Dinamica apăsării tastelor

Laurențiu - Iulian Iordache - Stoicescu

***Abstract*— În ziua de astăzi, calculatoarele sunt folosite peste tot pentru a stoca și procesa o gamă largă de date. De asemenea, numărul atacurilor cibernetice a crescut și el. Pentru a putea proteja aceste sisteme de intruși, utilizarea unui sistem de securitate adecvat reprezintă o prioritate. În prezent sunt utilizate cu succes mai multe metode de securitate bazate pe măsuri biometrice, cum ar fi analiza amprentelor sau recunoașterea facială. Această cercetare se va axa însă pe o metodă prea puțin folită și anume, dinamica apăsării tastelor, cunoscută în literatură drept „Keystroke dynamics".**

***Index Terms*— SAAS - Software as a service, PP - Press to press, RR - Release to release, RP - Release to press, wpm - words per minute, RN - Rețele neurale, BP - Backpropagation, FP - funcție de potențial, RBFN - Radial Basis Function Network, SVM - Suport Vector Machine, ADP - Arbori de decizie paraleli, GMM - Gaussian mixture models, PCA - Principal Component Analysis**

I. Introducere

A. Necesitate

Calculatoarele au devenit omniprezente în societatea modernă, conform statisticilor oferite de Internet World Stats [1], numărul utilizatorilor unici de internet calculat la sfârșitul lunii Iunie 2018 este de aproximativ 4.2 miliarde. Pe cât este de mare numărul de utilizatori ce au acces la internet, pe atât de mare este numărul de atacuri cibernetice ce îi vizează. Conform datelor oferite de ..., aproximativ 63% din toate intruziunile în rețele și furturile de informații se datorează compromiterii datelor de autentificare. Un atac cibernetic cunoscut care a constat în furtul datelor personale ale aproximativ 500 de milioane de conturi este cel ce a vizat site-ul „yahoo.com".

Din moment ce depindem din ce în ce mai mult de calculatoare, iar riscurile folosirii acestora cresc de la o zi la alta, este normal ca și nivelul de securitate să fie sporit pentru a putea face față atacurilor. Utilizarea de metrici biometrice în procesuld de autentificare este unul dintre pașii făcuți pentru sporirea securității în ceea ce privește autentificare unui utilizator. În acest caz se merge pe ideea că un atacator poate fura identitatea digitală a unui utilizator (utilizator, parola, token etc.) dar nu poate fura sau replica ceea ce este utilizatorul.

În prezent sunt implementate cu succes mai multe metode de autentificare pe baza de metrici biometrice cum ar fi recunoașterea utilizatorului pe bază de amprentă papilară, recunoașterea facială sau a irisului. Aceste metode însă necesită componente hardware suplimentare pentru achiziția datelor biometrice. O altă metodă de identificare a unui utilziator, mai putin populară, o reprezintă analzia dinamicii apăsării tasterlor. Această metodă are avantajul că nu necesită componente hardware adiționale, deoarece orice calculator are o tastatură. Pe lângă acesta mai are un avantaj semnificativ și anume că poate realiza achiziția metricilor în timp ce utilizatorul își îndeplinește sarcinile uzuale fără a-l deranja și fără a se face sesizată achiziția de date, aceasta fiind o metodă neintruzivă.

B. Generalități biometrie

Cuvântul biometrie provine din cuvintele grecești „bios" (viață) și „metrikos" (măsurătoare). Aceasta reprezintă știința care se ocupă cu analiza statistică a caracteristicilor biometrice și este folosită cu succes în aplicații de securitate pentru verificarea sau identificarea persoanelor.

Trăsăturile biometrice pot fi împărțite în două categorii principale - fizice (fiziologice), aceastea reprezintă măsurile extrase de la părți ale corpului uman ex: amprenta, irisul, fața etc. și comportamentale, acestea reprezintă măsurile extrase din acțiuni realizate de utilizator, ex: semnătura, dinamica de apăsare a tastelor, vocea(amprenta vocală).

C. Dinamica apăsării tastelor

Măsura biometrică a tastării este referită în literatură sub forma de dinamica apăsării tastelor („Keystroke dynamics"). Dinamica apăsării tastelor se referă la felul în care o persoană apasă tastele unei tastaturi. Această metodă este bazată pe caracteristicile de scriere ale persoanelor cum ar fi durata apăsării unei taste, latența dintre apăsări consecutive ale tastelor, timpul dintre două apăsări consecutive și în cazul în care se poate, forța apăsării. Cele mai folositoare metrici sunt: timpul de apăsare, acesta reprezintă durata de timp a menținerii unei taste apăsate și timpul de pauză, care reprezintă durata de timp dintre eliberarea unei chei și apăsarea alteia.

1) Scurt istoric: Această metodă este derivată din ideea de indentificare a expeditorului unui cod Morse ce folosește un telegraf. Această tehnică a fost analizată în timpul celui de-al doilea război mondial și poartă numele de „fist of the sender" (pumnul expeditorului). Aceasta a fost utilizată cu succes pentru a monitoriza deplasarea trupelor pe baza recunoașterii tiparului de transmisie a expeditorului mesajului [2].

2) Mecanismul psihologic: Avantajul utilizării de măsuri biometrice comportamentale cum ar fi dinamica apăsării tastelor îl reprezintă faptul că acestea pot fi colecate fără ca utilizatorul în cauză să își dea seama.

Experimentele psihologice efectuate în ultimul secol au demostrat că sarcinile repetitive cum ar fi vorbitul, scrisul, tastarea, cântatul la pian etc. sunt controlate de un set de acțiuni. Aceste acțiuni pot fi prezise folosind un model care descrie seria de pași efectuați pentru a realiza o sarcină [3]. Sistemul motor planifică și controlează mișcarea pe baza informațiilor primite ca stimuli. Acesta poate fi privit ca pe un caz special de sistem auto-organizat [**?**]. Pe parcursul secolului 20, s-au efectuat studii pentru înțelegerea fiziologiei și psihologiei deprinderilor sistemului motor. Studiile s-au axat asupra transmiterilor telegrafice. Acestea s-au efectuat pe 32 de subiecți cu nivel variat al deprinderii de telegrafiere. S-a observat că operatorii erau capabili să îi recunoască pe ceilalți operatori cu care au mai lucrat doar prin ascultarea caracteristicii de tastare, de asemenea, mulți dintre aceștia au susținut că erau capabili să determine și sexul operatorului [3].

În figura 1 este prezentat modelul propus de W. E. Cooper, acesta reprezintă primul model general de tranzit al informației primite de un dactilograf, fiind împărțit pe etape pornind de la citirea textului până la transpunerea acestuia prin intermediul tastaturii. Prima etapă o reprezintă recunoașterea caracterelor. În această etapă s-a observat că dactilograful tinde să analizeze în avans materialul pe care îl citește.

A doua etapă constă în analizarea informațiilor citite. În această etapă, informațiile citite sunt stocate în memorie pentru o perioadă

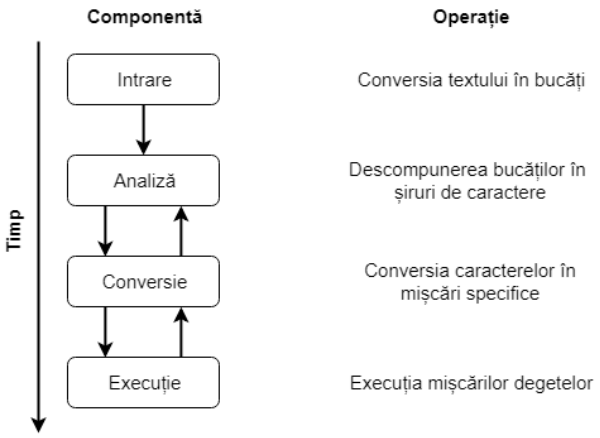

Fig. 1: Modelul general de tranzit al informației [5].

scurtă înainte de a fi scrise la tastatură. Conversia textului citit în acțiunea de tastare constă într-o combinație de procesări seriale și paralele ale informației. S-a observat că dactilografii împart textul în grupuri mici și predictibile datorită limitării memoriei [6].

A treia etapă o reprezintă translația caracterelor discrete în comenzi. Aceasta cuprinde acțiunea musculară efectuată pentru a executa mișcarea corectă a mâinii și a degetelor. S-a observat că succesiunea acțiunilor de mișcare a dactilografilor supuși experimentului este organizată înainte ca aceasta să fie executată și că aceasta este puternic influențată de starea actuală a subiectului.

A patra etapă constă în execuția efectivă a tastării. S-a observat și analizat semi-autonomia mișcării degetelor, aceasta fiind o acțiune care odata inițiată, nu mai poate fi oprită, ritmul și caracteristicile acesteia neputând fi modificate voit.

II. Achiziția datelor

Pentru achiziția datelor se pot utiliza mai multe dispozitive cum ar fi: tastatura clasică utilizată pentru scriere, un tip special de tastatură care poate înregistra presiunea apăsării, sau utilizarea unui ecran cu touchscreen, în cazul dispozitivelor mobile. Pe parcursul procesului de înrolare în sistem vor fi salvate diverse măsuri de timp precum și fraza unică tastată de utilizator (parola). După înrolarea acestuia, utilizatorul se va putea autentifica folosind parola și alți identificatori. Această informație va fi comparată de sistem cu parola existentă ca în cazul unui sistem clasic de autentificare dar se va verifica să coincidă și modul în care acesta a scris parola.

A. Introducerea textului

Analiza dinamicii apăsării tastelor poate fi clasificată pe larg în două tipuri în funcție de tipul textului introdus - analiză a textului static sau structurat și analiza textului liber.

Analiza textului static implică analiza comportamentului de tastare a unui individ pentru fraze predeterminate într-un interval de timp bine definit. De exemplu, se poate considera analiză statică în momentul autentificării unui utilizator în sistem pe baza analizei dinamicii tastării id-ului și a parolei acestuia. Dar se mai poate efectua și analiza unei fraze particulare care este comună pentru fiecare utilizator al sistemului. Utilizarea analizei textului static se folosește în mod normal în sistemele în care odată autentificat, nu mai sunt necesare introduceri suplimentare de date de către utilizator. Un exemplu de astfel de sistem îl reprezintă sistemul de autentificare într-un cont bancar. După autentificare, utilizatorul poate să citească datele tranzațiilor, să transfere sume etc.

Analiza textului liber sau dinamic implică o analiză continuă periodică a comportamentului de tastare. Analiza are loc inițial la autentificarea în sistem și mai are loc pe parcursul utilizării acestuia. De exemplu, se poate utiliza un sistem de analiză a textului liber pentru a garanta dacă un anumit cont sau o anumită licență sunt folosite de mai mult de o persoană. Această metodă având mai multă aplicabilitate în cazul aplicațiilor de tip serviciu sub formă de software, SAAS („Software as a service”). Datorită naturii intruzive a acestei metode pot apărea probleme de intimitate. Pentru a le evita, în literatură a fost propusă utilizarea combinațiilor de patru taste consecutive stocate în matrici în loc de logare integrală a informației tastate în ordine cronologică.

B. Mediul de achiziție

Mediul de achiziție joacă un rol important în determinarea caracteristicilor de scriere alea unui subiect. Mediul poate fi clasificat pe larg în două tipuri - mediu controlat și modiu necontrolat.

S-au efectuat mai multe experimente pentru analiza comportamentului de scriere al utilizatorilor, pentru acestea s-a utilizat același sistem pentru fiecare utilizator într-un mediu cu o luminozitate și temperatură controlată. Un astfel de mediu este cunoscut ca mediu controlat. De asemenea pentru a se asigura că toți utilizatorii sunt acomodați cu tipul de tastatură utilizat, au fost lăsați să exerseze pe aceasta. Datele astfel colecate pot să nu fie relevante pentru condițiile actuale în care un utilizator scrie.

Un mediu necontrolat poate fi definit ca un mediu al căror caracteristici sunt cunoscute parțial sau deloc. În astfel de medii, subiecților li s-a cerut esă realizeze achiziția datelor fie pe calculatoarele personale sau să completeze un formular online. Este preferată analiza și testarea sistemelor în medii necontrolate pentru perfecționare, chiar dacă aceste date sunt mai greu de analizat datorită numărului mare de variabile ce pot interveni. Astfel se poate obține un sistem robust care să fie folosit de utilizatorul normal în propriul mediu.

C. Caracteristici

În timpul scrierii, calculatorul poate înregistra tasta apăsată și timpul la care aceasta a fost apăsată, de asemenea, mai poate înregistra și timpul la care aceasta a fost eliberată și perioada de timp cât a fost menținută apăsarea, aceasta reprezentând diferența dintre primele două. În figura 2 sunt prezentate aceste informații temporale. Toate aceste informații pot fi stocate în timp ce un utilizator tastează.

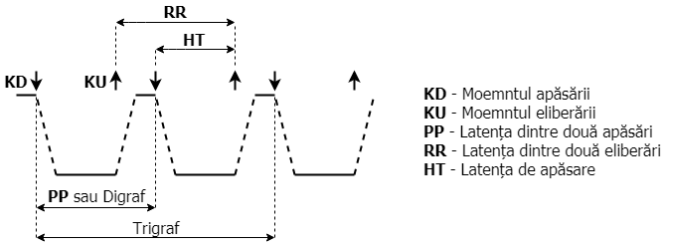

Fig. 2: Caracteristici extrase din dinamica apăsării tastelor [3]

Latența reprezintă una dintre cele mai folosite caracteristici. Există trei astfel de latențe definite [9] - latența dintre două apăsări consecutive, PP („press-to-press”), latența dintre două eliberări consecutive RR („release-to-release”), latența dintre o eliberare și o apăsare consecutivă RP („release-to-press”). Acestea trei mai sunt referite în literatură și sub alte nume, de exemplu, timpul dintre două apăsări consecutive mai este întâlnit sub forma de digraf, perioada de timp dintre eliberarea unei taste și apăsarea alteia se mai numește și timp de zbor (FT) („flight-time”). S-a observat că este mai importantă latența

de apăsare decât decât latența dintre apăsarea a două taste succesive [3]. De asemenea, latența dintre eliberarea unei taste și apăsarea alteia este de asemenea importantă deoarece aceasta este adesea dependentă de distanța dintre taste.

Pe baza acestor latențe se pot compune alte caracteristici, cum ar fi trigraful care reprezintă durata de timp necesară apăsării a trei taste consecutive, N-graful care reprezintă durata de timp necesară apăsării a N taste consecutive. S-a observat că utilizarea trigrafurilor oferă rezultate mai bune de clasificare decât în cazul utilizarii digrafurilor sau a N-grafurilor.

Pe lângă caracteristicile de timp, mai pot fi extrase și caracteristici de presiune. Acestea sunt extrase cu ajutorul unor tastaturi speciale ce permit măsurarea presiunii exercitate de utilizator la apăsarea unei taste. Aceste metrici au fost utilizate în încercarea de analizare a emoțiilor utilizatorului.

Alte caracteristici secundare care pot fi derivate din caracteristicile de timp sunt viteza de scriere, aceasta este măsurată în cuvinte pe minut (wpm), viteza maximă și cea minimă, media și deviația standard a caracteristicelor și entropia acestora.

D. Metrici de performanță ale sistemului

În procesul de autentificare/verificare, se extrag datele din șirul de caractere introdus de utilizator pentru a crea un model al modului de tastare. Acest model este apoi comparat cu unul existent pentru utilizatorul respectiv, creat în momentul înrolării în sistem. Pentru comparare se folosește un algoritm care determină cât este de similar modelul nou creat cu cel existent în baza de date. Iar pentru procesul de identificare, compararea realizată este un proces de tip one-to-many, acesta constă în compararea modelului nou creat cu toate modelele din baza de date pentru a realiza identificarea utilizatorul.

Pentru determinarea preciziei unui sistem biometric de autentificare se utilizează două rate de eroare importante - Rata de respingere falsă ,FRR („False Rejection Rate”) și rata de acceptare falsă, Far („False Acceptance Rate”).

FRR reprezintă probabilitatea ca un sistem biometric să nu recunoască identitatea unui utilizator autorizat. Se definește ca fiind raportul dintre numărul de respingeri false și numărul total de încercări legitime de identificare a unui utilizator autentificat [7].

$$FRR = \frac{f_n}{t_p + f_n} \tag{1}$$

FAR reprezintă probabilitatea ca un sistem biometric să confirme în mod eronat identitatea unui utilizator neautorizat. Se definește ca fiind raportul dintre numărul de acceptări false și numărul total de încercări de identificare a unui impostor [7].

$$FAR = \frac{f_p}{t_n + f_p} \tag{2}$$

Unde: f_n - numărul de respingeri false; f_p - numărul de acceptări false; t_n - numărul de respingeri corecte; t_p - numărul de acceptări corecte.

Se mai utilizează și măsura EER „Equal Error Rate” care reprezintă punctul de intersecție al curbelor trasate pentru FAR și FRR. Cu alte cuvinte $ERR = FAR = FRR$. Aceasta este utilizată pe post de indicator al preciziei sistemului biometric.

Uzual, pentru sisteme biometrice în care securitatea nu este o prioritate, se permit valori mai mari ale FAR-ului față de cele ale FRR-ului însă în aplicații cu un nivel ridicat de securitate se preferă valori ridicate ale FRR-ului dar foarte scăzute lare FAR-ului. Cu cât este mai scăzută valoarea ERR-ului, cu atât sistemul biometric este mai performant.

E. Baze de date

Lipsa de standardizare a achiziției datelor reprezintă pentru acest tip de măsură biometică un impediment în dezvoltare. Adoptarea de standarde ar trebui să faciliteze schimbul de informații între cercetători și să ofere o mai bună metodă de comparare a diferiților algoritmi. Astfel se va reduce cu siguranță duplicarea efortului depus. În tabelul 1 sunt listate o parte din bazele de date făcute disponibile de anumiți cercetători.

Bază de date	TT	S	M
Jugurta and Freire [8]	S	32	320
Jugurta and Freire [8]	D	15	150
Killourhy and Maxion [9]	S	51	20400
Giot et al. [10]	S	133	7555
Allen [11]	S	104	2379
Bello et al. [12]	S	54	282020
CMU [13]	S	51	20400
CMU Free vs. Transcribed Text [14]	D/S	20	-
BeiHang [15]	S	117	2661
Stonybrook [16]	D	196	-
Keystroke100 Dataset [17], [18]	S	100	2000

TABLE I: Baze de date făcute publice de anumite colective de cercetători.
TT - tipul de text; S - Subiecți; M - mostre;
S - static; D - dinamic;

III. Abordări și rezultate

Ulterior extragerii caracteristicilor sunt realizate modele capabile să diferențieze utilizatorii pe baza acestora. Clasificarea utilizatorilor este realizată pe baza asemănărilor și deosebirilor dintre aceste modele. În literatură s-au utilizat de la metode simple derivate din statistică cum ar fi media și deviația standard provenite din statisica până la metode complexe de recunoaștere a acestora pentru clasificarea scriitorului. Algoritmii de clasificare pot fi împărțiți în trei categorii [3].

A. Algoritmi statistici

Cea mai simplă metodă statistică constă în calculul mediei și deviației standard ale caracteristicilor modelului. Acestea pot fi folosite pentru compararea utilizatorilor folosind diverse modalități de testare cum ar fi: testarea ipotezei, t-tests și folosirea distanțelor cum ar fi distanța absolută, distanța Euclideană și a altor tipuri de distanțe. În tabelul 2 pot fi vizualizate diverse abordări cu rezultatele acestora.

Clasificare	TT	Env	S	M	Error rate (%)		
					FAR	FRR	EER
Distanță abs.	S	C	33	975	0.25	16.67	-
Distanță min.	S	C	39	171	2.8	8.1	-
Distanță	D	N	30	-	8.33	3.33	-
Statistică	D	C	30	60	15	0	-
Statistică	S	C	44	220	0	2.3	-
Statistică	S	C	30	553	1.89	1.45	-

TABLE II: Metode statistice utilizate și rezultate obținute [3].
TT - tipul de text; Env - mediu; S - Subiecți; M - mostre;
S - static; D - dinamic; C - Controlat; N - Necontrolat.

B. Rețele neurale

Rețelele neurale reprezintă mijloace de modelare statistică neliniară adaptivă, acestea sunt inspirate din interconexiunea biologică a neuronilor. Există două modalități în care ponderile pot fi asociate (învățate) - înățare supervizată și învățare nesupervizată. Pentru clasificarea măsurătorilor extrase din dinamica de apăsare a tastelor, în literatură s-au utilizat metode cum ar fi: perceptron, suma de produse (SOP), Adaline, arhitecturi de rețele neurale, backpropagation etc. [3]. În tabelul 3 sunt prezentate metrici de performanță pentru diverși algoritmi utilizați pentru realizarea comparației.

Clasificare	TT	Env	S	M	Error rate (%)		
					FAR	FRR	EER
Perceptron	S	C	24	1400	8	9	-
Adaline (RN)	S	C	46	1867	17.4	0	-
Rețele neurale	S	C	15	2100	8	7	-
Rețele neurale	D	C	22	-	0.015	4.82	-
BP-RN	S	C	151	-	1.11	0	-
FP-RN	S	C	30	6750	2.2	4.7	-
BP-RN	S	C	151	-	1.11	0	-
RBFN	S	C	30	180	2	-	-

TABLE III: Rezultate obținute în urma utilizării diverselor arhitecturi de rețele neurale [3].
TT - tipul de text; Env - mediu; S - Subiecți; M - mostre;
S - static; D - dinamic; C - Controlat; N - Necontrolat;
RN - Rețele neurale, BP - Backpropagation, FP - funcție de potențial, RBFN - Radial Basis Function Network.

C. Recunoaștere de tipare

Recunoașterea de tipare constă în clasificarea în diverse categorii a tiparelor sau a obiectelor folosind diverși algoritmi. Acesta utilizează de la algoritmi simpli de învățare precum „nearst neighbor", clusterizare până la algoritmi mult mai complecși de analiză a datelor („data mining") cum ar fi clasificatorii Bayes, discriminantul linear Fisher (FLD), support vector machine (SVM) și teoria grafurilor. Unul dintre cele mai mari avantaje alte utilizării algoritmilor de învățare probabilistici constă în faptul că aceștia oferă asociază o valoare de încredere deciziei luate. Aceștia pot de asemenea să reducă problemele introduse de propagările de eroare prin ignorarea răspunsurilor cu o valoare scăzută de încredere. În tabelul 4 sun prezentate rezultate obținute utilizând aceste metode.

Clasificare	TT	Env	S	M	Error rate (%)		
					FAR	FRR	EER
SVM	S	C	10	-	0.02	0.1	-
SVM	S	C	16	9600	0.69	0.1	-
SVM & AG	S	C	16	9600	0.69	0.1	-
Random Forest	S	U	41	8775	3.2	5.5	-
ADP	S	U	43	387	0.88	9.62	-
GMM	S	U	8	80	2.1	2.4	-
PCA	S	C	25	2400	-	-	-
Logică Fuzzy	S	C	10	200	3.4	2.9	-

TABLE IV: Rezultate obținute în urma utilizării diverșilor algoritmi de recunoaștere a tiparelor [3].
TT - tipul de text; Env - mediu; S - Subiecți; M - mostre;
S - static; D - dinamic; C - Controlat; N - Necontrolat; SVM - Suport Vector Machine, AG - Algoritm genetic, ADP - Arbori de decizie paraleli, GMM - Gaussian mixture models, PCA - Principal Component Analysis.

IV. Introducere detectori de caracteristici

În continuare vor fi prezentați succit un număr de detectori utilizați în literatură. Aceștia necesită o fază de antrenare în care se utilizează seturi de date de la utilizatorul original pentru a putea realiza un model caracteristic al acestuia. Ulterior etapei de antrenare are loc o etapă de testare în care se compară rezultatele obținute în urma folosirii acestui modelul pentru a compara seturi de date venite atât de la utilizatorul original cât și de la alți utilizatori.

A. Euclidean

Acesta algoritm este folosit pentru a modela fiecare șir de caractere sub forma unui punct în spațiul p-dimensional, unde p reprezintă numărul de caracteristici din vectorul de timpi. Acesta tratează datele de antrenare ca pe un nor de puncte, calculul gradului de similitudine pentru un vector de test fiind calculat în funcție de distanța acestuia față de centrul norului de puncte. Specific etapei de antrenare este calculul unui vector mediu pentru vectorii de timp ai utilizatorului iar în faza de testare, scorul este calculat în funcție de distanța Euclidiană dintre vectorul mediu și vectorul de test.

Fie $x = [x_1, x_2, ..., x_n]$ vectorul mediu calculat în faza de antrenare și $y = [y_1, y_2, ..., y_n]$ vectorul testat, atunci distanța Euclidiană este dată de formula 3.

$$dist_E(a, b) = \sqrt{\sum_i^n (x_i - y_i)^2} \tag{3}$$

B. Euclidean (normată)

Acest detector se mai numește și „clasificatorul de distanță minimă normalizat" [13]. În etapa de antrenare, vectorul mediu de valori este calculat, întocmai ca la detectorul Euclidean. Faza de testare este de asemenea similară, diferența constă în faptul că distanța rezultată este normată la produsul normelor celor doi vectori [13].

Fie x vectorul mediu, y vectorul testat și d distanța Euclideană, atunci scorul rezultat va fi dat de formula (4).

$$S = \frac{d}{\|x\|_1 \|y\|_1} \tag{4}$$

Unde $\|x\|_1$ este calculat cu formula (5).

$$\|x\|_k = (\sum_i^n |v_i|^k)^{1/k} \tag{5}$$

C. Manhattan

Acest algoritm este similar cu detectorul Euclidean cu excepția că distanța nu este Euclidiană ci Manhattan (sau „city-block"). În faza de antrenare se realizează vectorul de antrenare prin medierea vectorilor de timp iar în faza de testare, scorul este calculat pe baza distanței Manhattan dintre vectorul mediu și vectorul de test.

Fie x vectorul mediu, y vectorul testat, atunci distanța Manhattan dintre acștia este dat de formula (6).

$$dist_{Man} = \sum_i^n |x_i - y_i| \tag{6}$$

D. Manhattan (filtrată)

Metoda filtrată este similă cu cea Manhattan originală, diferind de acesta prin faptul că valorile extreme sunt filtrate în etapa de antrenare. Pentru aceasta se calculează deviația standard a fiecărui element, pe baza acesteia sunt eliminate valorile care sunt mai mari decât suma dintre media și de trei ori valoarea deviației standar.

Eliminarea valorilor extreme conduce la obținerea unui vector mediu de valori mai precis. Testarea se execută similar metodei nefiltrate, folosindu-se calculul distanței Manhattan dintre valorile vectorului mediu și velorile vectorilor testați.

E. Manhattan (scalată)

În etapa de antrenare se calculează doi vectori. Unul este cel al valorilor medii ale datelor de antrenare de la utilizator iar celălalt reprezintă deviația absolută medie a fiecărui element. În faza de testare, se procedează similar ca în cazul metodei clasice doar că valorile absolute ale diferenței dintre elementele vectorului mediu și elementele vectorului testat sunt scalate la deviația absolută medie a elementului respectiv calculată la momentul antrenării. Pentru clarificare este prezentată următoarea exemplificare.

Fie x vectorul mediu de calori și a vectorul de deviații absolute medii asociate valorilor vectorului x iar y vectorul testat. Avem următoarea formulă de calcul a distanței dintre aceștia, formula (7).

$$dist_{Man} = \sum_{i}^{n} \frac{|x_i - y_i|}{a_i} \tag{7}$$

F. Mahalanobis

Metoda Mahalanobis seamănă cu cei doi detectori prezentați anterior (Euclidian și Manhattan) dar calculul distanței este mai complicat. Distanța Mahalanobis poate fi văzută ca o extensie a distanței Euclidiene privind corelația dintre caracteristici. În faza de antrenare, sunt calculați vectorul mediu de valori al datelor de antrenare precum și matricea de covariație a acestora. În faza de testare, scorul este calculat între vectorul mediu și cel de test în felul următor:

Dacă x este vecorul de valori medii obținut la antrenare iar y vectorul de test și S matricea de covariație, distanța Mahalanobis, formula (8)

$$dist_{Mh} = (x - y)^T S^{-1} (x - y) \tag{8}$$

G. Mahalanobis (Normată)

Acest detector mai este numit și clasificatorul Bleha normat [**?**]. Partea de antrenare este identică cu cea a detectorului Mahalanobis. Diferența constă în procesul de testare, scorul fiind calculat prin normarea distanței Mahalobis folosind același divizor ca în cazul distanței Euclidiene normate.

H. Cel mai apropiat vecin (Mahalanobis)

Cunoscută în literatură sub denumirea de „Nearest-neighbor”, acest algoritm saleavează vectorii de antrenare și calculează matricea de covariație a acestora. Pentru testare detectorul calculează distanța Mahalanobis dintre toți vectorii de antrenare și cei de testare. Scorul final fiind dat de distanța de la vectorul de test la cel mai apropiat vector de antrenare.

I. Rețele neurale (standard)

Acest detector este alcătuit dintr-o rețea neurală cu feed-forward antrenată cu algoritmul „back-propagation”. În faza de antrenare, o rețea este construită cu p noduri de intrare, un nod de ieșire și $[2p/3]$ noduri ascunse. Detectorul este constituit din 6 noduri de intrare și 4 noduri ascune deoarece s-au considerat doar vectori de date a câte 6 caracteristici. Vectorii utilizați pentru antrenare în cazul [**?**] sunt constituiți din 31 de caracteristici astfel fiind nevoie de o extindere a modelului inițial utilizat pentru a suporta 31 de noduri de intrare în timp ce se menține rata de două noduri ascune pentru fiecare 3 noduri de intrare.

Ponderile rețelei au fost toate inițializate cu valoarea 0.1, și bias-ul a fost inițializat cu valori aleatoare. Detectorul a fost antrenat pentru a produce 1.0 pe nodul de ieșire pentru fiecare vector de testare. S-au antrenat 500 de generații folosind rata de învațare egală cu 0.0001. Pe parcursul procesului de testare, vectorii testați au fost trecuți prin rețea și s-au înregistrat valorile produse la ieșirea din rețea [**?**].

Dacă s reprezintă ieșirea rețelei, atunci scorul a fost calculat ca fiind $1 - s$ deoarece s are valori apropiate de 1.0 pentru vectorii de test similari cu vectorii de antrenare și apropiate de 0.0 în cazul vectorilor diferiți.

J. Rețele neurale (auto-asoc)

Acest tip de arhitectură mai este referit ca fiind „auto-asociativ, perceptron multistrat”. Incorporează de asemenea rețeaua neurală cu feed-forward antrenată folosind algoritmul de „back-propagation” dar spre deosebire de structura clasică a rețelei neurale este concepută pentru a fi folosită ca detector de diferențe [**?**]. Faza de antrenare învață rețeaua să producă vectori de ieșire apropiați de valorile intrării pentru vectorii de antrenare (de aceea denumirea de descriptor auto-asociativ). Apoi pe parcursul fazei de testare, se obțin scoruri mari pentru vectorii de test ce diferă de cei folosiți în procesul de antrenare și invers pentru cei similari.

K. Logică fuzzy

Acest detecor incorporează o procedură de deducție bazată pe logica fuzzy. Pricipiul de bază al acestui algoritm constă în împărțirea în seturi fuzzy (permisive, de exemplu timpii cuprinși între 210 și 290 milisecunde fac parte dintr-un set numit „foarte rapid”). Seturile sunt numite fuzzy deoarece elementele acestora pot să aparțină parțial unui set (de exemplu durata de 255 de milisecunde aparține cu precis de acest se în timp ce valorile existente mai spre extremități pot să aparțină parțial de aceste seturi cum ar fi valoarea 290 pentru setul considerat anterior).

În etapa de antrenare, detectorul determină cât de puternic aparține fiecare caracteristică anumitor seturi și marchează caracteristicile cu indicele setului de care aparțin cel mai puternic. În etapa de testare verifică dacă apartenența fiecărei caracteristici de timp este aceeași cu cea a setului de date de antrenare. Scorul asociat este alcătuit din valoarea medie a lipsei apartenenței membrilor vectorului la o anumită clasă [13].

L. z-score

Acest detector reprezintă o tehnică statistică. În faza de antrenare, detectorul calculează media și deviația standar a fiecărui vector de timp. În faza de testare, acesta calculează valoarea absolută z-score a fiecărui vector de date de test.

Dacă x este vecorul de valori medii și s reprezintă valorile deviației standar obținute la antrenare iar y vectorul de test, valoarea scorului z este calculată cu formula (9).

$$S_z = \frac{|x_i - y_i|}{s_i} \tag{9}$$

M. SVM cu o singură clasă

Acest detector folosește algoritmul Suport Vector Machime (SVM), acesta proiectează două clase de date într-un spațiu multi-dimensional și identifică separatorul linear dintre aceste două clase. O variantă de detector SVM cu o singură clasă proiectează date de la o singură clasă și identifică separatorul linear dintre origine și proiecție.

În faza de antrenare, detectorul construiește o singură clasă folosind datele de antrenare. În faza de testare, vectorul de test este proiectat în același spațiu multi-dimensional apoi este calculată distanța dintre noul separator linear obținut și cel anterior.

N. k-means

Acest detector folosește algoritmul de clusterizare k-means pentru a identifica clustere în datele de antrenare apoi calculează dacă vectorii de test sunt apropiați de oricare cluster obținut în etapa de antrenare.

V. Factori ce afectează performanța sistemului

În cazul scrierii libere, o problemă pentru utilziatorul normal o reprezintă variația vitezei de scriere. S-au observat variații puternice în compararea aceluiași utilizator care realizează scriere liberă și scierea unui text static predefinit. Un motiv al acestei variații o reprezintă latența introdusă de incertitudinea în planificarea datelor ce urmează a fi scrise. În schimb, la dactilografii profesioniști s-a remarcat o bună comparare a celor două cazuri [3].

Starea emoțională a utilizatorului poate influența viteza de scriere. S-a observat o scădere de 70% a vitezei de scriere în cazul subiecților cu o stare negativă și o creștere de 83% în cazul utilizatorilor care au o stare pozitivă [3]. De asemenea, viteza de scriere mai poate fi influențată și de starea de sănătate, poziția în care se realizează tastarea.

Deși teoretic nu se poate copia tiparul de scriere al unui utilizator, în practică s-a demonstrat că parolele cu puține caractere pot fi imitate cu succes. O bună analogie poate fi realizată cu semnătura realizată de o persoană cu o ustensilă de scris. Aceasta cu cât este mai complexă, cu atât este mai greu de replicat. În concluzie este necesară o parolă suficient de complexă pentru a nu putea fi falsificată cu ușurință. De asemenea, utilizarea unei parole complexe, permite o mai bună analiză a utilizatorului în momentul înrolării în sistem.

VI. Concluzii

Se poate realiza un sistem suficient de precis pentru a adăuga un nou start de securitate împotriva amenințărilor la adresa securității. Această metodă bazându-se pe o măsură biometrică psihică, nu poate fi însușită de alte persoane. În această cercetare au fost prezentate tehnicile de achiziție și metricile achiziționate, s-au prezentat rezultate obținute în literatură la aplicarea anumitor metode de analiză și comparare a datelor.Analiza dinamicii tastării textului liber reprezintă un domeniu puțin studiat care are potențial pentru diverse aplicații SAAS.

Referinte

[1] https://www.internetworldstats.com/stats.htm

[2] L. F. Coppenrath and Associates. „Biopassword Technology Overview”, http://www.lfca.net/Reference%20Documents/Biometric%20Solutions%20By%20Classification.pdf

[3] Salil P. Banerjee, Damon L. Woodard: „ Biometric Authentication and Identification using Keystroke Dynamics: A Survey”

[4] L. Shaffer. Tutorials in Motor Neuroscience, chapter Cognition and Motor Programming. Kluwer Academic Publishers, 1991.

[5] T. A. Salthouse. Perceptual, Cognitive, and Motoric Aspects of Transcription Typing. Psychological Bulletin, 99(3):303 – 319, 1986.

[6] K. S. Balagani, V. V. Phoha, A. Ray, and S. Phoha. On the Discriminability of Keystroke Feature Vectors Used in Fixed Text Keystroke Authentication. Pattern Recognition Letters, 32:10701080, 2011.

[7] Burileanu Dragoș, „Tehnologii biometrice. Recunoaşterea semnăturii dinamice.”

[8] Jugurta R.M.F. and Freire O.E. On the equalization of keystroke timing histograms. Pattern Recognition Letters, 27:1440–1446, October 2006.

[9] K. S. Killourhy and R. A. Maxion. Comparing Anomaly-Detection Algorithms for Keystroke Dynamics. In IEEE/IFIP International Conference on Dependable Systems Networks, pages 125–134, July 2009

[10] R. Giot, M. El-Abed, and C. Rosenberger. GREYC keystroke: A benchmark for keystroke dynamics biometric systems. In IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems, pages 1 –6, Sept. 2009.

[11] J. D. Allen. An Analysis of Pressure-Based Keystroke Dynamics Algorithms. Master's thesis, Southern Methodist University, Dallas, TX, U.S.A., May 2010.

[12] L. Bello, M. Bertacchini, C. Benitez, J. C. Pizzoni, and M. Cipriano. Collection and Publication of a Fixed Text Keystroke Dynamics Dataset. In CACIC'10, October 2010.

[13] Kevin S. Killourhy and Roy A. Maxion. "Comparing Anomaly Detectors for Keystroke Dynamics," in Proceedings of the 39th Annual International Conference on Dependable Systems and Networks (DSN-2009), pages 125-134, Estoril, Lisbon, Portugal, June 29-July 2, 2009. IEEE Computer Society Press, Los Alamitos, California, 2009.

[14] Kevin S. Killourhy and Roy A. Maxion. Free vs. transcribed text for keystroke-dynamics evaluations. In Learning from Authoritative Security Experiment Results (LASER-2012), July 18–19, 2012, Arlington, VA, 2012. ACM Press.

[15] Yilin Li, Baochang Zhang, Yao Cao, Sanqiang Zhao, Yongsheng Gao, Jianzhuang Liu. "Study on the BeiHang Keystroke Dynamics Database". International Joint Conference on Biometrics (IJCB), pp.1-5, 2011.

[16] Keystroke Patterns as Prosody in Digital Writings: A Case Study with Deceptive Reviews and Essays Ritwik Banerjee, Song Feng, Jun S. Kang, Yejin Choi Empirical Methods on Natural Language Processing (EMNLP). 2014.

[17] Keystroke Patterns Classification using the ARTMAP-FD Neural Network C. C. Loy, W. K. Lai, and C. P. Lim International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Taiwan, 2007 (IIHMSP 2007)

[18] Pressure-based Typing Biometrics User Authentication Using The Fuzzy ARTMAP Neural Network C. C. Loy, C. P. Lim, and W. K. Lai International Conference on Neural Information Processing, Taiwan, 2005 (ICONIP 2005)

[19] John V. Monaco, Charles C Tappert. The Partially Observable Hidden Markov Model and its Application to Keystroke Dynamics. Pattern Recognition, 2017.