UNIVERSITATEA BABEŞ-BOLYAI

FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ

Securitate în reţele şi în tehnologii wireless

Rezumatul tezei de doctorat

Conducător ştiinţific: **Doctorand:**

Prof. Univ. Dr. Florian Mircea Boian **Bogdan Călin Crainicu**

Cluj Napoca, 2015

Cuprinsul tezei de doctorat

Cuvinte cheie

proprietate de a fi aleatoriu, probabilitate algoritmică, teoria informaţiei algoritmice, secvenţe/şiruri aleatorii, generator de numere pseudo-aleatorii, algoritm de criptare secvenţial, *RC*4, *KSA*, *RC*4*m*, *KSAm*, vector de iniţializare *IV*, vulnerabilitate invariantă, vulnerabilitate *IV*, diferenţiator, diferenţiator$_{\text{indicator}}$, corelare, corelare$_{\text{diferenţiator}}$, corelare pe şablon, diferenţiator$_{\text{puternic}}$, diferenţiator$_{\text{slab}}$, stare X_N-incertă, cheie X_N-slab-incertă.

Lista de publicații

1. [CRA05] **Crainicu, B.**, "An Overview of the IPsec Extensions Header – AH (Authentication Header and ESP (Encapsulating Security Payload)", *Education/Training and Information/Communication Technologies – RoEduNet'05*: Proceedings of the 4th International Conference RoEduNet Romania: Târgu Mureş – Sovata, 20-22 May 2005, Editura Universităţii "Petru Maior" din Târgu Mureş, 2005, ISBN 973-7794-29-X, pp. 245-254.

2. [CRAI05] **Crainicu, B.**, "Web Security: Secure Socket Layer and Transport Layer Security", *Interdisciplinarity in Engineering: Proceedings of the Scientific Conference Inter-Ing 2005*: Târgu Mureş, "Petru Maior" University, Faculty of Engineering, 10-11 November 2005, Editura Universităţii "Petru Maior" din Târgu Mureş, 2005, ISBN 973-7794-41-9, pp. 787-800.

3. [CM06] **Crainicu, B**., Măruşteri, M., "PGP Cryptographic Keys and Key Rings", *Proceedings of the 5th RoEduNet IEEE International Conference*: Sibiu, 1-3 June 2006, Romania, Editura Universităţii "Lucian Blaga" din Sibiu, 2006, ISBN (10) 973-739-277-9, (13) 978-973-739-277-0, pp. 138-141.

4. [CRA08] **Crainicu, B.**, "Wireless LAN Security Mechanisms at the Enterprise and Home Level", *Novel Algorithms and Techniques In Telecommunications, Automation and Industrial Electronics*, Springer Netherlands, ISBN978-1-4020-8736-3 (Print), 978-1-4020-8737-0 (Online), 2008, pp. 305-310.

5. [CI08] **Crainicu, B.**, Iantovics, B.L., "Cryptanalysis of KSAm-like Algorithms", *First International Conference on Complexity and Intelligence of the Artificial and Natural Complex Systems. Medical Applications of the Complex Systems. Biomedical Computing. CANS 2008*, Published by IEEE Computer Society, 2008, IEEE Computer Society Order Number P3621, ISBN 978-0-7695-3621-7, Library of Congress Number 2009900732, pp. 130-148.

6. [CRI08] **Crainicu, B.**, Iantovics, B., "On A New RC4 Key Scheduling Algorithm", *Proceedings of the International Conference Complexity and Intelligence of the Artificial and Natural Complex Systems. Medical Applications of the Complex Systems. Biomedical Computing, CANS 2008*, 8-9 November, Târgu Mureş, 2008, Editura Universităţii "Petru Maior" Târgu-Mureş, 2008, ISSN 2065-0426, pp. 16-25.

7. [CRA09] **Crainicu, B.**, "A Local Search Approach for Recovering an Internal State of RC4 Stream Cipher", *11th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2009*, Timisoara, Romania, September 26-29, 2009.

8. [CB10] **Crainicu, B.**, Boian, F., "KSAm – An Improved RC4 Key-Scheduling Algorithm for Securing WEP", *Novel Algorithms and Techniques in Telecommunications and Networking,* Springer Netherlands, 2010, ISBN 978-90-481-3661-2 (Print), ISBN 978-90-481-3662-9 (Online), pp. 391-396.

9. [CRB10] **Crainicu, B.**, Boian, F., M., "Some Combinatorial Aspects of the KSAm-like Algorithms Suitable for RC4 Stream Cipher", *Studia Universitatis Babes-Bolyai*, Series Informatica, Volume LV, Number 1, 2010, ISSN 1224-869x (paper version), ISSN 2065-9601 (online version), pp. 105-114.

10. [CE11] **Crainicu,B.**, Enăchescu, C., "A Metaheuristic Tabu Search Approach for Internal State Reconstruction of RC4", Proceedings *of 10th RoEduNet IEEE International Conference*, Iaşi, Romania, 23-25 June 2011, Published by Stef, 2011, ISSN 2247-5443, pp. 164-167.

11. [CI11] **Crainicu,B.**, Iantovics, B., "An Agent-based Security Approach for Intrusion Detection Systems", *7th International Workshop on Grid Computing for Complex Problems*, *GCCP2011*, Bratislava, Slovakia, October 24 - 26, 2011, Institute of Informatics, Slovak Academy of Sciences, ISBN 978-80-970145-5-1, pp. 126-133.

12. [CRA14] **Crainicu, B**., "On Invariance Weakness in the KSAm Algorithm", *8th International Conference Interdisciplinarity in Engineering, INTER-ENG 2014*, 9-10 October 2014, Tirgu-Mures, Romania, Elsevier, ISSN 2212 – 0173, pp. 850-857.

13. [IC08] Iantovics, B., L., **Crainicu, B**., "Complex Mobile Multiagent Systems", *First International Conference on Complexity and Intelligence of the Artificial and Natural Complex Systems. Medical Applications of the Complex Systems. Biomedical Computing. CANS 2008*, Published by IEEE Computer Society, 2008, IEEE Computer Society Order Number P3621, ISBN 978-0-7695-3621-7, Library of Congress Number 2009900732, pp. 21-30.

14. [IAC08] Iantovics, B., L., **Crainicu, B.**, "Security in Mobile Multiagent Systems", *Proceedings of the International Conference Complexity and Intelligence of the Artificial and Natural Complex Systems. Medical Applications of the Complex Systems. Biomedical Computing, CANS 2008*, 8-9 November, Târgu Mureş, 2008, Editura Universităţii "Petru Maior" Târgu-Mureş, 2008, ISSN 2065-0426, pp. 183-191.

15. [IM10] Iantovics, B., L., Marusteri, M., Kountchev, R., Zamfirescu, C., B., **Crainicu, B.**, "Intelligent CMDS Medical Agents with lerning Capacity", *Proceedings of the International Conference on Virtual learning. ICVL 2010*, October 29-October 31, 2010, Targu Mures, Romania, Bucharest University Press, 2010, ISSN 1844-8933, pp. 325-331.

16. [IC13] Iantovics, B., L., **Crainicu, B,** "Sisteme multiagent: o abordare modernă în inteligenţa artificială**",** Editura Universităţii "Petru Maior, ISBN 978-606-581-099-0, 2013.

17. [IC14] Iantovics, B., L., **Crainicu, B.**, "A Distributed Security Approach for Intelligent Mobile Multiagent Systems", *Advanced Intelligent Computational Technologies and Decision Support Systems, Studies in Computational Intelligence*, Volume 486, Springer International Publishing, 2014, ISBN 978-90-481-3661-2 (Print), ISBN 978-90-481-3662-9 (Online), pp 175-189.

18. [MCS061] Măruşteri, M., **Crainicu, B.**, Şchiopu, A., "ROBIOCLUSTER – an open source platform for HPC (high performance computing)/Linux clusters in the biomedical field", *Integrating Biomedical Information: From eCell to ePatient, Proceedings of the European Federation for Medical Informatics, Special Topic Conference*, April 6-8, 2006, Timişoara, Romania, Akademische Verlagsgesellschaft Aka GmbH, Berlin, ISBN 3-89838-072-6 (Aka), ISBN-10 1-58603-614-9 (IOS Press), ISBN-13: 978-1586036140 (IOS Press), ISBN 973-625-303-1 (Editura Politehnica), 2006, pp. 174-177.

19. [MCS062] Măruşteri, M., Ş., **Crainicu, B.**, Şchiopu, A., "New trends in Open Source Educational Platforms – The ROSLIMS Linux Live CD Paradigm", *Proceedings of the 5th RoEduNet IEEE International Conference*: Sibiu, 1-3 June 2006, Romania, Editura Universităţii "Lucian Blaga" din Sibiu, 2006, ISBN (10) 973-739-277-9, (13) 978-973-739-277-0, pp. 82-86. *Acta Universitatis Cibiniensis*, Vol. LV, Technical Series, Editura Universităţii "Lucian Blaga" din Sibiu, 2007, ISSN 1583-7149, pp. 136-140.

20. [MCS063] Măruşteri, M., **Crainicu, B.**, Şchiopu, A., *ROSLIMS Linux Live CD – all-in-one cross platform solution for running biomedical software*, Integrating Biomedical Information: From eCell to ePatient, Proceedings of the European Federation for Medical Informatics, Special Topic Conference, April 6-8, 2006, Timişoara, Romania, Akademische Verlagsgesellschaft Aka GmbH, Berlin, ISBN 3-89838-072-6 (Aka), ISBN-10 1-58603-614-9 (IOS Press), ISBN-13: 978-1586036140 (IOS Press), ISBN 973-625-303-1 (Editura Politehnica), 2006, pp. 178-181.

Contribuții originale / Rezultate ale cercetării:

1. Un model formal unificat de algoritm de criptare secvențial sincron şi auto-sincronizat, *AlgStream*(S_0, *S*, *k*, F_0, *F*, *f*, *crypt*, *Z*, *M*, *C*), bazat pe definirea unui generator *G*(S_0, *S, k,* F_0, *F, f, Z*) de cheie de criptare (*keystream*), și care integrează componentele: componenta care inițializează şi modifică, pe baza unei chei secrete inițiale *k* sau a unei sămânțe secrete k_s (concatenarea dintre cheia secretă inițială şi o valoare suplimentară), un vector de stare internă *S* (*componenta*_1), componenta care produce cuvintele secrete de criptare ale textului în clar, şi anume generatorul de cheie de criptare (*componenta*_2), şi componenta care implementează procesul de criptare propriu-zisă (*componenta*_3). (Secțiunea 4.3.)

2. Definirea dimensiunii stării interne a unui generator de cheie de criptare (*keystream*) implementat de către o maşină autonomă cu stări finite. (Secțiunea 4.3.1.)

3. Un model general de atac al algoritmilor de criptare secvențiali de tip cunoaşterea textului în clar (eng. *known-plaintext attack – KPA*), *ModelAtacB*, care ia în considerare gradul cel mai ridicat de vulnerabilitate al algoritmului de criptare, stabilit de cantitatea maximă de informaţie la care poate avea acces în mod legitim sau nelegitim atacatorul. (Secțiunea 4.3.2.)

4. O metodă de criptanaliză/atac a algoritmului de criptare secvențial *RC*4, *TabuStateTable* (*TST*), care urmărește reconstruirea tabelei de stare inițială *S* pe baza abordării atacului *Knudsen*, a reprezentărilor arborescente şi a condiţiilor generale definite în atacul *Tomašević*, aplicând o strategie de căutare de tip *Tabu*. Rezultatele obținute sunt superioare celor din atacurile *Knudsen* și *Tomašević*. (Secțiunea 5.3.7.2.)

5. Un algoritm de planificare a cheii secrete pentru algoritmul de criptare secvențial *RC*4, *KSAm* (*Key Scheduling Algorithm modified / KSA modified*), care elimină vulnerabilitatea invariantă *Fluhrer-Mantin-Shamir* a *KSA*, distruge condiția rezolvată și, în consecință, anulează vulnerabilitatea vectorului de inițializare *IV* descoperită în criptanaliza *Fluhrer-Mantin-Shamir*, oferă protecție împotriva atacurilor *Ohigashi-Shiraishi-Morii* și *Klein/PTW* în modul de operare *WEP*, determină ineficiența cheilor slabe de tip *Roos* şi reduce semnificativ corelarea statistică *Mantin-Shamir* a celui de-al doilea cuvânt de ieşire *Z* al fluxului/șirului de criptare (*keystream*). (Secțiunile 6.1., 6.2., 6.3.3., 6.3.4.)

6. Un model (general) de abordare algoritmică de determinare a mapărilor biţilor din cheia secretă *K* în şabloane identificabile din permutarea inițială *S*, model care poate fi utilizat pentru testarea oricărui mecanism de tip *Scrambling* (de exemplu, *KSA/KSAm*) aplicat unei permutări căreia i se inter-schimbă la fiecare pas valorile a două intrări pe baza a doi indici. (Secțiunea 6.2.4.)

7. Un model de criptanaliză a stării permutării *S* aferente unui algoritm de criptare secvențial, bazat pe cunoaşterea stării interne iniţiale a permutării *S*, pe probabilitatea păstrării stării interne iniţiale a permutării *S* şi pe probabilitatea (*b*)-*conservativităţii* permutării *S*, în urma succesiunii unor cicluri *Scrambling* de permutare a valorilor elementelor permutării *S*. (Secțiunea 6.2.5.)

8. Definirea conceptelor de *diferențiator*, *diferenţiator*$_{indicator}$, *corelare*$_{generală}$, *corelare*$_{diferențiator}$, *corelare pe șablon*, *şablon asociat secvenţei de ieşire*, precum și a unei clase de diferenţiatori pentru algoritmii de criptare secvențiali, și construirea unor mecanisme de atac asupra stării interne aferente algoritmilor de criptare secvențiali pe baza valorilor diferențiatorilor, cu aplicare și testare asupra algoritmului *RC*4*m* (*RC*4 cu *KSAm* ca și algoritm de planificare a cheii). (Secțiunile 6.3.1, 6.3.2)

9. Definirea noțiunii de factor de corelare r_c cu scopul redefinirii conceptelor de *a-stare* şi *b-predictivitate* din [MS02], astfel încât să fie luate în considerare corelările cu stările iniţiale, şi al modelării procesului de identificare a diferenţiatorilor bazaţi pe stări predictive din punct de vedere statistic. Factorul de corelare devine un element de criptanaliză deosebit de util în contextul în care starea iniţială secretă a unui algoritm *PRGA* este obţinută dintr-o succesiune de cicluri diferite bazate pe constrângeri diferite şi pe o cheie secretă – *RC*4*m* reprezintă un astfel de caz datorită algoritmului *KSAm* care conţine două secvenţe diferite de tip *Scrambling*. (Secțiunea 6.3.3.)

10. Definirea conceptelor de *diferenţiator*$_{puternic}$ şi *diferenţiator*$_{slab}$ ca instrumente de criptanaliză a corelărilor $S_i \rightarrow s_i$ şi/sau $S_i \rightarrow T_i$. (Secțiunea 6.3.3.)

11. Definirea conceptelor de a_m-*stare* și *stare* r_c-*b-predictivă*, și identificarea diferențiatorilor pentru a_m-*stările* care sunt r_c-*b-predictive*. (Secțiunea 6.3.4.)

12. Determinarea coeficientului de avantaj sau a coeficientului de distingere/diferenţiere dintre un flux *RC*4/*RC*4*m* şi unul (pseudo)-aleatoriu. (Secțiunea 6.3.4.)

13. Stabilirea performanței atacurilor de identificare a *diferenţiatorilor*$_{indicator}$ / *diferenţiatorilor*$_{slabi}$ şi a *diferenţiatorilor*$_{puternici}$ din mulţimea *diferenţiatorilor*$_{slabi}$, bazate pe stări *neprevăzute* şi pe relaţia de corelare r_c. (Secțiunea 6.3.4.)

14. Definirea stării X_N-*incertă* și a clasei de chei slabe de criptare X_N-*slab-incertă* care permit identificarea și analiza celei mai puțin probabile permutări *S*. (Secțiunea 6.3.5.)

Doresc să mulțumesc în mod deosebit conducătorului științific, Prof. Dr. Florian Mircea Boian pentru deschiderea, disponibilitatea și suportul acordat pe întreaga perioadă de concepere și elaborare a tezei.

Adresez sincere mutumiri Prof. Dr. Călin Enăchescu pentru încurajările și ajutorul acordate pe parcursul acestor ani.

Mulțumesc familiei mele pentru pentru sprijinul enorm pe care mi l-au oferit.

În același timp, doresc să le mulțumesc colegilor din Departamentul de Informatică al Universității "Petru Maior" din Tg. Mureș, precum și colegilor din Facultatea de Matematică și Informatică a Universității "Babeș-Bolyai" din Cluj-Napoca, pentru opiniile și sugestiile constructive.

Rezumat

Teza de doctorat **“Securitate în reţele şi în tehnologii wireless”** este axată pe analiza principiilor şi mecanismelor care stau la baza protocoalelor şi algoritmilor de criptare secvențiali/șir (eng. *stream*), ca parte componentă a criptografiei simetrice. Ținând cont de faptul că algoritmii de criptare secvențiali sunt intrinsec legați de generatorii de numere/șiruri aleatorii şi pseudo-aleatorii, pe lângă detalierea noțiunilor, modelelor și formalismelor necesare construirii sistemelor criptografice secvențiale, teza cuprinde o sinteză a conceptelor fundamentale care definesc proprietatea de a fi aleatoriu (eng. *randomness*) din perspectiva teoriei clasice a probabilității, a teoriei Shannon a informației și, în special, a probabilității algoritmice și a teoriei informației algoritmice, prezentând principalele abordări algoritmice ale proprietății de a fi aleatoriu. Din punct de vedere criptografic, proprietatea de a fi aleatoriu reprezintă elementul central în proiectarea generatorilor de numere/șiruri (pseudo)-aleatorii care produc fluxul/șirul de criptare (eng. *keystream*) sau cheia finală de criptare în cadrul cifrurilor secvențiali.

În comparația dintre cifrurile secvențial şi bloc, metrica eficienței procesului de criptare trebuie definită extrem de clar. În [PP10] se specifică faptul că, pentru cifruri secvențiale optimizate în software, eficiența înseamnă că sunt necesare mai puține cicluri procesor pentru criptarea unui bit, iar pentru cifruri secvențiale optimizate în hardware, eficiența înseamnă că sunt necesare mai puține porți hardware pentru criptarea la aceeași rată a datelor. Analiza eficienței procesului de criptare nu trebuie să neglijeze faptul că modul de operare al anumitor cifruri bloc îi transformă în cifruri secvențiali sincroni (de exemplu, modurile de operare *CTR* – eng. *Counter* şi *OFB* – eng. *Output feedback*).

Pe lângă avantajele care decurg din simplitatea în implementare şi viteza mare de criptare, cifrurile secvențiale sunt liniare în timp şi constante în spațiu, şi, un aspect deloc de neglijat, propagarea erorilor este foarte scăzută – o eroare apărută la criptarea unui octet sau simbol nu afectează criptarea octeților sau simbolurilor care urmează (cifrul secvențial sincron) sau o eroare apărută în textul criptat afectează cel mult *t* octeți sau simboluri din textul în clar decriptat (cifrul secvențial asincron), astfel că cifrurile secvențiale își găsesc aplicabilitatea în situațiile în care erorile de transmisie apar cu mare probabilitate. Cifrurile secvențiale expun două mari dezavantaje: difuzie scăzută datorată faptului că informația dintr-un octet sau simbol al textului în clar este conținută într-un singur octet sau simbol al textului criptat, şi, comparativ cu cifrurile

bloc, posibilitatea mai facilă ca un atacator care a spart algoritmul să insereze text falsificat ce pare autentic.

Dintre cele mai răspândite și mai utilizate cifruri secvențiale se pot aminti: *A*5/1, *CryptMT*, *ISAAC*/*ISAAC*+, *Rabbit*, *RC*4, *Scream*, *SEAL*, *SNOW*, *Trivium*, *Turing*, *VEST*.

Teza se concentrează pe analiza cifrului secvențial/șir *RC*4 (eng. *Rivest Cipher* 4 sau eng. *Ron's Code* 4), prezentând principalele rezultate criptanalitice apărute în literatura de specialitate, cu un accent deosebit pus pe vulnerabilitățile descoperite în cadrul componentei de planificare a cheii, atât din punctul de vedere al arhitecturii algoritmului, cât și din cel al modurilor sale de operare. Fiind considerat unul dintre cei mai rapizi algoritmi de criptare la nivel software, *RC*4 rămâne în continuare cel mai utilizat cifru secvențial, regăsindu-se în implementări criptografice răspândite: *SSL*/*TLS*, *IPsec-ESP* (*Encapsulating Security Payload*), *Kerberos*, *RDP* (*Remote Desktop Protocol*), *Microsoft Point-to-Point Encryption*, *SSH* (*Secure Shell*), *SASL* (*Simple Authentication and Security Layer*) etc. Studiile și cercetările efectuate de-a lungul timpului asupra algoritmului *RC*4 au dezvăluit faptul că vulnerabilitățile critice ale acestuia nu decurg primordial din arhitectura sa, ci mai ales din felul în care acesta este implementat în diverse moduri de operare. Cu siguranță, exemplul cel mai elocvent este *WEP* (eng. *Wired Equivalent Privacy*), un protocol considerat actualmente extrem de vulnerabil pentru transmisiile fără fir [CRA08]. Pornind de la criptanaliza lui Fluhrer, Mantin și Shamir din [FMS01], Klein în [KLA08] și Tews, Weinmann și Pyshkin în [TW08] implementează practic un atac care permite recuperarea în proporție de 95% a unei chei *WEP* de 104 biți prin capturarea a cel puțin 85000 de pachete cu vectori de inițializare *IV* comuni. Însă, reușita atacului speculează utilizarea defectuoasă a vectorului de inițializare *IV* (eng. *Initialization Vector*) de 24 biți (o dimensiune mult prea mică), nesecretizat, care se concatenează cu cheia secretă de 104 biți, rezultând o sămânță de 128 biți ce devine blocul de date de intrare pentru generatorul de numere pseudo-aleatorii *WEP PRNG* (eng. *WEP Pseudorandom Number Generator*); *WEP PRNG* generează apoi o secvenţă pseudo-aleatorie de octeţi (eng. *keystream*) cu ajutorul căreia se criptează mesajul. Cum era firesc, industria a reacționat rapid, însă înlăturarea vulnerabilităților dovedite ale protocolului *WEP* s-a realizat cu costuri însemnate: în loc să se modifice modul de operare al algoritmului *RC*4, s-a preferat înlocuirea acestuia cu cifrul bloc *AES* (eng. *Advanced Encryption Standard*), împreună cu schimbarea algoritmului de verificare a integrității mesajelor, rezultând mecanismul *WPA*2 (eng. *Wi-Fi Protected Access II*), proces ce a necesitat înlocuirea completă a hardware-ului existent care nu avea suport pentru *AES*. În acest context, în teză se propune cel puțin o soluție care întărește considerabil nivelul de securitate oferit de protocolul *WEP*.

În cazul în care se dorește păstrarea vectorilor de inițializare *IV* în modul de opeare al cifrului *RC*4, criptologii sunt unanimi de acord că impunerea unor condiții de securitate stricte acestor vectori *IV* (irepetabilitate, caracter aleatoriu, dimensiune mărită), combinată cu obligativitatea utilizării unei chei secrete de minim 256 biți și cu eliminarea primilor 256 (cel puțin) octeți ai fluxului/șirului secret de criptare (eng. *keystrem*), determină practic impenetrabilitatea cifrului *RC*4. În aceste condiții, luând în considerare și proprietățile statistice ale generatorului pe care îl incorporează, combinate cu viteza mare de operare/criptare, considerăm că arhitectura cifrului *RC*4 poate rămâne în continuare o opțiune importantă de criptare secvențială, mai ales în transmisiile fără fir în care distribuția și mobilitatea nodurilor este ridicată, și în sistemele multiagent mobile [IC08], [IAC08], [IC14], [CI11], [IM10].

Teza propune un mecanism îmbunătățit de planificare a cheii, *KSAm* (eng. *Key Scheduling Algorithm modified*), pentru cifrul secvențial *RC*4 și o nouă metodă de atac asupra *RC*4 care urmărește reconstruirea tabelei secrete de stare inițială. Se avansează noi modele formale de criptanaliză (metode criptanalitice de atac) a componentei de planificare a cheii care se bazează pe modificarea secvențială a câte două valori dintr-un vector ce reprezintă starea internă a algoritmului de criptare secvențial, modele testate pe *RC*4 și pe *RC*4*m* (cifrul *RC*4 cu *KSAm* ca și algoritm de planificare a cheii). Un set de atacuri reușite asupra *RC*4 sunt testate pe *RC*4*m*, rezultatele acestor teste demonstrând viabilitatea mecanismului *KSAm* propus.

Teza este împărțită în 6 capitole și 2 anexe.

Capitolul 1, **Proprietatea de a fi aleatoriu. Secvenţe/şiruri aleatorii**, include o sinteză a conceptelor *aleatoriu* (eng. *random*) şi *proprietatea de a fi aleatoriu* (eng. *randomness*). Aceste concepte reprezintă componente fundamentale ale criptografiei, în special în studiul și proiectarea algoritmilor criptografici secvențiali. Din acest motiv, pornind de la teoria clasică a probabilităţii şi a teoriei Shannon a informaţiei, capitolul se concentrează pe sinteza detaliată a noțiunii de *aleatoriu* și a *proprietății de a fi aleatoriu* din perspectiva probabilităţii algoritmice şi a teoriei informaţiei algoritmice.

Proprietatea de a fi aleatoriu asociată unui fenomen/proces/eveniment/sistem/comportament este direct legată un set de alte concepte (interdependente) precum *şansă*, *entropie*, (*ne*)*predictibilitate*, (*ne*)*determinism*, *stocastică, haos/dezordine, hazard*, *incertitudine*, (*ne*)*tipicitate*, care permit crearea unui cadru de definire, mai mult sau mai puţin complet, a *proprietăţii de a fi aleatoriu*. În mod logic, *proprietatea de a fi aleatoriu* implică lipsa

predictibilităţii, a *coerenţei*, a *determinismului*, a *ordinii*, a *corelării* elementelor componente din cadrul unei secvenţe de caractere/simboluri/numere/biţi/paşi. Orice proces/fenomen care nu poate fi prevăzut trebuie tratat din *perspectivă aleatorie*. Conform [EAG05], nu se poate pune semnul de egalitate între *proprietatea de a fi aleatoriu* şi *nepredictibilitate* (deşi una din abordările matematice de studiu ale *proprietăţii de a fi aleatoriu* este din perspectiva *nepredictibilităţii*), *proprietatea de a fi aleatoriu* fiind un caz special al conceptului de *nepredictibilitate a unui proces/fenomen*; prin urmare, se poate afirma că *proprietatea de a fi aleatoriu* este indiscutabil *nepredictibilă*. Însă, indiferent de abordările de analiză ale *proprietăţii de a fi aleatoriu* (stocastică/nepredictibilă, haotică/incompresibilă, tipică), provocarea din punct de vedere matematic este extrem de ridicată din perspectiva formalizării/definirii, sau a încercării de formalizare/definire a noţiunii de *aleatoriu*.

Cerinţa ca rezultatele unui *fenomen/proces/eveniment aleatoriu* să nu poată fi integrate în nici o structură deterministă nu înseamnă că o astfel de structură nu există, doar că, cel puţin, ea nu a fost găsită – de aici porneşte îndoiala privind existenţa reală *proprietăţii de a fi aleatoriu*. Acest aspect, combinat cu noţiunea de *aparenţă* prin care o secvenţă de numere/biţi care ‘‘pare” aleatorie pentru un algoritm/adeversar, poate să nu ‘‘pară” aleatorie pentru un alt algoritm/adversar, conduce la necesitatea formulării unor abordări diferite, *algoritmice*, ale *proprietăţii de a fi aleatoriu*, construite deasupra definiţiilor intuitive anterioare sub forma unor superseturi de specificaţii parametrizate care să permită în final crearea unei/unor definiţii formale pentru *proprietatea de a fi aleatoriu*.

Muchnik, Semenov şi Uspensky subliniază în [MSU98] axa abordării intuitive a *proprietăţii de a fi aleatoriu*, precizând că asocierea caracterului aleatoriu unei secvenţe depinde de *modelul probabilistic ales şi acceptat în avans* – o secvenţă, pentru care s-a definit în avans o măsură (metrică) pentru *proprietatea de a fi aleatoriu*, trebuie declarată aleatorie sau nealeatorie pe baza acelei măsuri (metrici). Din acest punct trebuie pornit pentru definirea granulară, algoritmică, ulterioară a *proprietăţii de a fi aleatoriu*.

Li şi Vitanyi subliniază în [LIV08] că teoria clasică a probabilităţii nu poate acoperi noţiunea de *proprietate aleatorie a unei secvenţe individuale*, poate doar exprima probabilităţi ale proprietăţilor asociate rezultatelor unui eveniment/proces aleatoriu, adică probabilităţile proprietăţilor mulţimii complete a secvenţelor pentru o anumită distribuţie, neputând defini ce înseamnă ca o secvenţă individuală să fie aleatorie.

Mai mult, teoria clasică a probabilităţii nu furnizează instrumentele de a pune la îndoială rezultatul unui eveniment după ce acesta a avut loc – singura opţiune disponibilă este de a exclude în avans ‘‘injusteţea” posibilului rezultat prin luarea unor măsuri mai mult sau mai puţin restrictive [VIT01]. Chaitin scrie în [CHAG75] că, deşi noţiunea de *aleatoriu* poate fi definită precis şi, mai mult, poate fi măsurată, teoria clasică a probabilităţii nu este capabilă să stabilească dacă un număr individual dat este aleatoriu sau nealeatoriu. Prin urmare, Chaitin sugerează că este necesară o definiţie mult mai ‘‘sensibilă” pentru *proprietatea de a fi aleatoriu*. Rezolvarea acestei probleme, adică furnizarea componentelor matematice care să permită definirea *proprietăţii de a fi aleatoriu*, constă în utilizarea funcţiei *măsură de probabilitate*, care specifică probabilitea observării oricărui eveniment dintr-un experiment, al cărui rezultat este incert, reprezentând noţiunea fundamentală a teoriei moderne a probabilităţii, ale cărei baze axiomatice au fost stabilite de Kolmogorov în 1933 [KOL56].

Din punct de vedere al teoriei informaţiei, *entropia* unei variabile aleatorii, definită de Shannon în [SHA48] în contextul distribuţiei de probabilitate a variabilei respective, reprezintă o *măsură a relativităţii/nesiguranţei/nedeterminismului/dezordinii* (*measure of uncertainty/disorder*) – Shannon defineşte *entropia* ca fiind *măsura informaţiei dintr-o distribuţie*. Shannon prezintă în [SHA48] legătura dintre *entropia termodinamică* şi *entropia teoriei informaţiei*, sugerând că este posibilă măsurarea *conţinutului informaţiei* unui mesaj şi cum se poate transmite corect un mesaj în prezenţa zgomotului.

Entropia Shannon reprezintă de fapt o mapare funcţională între variabile aleatorii (distribuţii ale variabilelor aleatorii) şi numere reale. Principala interpretare a entropiei Shannon constă în stabilirea numărului de biţi necesari reprezentării/codării valorilor variabilelor aleatorii. Una din observaţiile importante ale lui Shannon este că aspectele semantice ale unui mesaj sunt irelevante în cadrul procesului ingineresc de manipulare a mesajului. Conform teoriei informaţiei formulate de Shannon, unui set/ansamblu de mesaje posibile i se atribuie o *cantitate de informaţie*, care reprezintă numărul de biţi necesari pentru a lua în considerare toate posibilităţile de reprezentare a mesajelor (se presupune că toate mesajele sunt egale). Prin urmare, mesajele pot fi manipulate, în ansamblu, utilizând acest număr de biţi. Însă, Shannon nu face nici o precizare în legătură cu numărul de biţi necesari manipulării/transmiterii unui mesaj individual din acel set/ansamblu. Exemplele prezentate în literatura de specialitate [GV03], [GW04] dezvăluie faptul că anumite şiruri de biţi (reprezentări ale mesajelor) pot fi compresate, însă cu riscul apariţiei unor regularităţi care înseamnă eliminarea *proprietăţii de a fi aleatoriu* pentru şirul (compresat) de biţi. În schimb, dacă orice element de regularitate lipseşte, devine extrem de dificilă

reprezentarea numerelor mari. Concluzia care se desprinde constă în necesitatea de a avea o *măsură a informaţiei* care, spre deosebire de abordarea Shannon, să nu se bazeze pe presupuneri probabilistice şi care să ia în considerare faptul că şiruri care conţin elemente de regularitate sunt comprimabile [GV03], [GW04]. Prin urmare, trebuie găsită/definită această măsură aplicabilă atât conţinutului informaţiei unui *obiect finit* individual (şir binar finit), cât şi *cantităţii de informaţiei* deţinute de un obiect finit cu privire la un alt obiect finit [KOL65].

Aşadar, teoria clasică a probabilităţii şi teoria informaţiei definită de Shannon nu pot oferi o definiţie riguroasă a *proprietăţii de a fi aleatoriu*. Teoria informaţiei lui Shannon măsoară probabilităţi în cadrul unui sistem de evenimente, unde apar o mulţime de rezultate posibile, însă nu poate analiza un singur eveniment extras din sistem şi izolat.

Dacă definiţiile (intuitive) date *proprietăţii de a fi aleatoriu* şi secvenţei aleatorii sunt clar formulate din punct de vedere logic şi conceptual, dificultăţile care apar rezidă în stabilirea exactă a măsurilor (metricilor) asociate *proprietăţii de a fi aleatoriu* (alegerea modelului probabilistic, conform [MSU98]), pe de o parte, şi în validarea şablonului nedeterminist al secvenţei aleatorii, pe de altă parte; de fapt, trebuie oferit răspuns, în mod real, la întrebarea: care sunt măsurile/metricile matematice pe baza cărora se stabileşte că o secvenţă este *aleatorie* ? Astfel, formalizarea definiţiei date *proprietăţii de a fi aleatoriu* şi *secvenţei aleatorii*, din perspectivă algoritmică, constă în stabilirea/definirea măsurilor/metricilor pentru *proprietatea de a fi aleatoriu*, crearea testelor aferente pe baza acestor măsuri/metrici, şi impunerea condiţiei de satisfacere/trecere a testelor respective.

Solomonoff propune şi dezvoltă în [SOL60], [SOL164], [SOL264], [SOL78], [SOL97] noţiunea de *probabilitate algoritmică* (eng. *Algorithmic* "*Solomonoff*" *Probability* – AP), prin care se alocă unui obiect o *probabilitate apriori* cu valoare universală. Având aplicabilitate teoretică în domenii importante (inteligenţă artificial, analiza complexităţii de timp a algoritmilor, teoria inferenţei inductive) şi facilitând o înţelegere superioară a *proprietăţii de a fi aleatoriu*, marele neajuns al AP constă în faptul că nu este calculabilă în practică, putând fi doar aproximată.

Pornind de la ideea de „*probabilitate personală*", Solomonoff consideră teorema lui Bayes ca punct de pornire pentru construcţie probabilităţii algoritmice. Deducţiile lui Solomonoff se bazează pe fenomene din viaţa reală [SOL97]: o persoană se naşte cu anumite cunoştinţe calitative despre o distribuţie apriori de probabilitate (obţinută printr-o evoluţie organică), face predicţii şi ia decizii pe baza acestei distribuţii, ulterior distribuţia se modifică în conformitate cu experienţa de viaţă a persoanei, iar în final se selectează acele organisme care au luat decizii pe

baza unei „bune” distribuţii apriori de probabilitate. Luând în considerare originea biologică a unei distribuţii apriori, Solomonoff identifică şi enumeră în [SOL97] corespondenţele dintre procesul de evaluare a probabilităţii şi învăţarea umană: predicţia viitorului pe baza informaţiilor/datelor obţinute în trecut; predicţia ca noţiune individuală are o valoare scăzută – predicţia trebuie să aibă asociat un element de precizie cantitativă înainte de a fi utilizată pentru luarea deciziilor; precizia predicţiei este dependentă de calitatea şi cantitatea informaţiilor/datelor obţinute în trecut; precizia predicţiei este dependentă de calitatea şi cantitatea resurselor de calcul (computaţionale) disponibile. Prin urmare, Solomonoff concluzionează că probabilitatea trebuie definită în termeni de resurse de calcul (computaţionale) necesare obţinerii/calculării sale.

Teoria informaţiei algoritmice (eng. *Algorithmic Information Theory – AIT*) este bazată pe conceptul lui Solomonoff de *probabilitate algoritmică*, rezultate fundamentale în acest domeniu fiind obţinute de Kolmogorov [KOL65] şi Chaitin [CHA66], [CHA69], [CHA74], [CHAG75], [CHA75], [CHA76], [CHAG76], [CHAG77].

În lucrarea [KOL65], Kolmogorov defineşte conceptul de informaţie din perspectivă combinatorială, probabilistică şi algoritmică. În cadrul abordării combinatoriale, Kolmogorov caracterizează entropia folosind alfabete de limbaj şi mulţimi de elemente. Pentru abordarea probabilistică, Kolmogorov analizează variabilele aleatorii cu o anumită distribuţie de probabilitate. În abordarea algoritmică, bazându-se pe funcţii recursive, Kolmogorov sugerează o metodă de descriere eficientă a lungimii $l(s) = n$ a unui şir s, pe $log_2\ n + log_2\ log_2\ n + log_2\ log_2\ log_2\ n$ + ... biţi, continuând recursiv până la ultimul termen pozitiv, metodă care permite descrierea cantităţii de informaţie [ZAW].

AIT poate fi considerată teoria informaţiei (teoria conţinutului informaţiei) unui obiect individual şi tratează, pe baza teoriei computaţionale, legăturile dintre informaţie, calcul computaţional şi *proprietatea de a fi aleatoriu*. Prin combinarea teoriei informaţiei cu teoria computaţională, *AIT* creează conceptul de *informaţie dintr-un obiect individual* sau *complexitate* (*algoritmică*) *a unui obiect individual*, şi, mai departe, a *proprietăţii de a fi aleatoriu* asociate obiectelor individuale, elemente care nu pot fi identificate în teoria informaţiei a lui Shannon. Dacă teoria informaţiei lui Shannon măsoară doar *cantitatea de informaţie*, partea algoritmică a *AIT* măsoară *conţinutul informaţiei* cu ajutorul algoritmilor (programelor).

Atât teoria informaţiei a lui Shannon, cât şi *AIT* consideră că volumul/cantitatea de informaţie despre un fenomen, obţinut în urma observării sale, poate fi măsurat prin *numărul minim de biţi*

necesari descrierii observării fenomenului [GW08]. Dacă Shannon tratează metodele de descriere a observării din punct de vedere al unei distribuţii de probabilitate date, *AIT* are o abordare diferită, neprobabilistică: orice program de calculator care, în prima fază, calculează şirul ce reprezintă procesul observării, iar apoi se termină, este considerat o descriere validă a observării – *cantitatea de informaţie din şir* (*măsura informaţiei asociate unui şir*) sau *complexitatea algoritmică a şirului* este definită ca fiind *dimensiunea în biţi a celui mai scurt program de calculator care produce acel şir, iar apoi se opreşte* [GW08]. Astfel, o primă definiţie a complexităţii algoritmice a unui şir (obiect) este dată de *dimensiunea celui mai scurt program sau set de algoritmi care generează sau descrie acel şir* (*obiect*) *pe o maşină universală Turing*. Această definiţie, bazată pe teoria algoritmilor, pare să depindă de tipul abordărilor algoritmice utilizate, abordări pentru care complexităţile obiectelor descrise sunt asimptotic optimale şi nu diferă între ele mai mult de valoarea unei constante aditive – dimensiunea minimă a unui program pe o maşină Turing va adăuga doar o constantă la dimensiunea minimă a programului pe fiecare altă maşină Turing (conform lui Chaitin, o maşină universală Turing poate simula funcţionarea oricărei maşini Turing) [CHA66].

Pornind de la conceptul de *probabilitate algoritmică* inventat de Solomonoff, au fost propuse mai multe abordări, echivalente până la un anumit nivel, de formalizare a *proprietăţii de a fi aleatoriu* asociată unui/unei obiect/secvenţe individual/individuale, cele mai importante abordări fiind următoarele:

- abordare din perspectivă stocastică (stabilitatea frecvenţelor) – *stocasticitate* (*nepredictibilitate*)
- abordare din perspectivă haotică (incompresibilitate) – *haoticitate*
- abordare din perspectivă tipică (mulţimea cu măsura efectivă 1) – *tipicitate*

Studiul *proprietăţii de a fi aleatoriu* din perspectivă *algoritmică* încearcă să cuprindă, în primul rând, secvenţele infinite (în cadrul abordării intuitive a *proprietăţii de a fi aleatoriu* s-a precizat faptul că, în mod normal, o secvenţă finită nu poate fi aleatorie tocmai datorită caracterului său finit). Un alt aspect important îl constituie *gradul* sau *nivelul* caracterului aleatoriu (*degree of randomness*) asociat secvenţelor în contextul în care două sau mai multe secvenţe sunt comparate din punct de vedere al *proprietăţii de a fi aleatoriu*. Astfel, dacă este posibilă şi fundamentată afirmaţia "*obiectul* (*secvenţa*) *A este "mai mult" aleatorie sau este "mai puţin" aleatorie decât obiectul* (*secvenţa*) *B*", atunci secvenţele pot fi incluse în clase care au grade/nivele diferite ale *proprietăţii de a fi aleatoriu*. Din nou, abordarea intuitivă este extrem de

simplă – o secvenţă este sau nu este aleatorie, în funcţie de nedetectarea sau detectarea unui şablon determinist care să caracterizeze valorile/însemnătatea elementelor componente.

Analiza stocastică a *proprietăţii de a fi aleatoriu* [Mises, Wald, Church, Kolmogorov, Loveland] presupune că, pentru a fi aleatorii, secvenţa în sine, împreună cu subsecvenţele sale trebuie să aibă *proprietatea stabilităţii frecvenţei*, adică să respecte toate testele statistice "rezonabile" – orice secvenţă arbitrară de lungime *k* trebuie să aibă aceeaşi limită de frecvenţă 2^{-k} sau acelaşi grad de *nepredictibilitate* (de exemplu, numerele zero dintr-o secvenţă să fie asimptotic egale cu numerele de unu). Apare, aşadar, şi noţiunea de *nepredictibilitate* în definirea *proprietăţii de a fi aleatoriu*, şi care poate fi tratată din punct de vedere a imposibilităţii construirii unei strategii de joc de succes. Prima încercare de a defini *proprietatea de a fi aleatoriu* pentru un obiect individual din perspectivă stocastică a fost făcută de von Mises în [MIS19], [MIS57], încercând să formalizeze matematic noţiunea intuitivă a unui şir care pare "mai mult" aleatoriu decât altul în urma analizei statistice a proprietăţilor unor evenimente repetitive. von Mises a fost interesat în aplicarea teoriei probabilităţii la studierea fenomenelor reale din natură şi a avansat ideea că studierea teoriei probabilităţii este intrinsec legată de studierea secvenţelor aleatorii. În încercarea de a depăşi critica lui Ville adusă abordării stocastice a *proprietăţii de a fi aleatoriu*, Kolmogorov [KOL63] şi Loveland [LOV66], în mod independent, admit regulile de selecţie calculabile, însă propun un model de selecţie nemonotonă. Aceste reguli de selecţie se numesc *reguli de selecţie admisibilă Kolmogorov-Loveland*, iar o secvenţă este *Kolmogorov-Loveland* (*KL*) *stocastică* dacă este stocastică în raport cu regulile de selecţie admisibilă Kolmogorov-Loveland.

Ideea analizei *proprietăţii de a fi aleatoriu* din perspectiva *incompresibilităţii* a fost propusă în mod independent de Solomonoff [SOL62], [SOL164], [SOL264], Kolmogorov [KOL63] şi Chaitin [CHA66], fiind punctul de pornire al dezvoltării conceptului de *complexitate algoritmică* sau *descriptivă* care stă la baza *AIT*. Încearcând să ofere definiţii algoritmice corecte pentru *proprietatea de a fi aleatoriu* , Kolmogorov a accentuat irelevanţa secvenţelor infinite pentru fundamentarea teoriei probabilităţilor, considerând că dezvoltarea unei măsuri a complexităţii doar pentru secvenţe finite are sens din perspectiva interpretării frecvenţelor, adică a secvenţelor finite. Din acest punct de vedere, Kolmogorov introduce conceptul de *complexitate a unui obiect finit* şi furnizează în [KOL63] o definiţie universală a acestei complexităţi ca fiind lungimea celui mai scurt program binar care permite reconstrucţia/decodarea obiectului, sau numărul minim de biţi care conţin toate informaţiile despre un obiect dat şi care este suficient pentru reconstrucţia/decodarea obiectului. Complexitatea Kolmogorov prezintă două mari avantaje:

clasifică secvenţele în secvenţe aleatorii şi nealeatorii, şi, extrem de important, permite atribuirea de grade/nivele caracterului aleatoriu asociat secvenţelor (eng. *degree of randomness*). Acest ultim avantaj devine util în procesul de analiza a secvenţelor infinite.

Fiind o abordare cantitativă a proprietăţii de a fi aleatoriu, *tipicitatea* este bazată pe noţiunea de măsură (eng. *measure-theoretic*): o secvenţă este aleatorie dacă nu există nici o metodă efectiv calculabilă de a găsi o mulţime de măsură zero care să conţină acea secvenţă [VS10]. O proprietate sau un atribut al unei secvenţe binare infinite este *special* dacă probabilitatea ca acea proprietate să aibă loc este zero, iar proprietatea este numită *tipică* dacă probabilitatea ca acea proprietate să aibă loc este unu. Un atribut este *special* dacă complementul său este *tipic*, şi invers [DAS11]. Conform lui Martin-Löf, datele sunt aleatorii în măsura în care pot fi analizate prin metode algoritmice. Formalizarea (algoritmică) propusă de Martin-Löf *proprietăţii de a fi aleatoriu* este considerată cea mai riguroasă şi satisfăcătoare. Dacă o secvenţă este aleatorie, atunci este *tipică*, *tipicitatea* fiind o condiţie necesară pentru ca o secvenţă să fie aleatorie. Martin-Löf încearcă să demonstreze că tipicitatea este şi condiţie suficientă pentru *proprietatea de a fi aleatoriu*. Astfel, conform lui Martin-Löf, o secvenţă binară este *aleatorie* dacă şi numai dacă este *tipică*. *Proprietatea de a fi Schnorr aleatoriu* se aplică în primul rând numerelor reale, considerate ca şiruri infinite, şi este strâns legată de *proprietatea de a fi Martin-Löf aleatoriu*. Schnorr a apreciat că proprietatea de a fi Martin-Löf aleatoriu este prea rigidă/puternică, nefiind suficientă definirea măsurii zero efective. Schnorr a demonstrat în **[**SC171**]** că proprietatea de a fi Martin-Löf aleatoriu a unei secvenţe poate fi definită prin intermediul complexităţii segmentelor iniţiale ale secvenţei, şi a introdus noţiunea de *proprietate de a fi aleatoriu calculabilă.*

Capitolul 2, Generatori de numere pseudo-aleatorii – perspectivă criptografică, este o continuare firească a capitolului anterior, prezentând aparatul teoretic care fundamentează proiectarea generatorilor de numere pseudo-aleatorii din punct de vedere criptografic. Se prezintă, cu scop comparitiv, conceptele de generator de numere aleatorii, generator de numere aleatorii în mod real, generator de numere pseudo-aleatorii şi generator de numere pseudo-aleatorii criptografic sigure. În practică, sistemele criptografice implementează generatori de numere aleatorii la nivel software*;* însă, generatorii software nu pot produce numere “perfect” aleatorii din cauza proprietăţilor de tip determinist ale algoritmilor software. Prin urmare, un generator software produce numere pseudo-aleatorii, adică secvențe de numere care “par” statistic aleatorii.

Din punct de vedere criptografic, definiţia noţiunii de *aleatoriu* are o interpretare (extrem de) practică: *valorile produse de o sursă sunt aleatorii dacă un adversar, chiar în cazul în care cunoaşte platforma hardware şi software pe care rulează acea sursă, inclusiv valorile generate anterior de sursă, nu poate prezice, pe baza informaţiilor cunoscute, valorile următoare produse de sursa respectivă*. Singura metodă disponibilă atacatorului rămâne astfel încercarea tuturor valorilor posibile – atac de tip *forţă brută*. Criptografia are nevoie de astfel de surse aleatorii pentru generarea cheilor/parolelor criptografice şi pentru ascunderea anumitor valori în protocoalele de comunicaţie.

Pe lângă problema factorizării numerelor întregi şi a tuturor aspectelor legate de complexitatea computaţională a mecanismelor criptografice, criptografia modernă se bazează fundamental pe *generatorii de numere pseudo-aleatorii*, la care se adaugă conceptul de *nedistingere*. În teoria complexităţii computaţionale, o distribuţie de probabilitate este pseudo-aleatorie peste o mulţime de adversari dacă nici un adversar din mulţime nu poate distinge/deosebi în mod semnificativ acea distribuţie de o distribuţie uniformă. . Conceptul de *nedistingere*, introdus de Goldwasse şi Micali [GOM82] şi dezvoltat de Blum, Micali şi Yao [BM84], [YAO82], care fundamentează matematic, împreună cu *proprietatea de a fi aleatoriu*, criptografia, permite crearea de generatori de numere pseudo-aleatorii de calitate din punct de vedere criptografic, folosind un nivel relativ scăzut de nepredictibilitate iniţială (procese şi fenomene fizice nepredictibile).

Un generator de numere aleatorii (eng. *random number generator – RNG*) reprezintă o entitate computaţională hardware sau software care generează o secvenţă de numere/biţi aleatorii. Generatorii *RNG* se raportează la cel puţin două numere generate, deoarece pentru un singur număr este imposibil de verificat dacă acesta a fost produs de un generator *RNG*. Analiza algoritmilor *RNG* se bazează pe teste statistice al căror rol este de a măsura real *calitatea* secvenţei generate, cu alte cuvinte, cât de aleatorii sau cât de (ne)predictibile sunt numerele/biţii secvenţei.

Fiind vorba de implementări reale ale algoritmilor criptografici în cadrul produselor software/hardware de securitate, aceştia sunt testaţi criptanalitic mult mai intens decât algoritmii care rămân la stadiul teoretic. Calificativul *perfect* sau *pur* dat caracterului aleatoriu al secvenţei de numere/biţi înseamnă calitatea supremă (maximă). Conceptul *perfect aleatoriu* sau *pur aleatoriu* constituie un subiect extrem de analizat, în context statistic, în literatura de specialitate. Unei secvenţe (şir) *perfect/pur* aleatorii de numere/biţi i se impun următoarele cerinţe: să nu fie predictibilă, să aibă o perioadă cât mai mare (ideal, perioadă infinită pentru obiecte/şiruri infinite), repartiţia elementelor componente (biţi sau numere) să fie uniformă (frecvenţa de

apariţie a elementelor componente să fie egală), să nu existe nici o corelare între elementele componente şi singura metodă de a fi ghicită, în cazul în care devine secretă, să fie doar forţa brută, adică încercarea tuturor combinaţiilor posibile. Pentru şirurile de biţi, condiţia de bază pentru repartiţia uniformă a biţilor este ca frecvenţa de apariţie a bitului 0 să fie egală cu frecvenţa de apariţie a bitului 1; apoi construcţia decurge în genul următor: frecvenţa de apariţie a secvenţei 00 trebuie să fie egală cu frecvenţa de apariţie a secvenţei 01, 10 şi 11, şi aşa mai departe. Testele statistice existente (Kolmogorov–Smirnov, Kuiper, Jarque-Bera, Crame-von Mises, Dvoretzky–Kiefer–Wolfowitz), care arată cât de aleatorii sunt numerele/biţii dintr-o secvenţă prin măsurarea distribuţiilor de probabilitate, includ metrici complexe de evaluare pornind de la astfel de construcţii de bază pentru fiecare cerinţă.

Dificultatea definirii conceptului de aleatoriu şi a declarării unei surse ca fiind aleatorii a determinat împărţirea numerelor aleatorii şi, implicit, a generatorilor de numere aleatorii, în două clase: numere *în mod real, cu adevărat aleatorii* (eng. *truly random numbers*) şi numere *pseudo-aleatorii* (eng. *pseudorandom numbers*).

Specialiştii în criptografie sunt unanimi de acord că numerele (biţii) *cu adevărat aleatorii* sunt produse în mod nedeterminist de generatori hardware (eng. *hardware random number generator – HRNG*) a căror funcţionare se bazează pe procese şi fenomene fizice cu sau fără proprietăţi cuantice aleatorii şi care, cel puţin teoretic, sunt total nepredictibile.

Un *generator de numere pseudo-aleatorii* (eng. *pseudorandom number generator – PRNG*) sau un *generator determinist de numere aleatorii* produce rezultate care "par" statistic aleatorii, având la bază un proces cauzal determinist – algoritm care generează numere ce aproximează caracteristicile numerelor cu adevărat aleatorii. Procesul determinist utilizează un algoritm care produce o secvenţă de biţi pornind de la o valoare iniţială produsă dintr-o sămânţă ce trebuie să conţină suficientă entropie pentru a garanta caracterul aleatoriu al secvenţei de biţi; însă generatorii software de secvenţe de numere/biţi aleatorii nu pot atinge criteriul perfect/pur din cauza proprietăţilor de tip determinist ale algoritmilor software, adică secvenţa de numere/biţi depinde în totalitate de un set relativ redus de valori iniţiale (de exemplu, perioada secvenţei este limitată la intervalul de numere care pot fi reprezentate în sistemul de calcul). Din cauza naturii deterministe a procesului cauzal, generatorul produce biţi *pseudo-aleatorii*. Conform [BK07], *PRNG* produce rezultate nepredictibile dacă sămânţa este secretă şi algoritmul este bine construit, iar securitatea întregului mecanism bazat pe *PRNG* constă în sursa de entropie a canalului de intrare.

Capitolul 3, **Elemente de bază pentru construirea sistemelor criptografice secvențiale**, descrie principiile, mecanismele şi elementele hardware şi software componente ale arhitecturilor sistemelor criptografice secvențiale.

Regiştrii de deplasare liniari cu reacţie (eng. *Linear Feedback Shift Register – LFSR*) sunt utilizaţi în mod extensiv în medii de testare, în designul sistemelor digitale şi în construirea metodelor de compresie. În ştiinţa calculatoarelor şi a comunicaţiilor, regiştrii *LFSR* prezintă un interes deosebit în următoarele tipuri de aplicaţii: generarea de biţi/numere/vectori (pseudo) aleatorii, criptare/decriptare, transmisii fără fir (*wireless*), calcularea sumelor de control al integrităţii datelor, compresia datelor. Regiştrii *LFSR*, prin posibilitatea de a genera numere pseudo-aleatorii, stau la baza construirii unei întregi clase de cifruri secvențiale. Datorită uşurinţei de construire a circuitelor electronice şi electromecanice, a perioadelor relativ lungi şi a distribuţie uniforme a fluxului de ieşire, regiştrii *LFSR* şi-au găsit un loc meritat în criptografie. Însă caracterul lor liniar implică anumite vulnerabilităţi demonstrate în cadrul proceselor de cripanaliză. Algoritmii secvențiali bazaţi de *LFSR* sunt vulnerabili în faţa atacurilor de tip *KPA* (*known-plaintext attack*) care exploatează vulnerabilităţile statistice rezultate în urma alegerii unor anumite funcţii boolene *f* integrate în cadrul *LFSR*. Atacul porneşte de la observarea corespondenţelor între starea ieşirii unui *LFSR* individual şi ieşirea funcţiei boolene *f* care combină starea ieşirilor tuturor generatorilor *LFSR*. Atacul porneşte sub forma unui atac în *forţă brută*, şi, treptat, o dată ce poate fi stabilită o corelare, se transformă într-un atac de tip *divide şi cucereşte*.

Pornind de la conceptul *cheie/umplutură de unică folosință* (eng. *One-Time Pad – OTP*), derivat din cifrul Vernam [VRM19], cifrurile secvenţiale sunt construite cu ajutorul regiştrilor de deplasare liniari cu reacţie *LFSR*, deoarece aceştia produc cuvinte (secvenţe de biţi) cu proprietăţi statistice foarte bune şi pot fi relativ uşor implementaţi la nivel hardware. Pe de altă parte, în contextul în care criptografia şi, implicit, cifrurile secvenţiale, se bazează în mod esenţial pe teoria complexităţii, un element extrem de important îl constituie *funcţiile într-un singur sens* (eng. *One-Way Function – OWF*). Mecanismul de criptare *OTP* preia fiecare caracter sau bit m_i al textului în clar pe care îl criptează printr-o operaţie de adunare *XOR* (modulo 2) cu o cheie secretă k_i generată aleatoriu (*keystream* sau *pad*), rezultând textul criptat c_i. Siguranţa algoritmului *OTP* constă într-un set de condiţionări impuse cheii de criptare (*keystream*): trebuie să rămână secretă, să fie generată perfect/pur aleator, mărimea cheii să fie cel puţin egală cu mărimea textului în clar, şi să nu fie reutilizată – cel puţin teoretic (matematic), îndeplinirea acestor condiţii garantează imposibilitatea decriptării textului criptat fără cunoşterea cheii de

criptare; din acest motiv, algoritmul *OTP* mai este denumit *cifrul perfect* şi este considerat *sigur în mod necondiţionat* în faţa unui atac de tip *COA* (*Ciphertext-Only Attack*); siguranţă necondiţionată înseamnă discreţie/ascundere perfectă (*perfect secrecy*) şi presupune că observarea/analizarea textului criptat nu oferă niciun fel de informaţie utilă.

OTP nu descrie modalitatea de generare a cheilor de criptare/decriptare şi nici felul în care trebuie schimbate aceste chei între partenerii de comunicaţie. Se presupune că aceste chei generate şi distribuite securizat doar sursei (criptorul) şi destinaţiei (decriptorul) comunicaţiei, şi păstrate în mod secret cât timp informaţia supusă procesului de criptare este confidenţială. Protecţia împotriva accesului neautorizat la chei, detectarea accesului neautorizat la chei şi asigurarea disponibilităţii cheilor sunt elemente obligatorii pentru un criptosistem *OTP* solid, însă ele sunt asigurate prin mecanisme externe.

O funcţie *f* este o *funcţie într-un singur sens* (eng. *One-Way Function – OWF*) dacă este *uşor* de calculat, însă *dificil* de inversat – *uşor*: $f(x)$ se calculează în timp polinomial (probabilistic); *dificil*: nu există nici un algoritm care să calculeze în timp polinomial (probabilistic) $f^{-1}(y)$. Funcţiile *OWF* sunt instrumente fundamentale în criptografie, în primul rând pentru construirea generatorilor de numere/secvenţe pesudo-aleatorii. Ţinând cont de faptul că este destul de uşor de creat o funcţie *OWF* dintr-un generator de numere/secvenţe pesudo-aleatorii, studiile de specialitate au demonstrat că un generator de numere/secvenţe pesudo-aleatorii există dacă şi numai dacă există o funcţie *OWF* ([BM84], [HIL99]). În [BM84] se tratează pentru prima oară creearea unui generator pseudo-aleator pe baza unei funcţii *OWF*, pornind de la presupunerea dificultăţii problemei logaritmice discrete. Ulterior, în [YAO82], [LEV87] şi [HIL99] se generalizează problema, abordându-se construirea unui generator pseudo-aleator din orice permutare *one-way.* Pe lângă problema logaritmilor discreţi, un alt principiu important luat în considerare pentru construirea generatorilor pseudo-aleatori este acela al dificultăţii factorizării. În cazul în care *f* este o permutare *one-way*, inversarea funcţie $f(x)$ constă în găsirea lui x. Dacă *f* nu este o permutare, atunci inversarea presupune găsirea oricărui x' astfel încât $f(x') = f(x)$.

Capitolul 4, **Arhitectura algoritmilor de criptare secvențiali**, prezintă componentele arhitecturale și modalitățile de operare ale algoritmilor de criptare secvențiali. Algoritmii de criptare de tip secvenţial (eng. *stream*) combină/adună la nivel de bit (eng. *bitwise*) textul în clar cu cheia de criptare (eng. *keystream* sau eng. *running-key*), care este o secvenţă de biţi pseudo-aleatorii. În cazul în care algoritmul generază pentru cheia de criptare o secvenţă ‘‘perfect” sau

‘‘pur” aleatorie de biţi, atunci cifrul obţinut este, cel puţin teoretic, invulnerabil. Comparativ cu algoritmii de criptare de tip bloc, algoritmii de criptare secvenţiali rulează la viteze superioare în cadrul implementărilor la nivel hardware şi necesită circuite hardware mai puţin complexe. Un alt mare avantaj al algoritmilor de criptare secvenţiali îl constituie propagarea extrem de redusă a erorilor, aspect deosebit de important atunci când mediul de comunicaţie generează erori de transmisie cu o probabilitate ridicată, precum şi cerinţe scăzute de stocare temporară la nivel de *buffere*.

Există un număr impresionant de studii, analize de specialitate şi propuneri ale algoritmilor de criptare secvenţiali, însă majoritatea celor folosiţi în practică (*RC*4, *SEAL*, *Scream*, *A*5/1, *ISAAC* etc.) sunt fie confidenţiali (implementări proprietare, secrete, realizate de companii comerciale), fie publici dar protejaţi prin patente (unii algoritmii de criptare secvenţiali au devenit publici prin metode de tip *reverse engineering* sau prin publicarea lor anonimă pe Internet (*RC*4, *A*5/1)).

Algoritmii de criptare secvenţiali pot fi cu cheie simetrică sau cu cheie publică (mecanismul de criptare probabilistic Blum-Goldwasser reprezintă un algoritm de criptare bazat pe cheie publică). Totuşi, marea majoritate a algoritmilor de criptare secvenţiali fac parte din clasa schemelor de criptare bazate pe chei secrete simetrice. Avantajul acestor algoritmi constă în faptul că elementele de criptare se modifică pentru fiecare cuvânt sau simbol care se criptează, şi că, în cazul erorilor de transmisie, propagarea erorilor este înlăturată. De asemenea, algoritmii secvenţiali devin utili în situaţia în care memoria disponibilă procesului de criptare este limitată, astfel încât se impune procesarea doar a unui singur simbol o dată.

Algoritmii de criptare secvenţiali se împart în două categorii:

1. Algoritm de criptare secvenţial sincron (cifru secvenţial sincron): secvenţa de biţi pseudo-aleatorii de criptare (*keystream*) este generată într-un mod independent de textul în clar şi textul criptat; pentru criptare, secvenţa de criptare generată se combină (în cazul algoritmilor aditivi binari, operaţii *XOR*) cu textul în clar, rezultând textul criptat, iar pentru decriptare, secvenţa de criptare generată se combină (*XOR*) cu textul criptat, rezultând textul în clar.
2. Algoritm de criptare secvenţial auto-sincronizat (cifru secvenţial asincron): mecanismul de obţinere a cheii/secvenţei de criptare (*keystream*) utilizează un număr de *n* biţi din textul criptat obţinut anterior; pentru criptare, secvenţa de criptare generată se combină (în cazul algoritmilor aditivi binari, operaţii *XOR*) cu textul în clar, rezultând textul criptat, iar pentru decriptare, secvenţa de criptare generată se combină (*XOR*) cu textul criptat, rezultând textul în clar.

Modelele de atac utilizate în mod real pentru criptanaliza cifrurilor/algoritmilor secvenţiali sunt următoarele:

- atac bazat doar pe textul criptat (*ciphertext-only attack*, *known ciphertext attack - COA*): model de atac care presupune accesul doar la o parte din textul criptat, urmărindu-se obţinerea textului în clar (necriptat) corespunzător textului criptat şi/sau a cheii secrete de criptare.
- atac bazat pe cunoaşterea textului în clar (necriptat) (*known-plaintext attack – KPA*): model de atac în care se cunoaşte o parte a textului în clar (necriptat) împreună cu versiunea criptată corespunzătoare textului în clar, urmărindu-se găsirea cheii secrete de criptare şi/sau a cărţii de cod (*codebook*).
- atac bazat pe selectarea textului în clar (necriptat) (*chosen-plaintext attack – CPA*): model de atac care presupune alegerea arbitrară a unor părţi din textul în clar (necriptat) din care, în urma criptării, se obţine versiunea criptată corespunzătoare textului în clar, urmărindu-se vulnerabilităţile mecanismului de criptare şi, implicit, găsirea cheii secrete de criptare; în cazul cifrurilor secvenţiale (aditive), atacul *CPA* este echivalent cu atacul *KPA* (nu acelaşi lucru se poate afirma în cazul cifrurilor bloc). Atacul este întâlnit frecvent în criptografia bazată pe chei publice.
- atac bazat pe selectarea textului criptat (*chosen-ciphertext attack – CCA*): model de atac care presupune decriptarea unei anumite părţi a textului criptat cu ajutorul unei chei necunoscute – se introduce în sistem text criptat, se obţine textul echivalent în clar (necriptat) şi se încearcă obţinerea cheii secrete folosite în procesul decriptării; modelul de atac nu este util în cazul cifrurilor secvenţiale aditivi, însă reprezintă o serioasă ameninţare pentru cifrurile secvenţiale auto-sincronizate şi pentru cifrurile secvenţiale care combină mecanismul de criptare şi de sutentificare în cadrul aceleiaşi funcţii.
- atac bazat pe cunoaşterea/alegerea vectorilor de iniţializare *IV* (*known/chosen IV attack – KIVA*): model de atac în care atacatorul poate manipula vectorii de iniţializare care se ataşează cheii secrete de criptare; atacatorul cunoaşte şi alege şirul vectorilor de inţializare IV_1 IV_2, ..., IV_n (vectorii *IV* sunt întotdeauna publici/cunoscuţi), reuşind să obţină fluxul de criptare corespunzător textului în clar.
- atac bazat pe detectarea diferenţelor dintre o secvenţă de biţi pseudo-aleatorii şi o secvenţă de biţi ‘‘perfect” aleatorii (*distingushing attack – DA*): ideal, cheia de criptare este o

secevenţă de biţi ‘‘perfect” aleatorii; în practică, însă, acest lucru este (aproape) imposibil de realizat [BD89]; criteriul principal de evaluare a unei secvenţe de biţi aleatorii îl reprezintă compararea cu o secvenţă de biţi ‘‘perfect” aleatorii – cu cât apropierea e mai mare, cu atât generatorul de biţi aleatori este mai bun.

Rueppel în [RUE86] şi Zenner în [ZEN04] definesc componentele unui model de bază al generatorului de cheie de criptare (eng. *keystream generator*), $G_{bază}(S, F, f)$, pentru cifrurile secvenţiale sincrone (cheia de criptare este generată independent de mesajul în clar şi de mesajul criptat), iar Zenner descrie în [ZEN04] mecanismul de integrare a generatorului $G_{bază}(S, F, f)$ în structura algoritmilor de criptare secvenţiali. De asemenea, Zenner extinde în [ZEN04] modelul $G_{bază}(S, F, f)$, introducând generatorul extins $G_{extins}(S, F, f, C)$ de cheie de criptare.

Zenner face în [ZEN04] o separare strictă a algoritmului de generarea a cheii de generatorul de cheie de criptare, însă într-un mod „neintegrat” – aspectele privind integrarea componentelor sunt neglijate. Însă, în timp ce separarea celor doua componente constituie o abordare utilă pentru designul şi studierea acestora, neincluderea modalitățile de integrare și a fluxurilor vehiculate între componente poate duce la omiterea anumitor aspecte de securitate privind interconecatera şi sincronizarea componentelor cifrului în ansamblu (pericolul propagării sau chiar al multiplicării unor vulnerabilități transmise între componentele cifrului). Din acest motiv, în cadrul tezei se propune un model formal unificat pentru algoritmii de criptare secvențiali sincron și auto-sincronizat, denumit $AlgStream(S_0, S, k, F_0, F, f, crypt, Z, M, C)$, format din trei componente: componenta care iniţializează şi modifică, pe baza unei chei secrete iniţiale k sau a unei sămânţe secrete k_s, un vector de stare internă S, componenta care reprezintă generatorul de cheie de criptare şi componenta care implementează procesul de criptare propriu-zisă. Noutatea adusă de *AlgStream* constă în faptul că face o delimitare mai clară între partea de planificare a cheii și partea de generare a fluxului de criptare (*keystream*), oferind astfel un grad mai ridicat de modularitate, și integrează și unifică toate fluxurile și modurile de operare posibile pentru un algoritm de criptare secvențial, inclusiv cele opționale: de exemplu, alegerea folosirii vectorilor de inițializare sau introducerea unui nou nivel de inițializare primară a vectorului de stare al algoritmului sau modificarea regulată a vectorului de stare finală care determină inițializarea generatorului fluxului de criptare.

Model formal AlgStream(S_0, S, k, F_0, F, f, crypt, Z, M, C) al algoritmului de criptare secvenţial sincron bazat pe un generator G(S_0, S, k, F_0, F, f, Z):

(i) (opţional) concatenarea cheii secrete iniţiale $k \in \{0, 1\}^l$ cu o valoare publică, suplimentară $sup \in \{0, 1\}^{l_{sup}}$ (vector de iniţializare *IV*, număr aleatoriu/pseudo-aleatoriu *nonce*) şi obţinerea unei sămânţe secrete $k_s \in \{0, 1\}^L$, $L = l + l_{sup}$ (*componenta*_1);

(ii) definirea vectorului de stare internă $S \subseteq \{0, 1\}^{N_t}$ şi setarea stării iniţiale $S_0 \in S$ a vectorului de stare internă (*componenta*_1):

- pe baza valorilor *k* sau k_s: $F_0 : \{0, 1\}^l \rightarrow S$, $s_i = F_0(k)$ sau $F_0 : \{0, 1\}^L \rightarrow S$, $s_i = F_0(k_s)$, $s_i \in S_0$;
- fără folosirea valorilor *k* sau k_s: $F_0 : S \rightarrow S$, $s_i = F_0(i)$ (de exemplu, alegerea permutării identice pentru intrările vectorului de stare *S*);

(iii) modificarea stării interne *S* pe baza cheii secrete iniţiale *k* sau a sămânţei secrete k_s, proces care determină obţinerea unei stări interne secrete: $F : \{0, 1\}^l \times S \rightarrow S$, $s_{i+1} = F(k, s_i)$ sau $F : \{0, 1\}^L \times S \rightarrow S$, $s_{i+1} = F(k_s, s_i)$, $s_i \in S_i$, $s_{i+1} \in S_{i+1}$, $S_i \in S$, $S_{i+1} \in S$ (*componenta*_1);

(iv) obţinerea spaţiului $Z \subseteq \{0, 1\}^{a_2}$ al cheilor finale de criptare (*keystream*), $Z = (z_0, z_1, ..., z_i, ...)$ (*componenta*_2):

- pe baza valorilor *k* sau k_s: $f : \{0, 1\}^l \times S \rightarrow \{0, 1\}^{a_2}$, $z_i = f(k, s_i)$ sau $f : \{0, 1\}^L \times S \rightarrow \{0, 1\}^{a_2}$, $z_i = f(k_s, s_i)$, $z_i \in \{0, 1\}^{a_2}$, $a_2 \leq N_t$;
- fără folosirea valorilor *k* sau k_s: $f : S \rightarrow \{0, 1\}^{a_2}$, $a_2 \leq N_t$, $z_i = f(s_i)$, $z_i \in \{0, 1\}^{a_2}$.

(v) procesul de criptare efectivă a textului în clar $M \subseteq \{0, 1\}^{a_2}$, $M = (m_0, m_1, ..., m_i, ...)$: *crypt* : $M \times Z \rightarrow C$, unde $C \subseteq \{0, 1\}^{a_2}$ este spaţiul cuvintelor criptate, $C = (z_0, z_1, ..., z_i, ...)$, *crypt* $(m_i, z_i) = m_i \oplus z_i = c_i$, $m_i \in \{0, 1\}^{a_2}$, $z_i \in \{0, 1\}^{a_2}$, $c_i \in \{0, 1\}^{a_2}$, unde m_i reprezintă cuvintele mesajului în text clar *M*, criptate succesiv cu cheile de criptare z_i (*keystream*) prin operaţii *XOR* (sunt posibile şi alte tipuri de operaţii) în urma cărora rezultă cuvintele criptate c_i (*componenta*_3).

Pentru modelul *AlgStream*(S_0, *S*, *k*, F_0, *F*, *f*, *crypt*, *Z*, *M*, *C*), se prezintă metoda de calcul pentru dimensiunea stării interne a algoritmului de criptare secvenţial.

Fie G un generator de tipul $G_{bază}(S, F, f)$ şi fie A o maşină AFSM care implementează generatorul G. Dimensiunea stării interne a generatorului G este definită ca fiind $\tilde{n} := \lceil \log_2(|A|) \rceil$, *unde* $|A|$ *este numărul stărilor interne ale lui A şi are valoarea:*

$$|A| = (2^n)! \times (2^{n_1})^{r_1} \times (2^{n_2})^{r_2} \times ... \times (2^{n_i})^{r_i} \times ...$$

Mărimea N_t de reprezentare a stării interne a vectorului S are valoarea $N_t = n \times 2^n$, unde n este dimensiunea (numărul de biţi) unei intrări s a vectorului de stare S. Mărimea totală N_f de reprezentare a stării interne a generatorului de cheie de criptare are valoarea $N_f = n \times 2^n + n_1 \times r_1 + n_2 \times r_2 + ... + n_i \times r_i + ...$ Se observă că $N_f \geq \tilde{n}$.

Pe baza acestei dimensiuni, se criptanalizează starea internă a algoritmilor de criptare secvenţiali şi se propune un model de atac al algoritmilor de criptare secvenţiali de tip cunoaşterea textului în clar (eng. *known-plaintext attack – KPA*), denumit *ModelAtacB*, care ia în considerare gradul cel mai ridicat de vulnerabilitate al unui algoritm de criptare secvenţial, stabilit de cantitatea maximă de informaţie la care poate avea acces un atacator.

ModelAtacB (***MAB***):

Date cunoscute (*date legitime*): se presupune că atacatorul cunoaşte toate detaliile algoritmului *stream* de criptare, însă nu are nici o informaţie despre cheia iniţială de criptare k şi nici despre stările interne S_i. Prin urmare, atacatorul cunoaşte funcţiile F, f şi F_0, valoarea *sup* (care este o valoare publică), numărul de paşi care determină starea internă finală, starea iniţială S_0 ($Iniţializare_{primară}$ – în cazul obţinerii stării iniţiale S_0 fără folosirea valorilor k sau k_s), şi, de asemenea, are acces la întregul flux criptat de date c_i.

Date cunoscute (*date nelegitime*): se presupune că atacatorul cunoaşte, din surse externe algoritmului de criptare, un număr de L biţi din cheia finală de criptare (*keystream*), unde $L << 2^l$, precum şi informaţii parţiale despre un set limitat de stări interne S_i. Fluxul criptat de date c_i poate fi inclus şi în categoria datelor nelegitime.

Tip de atac: se consideră toate tipurile de atac (*COA*, *KPA*, *CPA*, *CCA*, *DA*).

Resurse de calcul: atacatorul este capabil să realizeze orice calcul computaţional care necesită mai puţini paşi decât căutarea exhaustivă în spaţiul cheii iniţiale de criptare. Se fixează o limită superioară de memorie de 2^b biţi care poate fi utilizată.

Noţiunea de succes: atacul este considerat un succes dacă poate prezice/găsi, în mod corect, biţi necunoscuţi din cheia finală de criptare (*keystream*), dacă poate face deosebire dintre biţii produşi de generatorul de cheie de criptare (*keystream generator*) şi un şir "perfect" aleatoriu de biţi, sau dacă poate reduce spaţiul stărilor interne *S* (detectează stări interne care nu loc), ţinând cont de faptul că se cunosc toate detaliile de funcţionare ale algoritmului *stream* de criptare, cu efect direct atât în găsirea unor biţi necunoscuţi din cheia finală de criptare (*keystream*), cât şi în detectarea diferenţelor dintre biţii produşi de generatorul de cheie de criptare (*keystream generator*) şi un şir "perfect" aleatoriu de biţi. Se spune că generatorul de cheie de criptare este "spart" dacă probabilitatea de succes a atacului diferă în mod semnificativ de găsirea/detectarea pur întâmplătoare a informaţiilor secrete (informaţii total secrete sunt cheia iniţială de criptare *k* şi stările interne S_i, în afară de starea iniţială S_0).

Capitolul 5, **Algoritmul *RC*4**, tratează criptanalitic cifrul secvențial *RC*4: stări secrete în care algoritmul nu poate intra, corelări dintre componentele secrete și publice ale vectorului de stare, mulțimi de chei și vectori de inițializare *IV* slabi, invarianța algoritmului, corelări ale primelor intrări ale vectorului de stare, corelări ale distribuției valorilor permutării inițiale ale vectorului de stare, diferențiatori ai șirului/fluxului produs de generatorul de numere pseudo-aleatorii al algoritmului, atacuri de recuperare a cheii, atacuri de reconstrucție a stării interne, atacuri bazate pe textul criptat și pe textul în clar. Capitolul se concentrează pe rezultatele semnificative din literatura de specialitate obținute în urma criptanalizei componentei *RC*4 de planificare a cheii, prin care se încearcă reconstrucția stării interne secrete pe baza caracteristicilor sale combinatoriale.

Fluhrer, Mantin şi Shamir descriu în [FMS01] o metodă de atac de tip *COA* (*Ciphertext-Only Attack*) împotriva algoritmului *RC*4 în forma de implementare a acestuia în cadrul protocolului *WEP*. Atacul *FMS* (*Fluhrer-Mantin-Shamir*) a dezvăluit primele slăbiciuni majore ale modului în care protocolul *WEP* (*Wired Equivalent Privacy*) utilizează *RC*4 ca mecanism de criptare a comunicaţiei.

Criptanaliza *FMS* se axazează pe componenta *KSA* (*Key Scheduling Algorithm*) de planificarea cheii de criptare. Autorii descoperă o clasă mare de chei slabe: o cantitate relativ redusă dintr-o cheie secretă determină un număr mare de biţi din permutarea iniţială secretă produsă de *KSA*, care, mai departe, se reflectă în prefixul fluxului de ieşire produs de componenta *PRGA* (*Pseudo*

Random Generation Algorithm). Astfel, fluxul iniţial produs de *PRGA* este puternic afectat de un număr redus de biţi ai unei chei slabe. O altă vulnerabilitate descoperită de Fluhrer, Mantin şi Shamir constă în observaţia că atunci când aceeaşi parte secretă a cheii este folosită în combinaţie cu diferite valori cunoscute de atacator, atunci este posibilă, relativ uşor, reconstrucţia părţii secrete a cheii prin analiza cuvântului iniţial al fluxului de criptare.

Fluhrer, Mantin şi Shamir definesc şi demonstrează existenţa *vulnerabilităţii invariante* pentru o versiune simplificată a *KSA*, numită KSA^*, iar apoi descriu modificările necesare pentru aplicarea şi analizarea vulnerabilităţii invariante în contextul *KSA*. Mantin şi Shamir descriu în [MS02] o polarizare/corelare statistică importantă în al doilea cuvânt de ieşire al algoritmului *RC*4, pe baza căreia construiesc un algoritm ce permite o diferenţiere clară între cuvintele de ieşire *RC*4 şi secvenţele cu adevărat aleatoare, analizând doar un singur cuvânt dintr-un număr de $O(N)$ fluxuri diferite de ieşire. Însă, Fluhrer, Mantin şi Shamir obţin în [FMS01] rezultate superioare în procesul de identificare a elementelor de distingere/diferenţiere între cuvintele de ieşire *RC*4 şi secvenţele cu adevărat aleatoare, chiar şi în situaţia de eliminare a primelor două cuvinte din fiecare flux de ieşire. Analizând măsura de securitate *Sampling Resistance* (rezistenţă de prelevare de mostre) a cifrurilor secvenţiale definită şi descrisă de Biryukov, Shamir şi Wagner în [BSW01], Fluhrer, Mantin şi Shamir consideră că legătura puternică dintre anumite clase de chei *RC*4 şi cuvintele de ieşire corespunzătoare acestor chei dovedeşte faptul că *RC*4 are o rezistenţă scăzută de prelevare de mostre, ceea ce conduce la uşurarea atacurilor de tip TMDT (eng. *Time-Memory-Data Trade-Off*).

Fluhrer, Mantin şi Shamir analizează modalităţi de atac asupra algoritmului *RC*4 în care se cunosc câteva cuvinte din componenţa cheii secrete de intrare, în special în cazul în care cheia secretă este concatenată cu o valoare vizibilă/publică de tipul vectorilor de iniţializare (*IV* – *Initialization Vector*). Autorii demonstrează că, în cazul în care aceeaşi cheie secretă este folosită în mod repetat cu vectoriii *IV*, iar atacatorul obţine primul cuvânt al fluxului de ieşire *RC*4 corespunzător fiecărui vector *IV*, atunci poate reconstrui cheia secretă cu o probabilitate foarte ridicată – atacul depinde de numărul vectorilor *IV* necesari, de dimensiunea vectorilor *IV* şi, uneori, de valoarea cheii secrete, fiind axat doar pe analiza primului cuvânt de ieşire.

Knudsen, Meier, Preneel, Rijmen şi Verdoolaege analizează în [KNU98] securitatea algoritmului *RC*4 şi a unor variante ale sale, dezvoltând algoritmi criptanalitici pentru atacuri de tip text în clar (*plaintext atack*) cu scopul găsirii stării iniţiale a algoritmului *RC*4, în condiţiile în care se presupune cunoaşterea doar a unui mic segment din textul în clar. Analiza propusă exploatează natura combinatorială a algoritmului *RC*4 şi demonstrează proprietăţi intrinseci ale sale care sunt

independente de metoda de planificare a cheii şi de dimensiunea cheii. Astfel, construirea corectă a stării iniţiale permite calcularea secvenţei de ieşire fără necesitatea cunoaşterii cheii secrete. De asemenea, autorii estimează complexitatea fiecărui tip de atac propus. Knudsen reuşeşte să îmbunătăţească metoda lui Golić [Golić] de detectare a deviaţiilor statistice şi pune în practică pentru prima dată un atac real de tip *known plaintext* de aflare a stării tabelei iniţiale a algoritmului *RC*4. Atacul reuşeşte să construiască tabela iniţială pornind de la cunoaşterea unei părţi reduse a textului în clar. Complexitatea atacului este calculată teoretic şi verificată riguros prin intermediul testelor experimentale. Conform rezultatelor obţinute de Knudsen, complexitatea atacului propus este mai mică decât timpul necesar căutării într-un interval egal cu rădăcina pătrată a tuturor posibilelor stări iniţiale. În plus, Knudsen identifică fluxuri de date particulare pentru care atacul devine mult mai performant, obţinând rezultate importante în condiţiile în care este cunoscut un anumit număr de valori ale intrărilor tabelei de stare. Cu toate că metodele de atac propuse de Knudsen nu sunt fezabile pentru cuvinte de opt biţi (dimensiunea normală pentru implementările reale ale *RC*4), acestea au constituit de fapt baza criptanalizelor *RC*4 ulterioare care sunt independente de mecanismul de planificare a cheii şi de dimensiunea cheii.

Ohigashi, Shiraishi şi Morii prezintă în [OS05] un atac *WEP* (atac *OSM*: *Ohigashi-Shiraishi-Morii*) care reuşeşte să evite vectorii *IV* slabi folosiţi în atacul *FMS* (*Fluhrer-Mantin-Shamir*). Atacul este de tip *vector IV cunoscut* şi transformă majoritatea vectorilor *IV* ai *WEP* în vectori *IV* slabi. Numărul acestor vectori *IV* slabi devine atât de mare, încât evitarea lor este impracticabilă. Pentru o cheie de sesiune de 128 biţi, eficienţa atacului este $2^{72.1}$, ceea ce înseamnă că atacul poate recupera o cheie de 128 biţi într-un timp realistic.

Shiraishi, Ohigashi şi Morii evaluează în [SOM03] eficienţa metodei Knudsen [KNU98], încercând să reconstruiască starea internă pentru cuvinte de $n = 8$ biţi şi o limită de timp computaţională de 2^{20}. Pe baza rezultatelor experimentale, autorii precizează că este nevoie de cel puţin $k = 112$ intrări cunoscute pentru refacerea cu succes a stării interne iniţiale ($k = 207$ pentru reconstrucţia cu succes a unui număr de 100000 de stări iniţiale alese aleator, $k = 167$ pentru reconstrucţia cu succes a unui număr de 50000 de stări iniţiale alese aleator şi $k = 112$ pentru reconstrucţia cu succes a unei singure stări iniţiale). În cazul în care $k < 112$ limita de timp computaţională este depăşită înainte de sfârşitul procedurii de refacere a stării.

Criptanaliza algoritmului *KSA* a demonstrat adevărate breşe ale primelor poziţii ale permutării *S*; observaţia care a stat la baza acestei abordări a fost aceea că, în cadrul primilor paşi ai *KSA*,

există o mare probabilitate ca poziţiile/intrările atinse de *j* să nu fi fost anterior implicate în nicio operaţie de tip *swap*. Roos observă în [ROO95] că:

1. *Fiind dată o cheie de lungime K octeţi, şi fie E < K. Probabilitatea ca elementul/intrarea E al tabelei de stare S să depindă numai de elementele/intrările* 0, 1,.., *E* (*inclusiv*) *ale cheii este de* 37 %.
2. *Cea mai probabilă valoare pentru elementul/intrarea E din tabela de stare S este S*[*E*] = *X*(*E*) + *E*(*E*+1)/2, *unde X*(*E*) *este suma intrărilor* 0, 1, ..., *E* (*inclusiv*) *ale cheii*.
3. *Pentru o cheie RC*4 *de forma K*[1]*K*[2]... *K*[*N*], *dacă K*[0] + *K*[1] = 0, *atunci cu o probabilitate semnificativă primul cuvânt* (*octet*) *generat de RC*4 *va avea valoarea K*[2] + 3.

Tomašević, Bojanić şi Nieto-Taladriz prezintă în [TBN07] un atac asupra algortimului *RC*4 de refacere a stării iniţiale. Atacul se bazează pe găsirea cantităţii maxime de informaţie disponibilă la un moment dat despre starea curentă, pornind de la o dimensiune redusă (cunoscută) a secvenţei de ieşire. Autorii propun reprezentarea arborescentă [YAG06] a algoritmului *RC*4, reprezentare care conţine un set de arbori, fiecare arbore corespunzând unui cuvânt de ieşire, iar apoi definesc o abstractizare analitică sub forma unor *condiţii generale* asociate informaţiei din arbori (fiecare condiţie generală suma tutror condiţiilor dintr-un subarbore), cu scopul de a extrage şi a folosi cantitatea de informaţie care să permită atacului să fie totuşi fezabil (cantitatea de informaţie din arbori este extrem de mare, neputând fi exploatată în întregime din punct de vedere practic). Condiţiile generale sunt, de asemenea, organizate sub forma unei structuri arborescente, iar algoritmul de atac propus caută în această structură arborescentă, pe baza une strategii de tip *hill-climbing*, valorile necesare reconstrucţie stării iniţiale – atacul Tomašević propune o modificare a algoritmului de tip *backtraking* utilizat de Knudsen în [KNU98]. Rezultatele obţinute de Tomašević sunt superioare celor obţinute de Knudsen în [KNU98]; însă complexitatea atacului Tomašević este prea mare pentru un atac fezabil din punct de vedere practic.

În continuare, se propune un nou atac criptanalitic, denumit *TabuStateTable* (*TST*), bazat pe o abordare metaeuristică de căutare de tip Tabu pentru reconstrucţia stării interne secrete a algoritmului *RC*4 (Figura – Algoritmul *TST*). *TST* utilizează o variantă modificată a metodei de căutare Tabu şi se bazează pe atacul criptanalitic *Knudsen* [KNU98] şi pe reprezentările arborescente a cuvântului de ieşire *Z* şi a condiţiilor generale din atacul *Tomašević* [TBN07]. La

momentul t, se consideră a_t ca fiind numărul de intrări în tabela de stare S care au alocate o valoare, şi P ca fiind mulţimea valorilor posibile pentru condiţiile generale.

Algoritm TST

Pasul 1: Se verifică dacă $S_{t-1}[i_t]$ are alocată o valoare (lista Tabu conţine toate intrările a alocate anterior):

a. dacă da, se trece la Pasul 2.
b. dacă nu, se alocă intrării $S_{t-1}[i_t]$ una din cele $2^n - a_t$ valori rămase, se mută această valoare în lista Tabu, se verifică condiţia generală cu cea mai mare probabilitate la nivelul t, mergând în jos; dacă această condiţie nu poate fi satisfăcută, se eliberează valoare din lista Tabu, se alocă o nouă valoare intrării $S_{t-1}[i_t]$, se include această nouă valoare în lista Tabu, şi se verifică din nou condiţia generală cu cea mai mare probabilitate la nivelul t, mergând în jos. Dacă nici una din ele nu este satisfăcută, apare o contradicţie. Dacă verificarea unei anumite condiţii are loc cu succes, se incrementează a_t, şi, în funcţie de condiţia generală care este verificată, se incrementează t şi se trece la Pasul 1 sau la Pasul 2. Lista Tabu conţine valorile care rămân, într-adevăr, alocate.

Pasul 2: Se verifică dacă Z_t are alocată o valoare:

a. dacă da, se calculează valoarea corespunzătoare intrării $S_{t-1}[j_t]$ din ecuaţia cuvântului de ieşire Z_t. Dacă nu apare o contradicţie, se incrementează t şi se trece la Pasul 1.
b. dacă nu, se trece la Pasul 3.

Pasul 3: Se verifică dacă $S_{t-1}[j_t]$ are alocată o valoare:

a. dacă nu, se verifică dacă condiţiile generale au fost examinate la Pasul 1 – dacă da, se actualizează informaţiile obţinute; în caz contrar, se alocă intrării $S_{t-1}[j_t]$ una din cele $2^n - a_t$ valori rămase, se include această valoare în lista Tabu, se verifică condiţia generală cu cea mai mare probabilitate la nivelul t, mergând în jos. Dacă această condiţie nu poate fi satisfăcută, se eliberează valoarea din lista Tabu, se alocă o nouă valoare intrării $S_{t-1}[j_t]$, se trimite această nouă valoare în lista Tabu şi se verifică din nou condiţia generală cu cea mai mare probabilitate la nivelul t, mergând în jos. Dacă nici una din condiţii mu e satisfăcută, apare o contradicţie. Dacă verificarea unei anumite condiţii are loc cu succes, se incrementează a_t şi se verifică dacă valorile date pentru i_t, j_t şi Z_t conduc la o contradicţie – dacă nu, se incrementează t şi se trece la Pasul 1.

Figura – Algoritmul *TST*

Pentru analiza complexităţii atacului *TST*, se utilizează acelaşi formalism prezentat de Knudsen în [KNU98], furnizându-se metoda specifică analizei complexităţii *TST*. Complexitatea atacului *TST* este aceeaşi cu complexitatea atacului *Tomašević*; pentru $n = 6$, $n = 7$ şi $n = 8$, se obţin pentru *TST* rezultate mai bune faţă de atacul *Tomašević*. Dacă se cunoaşte un număr semnificati/suficient de cuvinte de ieşire Z_t, atunci spaţiul de căutare poate fi redus şi atacul devine fezabil.

Complexitatea strategiei *TST* este mai mică decât căutarea exhaustivă şi rădăcina pătrată a tuturor stărilor iniţiale posibile (rădăcina pătrată a tuturor stărilor iniţiale posibile reprezenta pragul complexităţii celui mai bun atac până la criptanaliza realizată de Knudsen). Deşi rezultatele analitice obţinute prin *TST* sunt superioare celor furnizate de Knudsen în [KNU98] şi Tomašević în [TBN07], atacul *TST* rămâne totuşi impracticabil datorită complexităţii ridicate; ca cercetare viitoare, devine promiţătoare abordarea *TST* aplicată atacurilor de tip distingere (eng. *distinguisher attacks*) – de exemplu, exploatarea *TST* a distribuţiei celui de-al doilea cuvânt Z_t produs de *RC*4.

***n* (dimensiunea cuvântului)**	**3**	4	**5**	**6**	**7**	**8**
Atac *Knudsen*	2^8	2^{21}	2^{53}	2^{132}	2^{324}	2^{779}
Atac *Tomašević*	2^5	2^{17}	2^{46}	2^{120}	2^{300}	2^{731}
Atac *TST*	2^5	2^{17}	2^{46}	2^{119}	2^{298}	2^{727}

Figura Aproximarea complexităţilor atacurilor criptanalitice *Knudsen*, *Tomašević* şi *TST* asupra algoritmului *RC*4

Capitolul 6, ***KSAm* (*Key Scheduling Algorithm modified*) – Algoritm de planificare a cheii pentru *RC*4**, reprezintă partea cea mai importantă a tezei, în care se propune un nou algoritm de planificare a cheii pentru *RC*4, denumit *KSA modified* (*KSAm*) (Figura – *KSA* vs *KSAm*), ale cărui scopuri principale constau în eliminarea vulnerabilității invariante a componentei *KSA* originale de planificare a cheii şi diminuarea, până la un nivel criptografic sigur, a vulnerabilităților bazate pe condiția rezolvată în condițiile utilizării vectorilor de inițializare *IV*, mai ales în modul de operare al protocolului *WEP*.

Faţă de versiunea *KSA* originală, *KSAm* include o buclă suplimentară de amestecare a elementelor vectorului S, denumită *Scrambling* _1 (Figura *KSA* vs *KSAm* – liniile (a), (b), (c) şi (d)): cheia secretă iniţializează un vector de indici $u_0, u_1, \ldots, u_{N-1}$; valorile secrete ale indicilor u_i nu sunt unici în cadrul vectorului de indici (repetiţiile sunt posibile). Apoi, urmează o operaţie în $N = 2^n$ paşi de interschimbare a valorilor elementelor $S[i]$ şi $S[u_i]$ ale vectorului S. Astfel,

secvenţa *Scrambling_*1 lasă vectorul *S* într-o stare care, cu o mare probabilitate, va fi diferită de permutarea identică. Secvenţa *Scrambling_*2 este de fapt secvenţa originală *KSA* în $N = 2^n$ paşi de permutare a vectorului S, însă starea vectorului S, la începutul operaţiilor *Scrambling_*2, nu mai este aceaşi ca şi în cazul *KSA* (stare diferită de permutarea identică cu o mare probabilitate).

***KSA* (*K*, *S*)**	***KSA*m (*K*, *S*)**	
Iniţializare:	***Iniţializare***:	
pentru $i = 0$ to $N - 1$	pentru $i = 0$ to $N - 1$	
$S[i] = i$;	$S[i] = i$;	
$j = 0$;	***Scrambling_*1**:	
Scrambling:	pentru $i = 0$ to $N - 1$	(a)
pentru $i = 0$ to $N - 1$	$u_i = (S[i] + K[i \bmod \ell]) \bmod N$;	(b)
$j = (j + S[i] + K[i \bmod \ell]) \bmod N$;	pentru $i = 0$ to $N - 1$	(c)
$swap(S[i],S[j])$;	$swap(S[i], S[u_i])$;	(d)
	$j = 0$;	
	***Scrambling_*2**:	
	pentru $i = 0$ to $N - 1$	
	$j = (j + S[i] + K[i \bmod \ell]) \bmod N$;	
	$swap(S[i],S[j])$;	(e)

Figura – *KSA* vs *KSAm*

Criptanaliza *KSAm*, fundamentată pe metodele de criptanaliză ale *KSA*, include pentru început analiza complexităţii de spaţiu şi de timp şi probabilitatea apariţiei permutării identice după terminarea rulării algoritmului. Dimensiunea stării interne a *KSAm* (complexitatea de spațiu) este de 3748 biţi, iar complexitatea de timp este $O(N)$. Evaluarea probabilităţii evenimentului ca tabela de stare *S*, după terminarea *KSAm*, să fie egală cu permutarea identică se integrează în cadrul analizei generale a probabilităţii ca o anumită valoare *b* să se găsească în poziţia *a* după rularea *KSAm* ($S_{2N}[a] = b$).

Presupunând că atât u_i în cadrul secvenţei *Scrambling_*1, cât şi *j* în cadrul secvenţei *Scrambling_*2 obţin valori aleatoare într-un mod uniform, permutarea identică se analizează pentru început din perspectiva unei secvenţe *Scrambling* generale ($Scrambling_{gen}$) care modelează comportamentul celor două secvenţe *Scrambling* (*Scrambling_*1 şi *Scrambling_*2) ale *KSAm*. Apoi, se studiază efectul combinat a două secvenţe $Scrambling_{gen}$ şi corelarea acestuia cu efectul combinat al secvenţelor *Scrambling_*1 şi *Scrambling_*2 din perspectiva permutării identice.

Se definesc noţiunile de *probabilitate minimă* şi *probabilitate maximă* asociate intrărilor vectorului de stare *S* ca valori de prag minim, respectiv prag maxim, cu scopul de a stabili exact intervalul de valori pe care le poate lua probabibilitatea evenimentului ca valoarea unei intrări să fie regăsită, după una sau după două secvenţe consecutive $Scrambling_{gen}$ (sau după secvenţele *Scrambling 1* şi *Scrambling_2*), în poziţia din faza de iniţializare, adică $S_N[i] = i$ sau $S_{2N}[i] = i$ (interesează distribuţia de probabilitate doar din perspectiva valorilor minimă şi maximă ale intervalului de valori). O dată stabilit acest interval, devin extrem de importante valorile probabilităţilor maxime raportate la valoarea $1/N$, valoare ce reprezintă limita maximă, ideală din punct de vedere teoretic. Un alt motiv pentru definirea celor două tipuri de probabilități, minimă şi maximă, este determinat de modelul de abordare a atacurilor criptografice: în general, un atac asupra unui cifru ține cont și speculează acele vulnerabilități care au loc cu probabilitatea cea mai mare. Raționamentul anterior este corect, numai că, uneori, un eveniment sau o vulnerabilitate cu probabilitate mică are un efect important fie în creșterea unei vulnerabilități deja cunoscută, fie, mai important, în propagarea (secvențială sau în evantai) a unor evenimente sau vulnerabilități care, cumulate, speculează natura combinatorială a acțiunilor care manipulează vectorul de stare internă al algoritmului de criptare.

Calcularea *probabilităţilor minime* presupune urmărirea valorilor intrărilor vectorului *S* la fiecare pas al permutării, şi evaluarea probabilităţii evenimentului ca valorile intrărilor vectorului *S* să rămână fixe permanent pe toată perioada derulării secvenţelor de tip *Scrambling*. Se constată că pentru *N* suficient de mare, probabilitatea minimă asociată unei intrări iniţiale sau permutării identice este extrem de redusă (pentru $N = 256$, $P_{\min\_S}(S_N[i] = i) = 0.0014034^{256}$).

Noţiunea de *probabilitate maximă* reprezintă probabilitatea evenimentului $S[a] = a$ în cazul cel mai defavorabil, adică valoarea de prag maxim a probabilităţii evenimentului $S[a] = a$, valoare care nu este depăşită pentru nici o intrare *a*, unde $a \in [0, N-1]$, în condiţiile în care se iau în considerare toate valorile posibile pentru cheia secretă *K.* Aşa cum remarcă şi Mironov în [MIR02], o dată ce valoarea unei intrări a vectorului *S* este indexată de *i*, atunci acea valoare poate fi găsită, după *N* paşi, pe orice poziţie în cadrul vectorului *S*. Cel puţin teoretic, această observaţie conduce la premiza că valorile obţinute pentru indexul *j* sunt (pseudo)aleatoare în intervalul $[0, N-1]$.

Testele au demonstrat că nici una din cheile *K* posibile de lungime 8, 16, 24 şi 32 de biţi nu lasă vectorul *S* într-o stare egală cu permutarea identică în urma rulării *KSAm* (timpii aproximativi de rulare a testelor au fost: $T_{K=8} \approx 0.003$ sec, $T_{K=16} \approx 0.63$ sec, $T_{K=24} \approx 157.84$ sec, $T_{K=32} \approx 39337$ sec). Se exclude cheia slabă $K = 0$.

Se analizează în continuare efectul *vulnerabilităţii invariante* definite de Fluhrer, Mantin şi Shamir în [FMS01], cu modificările corespunzătoare, asupra *KSAm* [CRA14]. În cadrul acestei analize, se demonstrează că permutarea $S = KSAm_{Scrambling\_1}(K)$ nu este *b-conservativă*. Singurul efect criptanalitic important, adică permutarea $S = KSAm_{Scrambling\_1}(K)$ să fie *b-conservativă*, este obţinut dacă $K = 0$ (singura clasă de chei slabe pentru *vulnerabilitatea invariantă* de tip [FMS01] aplicată mecanismului $KSAm_{Scrambling\_1}$). Chiar şi în condiţiile folosirii cheii slabe $K = 0$, trebuie ţinut cont de faptul că algoritmul KSA_m conţine, pe lângă *Scrambling*_1, secvenţa *Scrambling*_2 (*KSA* original), ceea ce reduce semnificativ "*b-conservativitatea*" permutării S ($S = KSA_m(K)$) – chiar şi în condiţiile $K = 0$, începând cu $i = 2$, *Scrambling*_2 modifică intrările permutării identice.

Se defineşte permutarea *aproape* b_m*-conservativă* şi se demonstrează relaţia:

$$\left(\frac{b-2}{N}\right)^b \times 2/5 \le P[KSAm(K)] \text{ este aproape } b_m\text{-conservativă}] \le \left(\frac{b-2}{N}\right)^b \times 2/5 + \left(\frac{1}{N}\right)^N$$

unde $q \le n$ şi l sunt numere întregi şi $b \overset{def}{=} 2^q$, $b \mid l$, K este o cheie specială *b-exactă* de dimensiune l cuvinte. Cu ajutorul termenului $\left(\frac{1}{N}\right)^N$ se stabileşte nivelul maxim al probabilităţii ca $KSAm(K)$] să fie *aproape* b_m*-conservativă*.

Pentru găsirea eventualelor mapări ale biţilor din cheie în şabloane identificabile din permutarea iniţială S se propune un model/cadru de abordare algoritmică a cărui formalizare poate fi utilizată pentru testarea oricărui mecanism de tip *Scrambling* aplicat unei permutări căreia i se inter-schimbă la fiecare pas valorile a două intrări pe baza a doi indici. Prin intermediul acestui model se poate determina numărul de cuvinte din permutarea S pe baza cunoaşterii unui număr m de cuvinte din cheia K:

$NR(m) = [2^n \times (m/l)] \times \alpha = [2^n \times (m/l)] \times (1/2^q)^{l-m}$, unde $\alpha = (1/2^q)^{l-m}$, n este dimensiunea cuvântului în biţi, l reprezintă numărul de cuvinte ai cheii K *b-exact*, m este numărul de cuvinte din cheia K ($0 \le m \le l$).

Pe baza demonstraţiilor lui Paul şi Maitra din [PMA07] referitoare la corelările dintre octeţii permutării S şi octeţii cheii secrete K la fiecare pas al algoritmului *KSA*, se analizează aceleaşi probabilităţi de corelare a valorilor intrărilor permutării S cu valorile cheii secrete K după fiecare pas al secvenţei *Scrambling*_2 din cadrul algoritmului *KSAm*. Se determină şi se demonstrează probabilitatea generală a (*b*)-*conservativităţii* permutării S la fiecare pas al *Scrambling*_2, cu

ipoteza existenţei (*b*)-*conservativităţii* la terminarea *Scrambling*_1 (*Scrambling*_2 primeşte la finalul *Scrambling*_1 o permutare *S b-conservativă*) pentru orice tip de clasă de cheie *K*. Permutarea identică *S* şi *b-conservativitatea* permutării *S* sunt noţiuni apropiate, principala diferenţă constă în faptul că aflarea probabilităţii rămânerii permutării *S* la permutarea identică după operaţii de tip *Scrambling* presupune ca stare de pornire însăşi permutarea identică, iar *b-conservativitatea* porneşte de la orice stare iniţială (poate fi şi permutarea identică), însă valorile intrările *i* sunt aduse la valorile *i mod b* prin orice tip de operaţii tip *Scrambling*.

Pornind de la probabilitatea generală a (*b*)-*conservativităţii* permutării *S* la fiecare pas al *Scrambling*_2, se determină și se propune un model de abordare a criptanalizei prin care se speculează natura algoritmului *Scrambling* combinat cu utilizarea de chei slabe *K*, astfel încât să se descopere o corelare proporţională între starea iniţială a permutării *S* şi a cheilor slabe *K* cu starea finală a permutării *S – criptanaliza stării permutării S, bazată pe cunoaşterea stării interne iniţiale a permutării S, pe probabilitatea păstrării stării interne iniţiale a permutării S şi pe probabilitatea* (*b*)-*conservativităţii permutării S, în urma succesiunii unor cicluri Scrambling de permutare a valorilor elementelor permutării S*. De asemenea, se calculează complexitatea de timp a acestui model. Modelul formalizează o strategie de criptanaliză a permutărilor folosite în cifrurile secvențiale, care asigură un avantaj minim care poate fi îmbunătăţit prin specularea caracteristicile algoritmilor *Scrambling*, a cheilor slabe *K* şi a stării de iniţializare a permutării *S*.

Pentru urmărirea propagării unor părţi din cheile *K* slabe de criptare în cadrul cuvintelor Z de ieşire ale *RC4m*, se pornește de la o variantă vulnerabilă a algoritmului RC4, $RC4^{*}_{KSAm}$, bazată pe *KSAm*, din care se elimină operaţiile de schimbare a valorilor intrărilor vectorului de stare *S* (operaţiile de tip *swap*). Urmând acelaşi raţionament ca cel din [FMS01], se poate afirma că fluxul de ieşire generat de *RC*4*m* nu poate fi corelat cu fluxul de ieşire generat de $RC4^{*}_{KSAm}$, deoarece *b-neconservarea* intrărilor permutării *S* presupune generare de flux (pseudo)-aleatoriu în ambele cazuri. Teoretic, corelarea care poate să apară este între un flux generat de $RC4^{*}_{KSAm}$ şi unul produs de *RC*4*m* dintr-o permutare *aproape* b_m*-conservativă*. Însă, corelarea trebuie legată de un şablon $\{x_t\}_{t=1}^{\infty}$ cu caracteristici cunoscute (secvenţă $\{x_t\}_{t=1}^{\infty}$ constantă), extras dintr-un flux generat de $RC4^{*}_{KSAm}$, şablon care nu a putut fi determinat. Chiar dacă această corelare ipotetică ar fi fost determinată, avantajul obţinut ar fi fost maxim doar pentru prima ieşire *Z*[1] a *RC*4*m* (avantaj propagat în *PRGA*), după care, pentru următoarele cuvinte de ieşire, avantajul scade proporţional cu un factor egal cu *b*, care reprezintă de fapt probabilitatea ca următorul

cuvânt de ieşire să coincidă (*mod b*) cu anteriorul, astfel încât secvenţa de ieşire $\{x_t\}_{t=1}^{\infty}$ să fie constantă (o dată cu avansarea producerii cuvintelor de ieşire *Z* are loc difuzia graduală a valorii *S*[(*S*[*i*] + *S*[*j*]) *mod N*], astfel încât valorile *Z* vor avea probabilitatea uniformă 1/*N* sau 1/(*N mod b*)). Analiza ia în considerare o variantă generală $PRGA_{gen}$ uşor modificată faţă de *PRGA*, care presupune că valorile *j* sunt alese în mod aleatoriu la fiecare pas, şi nu sunt obţinute neapărat conform relaţiei $j = (j + S[i]) \ mod \ N$ (diferenţa $\Delta(PRGA - PRGA_{gen})$ nu afectează analiza teoretică).

Se determină corelarea cuvintelor de ieşire *Z* cu o secvenţă $\{x_t\}_{t=1}^{\infty}$ constantă, precum şi probabilităţile maxime de determinare a primelor patru cuvinte constante *Z*. Valorile obţinute pentru *Z*[1] şi *Z*[2] nu pot fi considerate corelări semnificative între cheia secretă *K* şi cuvintele de ieşire *Z* care să constituie un avantaj criptanalitic important.

Una din cele mai importante observaţii de ordin statistic a fost realizată de Mantin şi Shamir în [MS02]: probabilitatea ca al doilea cuvânt de ieşire *Z* să fie 0 este de 2/*N*. În aceleaşi condiţii, pentru *KSAm*, se urmăreşte probabilitatea pentru care se păstrează corelările celui de-al doilea cuvânt de ieşire *Z*[2] = 0 cu probabilităţile 2/*N* şi 1. O observaţie importantă este aceea că analiza se bazează pe probabilitatea ca *S*[2] = 0 şi *S*[1] ≠ 2 după *KSAm*, în contextul în care se consideră că toate valorile intrărilor *S* sunt aleatorii după terminarea *KSAm*. Se demosntrează că probabilitatea cumulată a evenimentelor $S_{0\_S2}[2] = 0$ şi $S_{0\_S2}[1] \neq 2$, pentru care există corelarea celui de-al doilea cuvânt de ieşire *Z*[2] = 0 cu probabilitatea 1, este mai mică de 1/*N*. În concluzie, corelarea descrisă de Mantin şi Shamir, deşi nu poate fi anulată (*RC*4 şi *RC*4*m* folosesc acelaşi *PRGA*), se reduce în condiţiile în care cele două cicluri *Scrambling* ale *KSAm* distribuie valori aleatorii intrărilor *S*[1] şi *S*[2].

În [FMS01] şi [MAN201] se detaliază efectul criptanalitic al combinării cheii secrete cu vectori de iniţializare *IV*. Această metodă de combinare este specifică protocolului *WEP*, iar studiile de specialitate, în marea lor majoritate, subliniază faptul că insecuritatea protocolului *WEP* nu este datorată algortimului *RC*4, ci felului în care este utilizat mecanismul de combinare a cheii secrete cu un vector de iniţializare. Atacul asupra criptosistemelor care folosesc concatenarea cheii secrete cu vectorul *IV* permite construirea cheii secrete prin urmărirea şi analizarea primului cuvânt al fluxului de ieşire produs de un număr redus de chei de sesiune (atacul porneşte de la presupunerea, cu o mare probabilitate corectă, că primul cuvânt al textului în clar este, în mare măsură, o valoare constantă). Vulnerabilitatea *IV* este analizată pentru *KSAm*. În cazul în care vectorul de inițializare *IV* precede cheia secretă, în situația cea mai defavorabilă, şi anume chei

secrete formate doar din vectori *IV*, deci chei a căror valoare este complet cunoscută, numărul planificat de vectori *IV* necesari obţinerii celor aproximativ 100 de vectori *IV* de formă corespunzătoare este de aproximativ 8000000. Excluzând ultima variantă (chei formate doar din vectori *IV*), atacul *IV* devine practic nefezabil. În cazul în care cheia secretă precede vectorul de iniţializare *IV*, se arată că nu se pot manipula prin intermediul vectorilor *IV* cuvintele de ieşire astfel încât să se poată observa starea permutării *S* care să determine, mai departe, obţinerea cheii secrete. În cazul în care vectorul *IV* este combinat XOR cu cheia secretă, complexitatea atacului este egală cu complexitatea căutării exhaustive. Însă, aşa cum remarcă şi Mantin în [MAN201], un atac care să ofere o oarecare certitudine de reuşită trebuie să coreleze alegerea şi utilizarea vectorilor din cele trei etape (în *KSAm* este dificil de realizat din perspectiva diferenţei algoritmilor *Scrambling*_1 şi *Scrambling*_2), iar numărul vectorilor *IV* folosiţi ar trebui să fie N^A (*A* reprezintă numărul de cuvinte ale cheii secrete), ceea ce înseamnă că complexitatea de timp devine N^ℓ (ℓ este dimensiunea cheii secrete) – rezultă, astfel, o căutare exhaustivă indiferent de dimensiunea cheii (cu cât cheia este mai mare, cu atât atacul devine mai nefezabil).

Criptanaliza *KSAm* descrie două proprietăţi combinatoriale ale *KSA*m în modul său normal de operare, şi nu din perspectiva unei implementări particulare. În primul rând, pe baza unui model de amestecare (*Scrambling*) prezentat de Mironov în [MIR02], în care este introdus un cifru *RC*4 presupus ideal, se calculează semnul permutării *S* după terminarea *KSAm*, ale cărui valori permit predicţia unui bit cu un avantaj de 0.91 % faţă de ghicirea aleatorie. Acest avantaj rămâne totuşi prea mic pentru a fi fezabil în cadrul unui atac. În al doilea rând, se analizează intrările tabelei de stare *S* în cursul rulării *KSAm*, cu acordarea unei atenţii sporite deducerii probabilităţii de avansare liniară, la fiecare pas al *KSAm*, a unei valori iniţiale *b* de-a lungul vectorului *S*, precum şi calculării probabilităţii ca valoarea *b* să ocupe la finalul *KSAm* poziţia corespunzătoare unei anumite intrări *a* (acest caz include şi probabilitatea obţinerii permutării identice, adică $a = b$) [CRB10]. Se demonstrează că evenimentul ca o valoare particulară *b* să urmeze o cale liniară prin vectorul *S* şi, în consecinţă, să se găsească într-o locaţie previzibilă după *Scrambling*_1, are probabilitatea de aproximativ $1/N$ – găsirea unei locaţii în *S*, în timpul unei asemenea deplasări liniare, a cărei valoare să fie prezisă cu o probabilitate semnificativ mai mare decât $1/N$, este extrem de improbabilă. Acest rezultat poate fi luat în considerare în contextul startării secvenţei *Scrambling*_2 cu o tabelă de stare *S* care este diferită de permutarea identică cu o probabilitate foarte mare.

Criptanaliza algoritmului *KSAm* şi, implicit a algoritmului *RC*4*m*, presupune adaptarea şi testarea metodelor de atac care au avut succes şi care au dezvăluit vulnerabilităţi semnificative ale

algoritmului *KSA*, respectiv *RC*4. S-au ales şi au fost adaptate trei metode de atac prin intermediul cărora au fost identificate breşe de securitate ale *KSA*/*RC*4: atacul *OSM* [OS05], atacul *Roos* [ROO95] şi atacul *Klein*/*PTW* [KLA08], [TW08]. Pentru atacul *OSM*, rezultatele obținute demonstrează nefezabilitatea atacului *OSM* asupra *WEP* cu *KSAm*, în condiţiile în care eficienţa minimă are valoarea $2^{103.92}$. În plus, valorile parametrului eficienţă sunt mult mai mari decât complexitatea căutării exhaustive a cheii. Deşi atacul este de tip *vector IV cunoscut* şi, astfel, un număr semnificativ de vectori *IV* ar putea fi transformaţi (teoretic) în vectori *IV* slabi, eficienţa atacului obţinută atât teoretic, cât şi experimental, asupra unei chei de 128 biţi (24 biţi vector *IV*, 104 biţi cheia secretă), nu permite recuperarea intrărilor cheii secrete într-un timp practic computaţional. Adaptarea abordării *Roos* la *KSAm* dezvăluie o uşoare corelare a primelor două ieşiri cu prima, respectiv cu suma primelor două valori ale cheii *K*. Fiind extrem de redusă, această corelare nu are un efect criptanalitic important, mai ales că, începând cu a patra ieşire, valorile produse sunt asimptotice valorii 0,39. Concluziile care se desprind în urma atacului *Klein*/*PTW* aplicat asupra *RC*4*m* sunt: în primul rând, se observă protecţia incomparabil superioară furnizată de *RC*4*m*; apoi, dimensiunea mărită a cheii iniţiale de criptare nu are o influenţă proporţională cu diferenţa de mărime dintre cheile de 104 biți şi 232 biți; o diferenţă atât de mare între dimensiunile cheilor ar trebui să se observe semnificativ în nivelul de protecţie a traficului. Astfel, deşi numărul biţilor recuperaţi din cheia iniţială de criptare nu pare să atingă un prag periculos, vectorii de iniţializare *IV*, datorită dimensiunii lor reduse, reprezintă principala cauză a vulnerabilităţii protocolului *WEP*-*RC*4*m*. O altă observaţie constă în faptul că existenţa protocolului de difuzare *ARP* determină scăderea protecţiei, dezactivarea acestuia reducând numărul biţilor recuperaţi din cheia de criptare.

Ultima parte a tezei se concentrează pe aspecte privind corelarea intrărilor stărilor unui algoritm *PRGA* bazată pe conceptul de diferenţiator şi pe mecanisme de atac asupra stării interne pornind de la valorile diferenţiatorilor. Se introduc şi se definesc următoarele noțiuni: *diferențiator*, *diferenţiator*$_{indicator}$, *corelare*$_{generală}$, *corelare*$_{diferenţiator}$, *corelare pe șablon*, *şablon asociat secvenţei de ieşire*. Pornind de la aceste noțiuni, se propun modele de atac asupra stării/stărilor interne (iniţiale/intermediare) care iau în considerare acele intrări din starea iniţială care au cea mai mare probabilitate de a fi corelate cu un anumit diferenţiator. Fiecărui atac îi este asociat formula complexității.

Se introduce factorul de corelare r_c cu scopul redefinirii conceptelor de *a-stare* şi *b-predictivitate* din [MS02], astfel încât să fie luate în considerare corelările cu stările iniţiale, şi al modelării procesului de identificare a diferenţiatorilor bazaţi pe stări predictive din punct de vedere

statistic. Factorul de corelare devine un element de criptanaliză deosebit de util în contextul în care starea iniţială secretă a unui algoritm *PRGA* este obţinută dintr-o succesiune de cicluri diferite bazate pe pe constrângeri diferite şi pe o cheie secretă – *RC*4*m* reprezintă un astfel de caz datorită algoritmului *KSAm* care conţine două secvenţe diferite de tip *Scrambling*. Factorul de corelare r_c poate fi considerat şi interpretat sub forma unui coeficient Pearson care măsoară corelarea între două variabile (legătură liniară), însă, în cazul de faţă, corelarea existentă între starea iniţială şi starea finală se defineşte din perspectiva diferenţiatorilor care permit separarea secvenţei/fluxului de ieşire (starea finală) de o secvenţă (pseudo)-aleatorie. În cazul coeficientului Pearson, corelarea nu înseamnă o legătură de cauzalitate, pe când corelarea $< stare\_iniţială > \leftrightarrow < stare\_finală >$ influenţează decisiv detectarea diferenţiatorilor – stări iniţiale diferite pot determina diferenţiatori diferiţi. Din punct de vedere criptanalitic, această observaţie devine deosebit de importantă, deoarece stările iniţiale care determină diferenţiatori pot fi declarate stări iniţiale slabe şi, eventual, eliminate (eliminarea unui set mare de stări iniţiale slabe înseamnă implicit un grad de vulnerabilitate ridicată a algoritmului, ceea ce ar putea însemna modificarea structurală a algoritmului de generare de secvenţe (pseudo)-aleatoarii). Pe de altă parte, în cazul în care algoritmul *PRGA*, produce câte un singur cuvânt de criptare pe ciclu, acesta devine un element al secvenţei/fluxului de ieşire (stării finale), care, cu cât este produs mai târziu, va fi influenţat cu o probabilitate superioară de un număr mai mare de intrări ale stării iniţiale comparativ cu un element produs anterior.

Se definesc şi se introduc noţiunile $diferenţiator_{puternic}$, diferenţiator a cărui valoare este validată în conformitate cu o valoare de prag ε stabilită printr-o anumită metrică asociată *proprietăţii de a fi aleatoriu*, şi pentru care nu mai sunt necesare confirmări suplimentare, şi $diferenţiator_{slab}$, diferenţiator a cărui valoare se consideră că face parte din intervalul de toleranţă al valorii de prag ε, dar care, în condiţiile în care se pot furniza verificări suplimentare pe baza altor metrici, poate fi declarat, cu probabilitate ridicată, $diferenţiator_{puternic}$. În practică, limitele de operare ale $diferenţiatorului_{slab}$ sunt uneori dificil de stabilit. Soluţia cea mai simplă ar consta în eliminarea acestui concept şi stabilirea unui prag ε unic, dependent de metrica aleasă pentru proprietatea de a fi (pseudo)-aleatoriu, care să facă diferenţa dintre un *diferenţiator* şi o valoare (pseudo)-aleatorie. Însă avantajul $diferenţiatorului_{slab}$ apare (nu întotdeauna) în cazurile în care acesta este $diferenţiator_{indicator}$ al unui şablon şi se află înaintea unui $diferenţiator_{puternic}$, permiţând identificarea mai facilă a şabloanelor (a funcţiei de *corelare internă* g) – dacă primul diferenţiator identificat este $diferenţiator_{puternic}$ şi scopul analizei constă în determinarea caracterului (pseudo)-aleatoriu a secvenţei de ieşire, atunci se termină verificarea celorlalte

intrări ale secvenţei de ieşire; dacă diferenţiatorul este $diferenţiator_{slab}$ şi nu poate fi validat ca $diferenţiator_{puternic}$, se trece la verificarea celorlaltor intrări ale secvenţei de ieşire până se găseşte un $diferenţiator_{puternic}$, iar dacă se reuşeşte identificarea unui $diferenţiator_{puternic}$, atunci, cu o anumită probabilitate, acesta face parte dintr-un şablon împreună cu $diferenţiatorul_{slab}$ identificat anterior care devine $diferenţiator_{indicator}$ al respectivului şablon (pot fi identificaţi mai mulţi $diferenţiatori_{slabi}$ până la identificarea $diferenţiatorului_{puternic}$). Determinarea şablonului şi a $diferenţiatorului_{indicator}$ corespunzător poate fi utilizată apoi într-un atac de recuperare a intrărilor stării iniţiale.

Se prezintă algoritmii de alegere a $diferenţiatorului_{puternic}$ şi a corelării, algoritmi care pot fi utilizaţi în atacurile asupra stării/stărilor interne (iniţiale/intermediare). Pe baza abordărilor din [MS02] şi [FM01], se propune extinderea şi nuanţarea definiţiei *a-stării*, cu aplicabilitate atât pentru *KSA*, cât şi pentru *KSAm*, din perspectiva stării iniţiale şi a stărilor intermediare care produc fiecare câte o valoare de criptare componentă a secvenţei de ieşire, incluzându-se corelarea (factorul de corelare r_c) dintre stările intermediare (starea algoritmului $PRGA_{RC4/RC4m}$ la fiecare pas) şi starea inţială. În felul acesta, abordarea propusă permite ca diferenţiatorii să poată fi construiţi fie pe baza *a-stărilor predictive* aşa cum sunt definite în [MS02], fie pe baza unor *a-stări intermediare predictive* din care se produc direct valorile secvenţei de ieşire, astfel încât pentru aceste valori de ieşire, printre care se pot afla şi diferenţiatori, se pot construi mult mai facil relaţii de corelare cu starea iniţială prin intermediul *a-stării predictive intermediare*. Din punct de vedere criptanalitic, corelarea cu starea iniţială de la care se porneşte, combinată cu vulnerabilităţile specifice şi cunoscute ale algoritmilor *PRGA* individuali (în particular, $PRGA_{RC4/RC4m}$), permite predicţii mai bune ale intrărilor secvenţei de ieşire T_i; pe de altă parte, diferenţiatorii îşi găsesc utilitatea în procesul analizării secvenţei de ieşire din perspectiva proprietăţii de a fi (pseudo)-aleatoriu (cazul general), şi în posibilitatea de eliminare a stărilor iniţiale corelate cu respectivii diferenţiatori (cel puţin, eliminarea stărilor iniţiale corelate cu $diferenţiatorii_{puternici}$ şi/sau $diferenţiatorii_{indicator}$). Pentru *KSAm* s-a efectuat un set limitat de teste bazat pe lexicul limbii române. Testele se bazează pe analiza statistică a operei lui Mihai Eminescu din [SEC74]. Conform [SEC74], cele mai utilizate cuvinte din opera eminesciană sunt pronumele personal *el*, *ea* (frecvenţă absolută 3058, frecvenţă relativă 4.65%), iar frecvenţa cea mai mare din categoria substantivelor o are cuvântul *ochi* (frecvenţă absolută 299, frecvenţă relativă 0.44%). S-au generat cu *RC*4*m* 100000 de secvențe/șiruri formate din patru cuvinte şi s-au folosit testele [DIE] *OPSO* (*Overlapping Pairs Sparse Occupance*), *C1s* (*Count the1s, stream*) şi *Serial*. Rezultatele au arătat că cel de-al treilea cel mai puțin semnificativ bit al

secvențelor/șirurilor produse are o corelare suplimentară de 0.01 pe valoarea 0, cu alte cuvinte valoare 0 pe poziția celui de-al treilea cel mai puțin semnificativ bit reprezintă un diferențiator. Valoarea 0.01 se adaugă la valoarea de corelare medie 0.02 obținută experimental anterior, rezultând o valoare de corelare de 0.03. Însă se poate afirma că testele au folosit seturi de date de volum mult prea mic pentru a putea valida rezultatele. În orice caz, valoarea de corelare obținută de 0.03 pentru un singur bit din secvența produsă reprezintă în practică un avantaj total neglijabil.

În finalul tezei, se propune un nou concept de cheie slabă, şi anume cheia *X_N-slab-incertă*. Unul din avantajele algoritmului *KSAm* constă în faptul că cele două secvenţe *Scrambling* sunt diferite, fiind astfel dificil de găsit clase de chei care să fie slabe atât pentru *Scrambling*_1, cât şi pentru *Scrambling*_2. În acest caz, primul nivel de criptanaliză constă în verificarea cheilor slabe *KSA* asupra secvenţei *Scrambling*_1 şi a comportamentului combinat asupra *KSAm* în ansamblu, iar apoi găsirea cheilor slabe aferente secvenţei *Scrambling*_1 şi testarea acestora asupra *Scrambling*_2. Concluzia finală constă în faptul că o clasă de chei de forma $K[0] = K[1] = \ldots = K[\ell - 1] = c$, unde $c > 0$, $c \in \mathbb{Z}^+$, care sunt totuşi chei slabe datorită alocării de valori egale tuturor elementelor vectorului *K*, nu este o clasă de chei *X_N-slab-incerte* şi, astfel, nu lasă permutarea *S* într-o stare *X_N-incertă* după două cicluri *Scrambling* diferite din punct de vedere al algoritmului implementat, dar care folosesc aceeaşi cheie slabă $K = c$. Chiar dacă după *Scrambling*_1 permutarea *S* atinge starea *X_N-incertă*, *Scrambling*_2 distruge modelul ciclului de deplasare (la dreapta), şi lasă în final valorile elementelor permutării *S* distribuite (pseudo)uniform. Acesta este şi motivul pentru care succesiunea a două secvenţe *Scrambling* diferite de *N* paşi fiecare, alese în mod corespunzător, sunt mult mai rezistente din punct de vedere criptanalitic decât o singură secvenţă de tip $Scrambling_{gen}$ care rulează în $2N$ paşi.

ANEXA 1 cuprinde bateria de teste Dieharder aplicată secvenţei formată din 3903299 ieşiri *Z*1 produse de *RC*4*m* pentru un număr de 3903299 chei Roos de 128 biţi care satisfac relația $<K[0] + K[1] = 0>$

ANEXA 2 cuprinde distribuţia statistică a celor 256 de valori posibile [0 - 255] ale primilor opt octeţi (*Z*1, *Z*2, *Z*3, *Z*4, *Z*5, *Z*6, *Z*7, *Z*8) ai fluxului/şirului de criptare (*keystream*) pentru 10000000 de chei de 128 biți generate aleator

Bibliografie

[AB13] Alfardan, N., Bernstein, D., J., Paterson, K., Poettering, B., Schuldt, J., "On the Security of RC4 in TLS", USENIX Security Symposium, USENIX, 2013.

[AS02] Arbaugh, W., A., Shankar, N., Justin Wan, Y., C., "Your 802.11 Wireless Network has No Clothes", *IEEE Wireless Communications*, Vol. 9, No. 6, 2002, pp. 44-51. http://www.cs.umd.edu/~waa/wireless.pdf.

[BAB95] Babbage, S., "Improved Exhaustive Search Attacks on Stream Ciphers", European Convention on Security and Detection, IEE Conference Publication no. 408, IEE, 1995, pp. 161–166.

[BS06] Barkan, E., Shamir, S., "Rigorous Bounds on Cryptanalytic Time/Memory Tradeoffs", *Advances in Cryptology - CRYPTO 2006*, Lecture Notes in Computer Science, Vol. 4117, ISBN 978-3-540-37432-92006, 2006, pp 1-21.

[BH09] Bienvenu, L., Hölzl, R., Kräling, T., Merkle, W., "Separations of non-monotonic randomness notions", 2009. http://arxiv.org/abs/0907.2324.

[BD89] Beth, T., Dai, Z., D., "On the Complexity of Pseudo-Random Sequences - or: If You Can Describe a Sequence It Can't be Random", Advances in Cryptology — EUROCRYPT '89, Lecture Notes in Computer Science, Vol. 434, Springer Berlin Heidelberg, ISBN 978-3-540-53433-4, 1990, pp. 533-543.

[BSW01] Biryukov, A., Shamir, S., Wagner, D., "Real time cryptanalysis of A5/1 on a PC", Proceedings of PKC 2001, Lecture Notes in Computer Science, Vol. 1978, Springer-Verlag, 2001, pp. 37-44.

[BIT03] Bittau, A., "Additional weak IV classes for the FMS attack", Department of Computer Science, University College London, 2003. http://www.cs.ucl.ac.uk/staff/a.bittau/sorwep.txt.

[BIT06] Bittau, A., Handley, M., Lackey, J., "The Final Nail in WEP's Coffin", in *Proc. 2006 IEEE Symposium on Security and Privacy, S&P'06*, 2006, pp. 386-400. http://tapir.cs.ucl.ac.uk/bittau-wep.pdf.

[BM84] Blum, M., Micali, S., "How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits", *SIAM Journal on Computing*, Vol. 13, 1984, pp. 850-864.

[BG01] Borisov, N., Goldberg, I., Wagner, D., "Intercepting mobile communications: The insecurity of 802.11", in. *Proc. 7th Annual International Conference on Mobile Computing and Networking, MobiCom '01*, Rome, pp. 180–189, 2001. http://www.cypherpunks.ca/~iang/pubs/wep-mob01.pdf.

[BK07] Barker, E., Kelsey, J., "Recommendation for Random Number Generation Using Deterministic Random Bit Generators (Revised)", *NIST Special Publication 800-90*, Computer Security Division Information Technology Laboratory, 2007.

[DIE] Braun, R., G., Dieharder: A Random Number Test Suite. http://www.phy.duke.edu/~rgb/General/dieharder.php.

[BRO52] Broad, C. D., "Ethics and the History of Philosophy", *New York: Humanities Press*, 1952.

[CAL03] Calhoun P., Loughney J., Guttman E., Zorn G., Arko J., “Diameter Base Protocol”, Request for Comments: 3588, Network Working Group, 2003.

[CHA66] Chaitin, G., “On the length of programs for computing finite binary sequences” , *Journal of the ACM*, 13, 1966, pp. 547-569.

[CHA69] Chaitin, G., “On the length of programs for computing finite binary sequences: statistical considerations”, *Journal of the ACM* 16, 1969, pp.145-159.

[CHA74] Chaitin, G., ”Information-theoretic Limitations of Formal Systems", *J. ACM* 21, 1974, pp. 403-424.

[CHA75] Chaitin, G., "A Theory of Program Size Formally Identical to Information Theory." *J. Assoc. Comput. Mach.*, 22, 1975, pp. 329-340.

[CHAG75] Chaitin, G., “Randomness and Mathematical Proof”, *Scientific American 232*, No. 5, 1975, pp. 47-52.

[CHA76] Chaitin, G., J., ”Algorithmic Entropy of Sets", *Comput. & Math. Appls.* 2, 1976, pp.233-245.

[CHAG76] Chaitin, G., ”Information-theoretic Characterizations of Recursive Infinite Strings", *Theoret. Comput. Sci.* 2, 1976, pp.45-48.

[CHA77] Chaitin, G., “Algorithmic information theory”, *IBM Journal of Research and Development*, vol. 21, 1977, pp. 350–359.

[CHAG77] Chaitin, G., ”Program Size, Oracles, and the Jump Operation", *Osaka J. Math.*, Vol. 14, No. 1, 1977, pp. 139-149.

[CHA82] Chaitin, G., “Gödel's theorem and information”, *Int. J. Theor. Phys.*, 21**,** 1982, pp. 941-954.

[CHA90] Chaitin, G., “A random walk in arithmetic”, *New Scientist 125*, No. 1709, 1990, pp. 44-46.

[CHU40] Church, A., “ On the concept of a random sequence”, *Bulletin of AMS*, vol.46, 1940, pp.130-135.

[COV06] Cover, T., M., Thomas, J., A., “Elements of Information Theory”, 2nd ed., New York, Wiley, 2006.

[CRA05] Crainicu, B., “An Overview of the IPsec Extensions Header – AH (Authentication Header and ESP (Encapsulating Security Payload)”, *Education/Training and Information/Communication Technologies – RoEduNet’05*: *Proceedings of the 4th International Conference RoEduNet Romania*: Târgu Mureş – Sovata, 20-22 May 2005, Editura Universităţii “Petru Maior” din Târgu Mureş, 2005, ISBN 973-7794-29-X, pp. 245-254.

[CRAI05] Crainicu, B., “Web Security: Secure Socket Layer and Transport Layer Security“, *Interdisciplinarity in Engineering: Proceedings of the Scientific Conference Inter-Ing 2005*: Târgu Mureş, “Petru Maior” University, Faculty of Engineering, 10-11 November 2005, Editura Universităţii “Petru Maior” din Târgu Mureş, 2005, ISBN 973-7794-41-9, 2005, pp. 787-800.

[CM06] Crainicu, B., Măruşteri, M., “PGP Cryptographic Keys and Key Rings“, *Proceedings of the 5th RoEduNet IEEE International Conference*: Sibiu, 1-3 June 2006, Romania, Editura Universităţii “Lucian Blaga” din Sibiu, 2006, ISBN (10) 973-739-277-9, (13) 978-973-739-277-0, pp. 138-141.

[CRA08] Crainicu, B., "Wireless LAN Security Mechanisms at the Enterprise and Home Level", *Novel Algorithms and Techniques In Telecommunications, Automation and Industrial Electronics*, Springer Netherlands, ISBN978-1-4020-8736-3 (Print), 978-1-4020-8737-0 (Online), 2008, pp. 305-310.

[CI08] Crainicu, B., Iantovics, B.L., "Cryptanalysis of KSAm-like Algorithms", *First International Conference on Complexity and Intelligence of the Artificial and Natural Complex Systems. Medical Applications of the Complex Systems. Biomedical Computing. CANS 2008*, Published by IEEE Computer Society, 2008, IEEE Computer Society Order Number P3621, ISBN 978-0-7695-3621-7, Library of Congress Number 2009900732, 2008, pp. 130-148.

[CRI08] Crainicu, B., Iantovics, B., "On A New RC4 Key Scheduling Algorithm", *Proceedings of the International Conference Complexity and Intelligence of the Artificial and Natural Complex Systems. Medical Applications of the Complex Systems. Biomedical Computing, CANS 2008*, 8-9 November, Târgu Mureş, 2008, Editura Universităţii "Petru Maior" Târgu-Mureş, 2008, ISSN 2065-0426, pp. 16-25.

[CIA08] Crainicu, B., Iantovics, B., "Securing WEP Cryptosystems through A New RC4 Key Scheduling Algorithm", *Complexity in Artificial and Natural Systems*, Editura Universităţii "Petru Maior" Târgu-Mure, ISBN 978-973-7794-76-5, 2008, pp 93-99.

[CRA09] Crainicu, B., "A Local Search Approach for Recovering an Internal State of RC4 Stream Cipher", *11th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2009*, Timisoara, Romania, September 26-29, 2009.

[CB10] Crainicu, B., Boian, F., "KSAm – An Improved RC4 Key-Scheduling Algorithm for Securing WEP", *Novel Algorithms and Techniques in Telecommunications and Networking,* Springer Netherlands, ISBN 978-90-481-3661-2 (Print), ISBN 978-90-481-3662-9 (Online), 2010, pp. 391-396.

[CRB10] Crainicu, B., Boian, F., M., "Some Combinatorial Aspects of the KSAm-like Algorithms Suitable for RC4 Stream Cipher", *Studia Universitatis Babes-Bolyai*, Series Informatica, Volume LV, Number 1, 2010, ISSN 1224-869x (paper version), ISSN 2065-9601 (online version), pp. 105-114.

[CE11] Crainicu,B., Enăchescu, C., "A Metaheuristic Tabu Search Approach for Internal State Reconstruction of RC4", Proceedings *of 10th RoEduNet IEEE International Conference*, Iaşi, Romania, 23-25 June 2011, Published by Stef, 2011, ISSN 2247-5443, pp. 164-167.

[CI11] Crainicu,B., Iantovics, B., "An Agent-based Security Approach for Intrusion Detection Systems", *7th International Workshop on Grid Computing for Complex Problems*, *GCCP2011*, Bratislava, Slovakia, October 24 - 26, 2011, Institute of Informatics, Slovak Academy of Sciences, ISBN 978-80-970145-5-1, 2011, pp. 126-133.

[CRA14] Crainicu, B., "On Invariance Weakness in the KSAm Algorithm", *8th International Conference Interdisciplinarity in Engineering, INTER-ENG 2014*, 9-10 October 2014, Tirgu-Mures, Romania, Elsevier, ISSN 2212 – 0173, pp. 850-857.

[EAG05] Eagle, A., "Randomness is Unpredictability", *The British Journal for the Philosophy of Science*, Vol. 56, Issue 4, 2005, pp. 749-790.

[EA03] Edney, J., Arbaugh, W., A., "Real 802.11 Security: Wi-Fi Protected Access and 802.11i", Addison Wesley, 2003.

[DAS11] Dasgupta, A., “Mathematical Foundations of Randomness”, *Philosophy of Statistics*, North Holland, 2011, pp. 641-710.

[CLP05] De Canniere, C., Lano, J., Preneel, B., "Comments on the rediscovery of time memory data tradeoffs," *eSTREAM, ECRYPT Stream Cipher Project, Report 2005/040*, 2005.

[DH03] Doraswamy, N., Harkins, D., “IPSec: the new security standard for the Internet, intranets, and virtual private networks”, Second Edition, Prentice-Hall, 2003

[DG02] Downey, R., Griffiths, E., “Schnorr randomness”, *Electronic Notes in Theoretical Computer Science*, 66(1), 2002. http://www.elsevier.nl/locate/entcs/volume66.html.

[FER14] Ferriman, B., “Cryptanalysis of the RC4 Stream Cipher using Evolutionary Computation Methods”, School of Computer Science, Master Thesis, University of Guelph, Canada, 2014.

[FIN94] Finney, H., “An RC4 cycle that can’t happen”, Post in sci.crypt, September 1994.

[FIS49] Fisher, R., A., Yates, F., “Statistical tables for biological, agricultural and medical research”, 3rd Edition, London, Oliver and Boyd, 1949.

[FM01] Fluhrer, S., McGrew, D., “Statistical analysis of the alleged RC4 keystream Generator”, in. *Proc. 7th International Workshop, FSE 2000*, New York, Lecture Notes in Computer Science, Vol. 1978, Springer-Verlag, 2001, pp. 66-71.

[FMS01] Fluhrer, S., Mantin, I., Shamir, A., “Weaknesses in the key scheduling algorithm of RC4”, in *Proc. 8th Annual International Workshop, SAC 2001*, Toronto, Lecture Notes in Computer Science, Vol. 2259, Springer-Verlag, 2001, pp. 1-24.

[FMS02] Fluhrer, S., Mantin, I., Shamir, A., “Attacks on RC4 and WEP”, *CryptoBytes (RSA Laboratories)*, Vol. 5, No. 2, 2002, pp. 26–34. http://www.rsa.com/rsalabs/cryptobytes/cryptobytes_v5n2.pdf

[FUN]Funes, P., “Complexity measures for complex systems and complex objects”. http://www.cs.brandeis.edu//~pablo/complex.maker.html

[GAC74] Gács, P., "On the symmetry of algorithmic information”, *Soviet Math. Dokl*., 15, 1974, pp. 1477–1480.

[GAS05] Gast, M., “802.11 Wireless Networks: The Definitive Guide”, Second Edition, O’Reilly, 2005

[GELL95] Gell-Mann, M., “What is Complexity?” *Complexity* 1/1**,** 1995, pp. 16-19.

[GELL96] Gell-Mann, M., Lloyd, S., “Information Measures, Efective Complexity, and Total Information.”, *Complexity* 2/1**,** 1996, pp. 44-52.

[GLO86] Glover, F., “Future Paths for Integer Programming and Links to Artificial Intelligence, *Computer and Operations Research*, Vol. 13, No. 5, 1986, pp. 533-549.

[GLO89] Glover, F., “Tabu Search”, Part I, *ORSA Journal on Computing*, vol. 1, no. 3, 1989, pp. 190-206.

[GLO90] Glover, F., “Tabu Search”, Part II, *ORSA Journal on Computing*, vol. 2, no. 1, 1990, pp. 4-32.

[GLOV90] Glover, F., “Tabu Search: A Tutorial”, *Interfaces*, vol. 20, no. 4, 1990, pp. 74-94.

[GL93] Glover, F., Laguna, M., "Tabu search", C. Reeves (ed.) *Modern Heuristic Techniques for Combinatorial Problems,* London, Blackwell, 1993, pp. 70-150.
http://www.dei.unipd.it/~fisch/ricop/tabu_search_glover_laguna.pdf.

[GOD31] Gödel, K., "Über Formal Unentscheidbare Sätze der Principia Mathematica und Verwandter Systeme, I.", *Monatshefte für Math. u. Physik* 38, 1931, pp.173-198.

[GM03] Goldstein D., Moews, D., "The identity is the most likely exchange shuffle for large n", *Aequationes Mathematicae*, Vol. 65, No. 1-2, 2003, pp. 3-30.

[GOM82] Goldwasser, S., Micali, S., "Probabilistic Encryption", *JCSS*, Vol. 28, No. 2, 1982, pp. 270-299.

[GMR89] Goldwasser, S., Micali, S., Rackoff, C., "The Knowledge Complexity of Interactive Proof Systems", *SIAM Journal of Computing*, 18(1), 1989, pp. 186–208.

[GOL97] Golic, J., Dj., "Linear statistical weakness of alleged RC4 keystream generator", in. *Proc. International Conference on the Theory and Application of Cryptographic Techniques, EUROCRYPT '97*, Konstanz, Lecture Notes in Computer Science, Vol. 1233, Springer-Verlag, 1997, pp. 226-238.

[GJD97] Golic, J., Dj., Cryptanalysis of Alleged A5 Stream Cipher, Advances in Cryptology — EUROCRYPT '97, Lecture Notes in Computer Science, Vol. 1233, Springer Berlin Heidelberg, ISBN 978-3-540-62975-7, 1997, pp. 239-255.

[GOL82] Golomb, S., W., "Shift Register Sequences", *Holden-Day, Inc., San Francisco, 1967, revised edition*, Aegean Park Press, Laguna Hills, CA, 1982.

[GG05] Gong, G., Gupta, K., C., Hell, M., Nawaz, Y., "Towards a General RC4-like Keystream Generator", in *Proc. First SKLOIS Conference, CISC 2005*, Beijing, Lecture Notes in Computer Science, Vol. 3822, Springer-Verlag, 2005, pp. 162-174.

[GV03] Grünwald, P., and Vitányi, P., "Kolmogorov Complexity and Information Theory With an Interpretation in Terms of Questions and Answers", *Journal of Logic, Language, and Information*, Vol.12, No. 4, 2003.

[GW04] Grünwald, P., Vitányi, P., "Shannon Information and Kolmogorov Complexity", 2004.
http://homepages.cwi.nl/~paulv/papers/info.pdf.

[GW08] Grünwald, P., Vitanyi, P., "Algorithmic Information Theory", *In Handbook of the Philosophy of Science*, Volume 8: Philosophy of Information. (edited by P. Adriaans and J. van Benthem), Elsevier Science Publishers, 2008, pp 289-325.

[GMA11] Gupta, S.,S., Maitra, S., Paul, G., Sarkar, S., "Proof of empirical RC4 biases and new key correlations", *Selected Areas in Cryptography*, Lecture Notes in Computer Science, Vol. 7118, ISBN 978-3-642-28495-3, 2011, pp. 151-168.

[GMA14] Gupta, S.,S., Maitra, S., Paul, G., Sarkar, S., " (Non-) Random Sequences from (Non-)Random Permutations – Analysis of RC4 Stream Cipher", *Journal of Cryptology*, Vol. 27, Issue 1, ISSN 0933-2790, 2014, pp. 67-108.

[GW00] Grosul A., Wallach, D., “A related key cryptanalysis of RC4”, Technical Report TR-00-358, Department of Computer Science, Rice University, 2000.
http://www.weizmann.ac.il/mathusers/itsik/RC4/Papers/GrosulWallach.ps

[HAN86] Hansen, P., "The Steepest Ascent Mildest Descent Heuristic for Combinatorial Programming", *Congress on Numerical Methods in Combinatorial Optimization*, Capri, Italy, 1986.

[HIL99] Hastad, J., Impagliazzo, R., Levin, L., A., Luby, M., “A Pseudorandom Generator form any One-way Function”, *SIAM Journal on Computing*, Vol. 28, pp. 1364-1396, 1999.

[HM05] He, C., Mitchell, J., C., “Security Analysis and Improvements for IEEE 802.11i*”, The 12th Annual Network and Distributed System Security Symposium (NDSS'05)*, 2005, pp. 90-110.

[HS05] Hong, J., Sakar, P., “New Applications of Time Memory Data Tradeoffs”, Advances in Cryptology - ASIACRYPT 2005, Lecture Notes in Computer Science, Vol. 3788, ISBN 978-3-540-30684-9, Springer Berlin Heidelberg, 2005, pp. 353–372.

[HUS05] Huang, J., Seberry, J., Susilo, W., Bunder, M., “Security Analysis of Michael: The IEEE 802.11i Message Integrity Code”, *Lectures Notes in Computer Science*, Vol. 3823, Springer Berlin/Heidelberg, November 2005, pp. 423-432.

[HUL01] Hulton, D., “Practical exploitation of RC4 weaknesses in WEP environments”, 2001.
http://www.datastronghold.com/security-articles/hacking-articles/practical-exploitation-of-rc4-weaknesses-in-wep-environments.html

[HUT07] Hutter, M., “Algorithmic Information Theory: a brief non-technical guide to the field”, 2007.
http://arxiv.org/abs/cs/0703024.

[HUT09] Hutter, M., “Open Problems in Universal Induction & Intelligence”, *Algorithms* 2(3), 2009, pp.879-906.

[IC08] Iantovics, B., L., Crainicu, B., “Complex Mobile Multiagent Systems”, *First International Conference on Complexity and Intelligence of the Artificial and Natural Complex Systems. Medical Applications of the Complex Systems. Biomedical Computing. CANS 2008*, Published by IEEE Computer Society, IEEE Computer Society Order Number P3621, ISBN 978-0-7695-3621-7, Library of Congress Number 2009900732, 2008, pp. 21-30.

[IAC08] Iantovics, B., L., Crainicu, B., “Security in Mobile Multiagent Systems”, *Proceedings of the International Conference Complexity and Intelligence of the Artificial and Natural Complex Systems. Medical Applications of the Complex Systems. Biomedical Computing, CANS 2008*, 8-9 November, Târgu Mureş, 2008, Editura Universităţii “Petru Maior” Târgu-Mureş, 2008, ISSN 2065-0426, pp. 183-191.

[ICR08] Iantovics, B., L., Crainicu, B., “Security Measures in Complex Multiagent Systems Compose from Mobile Agents“, *Complexity in Artificial and Natural Systems*, Editura Universităţii “Petru Maior” Târgu Mureş, ISBN 978-973-7794-76-5, 2008, pp. 47-55.

[IM10] Iantovics, B., L., Marusteri, M., Kountchev, R., Zamfirescu, C., B., Crainicu, B., “Intelligent CMDS Medical Agents with lerning Capacity”, *Proceedings of the International Conference on Virtual learning. ICVL 2010*, October 29-October 31, 2010, Targu Mures, Romania, Bucharest University Press, 2010, ISSN 1844-8933, pp. 325-331.

[IC13] Iantovics, B., L., Crainicu, B, “Sisteme multiagent: o abordare modernă în inteligenţa artificială”, Editura Universităţii "Petru Maior, ISBN 978-606-581-099-0, 2013.

[IC14] Iantovics, B., L., Crainicu, B., “A Distributed Security Approach for Intelligent Mobile Multiagent Systems”, *Advanced Intelligent Computational Technologies and Decision Support Systems, Studies in Computational Intelligence*, Volume 486, Springer International Publishing, ISBN 978-90-481-3661-2 (Print), ISBN 978-90-481-3662-9 (Online), 2014, pp. 175-189.

[IEEE1] *IEEE Standard for Local and metropolitan area networks - Port-Based Network Access Control*, IEEE Std 802.1X-2004.

[IEEE2] *IEEE Standard for Information technology- Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 6: Medium Access Control (MAC) Security Enhancements*, IEEE Std 802.11i-2004

[ITO2014] Isobe, Takanori, Ohigashi, Toshihiro, "Security of RC4 Stream Cipher", Hiroshima University, 2014.

[JEN98] Jenkins, R., “Isaac and RC4”, 1998. http://burtleburtle.net/bob/rand/isaac.html.

[JOH99] Johansson, T., Jonsson, F., “Improved Fast Correlation Attacks on Stream Ciphers via Convolutional Codes”, *Advances in Cryptology — EUROCRYPT ’99*, Lecture Notes in Computer Science, Vol. 1592, 1999, pp. 347-362.

[KM12] Kamble, B., H., Meshram, B., B., Robustness of RC4 against Differential attack, International Journal of Advanced Research in Computer Engineering & Technology, Vol. 1, Issue 4, 2012, pp. 661-665.

[KO04] Kang, Y., S., Oh, K., Chung, B., Chung, K., Nyang, D., ”Analysis and Countermeasure on Vulnerability of WPA Key Exchange Mechanism”, *Lectures Notes in Computer Science*, Vol. 3090, Springer Berlin/Heidelberg, August 2004, pp. 915-924.

[KL10] Kastermans, B., Lempp, S., “Comparing notions of randomness”, *Theoretical Computer Science,* 411(3)*,* 2010*,* pp. 602-616.

[KLE43] Kleene, S., C., “Recursive predicates and quantifiers”, *Trans. Amer. Math. Soc.,* 53, 1943, pp. 41-73.

[KLA08] Klein, A., “Attacks on the RC4 stream cipher”, *Designs, Codes and Cryptography*, Vol. 48, No. 3, Springer-Verlag, 2008, pp. 269-286. http://cage.ugent.be/~klein/RC4/RC4-en.ps.

[KNU98] Knudsen, L., R., Meier, W., Preneel, B., Rijmen, V., Verdoolaege, S., “Analysis Methods for (Alleged) RC4”, in *Proc. International Conference on the Theory and Application of Cryptology and Information Security, ASIACRYPT’98*, Beijing, Lecture Notes in Computer Science, Springer-Verlag, Vol.1514, 1998, pp. 327–341.

[KNT97] Knuth, D., E., “*The Art of Computer Programming: Seminumerical Algorithm*”, Third edition, Volume 2, Addison-Wesley, 1997.

[KI06] Kobara, K., Imai, H., “Key-Dependent Weak IVs and Weak Keys in WEP – How to Trace Conditions Back to Their Patterns –“, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E89-A, No. 8, 2006, pp. 2198-2206.

[KI08] Kobara, K., Imai, H., “IVs to Skip for Immunizing WEP against FMS Attack”, *IEICE Transactions on Communications*, Vol.E91–B, No.1, 2008, pp. 218-227.

[KOL56] Kolmogorov, A. N. *Foundations of the theory of probability*. Translation, Edited by Nathan Morrison, with an added bibliography by A. T. Bharuch-Reid, Chelsea Publishing Co., New York, 1956.

[KOL63] Kolmogorov, A., N., “On the tables of random numbers”, *Sankhya Ser. A 25* 1963, pp. 369-376.

[KOL65] Kolmogorov, A., N., “Three approaches to the definition of the quantity of information”, *Problems of Information Transmission (Problemy Peredachi Informatsii)*, No. 1, 1965, pp. 3–11.

[KOL83] Kolmogorov, A., N., “On logical foundation of Probability Theory”, *Proceedings, 4th USSR-Japan Symposium on Probability Theory and Statistics*, Lecture Notes in Math., Vol. 1021, Springer-Verlag, New York/Berlin, 1983, pp. 1-5.

[KOR104] KoreK, *Need security pointers*, 2004.
http://www.netstumbler.org/showthread.php?postid=89036#post89036

[KOR204] KoreK, *Next generation of WEP attacks?*, 2004.
http://www.netstumbler.org/showpost.php?p=93942&postcount=35

[LEG97] Legg, S., “Solomonoff induction”, Technical Report 30, Centre for Discrete Mathematics and Theoretical Computer Science. University of Auckland, 1997.

[LEG06] Legg, S., “Is there an Elegant Universal Theory of Prediction?”, In *Proc. 17th International Conf. on Algorithmic Learning Theory (ALT’06),* Barcelona, 2006, pp. 274-287.

[LEV173] Levin, L., A., “Universal search problems”, *Problems of Information Transmission*, 9(3), 1973, pp. 265–266.

[LEV273] Levin, L., A., “On the notion of a random sequence”, *Soviet Math. Dokl.*, 14(5), 1973, pp.1413–1416.

[LEV74] Levin, L., A., “Laws of information conservation (non-growth) and aspects of the foundation of probability theory”, *Problems Information Transmission*, 10(3), 1974, pp. 206-210.

[LEV76] Levin, L., A., “Measures of complexity for finite objects (axiomatic description)”, *Sov.Math. Dokl.*, 17**,** 1976, pp. 552-526.

[LEV87] Levin, L.A., “One-way Function and Pseudorandom Generators”, *Combinatorica*, Vol. 7, No. 4, pp. 357-363, 1987

[LIV08] Li, M, Vitanyi, P., “An Introduction to Kolmogorov Complexity and Its Applications”, Third Edition, Springer Verlag, 2008.

[LOV66] Loveland, Z., *Math. Logik Grundl. Math.,* 12, 1966, pp.279-294.

[MG08] Maitra, S., Gouta Paul (2008-09-19), "Analysis of RC4 and Proposal of Additional Layers for Better Security Margin", *Progress in Cryptology – INDOCRYPT 2008*, Lecture Notes in Computer Science, Vol. 5365, Springer-Verlag Heidelberg, ISBN 3-540-89753-4, 2008 pp. 27-39.

[MAN101] Mantin, I., “The Security of the Stream Cipher RC4”, Master Thesis, The Weizmann Institute of Science, 2001.

[MAN201] Mantin, I., “Analysis of the Stream Cipher RC4”, Faculty of Mathematics and Computer Science, The Weizmann Institute of Science, 2001.

[MS02] Mantin, I., Shamir, A., “A practical attack on broadcast RC4”, in *Proc. 8th International Workshop, FSE 2001*, Yokohama, Lecture Notes in Computer Science, Springer-Verlag, Vol. 2355, 2002, pp. 87-104.

[MAN105] Mantin, I., “Predicting and Distinguishing Attacks on RC4 Keystream Generator”, in. *Proc. 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, EUROCRYPT 2005*, Aarhus, Lectures Notes in Computer Science, Vol. 3494, Springer-Verlag, 2005, pp. 491-506.

[MAN205] Mantin, I., “A Practical Attack on the Fixed RC4 in the WEP Mode”, in *Proc. 11th International Conference on the Theory and Application of Cryptology and Information Security, ASIACRYPT 2005*, Chennai, Lecture Notes in Computer Science, Springer-Verlag, Vol. 3788, 2005, pp. 395-411.

[MAR98] Marcus, S., “Imprecision, between variety and uniformity: the conjugate pairs”, *J.J. Jadacki, W. Strawinski, eds. In the World of Signs, Pozna n Studies in the Philosophy of the Sciences and the Humanities 62*, 1998, pp. 59–72.

[MLO66] Martin-Löf, P., “The definition of random sequences”, *Information and Control*, 9 (6), 1966, pp. 602–619.

[MLO68] Martin-Löf, P., “On the notion of randomness”, *Intuitionism and Proof Theory*, (Proc. Conf., Bu_alo, N.Y., 1968), pp. 73-78, North-Holland, Amsterdam, 1970 and Proof Theory, New York, 1968. pp. 73-78.

[MLO69] Martin-Löf, “The literature on von Mises’ kollectives revisited”, *Theoria*, 35, 1969, pp. 12-37.

[MCS061] Măruşteri, M., Crainicu, B., Şchiopu, A., *ROBIOCLUSTER – an open source platform for HPC (high performance computing)/Linux clusters in the biomedical field*, Integrating Biomedical Information: From eCell to ePatient, Proceedings of the European Federation for Medical Informatics, Special Topic Conference, April 6-8, 2006, Timişoara, Romania, Akademische Verlagsgesellschaft Aka GmbH, Berlin, ISBN 3-89838-072-6 (Aka), ISBN-10 1-58603-614-9 (IOS Press), ISBN-13: 978-1586036140 (IOS Press), ISBN 973-625-303-1 (Editura Politehnica), 2006, pp. 174-177.

[MCS062] Măruşteri, M., Ş., Crainicu, B., Şchiopu, A., “New trends in Open Source Educational Platforms – The ROSLIMS Linux Live CD Paradigm“, Proceedings of the 5th RoEduNet IEEE International Conference: Sibiu, 1-3 June 2006, Romania, Editura Universităţii “Lucian Blaga” din Sibiu, 2006, ISBN (10) 973-739-277-9, (13) 978-973-739-277-0, pp. 82-86. *Acta Universitatis Cibiniensis*, Vol. LV, Technical Series, Editura Universităţii “Lucian Blaga” din Sibiu, 2007, ISSN 1583-7149, pp. 136-140.

[MCS063] Măruşteri, M., Crainicu, B., Şchiopu, A., *ROSLIMS Linux Live CD – all-in-one cross platform solution for running biomedical software*, Integrating Biomedical Information: From eCell to ePatient, Proceedings of the European Federation for Medical Informatics, Special Topic Conference, April 6-8, 2006, Timişoara, Romania, Akademische Verlagsgesellschaft Aka GmbH, Berlin, ISBN 3-89838-072-6 (Aka), ISBN-10 1-58603-614-9 (IOS Press), ISBN-13: 978-1586036140 (IOS Press), ISBN 973-625-303-1 (Editura Politehnica), 2006, pp. 178-181.

[MEO96] Menezes, A., J., van Oorschot, P., C., “Handbook of Applied Cryptography (Discrete Mathematics and Its Applications)”, CRC Press, ISBN 10: 0849385237, 1996.

[MRK08] Merkle, W., “ The complexity of stochastic sequences”, *Journal of Computer and System Sciences,* 74, 2008, pp. 350-357.

[MIR02] Mironov, I., “(Not So) Random Shuffles of RC4”, in *Proc. 22nd Annual International Cryptology Conference, Advances in Cryptology, CRYPTO 2002*, Santa Barbara, Lecture Notes in Computer Science, Springer-Verlag, Vol. 2442, 2002, pp. 304–319.

[MT99] Mister S., Tavares, S., E., “Cryptanalysis of RC4-like Ciphers”, in *Proc. 5th Annual International Workshop, SAC 1998*, Kingston, Lecture Notes in Computer Science, Springer-Verlag, Vol.1556, 1999, pp. 131–143.

[MRH04] Moen, V., Raddum, H., Hole, K., J., “Weaknesses in the Temporal Key Hash of WPA”, *Mobile Computing and Communications Review*, Vol. 8, No. 2, April 2004, Papers from MC^{2}R Open Call, pp. 76-83.

[MSU98] Muchnik, A., A., Semenov, A., L., Uspensky, V., A., “Mathematical Metaphysics of Randomness”, *Theor. Comput. Sci.* 207(2), 1998, pp.263-317.

[OSM05] Ohigashi, T., Shiraishi, Y., Morii, M., “Most IVs of FMS Attack-Resistant WEP Implementation Leak Secret Key Information”, in *Proc. 2005 Symposium on Cryptography and Information Security*, Maiko, Vol. 4, 2005, pp. 1957–1962.

[OS05] Ohigashi, T., Shiraishi, Y., Morii, M., “FMS Attack-Resistant WEP Implementation Is Still Broken – Most IVs Leak a Part of Key Information – “, in *Proc. International Conference, CIS 2005*, Xi’an, Lecture Notes in Computer Science, Springer-Verlag, Vol. 3802, 2005, pp. 17-26.

[OSM08] Ohigashi, T., Shiraishi, Y., Morii, M., “New Weakness in the Key-Scheduling Algorithm of RC4”, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E91-A, No. 1, 2008, pp. 3-11.

[OWF] **One-Way Function.* http://mathworld.wolfram.com/One-WayFunction.html.

[OP13] Orumiehchiha, M., A., Pieprzyk, J., Shakour, E., Steinfeld, R., “Cryptanalysis of RC4(n,m) Stream Cipher”, in *Proc. 6th International Conference on Security of Information and Networks*, *SIN ’13*, Aksaray, ACM New York, NY, USA, 2013, pp. 165-172.

[OUG05] Ou, G., “Wireless LAN security guide, Security for any organization large or small”, January 2005, http://www.lanarchitect.net/ Articles/ Wireless/SecurityRating/

[PP10] Paar, C., Pelzl, J., ”Understanding Cryptography”, Springer-Verlag Berlin Heidelberg, ISBN 978-3-642-04101-3, 2010.

[PAP03] Paul S., Preneel, B., “Analysis of Non-fortuitous Predictive States of the RC4 Keystream Generator”, in *Proc. 4th International Conference on Cryptology in India*, *INDOCRYPT 2003*, New Delhi, Lecture Notes in Computer Science, Springer-Verlag, Vol. 2904, 2002, pp. 52-67.

[PAP04] Paul S., Preneel, B., “A New Weakness in the RC4 Keystream Generator and an Approach to Improve the Security of the Cipher”, in *Proc. 11th International Workshop, FSE 2004*, Delhi, Lecture Notes in Computer Science, Springer-Verlag, Vol. 3017, 2004, pp. 245–259.

[PMA07] Paul, G., Maitra, S., “RC4 State Information at Any Stage Reveals the Secret Key”, *14th Annual Workshop On Selected Areas in Cryptography, SAC 2007*, Ottawa, Canada, 2007, Lecture Notes in Computer Science, Vol. 4876, Springer Berlin Heidelberg, ISBN 978-3-540-77359-7, 2007, pp. 360-377.

[PRM08] Paul, G., Rathi, S., Maitra, S., “On non-negligible bias of the first output bytes of RC4 towards the first three bytes of the secret key”, *Designs, Codes and Cryptography*, Vol. 49, No. 1-3, Springer-Verlag, 2008, pp. 123-134.

[PEI10] Peikert, C., “Indistinguishability, Pseudorandomness“, *Theoretical Foundations of Cryptography*, Georgia Tech, Spring 2010.

[STC] **Private-key cryptography. Stream Ciphers, Computational Security*, October 5, 2010. http://www.deic.uab.es/material/20375-4-stream-pf.pdf.

[PSG] **Pseudo Random Generators*, School of Computer Science and Communication, CSC, Stockholm University. http://www.nada.kth.se/kurser/kth/2D1441/semteo03/lecturenotes/prg.pdf.

[PRF] **Pseudo Random Functions,* School of Computer Science and Communication, CSC, Stockholm University. http://www.nada.kth.se/kurser/kth/2D1441/semteo03/lecturenotes/prf.pdf.

[RIJ14] Rijmenants, D., “One-time Pad”, *Cipher Machines & Cryptology*, 2014. http://users.telenet.be/d.rijmenants/en/onetimepad.htm.

[RIV92] Rivest, R., L., “The RC4 Encryption Algorithm”, *RSA Data Security, Inc*., 1992. (Proprietary).

[RIV01] Rivest, R., L., “RSA security response to weaknesses in key scheduling algorithm of RC4”, Tech Notes, RSA Laboratories, 2001. http://www.rsasecurity.com/rsalabs/node.asp?id=2009

[RS14] Rivest, R., L., Schuldt, J, C., N., “Spritz – a spongy RC4-like stream cipher and hash function”, MIT Talk. 2014. http://people.csail.mit.edu/rivest/pubs/RS14.pdf.

[ROB81] Robbins D., Bolker, E., “The bias of three pseudo-random shuffles”, *Aequationes Mathematicae*, Vol. 22, 1981, pp. 268-292.

[ROO95] Roos, A., “Class of weak keys in the RC4 stream cipher”, Two posts in sci.crypt, message-id 43u1eh$1j3@hermes.is.co.za and 44ebge$llf@hermes.is.co.za, 1995.

[RUE86] Rueppel, R., A., “Analysis and Design of Stream Ciphers”, Communications and Control Engineering Series, , ISBN: 978-3-642-82867-6, Springer Berlin Heidelberg, 1986.

[SS92] Schmidt, F., Simion, R., “Card shuffling and a transformation on S_n”, *Aequationes Mathematicae*, Vol. 44, 1992, pp. 11-34.

[SC171] Schnorr, C., P., “Zufalligkeit und Wahrscheinlichkeit. Eine algorithmische Begrundung der Wahrscheinlichkeitstheorie”, *Lecture Notes in Mathematics*, Vol. 218, Springer-Verlag, Berlin, 1971.

[SC271] Schnorr, C. P., “A unified approach to the definition of a random sequence”, *Mathematical Systems Theory*, Vol. 5, 1971, pp. 246–258.

[SC73] Schnorr, C., P., “Process complexity and effective random tests”, *Journal of Computer and System Sciences*, 7(4), 1973, pp. 376–388.

[SEC74] Seche, L., “Lexicul artistic eminescian în lumină statistică”, Editura Academiei RSR, Bucureşti, 1974.

[SVV11] Sepehrdad, P., Vaudenay, S., Vuagnoux, M., "Discovery and Exploitation of New Biases in RC4", *Lecture Notes in Computer Science*, Vol. 6544, Springer Heidelberg, 2011, pp. 74–91

[SHA48] Shannon, C.E., "A Mathematical Theory of Communication", *Bell Syst. Tech. J.*, 27, 1948, pp.379-423, pp.623-656.

[SOM03] Shiraishi, Y., Ohigashi, T., Morii, M., “An improved Internal-State Reconstruction Method of a Stream Cipher RC4”, in *Proc. IASTED International Conference on Communication, Network, and Information Security, CNIS 2003*, New York, 2003, pp. 132-135.

[SHI10] Shirayev, A., N., “On the evolution of the von Mises’ notion of Randomness”, Mathematical Institute, University of Freiburg, 2010.

http://wochenprogramm.mathematik.uni-freiburg.de/kabstract.en.html?LfdNr=8&Semester=WS2009-2010

[SIE84] Siegenthaler, T., “Correlation-Immunity of Nonlinear Combining Functions for Cryptographic Applications”, *IEEE Transactions on Information Theory*, Vol. 30 (5), 1984, pp. 776–780.

[SIE85] Siegenthaler, T., “Decrypting a class of stream ciphers using ciphertext only”, *IEEE Transactions on Computers*, Vol. C-34, 1985, pp. 81-85,

[SR11] Singhal, N., Raina, J., P., S., "Comparative Analysis of AES and RC4 Algorithms for Better Utilization", *International Journal of Computer Trends and Technology (IJCTT)*, ,Vol. 1, Issue 3, 2011, pp. 259-263.

[STW03] Shunman W., Tao, R., Wang, Y., Zang, J., "WLAN and it's security problems”, in *Proceedings of the Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2003, pp 241-244.

[SOL60] Solomonoff, R., J., “A Preliminary Report on a General Theory of Inductive Inference”, (Revision of Report V–131), *Contract AF* 49(639)–376, Report ZTB–138, Zator Co., Cambridge, Mass., Nov. 1960.

[SOL62] Solomonoff, R., "Training Sequences for Mechanized Induction", *Self- Organizing Systems*, M. Yovits, ed., 1962, pp. 425-434.

[SOL164] Solomonoff, R., “A formal theory of inductive inference”, *Information and Control*, Part I, Vol. 7, No. 1, 1964, pp.1-22.

[SOL264] Solomonoff, R., “A formal theory of inductive inference*”, Information and Control*, Part II, Vol. 7, No. 2, 1964, pp.224-254.

[SOL78] Solomonoff, R., J., “Complexity-Based Induction Systems: Comparisons and Convergence Theorems”, *IEEE Trans. on Information Theory*, 24:4, 1978, pp.422–432.

[SOL97] Solomonoff, R., "The discovery of algorithmic probability", *Journal of Computer and System Sciences*, Vol. 55, No. 1, 1997, pp. 73-88.

[SON08] Song, D., Computer Security, Notes 4, CS 161, Fall 2008. http://inst.eecs.berkeley.edu/~cs161/fa08/Notes/random.pdf.

[STU01] Stubblefield, A., Ioannidis, J., Rubin, A., "Using the Fluhrer, Mantin, and Shamir attack to Break WEP", Technical Report TD-4ZCPZZ, AT&T Labs, 2001.

[SIR04] Stubblefield, A., Ioannidis, J., Rubin, A., "A key recovery attack on the 802.11b wired equivalent privacy protocol (WEP)", *ACM Transactions on Information and System Security (TISSEC)*, Vol. 7, No. 2, 2004, pp. 319–332.

[TW08] Tews, E., Weinmann, R., P., Pyshkin, A., "Breaking 104 bit WEP in less than 60 seconds", in *Proc. 8th International Workshop, WISA 2007*, Jeju Island, Lecture Notes in Computer Science, Vol. 4867, Springer-Verlag, 2008, pp. 188-202. http://eprint.iacr.org/2007/120.pdf.

[THO12] Thomsen, M., "Linear Feedback Shift Registers, Galois Fields, and Stream Ciphers", *Cryptography II*, 2012. http://www.cs.rit.edu/~mxt4877/.

[TBN07] Tomašević, V., Bojanić, S., Nieto-Taladriz, O., "Finding an internal state of RC4 stream cipher", *Information Sciences*, Vol. 177, Elsevier, 2007, pp. 1715-1727.

[TUR36] Turing, A., "On computable numbers, with an application to the Entscheidungsproblem", *Proceedings of the London Mathematical Society*, ser. 2. vol. 42 (1936-7), pp.230-265.

[USP06] Uspensky, V. A., "Four algorithmic physiognomies of randomness", *Matematicheskoe prosveshchenie, 10, MCCME,* Moscow, 2006, pp. 71–108.

[LA187] van Lambalgen, M., "Random Sequences", *PhD Thesis*, Department of Mathematics, University of Amsteram, Amsterdam, 1987.

[LA287] van Lambalgen, M., "Von Mises' Definition of Random Sequences Reconsidered", *J. Symb. Log.,* 52(3), 1987, pp.725-755.

[VV07] Vaudenay, S., Vuagnoux, M.,"Passive-only Key Recovery Attacks on RC4", in *Proc. 14th International Workshop, SAC 2007*, Ottawa, Lecture Notes in Computer Science, Vol. 4876, Springer-Verlag, 2007, pp. 344-359. http://infoscience.epfl.ch/record/115086/files/VV07.pdf.

[VER09] Vereshchagin, N., "Kolmogorov Complexity and Model Selection", *Computer Science - Theory and Applications*, Fourth International Computer Science Symposium in Russia, CSR 2009, Novosibirsk, Russia, 2009, LNCS 5675, Springer-Verlag, pp. 19-24.

[VRM19] Vernam, G., S., Secret signaling system, US 1310719 A (US patent), 1919.

[VIL36] Ville, J., "Sur la notion de collectif", *Comptes rendus* 203, 1936, pp.26–27.

[VIL39] Ville, J., "Etude critique de la notion de collectif", *Gauthier-Villars*, Paris, 1939.

[VIT01]Vitányi, P., "Randomness", *CoRR math.*PR/0110086, 2001. http://arxiv.org/abs/math/0110086

[VLE10] Vlek, C., “Definability in the degrees of randomness”, *MSc Dissertation*, University of Amsterdam, 2010.

[VOL02] Volchan, S., B., “*What Is a Random Sequence?*“, *The American Mathematical Monthly*, Vol. 109, 2002, pp. 46–63.

[MIS19] von Mises, R., “Grundlagen der Wahrscheinlichkeitsrechung”, *Math. Z.*, vol. 5, 1919, pp. 52-99.

[MIS57] von Mises, R., “Probability, statistics and truth”, 2nd English edition, George Allen and Unwin, London, 1957.

[VS10] Vovk, V., Shen, A., “Prequential randomness and probability”**,** *Theoretical Computer Science*, Vol. 411, No. 29-30, 2010, pp. 2632-2646.

[VYU98] V'yugin, V., V., “Non-stochastic infinite and finite sequences”, *Theoretical Computer Science*, 207(2), 1998, pp.363-382.

[WAG95] Wagner, D., “My RC4 weak keys”, Post in sci.crypt, message-id 447o1l$cbj@cnn.princeton.edu, 1995.

http://www.cs.berkeley.edu/~daw/my-posts/my-rc4-weak-keys.

[WAL36] Wald, A., "Sur la notion de collectif dans le calcul des probabilités", *Comptes Rendus des Séances de l'Académie des Sciences*, 202, 1936, pp. 1080-1083.

[WAL37] Wald, A., “Die Widerspruchsfreiheit des Kollektivbegriffes der Wahrscheinlichkeitsrechnung”, *Ergebnisse eines Math. Kolloquiums*, Vol. 8, 1937, pp. 38-72.

[WLK00] Walker, J., “Unsafe at any key size: an analysis of the WEP encapsulation,” *Tech. Rep. 03628E, IEEE 802.11 committee*, March 2000, http://grouper.ieee.org/groups/802/11/Documents/ DocumentHolder/0-362.zip.

[WIL70] Willis, D., G., “Computational Complexity and Probability Constructions," *Journal of the Assoc. of Comp. Mach.,* 1970, pp. 241-259.

[WOOL] Wool, A., “Lightweight key management for IEEE 802.11Wireless LAN’s with key refresh and host revocation”, *IEEE 802.11 TGi working group*, http://grouper.ieee.org/groups/802/11/Documents/ DocumentHolder/2-411.zip.

[WO04] Wool, A., “A Note on the Fragility of the “Michael” Message Integrity Code”, *IEEE Transactions on Wireless Communications*, Vol. 3, No. 5, September 2004, pp 1459-1462.

[WOO81] Wootters, W., K., “Statistical distance and Hilbert space”, *Physical Review D 23*, 1981, pp. 357-362.

[YAO82] Yao, A., C., “Theory and Applications of Trapdoor Functions”, *Proc. of the 23rd IEEE, Symp. on Foundation of Computer Science (FOCS)*, 1982, pp. 80-91.

[YAG06] Yager, R., R., “OWA trees and their role in security modeling using attack trees”, *Information Science*, Vol. 176, No. 20, 2006, pp. 2933–2959

[ZAW] Zawada, K., “Kolmogorov Complexity”, *University of Illinois at Chicago*. http://www.ece.uic.edu/~devroye/courses/ECE534/project/project_Krzysztof_Zawada.pdf.

[ZEN04] Zenner, E., On the Role of the Inner State Size in Stream Ciphers, Reihe Informatik, 01-2004. http://www.erikzenner.name/docs/2004_state_wosis.pdf.

[ZOL04] Zoltak, B., “VMPC One-Way Function and Stream Cipher”, in *Proc. 11th International Workshop, FSE 2004*, Delhi, Lectures Notes in Computer Science, Vol. 3017, Springer-Verlag, 2004, pp. 210–225.

[ZVO70] Zvonkin, A., K., Levin, L., A., “The Complexity of Finite Objects and the Development of the Concepts of Information and Randomness by Means of the Theory of Algorithms”, *Russian Mathematical Surveys*, Vol. 25, No. 66, 1970, pp. 83-124.