rezultând astfel independența-instanță a două modele protocol.

Definiția 7. Două modele protocol sunt independente-instanță dacă toate perechile de modele participant sunt independente instanță.

Din aplicarea acestei cerințe asupra a două modele protocol, rezultă că un termen confidențial într-un protocol rămâne confidențial și în celălalt protocol. De aici, rezultă că intrusul nu-și poate folosi puterile de decriptare și criptare pentru a construi atacuri asupra celor două protocoale. De menționat, că această cerință este satisfăcută numai dacă protocoalele în cauză sunt corecte, ceea ce înseamnă că proprietățile date au fost verificate în prealabil. Aceste proprietăți pot fi verificate “off-line” prin utilizarea unui utilitar cum ar fi Scyther [Cre06a], Casper [Low97b], FDR [Low96b] sau Athena [SBP01].

Independența canonică presupune că termenii criptați a două sau mai multe modele participant au o structură disjunctă. Pentru a determina dacă un termen dintr-un model protocol este acceptat în locul unui termen dintr-un alt model protocol, definim predicatul termen canonic acceptat $TCANACC : \mathcal{T} \times \mathcal{T} :$

$$TCANACC(t, t') = \begin{cases} \text{Adevărat,} & \text{dacă } t = t' \vee (t' = \mathbf{u} \wedge t \in \text{TipDeBază}) \vee \\ & (t = \mathbf{u} \wedge t' \in \text{TipDeBază}), \\ TCANACC(t_1, t'_1) \wedge & \text{dacă } (t = (t_1, t_2) \wedge t' = (t'_1, t'_2)) \vee \\ TCANACC(t_2, t'_2), & (t = \{t_1\}_{f(t_2)} \wedge t' = \{t'_1\}_{f(t'_2)} \wedge (t_2 = t'_2 \vee t'_2 = \mathbf{u})), \\ \text{Fals,} & \text{în caz contrar.} \end{cases}$$

Definiția 8. Fie $c_emisCriptat : MPART-C \rightarrow \mathcal{T}^*$ o funcție ce mapează toți termenii emiși criptați pentru un model participant canonic și fie $c_receptCriptat : MPART-C \rightarrow \mathcal{T}^*$ o funcție ce mapează toți termenii recepționați criptați de către lanțurile de clasă procesare. Atunci două modele participant canonice $\zeta_1, \zeta_2 \in MPART-C$ sunt independente-canonic dacă pentru toți termenii $t_1 \in c_emisCriptat(\zeta_1)$ și toți termenii $t_2 \in c_receptCriptat(\zeta_2)$, predicatul $TCANACC(t_1, t_2)$ este fals.

Demonstrarea proprietății de independență s-a realizat prin utilizarea următoarei propoziții.

Propoziția 1. Fie $\xi_1, \xi_2 \in MPROT$ două modele protocol și $\xi'_1, \xi'_2 \in MPROT-C$, modelele canonice corespunzătoare astfel încât $\xi'_1 = c_mprot(\xi_1)$ $\xi'_2 = c_mprot(\xi_2)$. Dacă ξ_1 și ξ_2 sunt independente-instanță iar ξ'_1 și ξ'_2 sunt independente-canonic, atunci ξ_1 și ξ_2 sunt independente.

Pentru validarea metodei de compunere pro puse s-au compus 17 perechi de protocoale din biblioteca SPORE [LSV08] și din biblioteca menținută de John Clark [CJ97]. Verificarea proprietăților de non-destructivitate s-a realizat cu ajutorul utilitarului Scyther [Cre06a]. Printre protocoalele compuse se numără protocoale precum ISO9798, Kerberos cu chei simetrice, sau

CCITTX.509, rezultatele procesului de compunere fiind ilustrate în tabelul 1. Rezultatul afirmativ al procesului compunere (PE sau T) s-a notat cu „DA” iar rezultatul negativ cu „NU”. Deasupra barei s-a reprezentat compunerea $\xi_{P_1} \prec_{\xi}^{PE} \xi_{P_2}$ sau $\xi_{P_1} \prec_{\xi}^T \xi_{P_2}$, iar sub bară compunerea inversă $\xi_{P_2} \prec_{\xi}^{PE} \xi_{P_1}$ sau $\xi_{P_2} \prec_{\xi}^T \xi_{P_1}$. După cum se poate observa, în unele cazuri nu se poate realiza compunerea PE iar în alte cazuri nu se poate realiza compunerea T. De exemplu, compunerea versiunii lui Lowe al protocolului BAN cu protocolul ISO9798 este posibilă doar în ordine inversă $\xi_{P_2} \prec_{\xi}^{PE} \xi_{P_1}$ datorită contextului considerat, unde cheia simetrică necesară protocolului Lowe-BAN nu există, acesta fiind schimbat de participanți prin protocolul ISO9798. Atacurile descoperite prin procesul de compunere T sunt confirmate de acest utilitar, însă, întrucât Scyther nu asigură compunerea secvențială a protocoalelor de securitate, eșuarea compunerii PE nu poate fi detectată prin acesta.

Tabelul 1. Rezultatul compunerii protocoalelor din biblioteci existente

Nr	Protocolul 1 (P1)	Protocolul 2 (P2)	Comp. PE (P1-P2/ P2-P1)	Comp. T (P1-P2/ P2-P1)	Verif. Scyther (P1-P2/ P2-P1)
1.	Lowe-BAN	ISO9798	NU/DA	DA/DA	DA/DA
2.	Lowe-BAN	CCITTX.509 v1	NU/NU	DA/DA	DA/DA
3.	ISO9798	CCITTX.509 v1	DA/DA	DA/DA	DA/DA
4.	ISO9798	CCITTX.509 v1c	DA/DA	DA/DA	DA/DA
5.	CCITTX.509 v1	CCITTX.509 v1c	DA/DA	DA/DA	DA/DA
6.	BAN Concrete RPC	Lowe-BAN	DA/DA	NU/NU	NU/NU
7.	Lowe-Denning-Sacco	Kao-Chow v1	DA/DA	NU/NU	NU/NU
8.	Kao-Chow v1	Kao-Chow v2	DA/DA	DA/DA	DA/DA
9.	Lowe-Denning-Sacco	Kerberos v5	DA/DA	NU/NU	NU/NU
10.	Lowe-Kerberos v5	N-S	DA/DA	NU/NU	NU/NU
11.	Hwang-N-S	Needham-Sch	DA/DA	DA/DA	DA/DA
12.	Needham-Schroeder	CCITTX.509 v1	DA/NU	DA/DA	DA/DA
13.	Lowe-Needham-Schroeder	ISO9798	DA/NU	DA/DA	DA/DA
14.	Otway-Rees	Lowe-BAN	DA/NU	DA/DA	DA/DA
15.	SPLICE/AS	N-S	DA/DA	DA/DA	DA/DA
16.	TMN	Andrew RPC	DA/NU	DA/DA	DA/DA
17.	Yahalom-Lowe	Kao-Chow v1	DA/DA	NU/NU	NU/NU

Capitolul 4 este dedicat evaluării performanțelor protocoalelor de securitate în contextul compunerii protocoalelor cu proprietăți de securitate egale. O asemenea compunere permite eliminarea protocoalelor ce asigură aceleași proprietăți de securitate. În această categorie intră protocoale de schimb de cheie, protocoale de autentificare sau protocoale de schimb de mesaje utilizator. Identificarea și eliminarea protocoalelor cu aceleași proprietăți va asigura nu numai o performanță mărită a protocolului rezultat dar și o mulțime consistentă de termeni rezultați din execuția protocoalelor, având în vedere faptul că, de exemplu, execuția a două protocoale de schimb de cheie va rezulta în două chei de sesiune.

Criteriul de alegere al protocolului inclus în secvența finală de protocoale se poate alege dintr-o varietate de criterii, cum ar fi tipul algoritmilor utilizați, dimensiunea termenilor sau unul mult mai complex, cum ar fi performanța protocoalelor implicate. În cadrul acestei lucrări, pentru procesul de compunere a protocoalelor cu proprietăți de securitate egale s-a utilizat ultimul criteriu, acela de evaluare a performanțelor protocoalelor.

Evaluarea performanțelor protocoalelor de securitate presupune de fapt evaluarea performanțelor operațiilor de construire și procesare a termenilor. Operațiile de construire presupun concatenarea termenilor, generarea cheilor sau a numerelor aleatoare precum și aplicarea algoritmilor criptografici pentru construirea termenilor criptați. Operațiile de procesare reprezintă inversul operațiilor de construire și implică utilizarea algoritmilor criptografici pentru decriptarea termenilor, împărțirea termenilor precum și verificarea validității termenilor recepționați.

Față de metodele existente, metoda propusă [GHIO08] asigură o evaluare comparativă a performanțelor, ce nu necesită detalii legate de implementarea protocoalelor. Metoda de evaluare propusă îmbogățește modelul canonic construit în capitolul precedent printr-o serie de clasificatori ce permit modelarea operațiilor efectuate de participanți pentru construirea și procesarea termenilor. O atenție specială este acordată evaluării performanțelor algoritmilor criptografici întrucât operațiile care implică aplicarea acestor algoritmi sunt cele mai costisitoare [Gut03, CDW06].

Costul fiecărei operații este mapat prin intermediul următoarelor funcții pe baza unei dimensiuni date:

$$f_{sk-c}, f_{sk-d}, f_{pk-c}, f_{pk-d}, f_{pk-s}, f_{pk-vs}, f_h, f_{hm}, f_{kg}, f_{ng}, f_c, f_i : \mathbb{R}^+ \rightarrow \mathbb{R}^+,$$

unde funcțiile date mapează costul operațiilor pentru criptare simetrică, decriptare simetrică, criptare asimetrică, decriptare asimetrică, semnătură digitală, verificare semnătură digitală, hash, hash cu cheie, generare cheie, generare numere aleatoare, concatenare și împărțire.

Definiția funcțiilor corespunzătoare algoritmilor criptografici este construită prin modelarea tipurilor de algoritmi și a operațiilor specifice, în capitolele următoare. În ceea ce privește celelalte funcții, vom considera că funcțiile de generare a cheilor și numerelor aleatoare, f_{kg} și f_{ng} , mapează același timp de execuție indiferent de dimensiunea cheii sau a numărului aleator, astfel încât $f_{kg}(d) = f_{ng}(d) = E_{gen}$, $\forall d \in \mathbb{R}^+$. Vom considera aceeași simplificare și în cazul funcțiilor de concatenare și împărțire, astfel încât $f_c(d) = f_i(d) = E_{ci}$, $\forall d \in \mathbb{R}^+$.

Dimensiunea termenilor canonici joacă un rol deosebit de important în performanța protocoalelor de securitate, fapt pentru care definim funcții de mapare a acestora la valori reale pozitive. Funcția $|_|_ : TipDeBază \rightarrow \mathbb{R}^+$ va determina dimensiunea corespunzătoare fiecărui tip de bază iar pentru maparea dimensiunii datelor criptate vom utiliza funcțiile:

$$\lambda_S, \lambda_A, \lambda_{SM}, \lambda_H, \lambda_{SM-H} : \mathcal{T} \rightarrow \mathbb{R}^+,$$

unde λ_S mapează dimensiunea termenilor canonici criptați simetric, λ_A mapează dimensiunea termenilor canonici criptați asimetric, λ_{SM} mapează dimensiunea semnăturii digitale, λ_H mapează dimensiunea termenilor canonici criptați printr-o funcție hash, λ_{SM-H} mapează dimensiunea semnăturii digitale ce include și hash-ul semnat iar λ_{HM} mapează dimensiunea termenilor canonici criptați cu o funcție hash cu cheie.

În procesul de evaluare a performanțelor protoalelor de securitate vom considera un prim criteriu compararea timpului de execuție a algoritmilor de criptare, decriptare, semnare, verificare a semnăturii, hash. Vom denumi acesta *criteriul algoritmilor criptografici*. În cazul în care nu se poate stabili protocolul cel mai performant pe baza acestui criteriu, vom considera un al doilea criteriu ce presupune compararea efortului criptografic de generare a cheilor criptografice și a numerelor aleatoare, denumit *criteriul generării criptografice*. Dacă în continuare nu se pot diferenția protoalele, se va considera *criteriul concatenării și împărțirii* ce presupune compararea efortului de concatenare și împărțire a termenilor.

Pentru aplicarea primului criteriu, definim funcția $\Phi_I : \text{MPROT-C} \rightarrow \mathbb{R}^+$ ce evaluează performanța unui protocol de securitate pe baza primului criteriu. Pentru criteriul generării criptografice definim funcția $\Phi_{II} : \text{MPROT-C} \rightarrow \mathbb{R}^+$, iar pentru criteriul concatenării și împărțirii, funcția $\Phi_{III} : \text{MPROT-C} \rightarrow \mathbb{R}^+$.

Definiția 9. Fie $\xi_1, \xi_2 \in \text{MPROT}$, două modele protocol și $\xi'_1, \xi'_2 \in \text{MPROT-C}$, corespondenții canonici ai acestora. Spunem că ξ_1 este mai performant decât ξ_2 , notat cu $\xi_1 \ll \xi_2$, dacă:

- $\Phi_I(\xi'_1) < \Phi_I(\xi'_2)$ sau
- $\Phi_I(\xi'_1) = \Phi_I(\xi'_2) \wedge \Phi_{II}(\xi'_1) < \Phi_{II}(\xi'_2)$ sau
- $\Phi_I(\xi'_1) = \Phi_I(\xi'_2) \wedge \Phi_{II}(\xi'_1) = \Phi_{II}(\xi'_2) \wedge \Phi_{III}(\xi'_1) < \Phi_{III}(\xi'_2)$.

Întrucât performanța unui protocol de securitate este influențată semnificativ de performanța algoritmilor criptografici utilizați, aplicarea criteriului algoritmilor criptografici necesită modelarea performanței algoritmilor criptografici și aplicarea modelului rezultat în procesul de evaluare.

Performanțele algoritmilor criptografici sunt influențate de o serie de factori precum: dimensiunea mesajelor asupra cărora se aplică, dimensiunea cheilor, modul de criptare (e.g. ECB, CBC, CFB, OFB). La acestea se adaugă contextul de implementare a algoritmilor, ce include biblioteca criptografică utilizată, resursele procesor, memorie, sistem de operare.

Performanțele algoritmilor criptografici pot fi evaluate prin măsurarea energiei consumate [VW01, Hir03] la aplicarea algoritmilor sau prin măsurarea timpului de execuție [Gut03]. Întrucât energia consumată este direct proporțională cu timpul de execuție [Kir05], în cadrul lucrării performanța algoritmilor s-a evaluat prin măsurarea timpului de execuție.

Implementarea algoritmilor diferă în funcție de biblioteca criptografică utilizată, de unde rezultă și performanțe diferite pentru aceiași algoritmi. Din acest motiv, s-a evaluat performanța algoritmilor în cazul a trei din cele mai cunoscute și larg răspândite biblioteci criptografice: *Cryptlib* [Gut08], *OpenSSL* [OSS08] și *Crypto++* [Cry08].

Modelul matematic corespunzător evoluției medii a performanțelor algoritmilor criptografici s-a construit utilizând metoda celor mai mici pătrate [WMW08, PPS06]. Pentru aproximarea performanțelor medii corespunzătoare operațiilor de criptare și decriptare din cele trei categorii de algoritmi, s-a construit un model polinomial de gradul 3:

$$f(x) = \alpha_4 x^3 + \alpha_3 x^2 + \alpha_2 x + \alpha_1.$$

Pentru fiecare clasă de algoritmi (i.e. simetric, hash, asimetric) și pentru fiecare tip de operație (i.e. criptare, decriptare – unde este cazul) s-au calculat coeficienții $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ prin utilizarea metodei celor mai mici pătrate. Utilizând valorile măsurate și cele estimate, s-a calculat

eroarea medie de estimare pentru toate clasele de algoritmi. Eroarea medie pătratică obținută a fost de 3.714 milisecunde, ceea ce este echivalent cu o deviație de 0.3963% din valoarea maximă măsurată.

Pentru validarea aplicabilității funcțiilor calculate în procesul de evaluare comparativă a performanțelor protocoalelor de securitate din perspectiva performanței algoritmilor criptografici, s-au generat automat 1000 de protocoale. Protocoalele au fost generate prin permutarea componentelor și prin punerea în evidență a celor trei clase de algoritmi de criptare. Structura protocoalelor rezultate conține construcții criptografice pe maxim 3 nivele, luându-se în calcul și generarea unor protocoale în cadrul cărora rezultatul unei criptări este criptat din nou.

Reprezentarea grafică a rezultatelor experimentale pentru un eșantion de 115 perechi de protocoale este dată în figura 1, unde s-a ales algoritmul simetric AES-CBC, algoritmul hash SHA-1 și algoritmul asimetric RSA. După cum se poate observa, funcțiile utilizate asigură o urmărire a traiectoriei implementării. Cu toate acestea, există cazuri în care decizia se dovedește a fi eronată. Aceste cazuri sunt ilustrate prin săgeți care arată către punctele în care raportul estimat diferă de raportul măsurat. De exemplu, în cazul perechii de protocoale cu numărul 50, valoarea raportului măsurat este 0.99, iar valoarea raportului estimat este 1.095. Aceeași situație se poate observa și în cazul perechii de pe poziția 52 sau a perechilor de pe pozițiile 95 și 98.

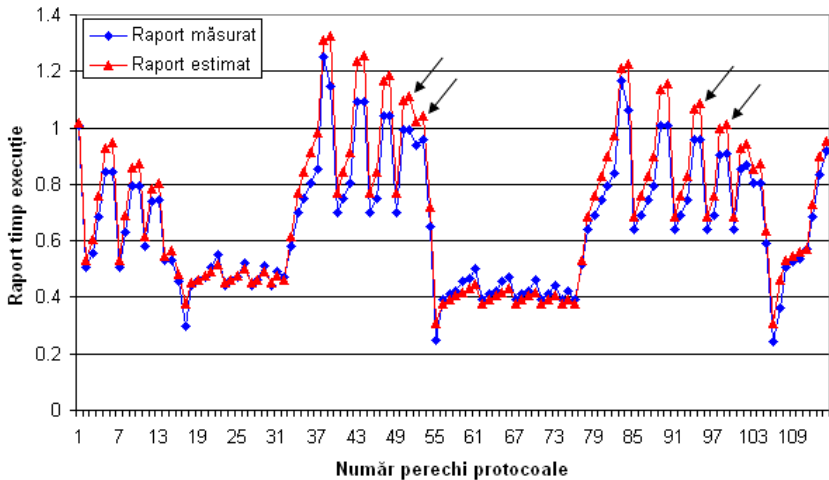


Figura 1. Reprezentarea grafică a raportului dintre timpii de execuție (algoritmii AES-CBC, SHA1, RSA) și estimări pentru termeni de dimensiune 80 octeți

Din calculele efectuate reiese și cauza erorilor de estimare. Aceasta este datorată performanțelor similare ale protocoalelor implicate. Din figura 1 se poate observa faptul că valorile rapoartelor estimate și măsurate se află în jurul valorii 1, fără să depășească valoarea de ± 0.2 . Prin repetarea măsurătorilor s-a constatat dispariția și revenirea erorilor pentru protocoalele ale căror performanță diferă cu cel mult ± 0.2 . Explicația acestei situații este dată prin însăși platforma pe care s-au efectuat măsurările, aceasta nefiind o platformă de timp real.

Cu toate acestea, luând în considerare informațiile lipsă pentru luarea unei decizii asupra performanțelor protocoalelor: lipsa algoritmilor, a dimensiunii cheilor de criptare, a modului de criptare, a dimensiunii termenilor, o eroare maximă de estimare de 6.88% (media pe cele trei

bibliotecii criptografice) poate fi considerată rezonabilă. În aplicații reale, însă, situațiile în care termenii să fie sub dimensiunea de 10 octeți sunt foarte rare. De exemplu, dacă luăm în considerare doar dimensiunea cheilor, acestea variază între 16 și 256 octeți, valori peste minimul de 10 octeți considerat, astfel încât în aplicațiile reale se poate discuta și despre erori sub 4%.

Rezultatele obținute din evaluarea comparativă a performanțelor protocoalelor generate au fost validate și din evaluarea comparativă a 18 perechi de protocoale din biblioteca SPORE [LSV08] și biblioteca menținută de John Clark [CJ97], de unde au rezultat 306 de combinații după eliminarea combinațiilor cu protocoale identice. Din totalul combinațiilor de 306, s-a constatat o estimare eronată pentru un număr de 20 de perechi, adică 6.53%.

Pornind de la modelul protocol propus, în **Capitolul 5** se construiește un model specificație care cuprinde suficiente informații pentru a asigura compunerea și execuția automată a protocoalelor de securitate. În construirea unei reprezentări a specificațiilor s-a ales utilizarea WSDL-S pentru descrierea secvențelor de mesaje (i.e. S-SEC) și OWL pentru descrierea semanticii adnotărilor (i.e. S-SEM) [GH08a, GH08, GH09a]. Structura acestei specificații este reprezentată grafic în figura 2, unde liniile solide cu săgeți reprezintă legături concept-sub-concept, liniile întrerupte reprezintă adnotările condiție-efect iar liniile punctate reprezintă adnotări corespunzătoare termenilor transmiși sau recepționați.

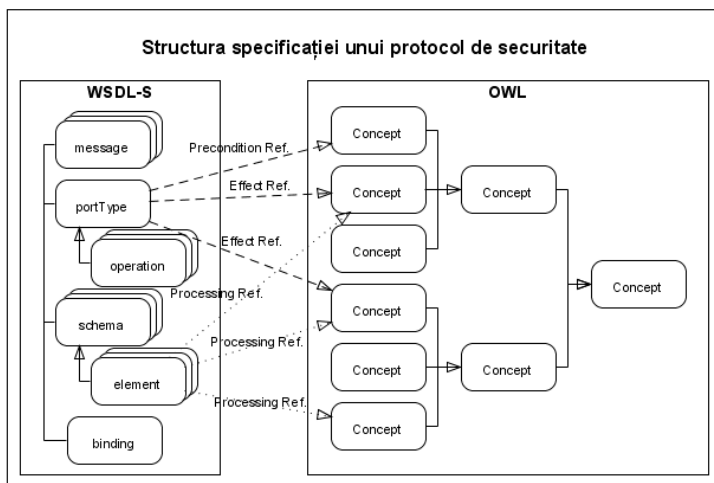


Figura 2. Structura specificației unui protocol de securitate utilizând WSDL-S și ontologiile construite prin limbajul OWL

Unitatea de lucru a specificației secvențiale o reprezintă adnotările prin care se leagă de specificația semantică. O adnotare este formată din două componente: o componentă de identificare [OAS04a] (en. „URI – Uniform Resource Identifier”) a specificației semantice și o componentă ce reprezintă un concept din specificația semantică. În continuare se definește URI ca fiind mulțimea tuturor identificatorilor, CONC ca fiind mulțimea tuturor conceptelor iar CONC* ca fiind mulțimea sub-mulțimilor cu elemente din CONC .

Definiția 10. O adnotare reprezintă o pereche $\langle uri, c \rangle$, unde $uri \in \text{URI}$ iar $c \in \text{CONC}$.

Mulțimea tuturor adnotărilor este notată cu ADNOT iar mulțimea sub-mulțimilor cu elemente din ADNOT este notată cu ADNOT^* .

Un mesaj este definit ca o pereche formată din direcția mesajului și o adnotare.

Definiția 11. Un mesaj reprezintă o pereche $\langle d, a \rangle$, unde $d \in \text{DIR}$ reprezintă direcția mesajului, iar $a \in \text{ADNOT}$ reprezintă o adnotare. Mulțimea tuturor mesajelor este notată cu MSG , iar mulțimea sub-mulțimilor cu elemente din MSG este notată cu MSG^* .

Pe baza definițiilor date, un model secvențial este definit ca fiind un cvadruplu format din precondiție, efect, mesaje și un tip de protocol transport.

Definiția 12. Un model secvențial reprezintă un cvadruplu $\langle \text{PREC}, \text{EFECT}, \text{MSGSEQ}, \text{tiptr} \rangle$, unde $\text{PREC} \in \text{ADNOT}^*$ reprezintă o mulțime ordonată de adnotări ce formează precondițiile modelului secvențial, $\text{EFECT} \in \text{ADNOT}^*$ reprezintă o mulțime ordonată de adnotări ce formează efectele modelului secvențial, $\text{MSGSEQ} \in \text{MSG}^*$ reprezintă o mulțime ordonată de mesaje, iar $\text{tiptr} \in \text{TIPTR}$ reprezintă un tip de transport.

Specificațiile semantice sunt construite pornind de la o ontologie de bază ce este extinsă cu concepte și proprietăți pentru fiecare protocol în parte. Ontologia de bază a fost proiectată utilizând principiile de proiectare întâlnite în numeroase lucrări din literatura de specialitate. De exemplu, principiile date în [Gru93, Gru94] sunt următoarele:

- claritatea;
- coerența;
- extensibilitatea;
- minimizarea dependenței de codificarea simbolurilor;
- minimizarea angajamentului față de ontologie.

La acestea se adaugă și alte principii formulate în [NM01, SWGM05, SGHB02, Sab06, MMS03], dintre care amintim:

- reutilizarea ontologiilor existente;
- generalitatea;
- determinarea momentului oportun pentru introducerea unei noi clase sau a unei noi instanțe;
- alegerea denumirilor claselor și proprietăților;
- utilizarea unui process repetitiv de proiectare și implementare.

Ontologia de bază este o organizare ierarhică a șapte sub-ontologii de domeniu [Sab06] utilizate la descrierea semantică a operațiilor și conceptelor specifice protocolelor de securitate. Ontologia de bază împreună cu conceptele rădăcină ale celor 7 sub-ontologii este ilustrată în figura 3.

Modelul specificației semantice reprezintă un model al OWL, utilizat în procesul de descriere a ontologiilor. În continuare, vom considera mulțimea conceptelor CONC , mulțimea INST ca fiind mulțimea tuturor instanțelor corespunzătoare conceptelor iar mulțimea INST^* ca fiind mulțimea sub-mulțimilor cu elemente din INST .

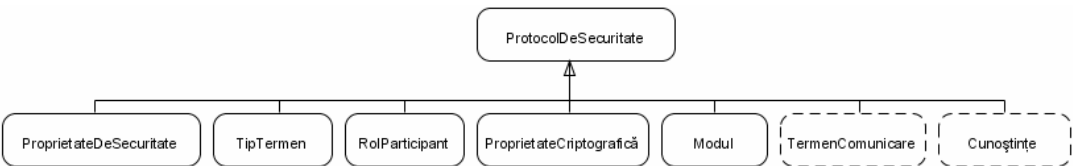


Figura 3. Reprezentarea grafică a ontologiei de bază

Definiția 13. O proprietate reprezintă o pereche $\langle \alpha, \beta \rangle$, unde α reprezintă un identificator unic, iar β reprezintă o construcție sintactică ce denotă denumirea proprietății. Definim $PROPR$ ca fiind mulțimea tuturor proprietăților, iar $PROPR^*$ ca fiind mulțimea sub-mulțimilor cu elemente din $PROPR$.

Definiția 14. Un model semantic reprezintă un triplet $\langle CONC, PROPR, INST \rangle$, unde $CONC \in CONC^*$, $PROPR \in PROPR^*$, iar $INST \in INST^*$.

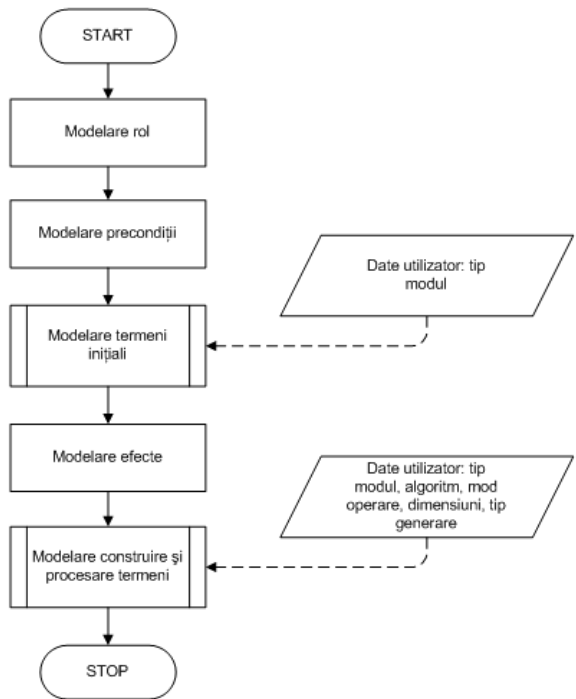


Figura 4. Diagrama de activitate pentru generarea S-SEC-SEM corespunzătoare unui participant

Datorită complexității specificațiilor SEC-SEM, generarea acestora trebuie realizată astfel încât să se asigure menținerea proprietăților de securitate ale protocoalelor inițiale precum și să se asigure toate informațiile necesare execuției protocoalelor pe baza specificațiilor rezultate.

Pentru generarea S-SEC-SEM, s-a considerat ca punct de plecare modelul protocol prezentat în capitolele anterioare. Pentru fiecare protocol astfel descris, generarea specificației SEC-SEM s-a realizat printr-un set de reguli care asigură transformarea componentelor și utilizarea unor algoritmi de transformare [GH08a, GH09a]. Informațiile care nu sunt disponibile în modelul protocol (e.g. algoritmi, dimensiuni termeni generați) sunt introduse de utilizator și sunt modelate prin funcții. Pașii pentru generarea S-SEC-SEM sunt ilustrați în figura 4. Procesul de generare începe cu modelarea rolului participantului și continuă cu modelarea precondițiilor și a termenilor inițiali. Pentru termenii inițiali, utilizatorul trebuie să introducă tipul modulelor utilizate în procesul de încărcare a termenilor, simbolizat în figură printr-o linie întreruptă. După modelarea efectelor se trece la modelarea construirii și prelucrării termenilor, proces care necesită o serie de informații din partea utilizatorului, precum tipul modulelor, algoritmi utilizați, modul de operare al algoritmilor simetrici, dimensiunea termenilor generați precum și tipul de generare aplicat.

Menținerea proprietăților de securitate ale protocolelor de securitate pentru care se construiește o S-SEC-SEM este asigurată din trei aspecte: transformarea structurilor modelului protocol în structuri similare din S-SEC-SEM, principiul *fail-stop* [Gon95] aplicat asupra termenilor recepționați și interogarea utilizatorului pentru informații ce nu se regăsesc în modelul protocol transformat, informații considerate a fi corecte.

Tabelul 2. Timpii de execuție a specificațiilor protocolelor de securitate

Rol participant	Timp procesare specificații (ms)	Timp construire mesaje (ms)	Timp procesare mesaje (ms)	Timp total participant (ms)	Timp total (ms)
Inițiator Lowe-BAN	14.58	11.81	3.68	30.08	48.6
Respondent Lowe-BAN	14.03	2.86	1.62	18.52	
Inițiator ISO9798	13.07	35.78	23.30	72.16	104.79
Respondent ISO9798	13.51	6.87	12.24	32.63	
Inițiator Kerberos 1	22.63	0.83	0.00	23.47	100.57
Inițiator Kerberos 2	12.61	0.55	1.58	14.76	
Inițiator Kerberos 3	2.23	3.34	0.94	6.52	
Respondent Kerberos 1	19.28	0.00	0.41	19.69	
Respondent Kerberos 2	10.81	3.37	1.67	15.87	
Respondent Kerberos 3	5.25	11.41	3.59	20.26	
Inițiator CCITT509	3.29	7.85	0.00	11.14	88.44
Respondent CCITT509	2.88	0.00	74.42	77.3	
Inițiator CCITT509 1c	10.24	65.21	0.00	75.45	161.92
Respondent CCITT509 1c	8.35	0.00	78.12	86.47	
Inițiator Andrew RPC	14.61	12.56	5.04	32.21	61.14
Respondent Andrew RPC	14.89	14.04	4.9	28.93	
Inițiator Guttman-Sim	3.52	10.05	2.14	15.71	43.67
Respondent Guttman-Sim	9.68	16.5	1.78	27.96	

Pornind de la regulile și algoritmi propuși s-au modelat peste 10 protocole de securitate. Protocolele modelate sunt reprezentative în sensul că utilizează clase criptografice diferite și

asigură proprietăți de securitate variate. Protocoalele modelate ne asigură asupra faptului că metoda propusă permite modelarea oricăror protocoale ce utilizează construcții criptografice similare sau care implementează proprietățile de securitate incluse în modelul protocol inițial.

Posibilitatea construirii și procesării automate a mesajelor protocoalelor conform specificațiilor date reprezintă o dovadă a faptului că specificațiile conțin informații suficiente pentru a asigura execuția automată a specificațiilor generate. O parte din rezultatele experimentale sunt date în tabelul 2.

Tabelul 2 include atât protocoale cu 2 participanți cât și protocoale cu 3 participanți. În prima categorie intră protocoalele Lowe-BAN, ISO9798, CCITT X.509 și versiunea CCITT X.509 1c, Andrew RPC, Guttman Simetric, iar în a doua categorie intră protocoalele Kerberos, Dennig-Sacco, TMN și WMF. Modelarea protocoalelor cu trei participanți s-a realizat prin utilizarea precondițiilor și efectelor pentru transferul termenilor dintr-o execuție în cealaltă. De exemplu, în cazul protocolului Kerberos, un prim sub-protocol presupune transmiterea mesajului A, Na de către participantul A , participantului B . Na este precondiție în următorul sub-protocol executat de B cu S . Printr-un asemenea mecanism de tranziție a termenilor între sub-protocoale s-au modelat și celelalte protocoale cu 3 participanți.

Capitolul 6 este dedicat proiectării și implementării unei platforme intermediare care să asigure compunerea și implementarea automată a protocoalelor de securitate. Platforma propusă [GH08b, OHSG08, GH09b] definește două nivele soft: nivelul mesagerie și nivelul serviciu. Nivelul mesagerie asigură implementarea operațiilor de transmitere și recepționare a mesajelor, asigurând și componentele necesare execuției protocoalelor de securitate. Nivelul serviciu oferă o implementare a capabilităților puse la dispoziție de servicii.

Comunicarea între serviciile platformei se realizează prin intermediul nivelului mesagerie, care asigură implementarea protocoalelor de securitate prin structuri XML specifice standardului SOAP [Box01, WWWC07a] și WS-Security [IBM02, OAS04b]. Întrucât structurile WS-Security sunt insuficiente pentru implementarea protocoalelor de securitate considerate, acest standard a fost extins de autor cu o serie de structuri noi, precum tipuri multiple de denumiri utilizator sau tipuri de chei criptografice [GH09c].

Rolul nivelului mesagerie nu se limitează însă doar la execuția unor protocoale predefinite. Acesta asigură și execuția automată a specificațiilor protocoalelor de securitate și a specificațiilor protocoalelor de comunicare. Prin utilizarea nivelului mesagerie, nivelul serviciu asigură interconectarea serviciilor ce implementează protocoale de securitate și de comunicare eterogene.

Nivelul serviciu oferă un set de capabilități ce asigură localizarea serviciilor, localizarea specificațiilor protocoalelor de securitate și a specificațiilor protocoalelor de mesagerie între servicii, compunerea serviciilor și compunerea protocoalelor de securitate precum și un set de capabilități specifice serviciilor de tip resursă. Noutatea principală adusă de nivelul serviciu o reprezintă compunerea automată a specificațiilor protocoalelor de securitate. Compunerea specificațiilor protocoalelor de securitate se realizează în contextul compunerii capabilităților serviciilor resursă. Compunerea capabilităților presupune definirea unui set de precondiții și efecte pentru fiecare serviciu și compunerea acestora în funcție de valorile date. Aceasta reprezintă o formă simplificată de compunere a serviciilor Web, pentru efectuarea unei compuneri mult mai complexe fiind nevoie de efectuarea unor cercetări într-o altă direcție [Cri01, SK03, AMZA04, FJW07, CS08, SCHG08] ce iese în afara scopului lucrării de față.

Platforma propusă beneficiază de o arhitectură orientată pe servicii. O asemenea arhitectură presupune existența unui set de capacități distribuite și de sine stătătoare implementate sub forma unor servicii [OAS06]. Avantajele utilizării unei asemenea arhitecturi sunt multiple. În primul rând, datorită implementării componentelor platformei sub forma unor servicii, acestea devin independente una de cealaltă și oferă o interfață bine definită pentru accesarea capacităților implementate. Totodată, o asemenea arhitectură permite interconectarea serviciilor în scopul creării de servicii noi, compuse, care beneficiază de compunerea capacităților.

Arhitectura generală a platformei este ilustrată în figura 5, unde sunt puse în evidență și tipurile de servicii din care este alcătuită platforma. Platforma presupune existența unui serviciu de nume, al unui serviciu de specificații, al unui serviciu de autorizare și compunere precum și al unuia sau mai multor servicii de tip resursă. Interconectarea serviciilor și accesarea acestora de către aplicațiile utilizator se realizează dinamic, utilizând protocoale de securitate, ale căror specificații sunt localizate la cerere. Pentru fiecare modul al unui serviciu ce pune la dispoziție o interfață de accesare se definește o pereche de specificații: specificația inițiator și specificația respondent. Specificațiile respondent sunt stocate local de către servicii și sunt încărcate la pornirea acestora. În schimb, specificațiile inițiator sunt stocate de partea serviciului de specificații și sunt accesate prin intermediul acestuia.

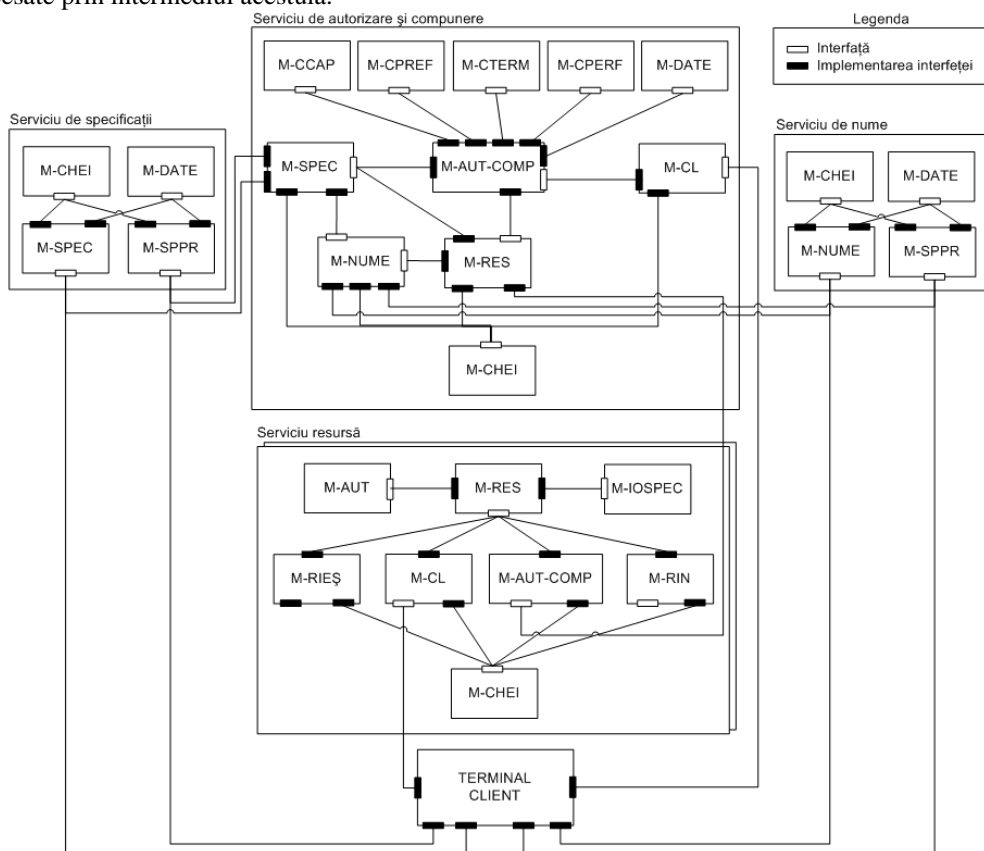


Figura 5. Arhitectura generală a platformei

Arhitectura orientată pe servicii a pus în evidență capacitățile și structura modulară a tipurilor de servicii definite pentru platforma propusă. Implementarea propriu-zisă a platformei presupune dezvoltarea unei colecții de softuri pornind de la standardele existente. Implementarea platformei presupune utilizarea tehnologiilor din domeniul serviciilor Web, domeniu în care există deja implementate o serie de standarde pentru descrierea și descoperirea serviciilor dar și pentru implementarea comunicațiilor între acestea. Protocoalele de comunicare utilizate de regulă în domeniul serviciilor Web, cum sunt HTTP, HTTPS sau FTP oferă independența implementărilor de platforma destinație precum și interoperabilitatea serviciilor. Utilizând asemenea protocoale, serviciile pot fi accesate în spatele zidurilor de protecție, oferă o interfață deschisă de acces peste care pot fi implementate o serie de protocoale mesagerie.

Nivelul mesajelor XML necesar transmiterii mesajelor este construit peste nivelul comunicații și ia forma protocolului SOAP. Standardul SOAP oferă realizarea unei mesagerii XML, utilizarea unei asemenea tehnologii oferind nu numai interoperabilitate ci și ușurință în extinderea protocoalelor cu mesaje noi.

Peste nivelul mesagerie XML s-a elaborat un nivel al protocoalelor de securitate. Implementarea acestor protocoale presupune utilizarea unor specificații adecvate și utilizarea standardului WS-Security extins de autor.

Comunicarea între servicii presupune execuția protocoalelor de securitate și transferul unor mesaje specifice fiecărui tip de serviciu implementat. Mesajele corespunzătoare serviciilor implementate de platformă sunt reprezentate prin intermediul nivelului protocoale serviciu. Aceste mesaje sunt implementate pe baza specificațiilor WSDL specifice serviciilor Web.

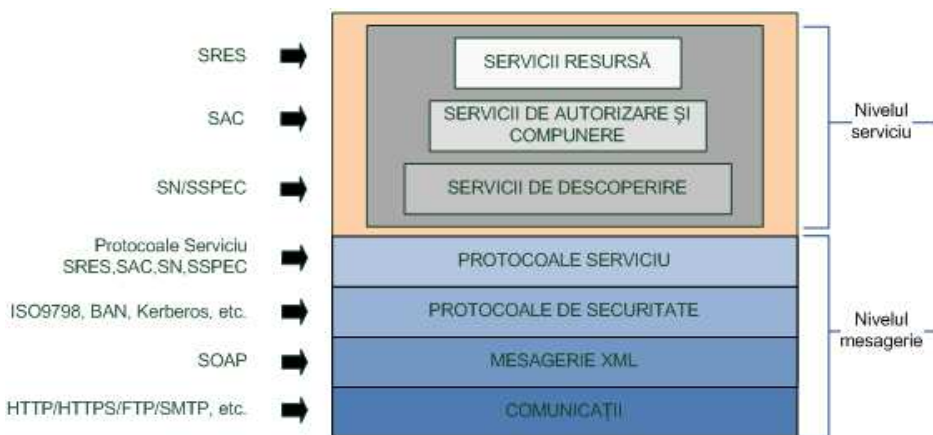


Figura 6. Stiva de softuri a platformei

Stiva soft rezultată este ilustrată în figura 6. Conform acestei figuri, nivelul mesagerie al platformei, este format din 4 sub-nivele: nivelul comunicații, nivelul mesagerie XML, nivelul protocoalelor de securitate și nivelul protocoalelor serviciu.

Gruparea serviciului de nume și a celui de specificații într-un singur tip de serviciu, denumit serviciu de descoperire, s-a realizat pentru reprezentarea capacităților de descoperire a unui anumit serviciu, fapt care constă din localizarea acestuia și accesarea specificației corespunzătoare. Utilizând serviciile de descoperire SN și SSEC, aplicațiile utilizator pot accesa serviciul de

autorizare și compunere, perspectivă ce este reprezentată printr-un nivel superior nivelului de descoperire. O dată autorizate, aplicațiile client vor accesa resursele sistemului.

Accesarea resurselor de către aplicațiile utilizator, presupune interogarea tipurilor de servicii disponibile, a locației SAC și a locației SSPEC prin intermediul SN. Pașii care trebuie efectuați pentru accesarea resurselor sunt în număr de 19, pornind de la localizarea serviciilor, descărcarea specificațiilor și până la accesarea serviciilor compuse, fiind ilustrați în figura 7. Acest număr se reduce semnificativ prin salvarea locațiilor și specificațiilor.

Pentru evaluarea performanței platformei propuse s-au implementat serviciile SN, SAC, SSPEC și 3 tipuri de resurse utilizând interfața de programare a aplicațiilor (en. „API”) NSPR [Moz08a] (en. „Netscape Portable Runtime”) pusă la dispoziție de platforma Mozilla. Pentru implementarea componentelor criptografice ale protocoalelor de securitate s-a utilizat OpenSSL [OSSLO8].

Resursele implementate oferă servicii video utilizatorilor de la camere de supraveghere, servicii de salvare precum și servicii de redare a imaginilor capturate. Implementarea reprezintă un prototip ce demonstrează posibilitatea utilizării compunerii on-line a serviciilor ce utilizează protocoale de securitate precum și a accesării dinamice a acestor servicii, fără cunoașterea în prealabil a protocoalelor de securitate, în domeniul supravegherilor la distanță cu camere video.

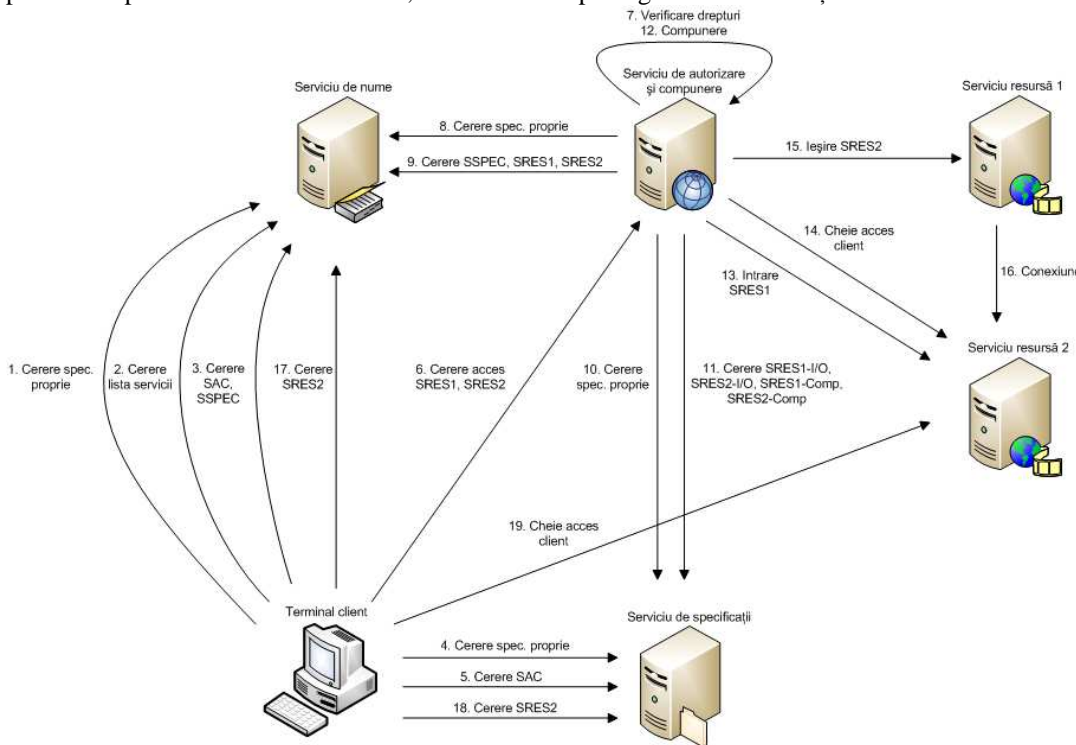


Figura 7. Pașii de accesare a unei resurse compuse

Un prim scenariu de utilizare a sistemului de supraveghere reprezintă accesarea directă a serviciilor sistemului, fără compunerea serviciilor video cu serviciul de salvare. Într-o primă fază s-

a realizat măsurarea timpului de răspuns al SAC în comparație cu timpul total necesar accesării unei resurse cerute, pentru cazul în care clientul nu salvează fișierele specificație (pentru SN, SSPEC și SAC), după care s-au efectuat o serie de măsurători pentru cazul salvării fișierelor. Din figura 8, se observă o scădere a timpului de accesare în al doilea caz cu până la 300ms.

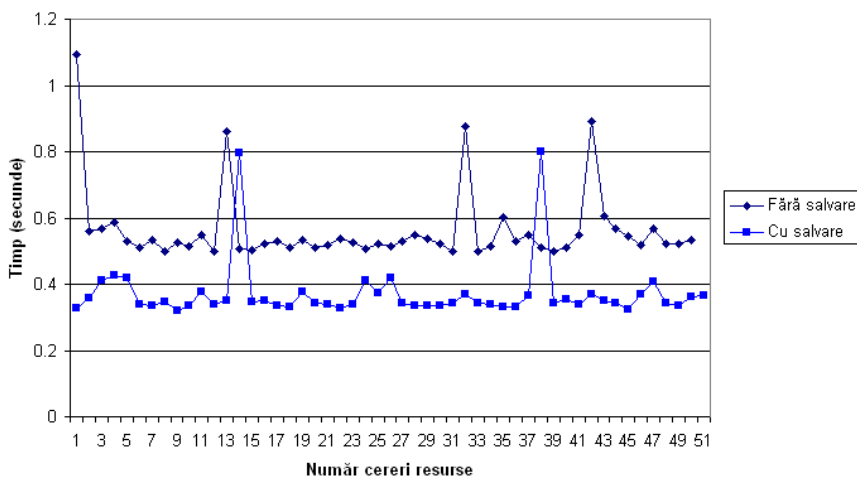


Figura 8. Timpul total de accesare fără salvarea specificațiilor comparativ cu timpul total de accesare cu salvarea specificațiilor

În cadrul sistemului de supraveghere dezvoltat, redarea imaginilor capturate poate fi efectuată doar dacă acestea sunt salvate de către serviciul de salvare. Legarea serviciilor video la acest serviciu se realizează prin intermediul procesului de compunere propus în lucrarea de față. Prin experimentele efectuate s-a măsurat timpul de compunere a specificațiilor, timpul de răspuns al SN, SSPEC și timpul de răspuns al serviciilor resursă compuse.

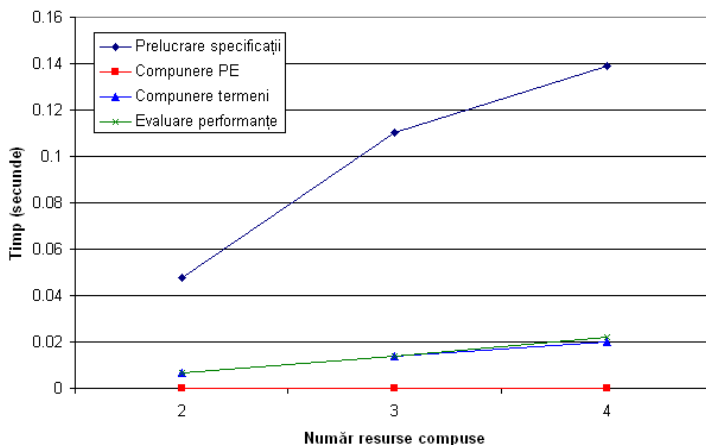


Figura 9. Timpul înregistrat de procesul de compunere a specificațiilor pentru 3 resurse video

Într-o primă fază s-a măsurat timpul necesar compunerii specificațiilor pentru un sistem format din 3 servicii video. Rezultatele obținute sunt afișate în figura 9, unde se poate observa o evoluție relativ liniară a performanței algoritmilor de compunere în funcție de numărul de resurse compuse. Timpul de compunere total este influențat însă și de timpul de prelucrare a specificațiilor, reprezentând operații complexe de prelucrare a fișierelor WSDL-S și OWL.

După cum se poate observa din figura 9, compunerea PE reprezintă o operație rapidă (sub 1 milisecundă) întrucât aceasta presupune doar prelucrarea precondițiilor și efectelor. Pe de altă parte, compunerea termenilor este o operație ce variază între 3 și 20 milisecunde, în funcție de numărul de specificații considerate. Față de compunerea PE, în acest caz trebuie verificate proprietățile non-distructive ale protocoalelor. Totodată, se construiește modelul canonic pe baza căruia se efectuează comparații sintactice. Compunerea specificațiilor cu proprietăți de securitate egale necesită utilizarea modelului canonic precum și modelele algoritmilor criptografici. Din acest motiv, evoluția acestui proces este similar cu cel al compunerii termenilor.

Timpul de accesare al aplicațiilor utilizator depinde însă și de alți factori. După cum se poate observa din figura 10, procesul de compunere reprezintă doar una din componentele ce influențează timpul de accesare. Pe primul loc, cu cel mai mare timp de execuție, se află operațiile de interogare a serviciilor resursă. Aceste operații presupun într-o primă fază execuția specificațiilor protocoalelor de securitate descărcate și transmiterea informațiilor de legare a resurselor precum și a celor de autorizare a utilizatorilor. Creșterea numărului de resurse presupune o creștere a numărului de specificații ce trebuie executate și o creștere a numărului de comenzi ce trebuie transmise, de unde rezultă evoluția din figură.

O influență semnificativă asupra timpului de accesare o are și comunicarea cu SSPEC. Dimensiunea fișierelor specificație descărcate reprezintă un factor cheie în acest caz. Din figură, se poate observa evoluția liniară a performanței interogărilor, dată de altfel și de numărul de specificații ce sunt accesate.

În timpul total de accesare intră și timpul de compunere a specificațiilor. Valorile afișate compunerii pentru cazul de față cuprind atât algoritmii de compunere cât și operațiile de prelucrare a specificațiilor.

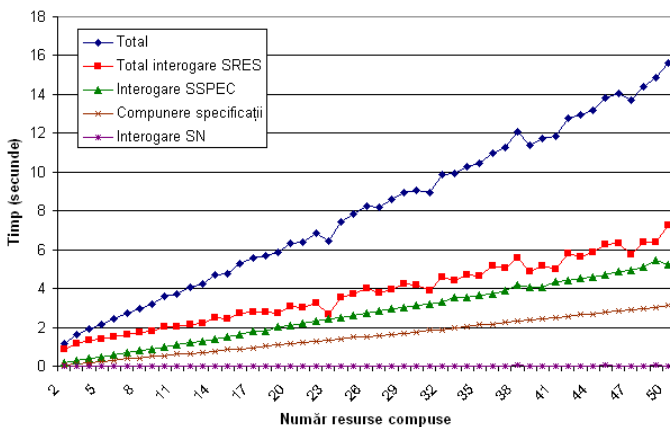


Figura 10. Timpul de accesare a resurselor compuse de către aplicațiile utilizator

Capitolul 7 este dedicat concluziilor, trecerii în revistă a contribuțiilor originale și identificării direcțiilor de continuare a cercetării. Scopul principal al lucrării de față a fost acela de a dezvolta o metodă care să asigure compunerea automată a protocoalelor de securitate și implementarea acestora fără intervenția operatorului uman. Pentru atingerea scopului propus s-a realizat un studiu elaborat asupra celor trei direcții principale identificate: compunerea, evaluarea performanțelor și implementarea protocoalelor de securitate.

Compunerea protocoalelor de securitate presupune combinarea mesajelor componente cu scopul acumulării proprietăților de securitate precum schimbul cheilor de sesiune, autentificarea sau confidențialitatea. În literatura de specialitate s-au identificat două forme de compunere ale acestor protocoale [DDMR07, ACG+08]: compunerea secvențială și compunerea paralelă. Compunerea secvențială presupune existența unei anumite ordini în execuția protocoalelor, față de compunerea paralelă ce nu necesită utilizarea unei asemenea condiții, caz în care protocoalele pot fi rulate și în paralel.

Metoda de compunere propusă se bazează pe un model formal al protocoalelor de securitate și conține toate informațiile necesare procesului de compunere. Metoda propusă poate fi aplicată atât în compunerea secvențială cât și în compunerea paralelă a protocoalelor deoarece compunerea paralelă necesită aplicarea unei sub-mulțimi a operațiilor specifice compunerii secvențiale.

Aplicabilitatea metodei s-a demonstrat prin compunerea a peste 17 perechi de protocoale, validarea rezultatelor obținute fiind realizată prin utilizarea utilitarului Scyther [Cre06a] ce asigură verificarea corectitudinii protocoalelor de securitate prin verificarea modelelor date. Întrucât utilitățile existente nu asigură compunerea protocoalelor, validarea compunerii s-a axat pe verificarea menținerii proprietăților de securitate a protocoalelor compuse. Pe de altă parte, utilitățile existente nu asigură un proces complet de compunere, motiv pentru care acestea nu pot înlocui metoda propusă în lucrarea de față.

Lucrarea de față îmbogățește domeniul compunerii protocoalelor de securitate cu o nouă direcție de cercetare: compunerea protocoalelor cu proprietăți de securitate egale. Importanța tratării acestui aspect rezultă din necesitatea implementării automate a secvenței de protocoale rezultate, existența mai multor protocoale cu aceleași proprietăți determinând scăderea performanțelor sistemului. În acest caz, s-a propus utilizarea unui criteriu de selecție pe baza performanțelor protocoalelor implicate. Deoarece metodele existente de evaluare a performanțelor nu pot fi aplicate în faza de compunere, întrucât acestea se bazează pe informații legate de implementarea protocoalelor, s-a elaborat o metodă nouă bazată pe evaluarea comparativă a performanțelor. Noutatea metodei constă în faptul că nu necesită informații legate de implementarea protocoalelor de securitate (e.g. algoritmi utilizați, biblioteci criptografice, caracteristici fizice ale sistemului).

Pentru automatizarea procesului de compunere aplicat în sisteme bazate pe servicii Web, s-a construit un model de specificație bazat pe tehnologii specifice acestor sisteme, precum WSDL-S [AFM+05, WWWC05] și OWL [WWWC04]. Utilizarea WSDL-S și OWL asigură integrarea cu ușurință a specificațiilor în sisteme bazate pe servicii Web și o serie de avantaje specifice acestor sisteme, precum extensibilitatea și flexibilitatea. WSDL-S asigură descrierea secvenței mesajelor transmise și recepționate, identifică precondițiile și efectele corespunzătoare participantului. OWL asigură o descriere detaliată a mesajelor, a operațiilor de construire și procesare a acestora, conținând toate datele din modelul protocol utilizat în procesul de compunere, dar și o serie de detalii legate de proprietățile criptografice ale termenilor, precum algoritmi utilizați, modul de criptare, dimensiunea cheilor sau a numerelor aleatoare generate.

Validarea modelului specificație, a regulilor și a algoritmilor de generare propuse s-a realizat prin modelarea a 10 protocoale reprezentative. Printre protocoalele modelate se numără Kerberos bazat pe criptografia simetrică, ISO9798 bazat pe criptografia asimetrică (semnături digitale, schimb de cheie Diffie-Hellman), CCITT X.509 bazat pe criptografia asimetrică (criptare și decriptare asimetrică, semnături digitale), Denning-Sacco bazat pe criptografia simetrică. Metoda propusă a făcut posibilă modelarea nu numai a protocoalelor cu doi participanți, ci și a protocoalelor cu trei participanți. Modelarea protocoalelor cu trei participanți este posibilă datorită informațiilor precondiție-efect prin care termenii pot fi transmiși de la un protocol la celălalt. Fiecare protocol este împărțit în sub-protocoale cu 2 participanți, iar pentru fiecare participant se construiește o specificație separată formată din cele două componente: WSDL-S și OWL. Pentru cele 10 protocoale modelate, s-au construit 32 de specificații corespunzătoare participanților cu rol de inițiator și respondent.

Pentru finalizarea procesului de automatizare a compunerii și a implementării protocoalelor de securitate în lucrarea de față s-a elaborat o platformă intermediară cu o arhitectură orientată pe servicii. Platforma propune utilizarea a 4 tipuri de servicii: servicii de nume, servicii de autorizare și compunere, servicii de specificații și servicii resursă.

Pornind de la platforma propusă, s-a construit un sistem de supraveghere video pentru care s-au definit trei tipuri de servicii resursă: servicii de captură video, servicii de salvare video și servicii de redare video. Noutatea adusă de acest sistem constă în compunerea automată a serviciilor captură cu serviciile de salvare, ce presupune atât o compunere a capabilităților cât și o compunere a protocoalelor de securitate utilizate de fiecare pereche de servicii. Întrucât compunerea capabilităților serviciilor Web este în afara scopului lucrării de față, această compunere s-a realizat prin simpla verificare a precondițiilor și efectelor definite pentru fiecare serviciu. În schimb, procesul de compunere a protocoalelor de securitate a fost implementat conform metodei propuse în cadrul lucrării.

Având în vedere conținutul lucrării de față, autorul consideră că nu au fost epuizate cercetările din cadrul direcțiilor propuse și își propune o serie de îmbunătățiri și dezvoltări viitoare, printre care se numără următoarele:

1. Elaborarea unei metode de compunere a protocoalelor cu proprietăți de securitate parțial diferite. În lucrarea de față s-a luat în considerare cazul în care proprietățile de securitate sunt egale în totalitate, însă pot exista protocoale ce asigură un set de proprietăți identice și un set de proprietăți diferite;
2. Elaborarea unei metode de reducere a dimensiunii protocoalelor compuse în sensul concatenării termenilor și a eliminării termenilor redundanți, o asemenea compunere asigurând protocoale mult mai performante;
3. Elaborarea unei metode de împărțire a protocoalelor cu mai mult de doi participanți în sub-protocoale cu doi participanți pentru construirea specificațiilor;
4. Dezvoltarea unui utilitar pentru generarea specificațiilor pornind de la modelul protocol de intrare;
5. Integrarea în cadrul platformei propuse a metodelor existente de compunere a capabilităților serviciilor;
6. Aplicarea platformei propuse în sisteme multimedia de streaming video, audio, informații definite de utilizator (e.g. partajare spațiu de lucru, partajare imagini), ce necesită compunerea serviciilor.

În continuare, sunt prezentate sumar principalele contribuții aduse de lucrarea de față:

1. Dezvoltarea unui model formal al protocoalelor de securitate ce permite analiza și compunerea protocoalelor de securitate;
2. Elaborarea unei metode noi complet automatizate, ce nu necesită intervenția operatorului uman, pentru compunerea secvențială și compunerea paralelă a protocoalelor de securitate;
3. Compunerea a 17 perechi de protocoale de securitate definite în biblioteca SPORE și cea menținută de John Clark;
4. Elaborarea unei metode noi, comparative, de evaluare a performanțelor protocoalelor de securitate;
5. Evaluarea comparativă a performanțelor a 18 protocoale de securitate din biblioteca SPORE și biblioteca menținută de John Clark;
6. Construirea unui model formal al specificațiilor protocoalelor de securitate pornind de la tehnologiile existente în domeniul serviciilor Web: WSDL-S și OWL;
7. Construirea unei ontologii de bază ce asigură modelarea proprietăților protocoalelor de securitate întâlnite în biblioteci existente;
8. Construirea specificațiilor a 10 protocoale de securitate reprezentative, cu un număr de 32 de specificații totale rezultate pentru sub-protocoalele inițiator-respondent identificate;
9. Proiectarea unei platforme intermediare ce asigură compunerea și implementarea automată a protocoalelor de securitate;
10. Implementarea în format XML a protocoalelor binare prin construirea unei noi stive soft bazate pe structurile XML existente ale protocolului SOAP;
11. Implementarea unui sistem de supraveghere bazat pe platforma propusă, cu un număr de 50 de servicii captură, un serviciu de salvare și un serviciu de redare a imaginilor salvate.

LISTA CONTRACTELOR DE CERCETARE ȘI A LUCRĂRILOR PUBLICATE

Contracte de cercetare:

1. “Contribuții la compunerea protocoalelor de securitate bazate pe criterii de performanță”, contract de cercetare intern cu Universitatea “Petru Maior” din Târgu Mureș, Nr. 2445/2007, **director grant**.
2. “Mobile Knowledge Workplace supporting Service Workers”, Contract pentru centrul de cercetare “Kompetenzzentrum für wissensbasierte Anwendungen und Systeme Forschungs- und Entwicklungsgesellschaft”, Austria, Nr. WV-2008-06, în **colaborare** cu IBS (Intelligent Building Solutions, Târgu Mureș).
3. “Monocular Head Mounted net-Conferencing for Service Workers”, Contract pentru centrul de cercetare “Kompetenzzentrum für wissensbasierte Anwendungen und Systeme Forschungs- und Entwicklungsgesellschaft”, Austria, Nr. WV-2007-09, în **colaborare** cu IBS (Intelligent Building Solutions, Târgu Mureș).
4. “Adaptation of net-Conferencing for Mobile Devices”, Contract pentru centrul de cercetare “Kompetenzzentrum für wissensbasierte Anwendungen und Systeme Forschungs- und Entwicklungsgesellschaft”, Austria, Nr. WV-2006-14, în **colaborare** cu IBS (Intelligent Building Solutions, Târgu Mureș).

Lucrări indexate în baze de date internaționale:

1. **Genge Bela**, Haller Piroska, Middleware for Automated Implementation of Security Protocols, 6th European Semantic Web Conference, Heraklion, Greece, Lecture Notes in Computer Science (LNCS 5554), Springer-Verlag, pages 476-490, 2009 (DBLP, EI Compendex, INSPEC, ACM Portal).
2. **Genge Bela**, Haller Piroska, Towards Automated Secure Web Service Execution, Networking 2009, Aachen, Germany, Lecture Notes in Computer Science (LNCS 5550), Springer-Verlag, pages 943-954, 2009 (DBLP, EI Compendex, INSPEC, ACM Portal).
3. **Genge Bela**, Iosif Ignat, Syntactic Sequential Composition of Security Protocols, Journal of Automation Computers Applied Mathematics (ACAM), Volume 17, No. 2, pages 169-178, 2008 (Mathematical Reviews).
4. **Genge Bela**, Haller Piroska, A Modeling Framework for Generating Security Protocol Specifications, 10th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC'08), Workshop on Global Computing Models and Technologies, Timisoara, Romania, pages 362-365, 2008, IEEE Computer Society Press (DBLP, INSPEC, EI Compendex).
5. **Genge Bela**, Haller Piroska, Ovidiu Ratoi, Constructing Security Protocol Specifications for Web Services Intelligent Distributed Computing, Italy, Appears in Studies In Computational Intelligence, Springer-Verlag Berlin Heidelberg, Sept. 2008, Volume 162/2008, pages 245-250, 2008 (DBLP, Ulrichs, SCOPUS, MathSciNet, Zentralblatt).
6. **Genge Bela**, Haller Piroska, Ovidiu Ratoi, Iosif Ignat, Term-based composition of security protocols, 2008 IEEE International Conference on Automation, Quality and Testing, Robotics, Cluj-Napoca, Romania, pages 233-238, May 2008 (IEEE Xplore, ISI Web of Knowledge).
7. **Genge Bela**, Haller Piroska, Iosif Ignat, Ovidiu Ratoi, Informal specification-based performance evaluation of security protocols, 4th IEEE International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Romania, pages 193-200, Aug. 2008 (IEEE Xplore, ISI Web of Knowledge).
8. **Genge Bela**, Iosif Ignat, An Abstract Model for Security Protocol Analysis, WSEAS Transactions on Computers, Issue 2, Volume 6, pages, 207-215, February 2007 (INSPEC, Zentralblatt, Ulrichs).

9. **Genge Bela**, Iosif Ignat, Verifying the Independence of Security Protocols, 3rd IEEE International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Romania, pages 155-163, September 2007 (IEEE Xplore, ISI Web of Knowledge).

10. **Genge Bela**, Iosif Ignat, A typed specification for security protocols, In the Proceedings of the 5th WSEAS Int. Conf. on Data Networks, Communications & Computers, Bucharest, Romania, pages 113-118, October 2006 (INSPEC, Zentralblatt, Ulrichs).

11. **Genge Bela**, Haller Pirooska, Towards a distributed authentication system in Coordinated Mobile Virtual Organizations, In the Proceedings of the 5th RoEduNet IEEE International Conference, Sibiu, Romania, pages 119-123, July 2006 (ISI Web of Knowledge).

Lucrări neindexate:

1. **Genge Bela**, Haller Pirooska, Extending WS-Security to Implement Security Protocols for Web Services, 1st International Conference on Recent Achievements in Mechatronics, Automation, Computer-Sciences and Robotics, Targu Mures, March 20, Appears in Acta Universitatis Sapientiae - Electrical and Mechanical Engineering, In Press, 2009.

2. Magyari Attila, **Genge Bela**, Haller Pirooska, Certificate-based Single Sign-On Mechanism for Multi-Platform Distributed Systems, 1st International Conference on Recent Achievements in Mechatronics, Automation, Computer-Sciences and Robotics, Targu Mures, March 20, Appears in Acta Universitatis Sapientiae - Electrical and Mechanical Engineering, In Press, 2009.

3. **Genge Bela**, Haller Pirooska, Middleware for Implementing Security Protocols, 8th International Conference on Computer Science and Energetics-Electrical Engineering, Sumuleu-Ciuc, Romania, pp. 139-144, October 2008.

4. Ovidiu Ratoi, Haller Pirooska, Ioan Salomie, **Genge Bela**, Component Based Platform for Multimedia Applications, 7th IEEE RoEduNet International Conference, Cluj-Napoca, Romania, pages 40-43, Aug. 2008.

5. **Genge Bela**, Haller Pirooska, A Chained Authentication Model for Virtual Organizations, Acta Universitatis Cibiniensis, VOL. LV, Technical Series, Sibiu, Romania, pages 60-68, 2007.

6. **Genge Bela**, Haller Pirooska, Bindings for Security Protocol Composition, In the Proceedings of the 6th IEEE RoEduNet International Conference, Craiova, Romania, pages 64-69, November 2007.

7. **Genge Bela**, Programming Manual for Smart Houses, Eliberatica – The benefits of Open and Free Technologies, Brasov, Romania, Locally edited booklet, June 2007.

8. **Genge Bela**, Haller Pirooska, Extending the Strand Space Model for Security Protocol Composition, International Scientific Conference "Interdisciplinarity in Engineering", Inter-Ing 2007, Târgu Mures, Romania, pages 1-7, November 2007.

9. **Genge Bela**, Haller Pirooska, Attacks in single and multi-protocol environments, 7th International Conference on Computer Science and Energetics-Electrical Engineering, Oradea, Romania, pages 54-58, October 2007.

10. Haller Pirooska, **Genge Bela**, Security Issues in Wireless Distance Vector Routing Protocols, International Scientific Conference "Interdisciplinarity in Engineering", Inter-Ing 2005, Târgu Mures, Romania, pages 662-668, 2005.

Cărți:

1. **Genge Bela**, Haller Pirooska – Proiectarea sistemelor dedicate și încorporate cu microcontrolerul PIC, editura Universității "Petru Maior", Târgu Mures, 2008.

BIBLIOGRAFIE SELECTIVĂ

- [ACG+08] S. Andova, Cas J.F. Cremers, K. Gjosteen, S. Mauw, S. Mjolsnes, and S. Radomirovic, A framework for compositional verification of security protocols, Information and Computation, Special issue on Computer Security: Foundations and Automated Reasoning, Volume 206, Issues 2-4, pages 425-459, Elsevier, 2008.
- [AFM+05] R. Akkiraju, J. Farrell, J. Miller, M. Nagarajan, M. Schmidt, A. Sheth, K. Verma, Web Service Semantics - WSDL-S, A joint UGA-IBM Technical Note, version 1.0, April 18, 2005.
- [AM03] I. Abdullah and D. Menascé, Protocol specification and automatic implementation using xml and cbse, IASTED conference on Communications, Internet and Information Technology, November 2003.
- [AMZA04] I. Budak Arpinar, Boanerges Aleman-Meza, Ruoyan Zhang, and Angela Maduko, Ontology-Driven Web Services Composition Platform, Proceedings of the IEEE International Conference on E-Commerce Technology, pages 146 – 152, 2004.
- [APS99] George Apostolopoulos, Vinod Peris, Debanjan Saha, Transport Layer Security: How much does it really cost?, In the Proc of IEEE Infocom '99, pages 717-725, 1999.
- [Bac06] Michael Backes, Real-or-random Key Secrecy of the Otway-Rees Protocol via a Symbolic Security Proof, Electronic Notes in Theoretical Computer Science, Vol. 155, pages 111-145, 2006.
- [BBD+03] C. Bodei, M. Buchholtz, P. Degano, H. Riis Nielson, F. Nielson: Automatic Validation of Protocol Narrations, In Proceedings of 16th IEEE Computer Foundations Workshop (CSFW 03), IEEE Computer Society Press, pages 126-140, 2003.
- [BBC+05] Chiara Bodei, Mikael Buchholtz, Michele Curti, Pierpaolo Degano, Flemming Nielson, Hanne Riis Nielson, Corrado Priami, On Evaluating the Performance of Security Protocols, Lecture Notes in Computer Science, Springer, Berlin, 2005.
- [Bla03] Bruno Blanchet, Automatic Verification of Cryptographic Protocols: A Logic Programming Approach, In 5th ACM-SIGPLAN International Conference on Principles and Practice of Declarative Programming, Sweden, pages 1-3, August 2003.
- [BLG+08] C. Blanco, J. Lasheras, R. Valencia-Garcia, E. Fernandez-Medina, A. Toval, and M. Piattini, A systematic review and comparison of security ontologies, Proc. of the Third International Conference on Availability, Reliability and Security, pages 813-820, 2008.
- [Box01] D. Box, A brief history of SOAP, At <http://webservices.xml.com/pub/a/ws/2001/04/04/soap.html>, 2001.
- [But01] Levente Buttyan, Building blocks for secure services: Authenticated key transport and Rational exchange protocols, Thesis, 2001.
- [Can01] Ran Canetti, Universally composable security: A new paradigm for cryptographic protocols, 42nd FOCS, 2001, Revised version (2005), available at eprint.iacr.org/2000/067, 2001.
- [CDW06] Cristian Coarfa, Peter Druschel and Dan S. Wallach, Performance Analysis of TLS Web Servers, ACM Transactions on Computer Systems, 24 (1), pages 39-69, 2006.
- [Cer01] Iliano Cervesato, The Dolev-Yao Intruder is the Most Powerful Attacker, 16th Annual Symposium on Logic in Computer Science, LICS'01, IEEE Computer Society Press, Boston, MA, 2001.
- [Cer02] Iliano Cervesato, Data Access Specification and the Most Powerful Symbolic Attacker in MSR, Software Security - Theories and Systems - ISSS, Springer-Verlag LNCS 2609, Tokyo, Japan, pages 384-416, 2002.
- [CDPW07] Ran Canetti, Yevgeniy Dodis, Rafael Pass, Shabsi Walfish, Universally Composable Security with Global Setup, Theory of Cryptography Conference (TCC), February 2007.
- [Cho06] Hyun-Jin Choi, Security protocol design by composition, Cambridge University, UK, Technical report Nr. 657, ISSN 1476-2986, 2006.
- [CJ97] John Clark and Jeremy Jacob, A Survey of Authentication Protocol Literature: Version 1.0, York University, 17 November 1997.
- [Cla96] John Clark, Attacking Authentication Protocols, www.cs.york.ac.uk/~jac/papers/newHISJ.ps, March 1996.
- [CMV06] C.J.F. Cremers, S. Mauw, E.P. de Vink, Injective Synchronization: an extension of the authentication hierarchy, TCS 6186, Special issue on ARSPA'05, Editors: P. Degano and L. Vigano, 2006, Elsevier.
- [CM05a] C. Cremers, S. Mauw, Checking secrecy by means of partial order reduction, In S. Leue and T. Systa, editors, Germany, september 7-12, 2003, revised selected papers LNCS, Springer, Vol. 3466, 2005.
- [CM05b] Cas Cremers, S. Mauw, Operational semantics of security protocols, In S. Leue and T. Systa, editors, Scenarios: models, transformations and tools, international workshop, Dagstuhl castle, Germany, september 7-12, 2003, revised selected papers LNCS, Springer, Vol. 3466, 2005.
- [CR03] Ran Canetti, Tal Rabin, Universal Composition with Joint State, In Proceedings of CRYPTO 2003, Lecture Notes in Computer Science, vol. 2729. Springer Verlag, New York, pages 265-281, 2003.
- [Cre05] C.J.F. Cremers, Verification of multi-protocol attacks, Computer Science Report CSR 05-10, Eindhoven University of Technology, 2005.

- [Cre06a] Cas Cremers, Scyther - Semantics and Verification of Security Protocols, Thesis, University Press Eindhoven, 2006.
- [Cre06b] Cas J. F. Cremers, Compositionality of Security Protocols: A Research Agenda, *Electr. Notes Theor. Comput. Sci.*, 142, pages 99-110, 2006.
- [Cri01] Valentin Cristea, A Collaborative Environment for High Performance Computing, *IWCC 2001*, pages 47-59, 2001.
- [Cry08] Crypto++ Software Distribution, Version 5.5.2, <http://www.cryptopages.com/>, 2008.
- [CS05] Iliano Cervesato, Mark-Oliver Stehr, Representing the MSR Cryptoprotocol Specification Language in an Extension of Rewriting Logic with Dependent Types. *Electr. Notes Theor. Comput. Sci.*, Vol. 117, pages 183-207, 2005.
- [CS08] V.R. Chifu and I. Salomie - Fluent calculus-Based Web service composition - From OWL-S to Fluent Calculus, *Proceedings of 4th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP 2008)*, Cluj-Napoca, Romania, Aug. 2008, pp. 161-168.
- [CW96] Edmund M. Clarke, Jeanette M. Wing, Formal methods: State of the art and future directions, School of Computer Science, Carnegie Mellon University, 1996.
- [DA99] Dierks T., Allen C., The TLS Protocol, Version 1.0, Request for Comments: 2246, Network Working Group, January 1999.
- [Das00] Neil Daswani, Cryptographic Execution Time for WTLS Handshakes on Palm OS Devices, *Certicom Public Key Solutions*, September 2000.
- [DBF+01] D. Dean, T. Berson, M. Franklin, D. Smetters, and M. Spreitzer, Cryptology as a network service, In *Proceedings of the 7th Network and Distributed System Security Symposium*, San Diego, California, Feb. 2001.
- [DDMP03] Anupam Datta, Ante Derek, John C. Mitchell, Dusko Pavlovic, Secure Protocol Composition, *Proceedings of the 2003 ACM workshop on Formal methods in security engineering*, pages 11-23, 2003.
- [DDMW06] Anupam Datta, Ante Derek, John C. Mitchell, and Bogdan Warinschi, Key Exchange Protocols: Security Definition, Proof Method and Applications, available at <http://eprint.iacr.org/2006/056>, 2006.
- [DDMR07] A. Datta, A. Derek, J. C. Mitchell, A. Roy, Protocol Composition Logic (PCL), *Electronic Notes in Theoretical Computer Science*, Vol. 172, pages 311-358, 2007.
- [Der06] A. Derek, Formal analysis of security protocols: Protocol Composition Logic, PhD Thesis, Computer Science Department, Stanford University, 2006.
- [DKF+03] Grit Denker, Lalana Kagal, Tim Finin, Massimo Paolucci and Katia Sycara, Security for DAML Web Services: Annotation and Matchmaking, *LNCS 2870*, pages 335-350, 2003.
- [DP01] P. Degano and C. Priami, Enhanced Operational Semantics, *ACM Computing Surveys*, 33, 2, pages 135-176, June 2001.
- [DY83] D. Dolev and A.C. Yao. On the security of public key protocols. *IEEE Transactions on Information Theory*, IT-29(2), pages 198-208, 1983.
- [FHG98] F. Javier Thayer Fabrega, Jonathan C. Herzog, Joshua D. Guttman, Strand Spaces: Why is a security protocol correct?, In *Proc. Of the 1998 Symposium on Security and Privacy*, Oakland, California, pages 66-77, 1998.
- [FHG99b] F. Javier Thayer Fabrega, Jonathan C. Herzog, Joshua D. Guttman, Strand spaces: Proving security protocols correct, *Journal of Computer Security* 7, pages 191-230, 1999.
- [FJW07] Feenstra, R. W., Janssen, M., and Wagenaar, R. W., Evaluating Web Service Composition Methods: the Need for Including Multi-Actor Elements, *The Electronic Journal of e-Government Volume 5 Issue 2*, pages 153 - 164, 2007.
- [Fos98] Jim Alves-Foss, Multi-Protocol Attacks and the Public Key Infrastructure, In the *Proceedings of the 21st National Information Systems Security Conference*, pages 566-576, October 1998, available at <http://csrc.nist.gov/nissc/1998/proceedings/toc.pdf>.
- [GF02] Joshua D. Guttman, F. Javier Thayer Fabrega, Authentication tests and the structure of bundles, *Theoretical Computer Science*, Vol. 283, No. 2, pages 333-380, June 2002.
- [GF00] Joshua D. Guttman, F. Javier Thayer Fabrega, Protocol Independence through Disjoint Encryption, *Appears in Proceedings, 13th IEEE Computer Security Foundations Workshop*, Cambridge, pages 24-34, July 2000.
- [GH06] **Genge Bela**, Haller Piroska, Towards a distributed authentication system in Coordinated Mobile Virtual Organizations, In the *Proceedings of the 5th RoEduNet IEEE International Conference*, Sibiu, Romania, pages 119-123, July 2006.
- [GH07a] **Genge Bela**, Haller Piroska, Bindings for Security Protocol Composition, In the *Proceedings of the 6th IEEE RoEduNet International Conference*, Craiova, Romania, pages 64-69, November 2007.
- [GH07b] **Genge Bela**, Haller Piroska, Extending the Strand Space Model for Security Protocol Composition, *International Scientific Conference "Interdisciplinarity in Engineering"*, Inter-Ing 2007, Târgu Mures, Romania, pages 1-7, November 2007.

- [GH07c] **Genge Bela**, Haller Piroska, Attacks in single and multi-protocol environments, 7th International Conference on Computer Science and Energetics-Electrical Engineering, Oradea, Romania, pages 54-58, October 2007.
- [GH07d] **Genge Bela**, Haller Piroska, A Chained Authentication Model for Virtual Organizations, Acta Universitatis Cibiniensis, VOL. LV, Technical Series, Sibiu, Romania, pages 60-68, 2007.
- [GH08a] **Genge Bela**, Haller Piroska, A Modeling Framework for Generating Security Protocol Specifications, 10th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC'08), Workshop on Global Computing Models and Technologies, Timisoara, Romania, IEEE Computer Society Press, pages 362-365, 2008.
- [GH08b] **Genge Bela**, Haller Piroska, Middleware for Implementing Security Protocols, 8th International Conference on Computer Science and Energetics-Electrical Engineering, Sumuleu-Ciuc, Romania, pp. 139-144, October 2008.
- [GH09a] **Genge Bela**, Haller Piroska, Towards Automated Secure Web Service Execution, Networking 2009, Aachen, Germany, May 11-15, Lecture Notes in Computer Science (LNCS 5550), Springer-Verlag, pp. 943-954, 2009.
- [GH09b] **Genge Bela**, Haller Piroska, Middleware for Automated Implementation of Security Protocols, 6th Eutopian Semantic Web Conference, Heraklion, Greece, Lecture Notes in Computer Science (LNCS 5554), Springer-Verlag, pp. 476-490, 2009.
- [GH09c] **Genge Bela**, Haller Piroska, Extending WS-Security to Implement Security Protocols for Web Services, 1st International Conference on Recent Achievements in Mechatronics, Automation, Computer-Sciences and Robotics, Targu Mures, March 20, Appears in Acta Universitatis Sapientiae - Electrical and Mechanical Engineering, 2009, In Press.
- [GHO08] **Genge Bela**, Haller Piroska, Ovidiu Ratoi, Constructing Security Protocol Specifications for Web Services, C. Badica et al. (Eds.), Intelligent Distributed Computing, Italy, Appears in Studies In Computational Intelligence, Springer-Verlag Berlin Heidelberg, pages 245-250, Sept. 2008.
- [GHO108] **Genge Bela**, Haller Piroska, Ovidiu Ratoi, Iosif Ignat, Term-based composition of security protocols, 2008 IEEE International Conference on Automation, Quality and Testing, Robotics, Cluj-Napoca, Romania, pages 233-238, May 2008.
- [GHIO08] **Genge Bela**, Haller Piroska, Iosif Ignat, Ovidiu Ratoi, Informal specification-based performance evaluation of security protocols, 4th IEEE International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Romania, pages 193-200, August 2008.
- [GI06] **Genge Bela**, Iosif Ignat, A typed specification for security protocols, In the Proceedings of the 5th WSEAS Int. Conf. on Data Networks, Communications & Computers, Bucharest, Romania, pages 113-118, October 2006.
- [GI07a] **Genge Bela**, Iosif Ignat, An Abstract Model for Security Protocol Analysis, WSEAS Transactions on Computers, Issue 2, Volume 6, pages 207-215, February 2007.
- [GI07b] **Genge Bela**, Iosif Ignat, Verifying the Independence of Security Protocols, 3rd IEEE International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Romania, pages 155-163, September 2007.
- [GI08] **Genge Bela**, Iosif Ignat, Syntactic Sequential Composition of Security Protocols, Journal of Automation Computers Applied Mathematics (ACAM), Volume 17, No. 2, pages 169-178, 2008.
- [GM05] Gorgan D., Melenti C., Parallel and distributed graphical processing on Grid structure of geographic and environment data, Ed Mediamira, 2005.
- [GMB+07] Gorgan D., Muresan, O., Bacu, V., Melenti, C., Safta, D.: Satellite Image Processing by MedioGRID Platform Kernel, CSCS-16: 16th International Conference on Control Systems and Computer Science, May 22-25, 2007, Bucharest, Vol. 2, ISBN: 978-973-718-743-7, pp. 142-149.
- [Gol04] O. Goldreich, Foundations on Cryptography, Cambridge Press, Vol 1 (2001), Vol 2 (2004).
- [Gon95] Li Gong, Fail-Stop Protocols: An Approach to Designing Secure Protocols, In Proceedings of the 5th IFIP Working Conference on Dependable Computing for Critical Applications, Dependable Computing and Fault-Tolerant Systems, pages 44-55, Urbana-Champaign, Illinois, September 1995.
- [Gru93] Gruber, T., A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition, 5(2), pages 199 – 220, 1993.
- [Gru94] Gruber, T., Toward Principles for the Design of Ontologies Used for Knowledge Sharing. IJHCS, 43(5/6), pages 907-928, 1994.
- [GSVG08] Gorgan D., Stefanut T., Veres M., Gabos I., Knowledge Assessment Based on Evaluation of 3D Graphics Annotation in Lesson Content, 4th Symposium of the WG HCI&UE of the Austrian Computer Society - USAB 2008, Nov. 2008, Graz, Austria. Published by Springer in the Lecture Notes in Computer Science (LNCS 5298) Series, pp. 145-160.
- [Gua97] Guarino, N., Understanding, building and using ontologies, International Journal of Human-Computer Studies, pages 293 – 310, 1997.
- [Gut01] J. D. Guttman, Key compromise and the authentication tests, Electronic Notes in Theoretical Computer Science, 2001.

- [Gut02] Joshua D. Guttman, Security protocol design via authentication tests, In Proceedings of the 15th IEEE Computer Security Foundations Workshop, IEEE CS Press, June, 2002.
- [Gut03] Peter Gutmann, Performance Characteristics of Application-level Security Protocols, 2003, available at http://www.cs.auckland.ac.nz/~pgut001/pubs/app_sec.pdf.
- [Gut08] Peter Gutmann, Cryptlib Encryption Toolkit, <http://www.cs.auckland.ac.nz/~pgut001/cryptlib/index.html>, 2008.
- [HAD07] Michael Herrmann, Muhammad Ahtisham Aslam, Oliver Dalferth, Applying Semantics (WSDL, WSDL-S, OWL) in Service Oriented Architectures (SOA), 10th Intl. Protégé Conference - July 15-18, 2007 - Budapest, Hungary.
- [HG05] Haller Pirooska, **Genge Bela**, Security Issues in Wireless Distance Vector Routing Protocols, International Scientific Conference "Interdisciplinarity in Engineering", Inter-Ing 2005, Târgu Mureş, Romania, pages 662-668, 2005.
- [Hir03] S. Hirani, Energy consumption of encryption schemes in wireless devices, Master's thesis, Telecommunications Program, University of Pittsburgh, Pittsburgh, Pennsylvania, 2003.
- [HLS00] James Heather, Gavin Lowe, Steve Schneider, How to Prevent Type Flaw Attacks on Security Protocols, In the Proc. of the 13th Computer Security Foundations Workshop, IEEE Computer Society Press, July 2000.
- [HM02] Alan Harbitter, Daniel A. Menasce, A methodology for measuring the performance of Authentication Protocols, ACM Transactions on Information and System Security, Vol. 5, No. 4, pages 458-491, November 2002.
- [Hol05] G. Hollestelle, Systematic Analysis of Attacks on Security Protocols, Master's Thesis, Department of Mathematics and Computer Science, Univ. of Eindhoven, Nov. 2005.
- [HPJ03b] Yih-Chun Hu, Adrian Perring, David B. Johnson, Packet Leashes: A Defence against Wormhole Attacks in Wireless Networks, INFOCOM 2003, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies, Vol. 3, pages 1976-1986, March 2003.
- [HT96] Nevin Heintze and J. D. Tygar, A Model for Secure Protocols and Their Compositions, IEEE Transactions on Software Engineering, Vol. 22, No. 1, Jan. 1996.
- [HY06] Hann-Jang Ho and Rong Jou Yang, A Comparison of Secure Mechanisms for Mobile Commerce, In the Proc. of the 7th WSEAS International Conference on Mathematics & Computers in Business & Economics, Cavtat, Croatia, June 13-15, pages 24-28, 2006.
- [IBM02] IBM Corporation and Microsoft Corporation, Security in a web services world: A proposed architecture and roadmap, available at <http://msdn.microsoft.com/library/en-us/dnwssecur/html/securitywhitepaper.asp>, April 2002.
- [IBM07] IBM Corporation, Web Services Federation Language (WS-Federation), available at <http://www.ibm.com/developerworks/library/specification/ws-fed/>, May 2007.
- [ISO94a] ISO/IEC 9798-2, Information technology – Security techniques – Entity authentication – Part 2: Mechanisms using symmetric encipherment algorithms, International Organization for Standardization, Geneva, Switzerland, 1994 (first edition).
- [ISO94b] ISO/IEC 9797, Information technology – Security techniques – Data integrity mechanism using a cryptographic check function employing a block cipher algorithm, International Organization for Standardization, Geneva, Switzerland, 1994 (second edition).
- [Kir05] Phongsak Kiratwintakorn, Energy efficient security framework for wireless Local Area Networks, PhD Thesis, University of Pittsburgh, 2005.
- [KLM05] Anya Kim, Jim Luo, and Myong Kang, Security Ontology for Annotating Resources, R. Meersman and Z. Tari (Eds.): CoopIS/DOA/ODBASE 2005, LNCS 3761, Springer-Verlag Berlin Heidelberg, pages 1483-1499, 2005.
- [KSW97] J. Kelsey, B. Schneier, and D. Wagner, Protocol interactions and the chosen protocol attack, In Proceedings of the 5th International Workshop on Security Protocols, pages 91-104, April 1997.
- [LLP+07] Holger Lausen, Ruben Lara, Axel Polleres, Jos de Bruijn and Dumitru Roman, Semantic Annotation for Web Services, Semantic Web Services – Concepts, Technologies and Applications, Rudi Studer, Stephan Grimm, Andreas Abecker (eds.), pages 179-209, Springer-Verlag, 2007.
- [Low96b] Lowe, G. Breaking and fixing the Needham-Schroeder public-key protocol using FDR. In Tools And Algorithms for the Construction and Analysis of Systems, vol. 1055 of Lecture Notes in Computer Science, Springer-Verlag, pages 147-166, 1996.
- [Low97a] Gavin Lowe, A Family of Attacks upon Authentication Protocols, Report 1997/5, Dept. of Mathematics and Computer Science, University of Leicester, 1997.
- [Low97b] Gavin Lowe, Casper: A compiler for the Analysis of Security Protocols, In Proc. CSFW '97, Rockport. IEEE, 1997.
- [Low98] Gavin Lowe, Towards a completeness result for model checking of security protocols. Technical Report 1998/6, Dept. of Mathematics and Computer Science, University of Leicester, 1998.
- [LSV08] Laboratoire Specification et Verification, Security Protocol Open Repository (2008), <http://www.lsv.ens-cachan.fr/spore/>.

- [Maf05] Matteo Maffei, Tags for Multi-Protocol Authentication, *Electronic Notes in Theoretical Computer Science*, Vol. 128, Issue 5, pages 55-63, 2005.
- [MBJ+02] L. Mengual, N. Barcia, E. Jimnez, E. Menasalvas, J. Setin and J. Ygez, Automatic implementation system of security protocols based on formal description techniques, *Proceedings of the Seventh International Symposium on Computers and Communications*, pages 355–401, 2002.
- [Mea03] Catherine Meadows, A Procedure for Verifying Security Against Type Confusion Attacks, 16th IEEE Computer Security Foundations Workshop (CSFW'03), page 62, 2003.
- [MMS03] A. Maedche and B. Motik and L. Stojanovic, Managing Multiple and Distributed Ontologies in the Semantic Web, *The VLDB Journal — The International Journal on Very Large Data Bases*, Volume 12 , Issue 4, pages 286 – 302, November 2003.
- [Moz08a] Mozilla Corporation, NSPR, Netscape Portable Runtime, <http://www.mozilla.org/projects/nspr/>, 2008.
- [NHR05] C. Neuman, T. Yu, S. Hartman, K. Raeburn, The Kerberos Network Authentication Service (V5), <http://www.ietf.org/rfc/rfc4120> (July 2005).
- [NM01] Natalya F. Noy and Deborah L. McGuinness, Ontology Development 101: A Guide to Creating Your First Ontology, Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001.
- [NS78] Roger M. Needham and Michael D. Schroeder. Using encryption for authentication in large networks of computers. *Communications of the ACM*, 21(12), pages 993-999, December 1978.
- [OAS04a] Organization for the Advancement of Structured Information Standards, Universal Description, Discovery and Integration, UDDI Spec Technical Committee Draft, available at http://www.uddi.org/pubs/uddi_v3.htm, 2004.
- [OAS04b] Organization for the Advancement of Structured Information Standards, Web Services Security: SOAP Message Security 1.0 (WS-Security 2004), Available at <http://www.oasis-open.org/committees/download.php/5941/oasis-200401-wss-soap-message-security-1.0.pdf>, March 2004.
- [OAS05a] Organization for the Advancement of Structured Information Standards, eXtensible Access Control Markup Language (XACML) TC version 2.0, available at http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml, February 2005.
- [OAS05b] Organization for the Advancement of Structured Information Standards, Security Assertion Markup Language (SAML), version 2.0, available at <http://saml.xml.org/saml-specifications>, March 2005.
- [OAS06] Organization for the Advancement of Structured Information Standards, OASIS Reference Model for Service Oriented Architecture 1.0, Official OASIS Standard, Oct. 12, available at <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf>, 2006.
- [OAS07a] Organization for the Advancement of Structured Information Standards, Web Service Trust (WS-Trust) version 1.3, available at <http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.html>, March 2007.
- [OAS07b] Organization for the Advancement of Structured Information Standards, Web Service Secure Conversation, version 1.3, available at <http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/ws-secureconversation-1.3-os.html>, March 2007.
- [OHSG08] Ovidiu Ratoi, Haller Piroska, Ioan Salomie, **Genge Bela**, Component Based Platform for Multimedia Applications, 7th IEEE RoEduNet International Conference, Cluj-Napoca, Romania, pages 40-43, Aug. 2008.
- [OSS08] OpenSSL Project, version 0.9.8h, available at <http://www.openssl.org/>, 2008.
- [Pat06] Victor Valeriu Patriciu, Semnături electronice și securitatea informatică, Editura All, București, 2006.
- [Pau01] Lawrence C. Paulson, Relations between secrets: two formal analyses of the Yahalom protocol, *Journal of Computer Security*, Volume 9, Issue 3, pages 197-216, 2001.
- [PPBC98] Victor Valeriu Patriciu, Monica Ene Pietroșeanu, Ion Bica, Costel Cristea, *Securitatea informatică în Unix și Internet*, Editura Tehnică, București, 1998.
- [PSW+02] Adrian Perrig, Robert Szewczyk, Victor Wen, David Culler, and J.D.Tygar, Spins: Security protocols for sensor networks, *Wireless Networks*, Vol. 8, pages 521 – 534, 2002.
- [SA99] Frank Stajano and Ross Anderson, The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks, *Security Protocols*, 7th International Workshop Proceedings, Lecture Notes in Computer Science, 1999.
- [Sab06] Marta Sabou, Building Web Service Ontologies, Thesis, Vrije University of Amsterdam, 2006.
- [SBP01] Song D., Berezin S., Perring A., Athena: a novel approach to efficient automatic security protocol analysis, *Journal of Computer Security*, 2001.
- [Sch96] Bruce Schneier, *Applied Cryptography – Second Edition*, John Wiley & Sons, 1996.
- [SCHG08] Ioan Salomie, Viorica Rozina Chifu, Ioana Harsa, Marius Gherga - Towards Automated Web service Composition with Fluent Calculus and Domain Ontologies, The 10th International Conference on Information Integration and Web-based Applications & Services (iiWAS2008), Linz (Austria), November 2008, To Appear.

- [SGHB02] Robert Stevens, Carole Goble, Ian Horrocks and Sean Bechhofer, Building a Bioinformatics Ontology Using OIL, IEEE Information Technology, Biomedicine special issue on Bioinformatics, vol. 6, pages 135-141, 2002.
- [SK03] B. Srivastava and J. Koehler. Web Service Composition - Current Solutions and Open Problems. ICAPS 2003 Workshop on Planning for Web Services, 2003.
- [SPP01] Song Dawn, Adrian Perrig, and Doantam Phan, AGVI - Automatic Generation, Verification, and Implementation of Security Protocols, In Proceedings of the 13th Conference on Computer Aided Verification (CAV), Paris, France, July 2001.
- [SRW05] Adam Stubblefield, Aviel D. Rubin, Dan S. Wallach, Managing the Performance Impact of Web Security, Electronic Commerce Research, No. 5, Springer Science + Business Media, Inc. Manufactured in the Netherlands, pages 99–116, 2005.
- [SR06] Rodan Sass, Razvan Rughinis, Implementing Layer 2 Attack Mitigation Techniques in a Cisco Switching Environment, 5th International RoEduNet IEEE international Conference, Sibiu, Romania, pages 129-133, June 2006.
- [SWG05] Marta Sabou, Chris Wroe, Carole Goble, Gilad Mishne, Learning domain ontologies for Web service descriptions: an experiment in bioinformatics, Proceedings of the 14th international conference on World Wide Web, pages 190-198, 2005.
- [Sv94] Paul Syverson, A taxonomy of replay attacks, In Proc. of the 7th IEEE Computer Security Foundations Workshop, 1994, pages 131-136.
- [TH05] Benjamin Tobler and Andrew C. M. Hutchison, Generating Network Security Protocol Implementations from Formal Specifications, IFIP International Federation for Information Processing, Springer Boston, pages 33-54, 2005.
- [Tsa02] Chii-Ren Tsai, Non-Repudiation in Practice, The Second International Workshop for Asian Public Key Infrastructures, Taipei, Taiwan, October 30-November 01, 2002.
- [Vog05] Harald Vogt, Exploring Message Authentication in Sensor Networks, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, Volume 3313, pages 19-30, 2005.
- [VW01] M. Viredaz and D. Wallach, Power evaluation of a handheld computer: A case study, Compaq Western Research Lab, Tech. Rep. 2001/1, 2001.
- [WDY+03] Wenliang Du, Jing Deng, Yunghsiang S. Han, Pramod K. Varshney, Jonathan Katz and Aram Khalili, A Pairwise Key Pre-Distribution Scheme for Wireless Sensor Networks, ACM, 2003.
- [Wei99] Christoph Weidenbach, Towards an automatic analysis of security protocols, In Harald Ganzinger, editor, Proceedings of the 16th International Conference on Automated Deduction, volume 1632 of LNAI, Springer, pages 378–382, 1999.
- [Wer07] Bengt Werstén, Implementing the Transport Layer Security Protocol for Embedded Systems, Masters Thesis, Department of Electrical Engineering, Sweden, LITH-ISY-EX-06/3985-SE, 2007.
- [Wie97] Wielemaker, J., SWI-Prolog 2.9.6, Reference Manual, University of Amsterdam, 1997.
- [Win98] D. Winer. RPC over HTTP via XML. At <http://davenet.scripting.com/1998/02/27/rpcOverHttpViaXml>, 1998.
- [WKO04] Tao Wan, Evangelos Krankis, P.C. van Oorschot, S-RIP: A Secure Distance Vector Routing Protocol, Applied Cryptography and Network Security, Second International Conference, Yellow Mountain, China, June 2004.
- [WMW08] Wolfram Math World, Least Squares Fitting, <http://mathworld.wolfram.com/LeastSquaresFitting.html>, 2008.
- [WP06] Jake Wu, Panos Periorellis, Evaluation of Authentication-Authorization Tools for VO security, Proceedings of the UK e-Science All Hands Meeting, 2006.
- [WTLS99] WAP Forum, Wireless Transport Layer Security Specification Version 1.1, 11.2.1999.
- [WWWC99a] World Wide Web Consortium, Resource Description Framework (RDF) Model and Syntax Specification, W3C Proposed Recommendation 22 February 1999.
- [WWWC99b] World Wide Web Consortium, Resource Description Framework (RDF) Schema Specification, W3C Proposed Recommendation 3 March 1999.
- [WWWC99c] World Wide Web Consortium, Extensible Stylesheet Language Transformations (XSLT), W3C Proposed Recommendation 16 November 1999.
- [WWWC01] World Wide Web Consortium, DAML+OIL Reference Description, W3C Note 18 December 2001.
- [WWWC03] World Wide Web Consortium, Web Services Description Language (WSDL) 1.2, <http://www.w3.org/TR/wsdl>, March 2003.
- [WWWC04] World Wide Web Consortium, OWL Web Ontology Language Reference, W3C Recommendation 10 February 2004.
- [WWWC05] World Wide Web Consortium, Web Service Semantics - WSDL-S, W3C Member Submission, 7 November, 2005.
- [WWWC06] World Wide Web Consortium, WS-Policy, available at <http://www.w3.org/Submission/WS-Policy/>, April 2006.

- [WWWC07a] World Wide Web Consortium, Simple Object Access Protocol (SOAP) 1.2, <http://www.w3.org/TR/soap/>, April 2007.
- [WWWC07b] World Wide Web Consortium, Semantic Annotations for WSDL and XML Schema, W3C Recommendation 28 August 2007.
- [XRC+04] Ning Xu, Sumit Rangwala, Krishna Kant Chintalapudi, Deepak Ganesan, Alan Broad, Ramesh Govindan, Deborah Estrin, „A wireless sensor network For structural monitoring”, Proceedings of the 2nd international conference on Embedded networked sensor systems, Baltimore, MD, USA, pages 13-24, 2004.
- [Ylo96] Tatu Ylonen, SSH-Secure login connections over the Internet. In Proceedings of the Sixth USENIX Security Symposium, pages 37-42, July 1996.
- [Ylo06] T. Ylonen, The Secure Shell (SSH) Protocol Architecture, SSH Communications Security Corp, RFC4251, January 2006.
- [Zav97] Zave, P., Classification of Research Efforts in Requirements Engineering, ACM Computing Surveys, 29(4), pages 315-321, 1997.
- [Zha05] Meiyuan Zhao, Performance Evaluation of Distributed Security Protocols Using Discrete Event Simulation, Dartmouth Computer Science Technical Report TR2005-559, PhD Thesis, Dartmouth College, Hanover, New Hampshire, October, 2005.
- [ZQ03] Zhang, N., Shi, Q., An Efficient Protocol for Anonymous and Fair Document Exchange, Computer Networks Journal, Elsevier Science Publisher, Vol. 41, No. 1, pages 19-28, 2003.

