DOI 10.15276/imms.v14.no1-2.97
УДК 004.056.5

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 1-2, pp. 97-107

РОЗРОБКА ПРОГРАМНОГО ЗАСТОСУНКУ ДЛЯ БІОМЕТРИЧНОЇ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА ЗА СТИЛЕМ НАБОРУ НА КЛАВІАТУРІ

Д. М. Слабенко[1], О.А. Стопакевич[1], А. О. Стопакевич[2]

[1] Національний університет «Одеська політехніка»,
1, Шевченка пр., м.Одеса, 65044, Україна
email: stopakevich@op.edu.ua
[2] Державний університет інтелектуальних технологій та зв'язку,
1, Кузнечна вул., м.Одеса, 65029, Україна
email: stopakevich@gmail.com

В статті розглядається розробка програмного забезпечення для біометричної ідентифікації користувачів за стилем набору на клавіатурі. Біометрична автентифікація, зокрема, методи, що базуються на поведінкових характеристиках, набувають все більшої популярності завдяки своїй здатності забезпечувати безпеку без необхідності запам'ятовування паролів. Розробка системи біометричної ідентифікації на основі набору тексту, введеного користувачем на клавіатурі, є привабливим рішенням, оскільки не лише потенційно забезпечує доволі надійну ідентифікацію, а й не потребує витрат на спеціальне обладнання. В роботі наведено перелік факторів, які впливають на процес набору тексту, таких як час натискання клавіш, швидкість друку, частота помилок та інші, які можуть варіюватися в залежності від індивідуальних особливостей користувача. Також обговорюються недоліки біометричної ідентифікації, зокрема, вплив зовнішніх факторів, таких як втома або відволікання, на точність аутентифікації. На основі аналізу параметрів, відомих методів та підходів до біометричної ідентифікації за стилем набору на клавіатурі запропоновано новий алгоритм перевірки, який базується на аналізі інтервалів часу натискання та пошуку клавіш, що дозволяє визначити відповідність між збереженим профілем користувача та його поточним набором тексту. Для зменшення впливу зовнішніх факторів рекомендується використовувати фіксований текст обсягом не менше 300 символів. Результати експериментів, проведених за допомогою розробленого програмного застосунку, підтверджують ефективність розробленого програмного забезпечення на основні запропонованого алгоритму ідентифікації. Результати демонструють достатню надійність та точність у процесі ідентифікації. Робота має практичне значення для розробки нових методів безпеки в інформаційних технологіях, шляхом впровадження біометричних систем у різних сферах.

Ключові слова*:* біометрія, поведінкова, ідентифікація, авторизація, клавіатура, набір, користувач, програмне, забезпечення, профіль, метод, алгоритм

Вступ. Розвиток технологій призводить до того, що все більше конфіденційної інформації, поширення якої є небажаним, зберігається на цифрових пристроях. Отже, актуальною стає проблема розробка безпечних та економічно ефективних механізмів автентифікації.

Одним з підходів, який розвивається останнім часом, є біометрична ідентифікація особи. Крім класичних відбитків пальців, розвиваються технології ідентифікації за фотографією, голосом тощо. Недоліком цих методів є необхідність в застосуванні окремих приладів, складність діагностики. Тобто якщо ідентифікація не проходить, то доволі складно зрозуміти причину й провести додаткові дослідження причин цього. Методи ідентифікації за мовою, письмом, ходьбою, рухом та набором на клавіатурі відомі, як поведінкова біометрія. Перевагою цих методів над фізіологічним аналогом є здатність працювати в прихованому режимі. Недоліком є мінливість характеристик в залежності від стану здоров'я, фізичних пошкоджень тощо, що впливає на точність ідентифікації. Ще одним методом біометричної ідентифікації є біометрія натискання

клавіш. Вона пов’язана з вимірюванням та оцінкою ритму друку людини на цифрових пристроях. Під таким пристроєм зазвичай мається на увазі комп'ютерна клавіатура, мобільний телефон або сенсорна панель. Форма цифрового сліду створюється при взаємодії людини з цими пристроями. Вважається, що ці сигнатури багаті когнітивними якостями, які достатньо унікальні для кожної людини й мають великий потенціал для ідентифікації користувача [1]. Застосування біометричної ідентифікації за набором на клавіатурі дозволяє реалізувати процедуру ідентифікації як одночасно достатньо надійну, так й таку яка не вимагає додаткових економічних витрат на придбання обладнання для біометричної ідентифікації.

Задачею статті є розробка та дослідження програмного застосунку для біометричної ідентифікації за набором на клавіатурі, який базується на запропонованій метриці оцінки подібності набору з еталонним набором користувача.

Особливості біометричної ідентифікації за набором. Біометрична ідентифікація за набором має принципові особливості, якими не можна зневажати. Вона поступається з точки зору точності аутентифікації через зміни в ритмі набору тексту, які можуть бути викликані такими зовнішніми факторами, як травма, втома або відволікання [2]. Характер набору тексту людиною може поступово змінюватися відповідно до звикання, розвитку навичок набору тексту, адаптації до пристроїв введення та інших факторів навколишнього середовища. Тому рекомендується постійно оновлювати збережений профіль натискання клавіш для порівняння [3, 4].

Параметри біометрії натискання клавіш. Серед факторів, які впливають на біометричну ідентифікацію користувача при наборі тексту на клавіатурі, можна виділити наступні [5]:

– час пошуку, тобто час, потрібний для знаходження кожної клавіши до її натискання;
– час натискання, тобто час утримання клавіши перед відпусканням [6, 7];
– час польоту, тобто час, коли натискається наступна клавіша;
– швидкість друку;
– час перерв / пауз при наборі тексту;
– кількість допущених помилок і врахування найпоширеніших помилок;
– техніка виправлення помилок користувачем при наборі тексту;
– тип локальної клавіатури, яка використовується (механічна та плівковій, мала та велика, пряма та вигнута клавіатури);
– набір проводиться правшею чи лівшою (аналіз частини клавіатури, яка ефективніше використовується);
– місце розташування клавіатури (на столі, на ногах, …);
– типову послідовності букв, яка найчастіше вживана в рідній мові користувача.

Аналіз літератури показує, що перші з перелічених параметрів є основними, а інші можуть бути використані для аналізу, як допоміжні.

Відмітимо, що ідентифікація користувача при біометричній ідентифікації може бути проведена тільки за набором тексту з достатньо великою кількістю символів. Згідно з дослідженням [8, 9], кількість символів має бути не меншою за 300, а в якості основних показників, які треба застосувати при ідентифікації, рекомендовано використати час польоту та час утримання, що видно з рис.1.

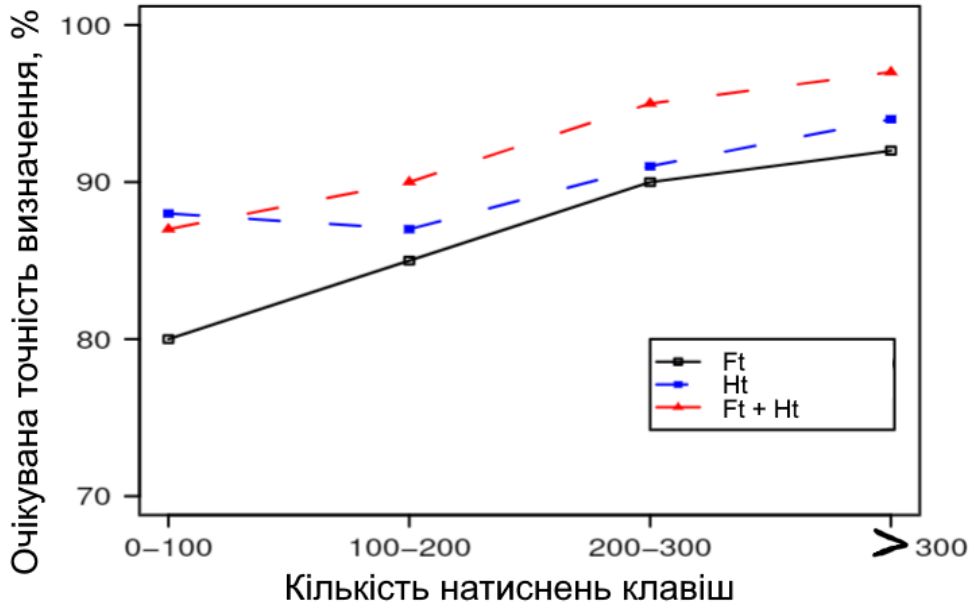

Рис. 1. Ефективність застосування основних параметрів: Ft – час польоту, Ht – час натиснення [8]

Запропонований алгоритм оцінки співпадіння набору еталонному профілю користувача. Звичайно алгоритми біометричної ідентифікації намагаються визначити подібність незалежно від тексту, який має набиратись. При наборі довільного тексту задача ідентифікації дуже складна. Орієнтація на диграми має свою раціональну мотивацію – людина набирає одне й теж слово в прямому й зворотному напрямку з різною швидкістю. Однак вона не враховує, що когнітивною одиницею набору тексту є слово, яке переводиться в натиснення окремих клавіш. Для людини характерна різна швидкість набору різних слів, яка обумовлена тим, наскільки часто таке слово набиралося людиною (прості слова, артиклі будуть набиратись швидше), наскільки людина грамотна і чи є мова набору її рідною мовою (треба замислюватись, як вірно набрати слово, чи ні), чи було визначено закінчення, чи викликає слово чи текст певні асоціації та думки. В такій мові, як англійська, є типові артиклі, закінчення, неправильні дієслова. Наприклад, зазвичай час набору артиклів “the”, “a”, прийменників “at”, “in”, закінчень “ing”, “ed”, дієслів типу “flet”, “was”, “been” тощо менший за час набору менш часто вживаних слів. Диграми частіше виділяються в наборі людину, яка застосовує сліпий десятипальцевий метод набору. Однак сліпий набор чітко виділяє й час натиснення окремих клавіш, оскільки в такому наборі є фіксоване положення пальців за замовченням, тому перехід з однієї клавіши на іншу обумовлений в першу чергу геометричною відстанню. Для людини, яка набирає просто кожний символ окремо й не тримає чіткої позиції рук, характеристики натиснення клавіш істотно залежать від слова.

Для зменшення впливу сторонніх факторів, біометричну автентифікацію ми пропонуємо визначати, використовуючи фіксований текст розміром понад 300 символів без регістру та розділювачів, й взяти за основу розкид параметрів користувача за кожною окремою клавішею.

Алгоритм перевірки вибраний наступний.

1. При формуванні профілю користувача – запам’ятати час пошуку та час натискання кожної літери тексту при наборі фіксованого тексту, що вводиться.
2. При ідентифікації – запропонувати ідентифікованої особі ввести той же текст і запам’ятати його.

Організувати цикл по всім літерам набраного тексту:

Якщо поточна літера по часу пошуку входить в діапазон часу еталонного набору

для даної літери

То додати таку літеру в тексті до лічильника g_1;

Інакше додати таку літеру в тексті до лічильника b_1;

Якщо поточна літера по часу натискання входить в діапазон часу еталонного набору для даної літери

То додати таку літеру в тексті до лічильника g_2;

Інакше додати таку літеру в тексті до лічильника b_2.

Критерієм проходження ідентифікації є умова $\left\{ \frac{b_1}{g_1} < 0.15 \ \& \ \frac{b_2}{g_2} < 0.15 \right\}$.

Програмна реалізація застосунку. Застосунок має включати дві програми, які реалізуємо мовою програмування Python для ОС Windows.

Перша програма реалізується з консольним інтерфейсом та аргументами командного рядка й має реалізовувати введення текстової фрази нижнім регістром без розділювальних знаків (про що користувача попереджують) та запис результатів цього введення в спеціальний JSON файл. При цьому застосовані такі бібліотеки для мови Python: sys, os, datetime, time, re, curses (windows), keyboard, beepy, json. Вибір введення натиснення клавіш саме в консольному застосунку обумовлено тим, що архітектура графічного інтерфейсу мови Python та взагалі ряду мов, які побудовані на концепції віртуальної машини, може призводити до значних похибок при отриманні часу натиснення клавіш в межах програмного інтерфейсу. Бібліотеки графічного інтерфейсу не націлені на отримання такого виду інформації, події звичайно орієнтуються на сам факт натиснення чи відтиснення клавіши. Нажаль, при введенні українською мовою в консолі виникають певні проблеми. Windows використовує розкладку cp866 для введення й в цій розкладці відсутня українська літера «і». Замість цього передбачається, що буква буде введена латинкою. Зміна кодування на cp1251 й Unicode призводить до інших проблем. В результаті, проблема біометрії клавіш латинкою розв'язується більш легко, а з українськими літерами виникають проблеми з бібліотеками curses та keyboard. З такої проблеми знайдено вихід в межах обмежень програмних бібліотек Python та програмного інтерфейсу Windows– вводити за фактом латинські літери, ігноруючи їх регістр, але відображати українські в консолі. Інтерфейс curses дозволяє відображати текст, що має бути набраний, й жовтим показує ту частину, яка вже набрана. В разі помилки подається за допомогою бібліотеки beep звуковий сигнал. Обробку кодів клавіш, подій натиснення та відтиснення реалізує бібліотека keyboard, ключовий метод – read_event(). Вимірювання часу між подіями клавіатури проводиться за допомогою time.perf_counter().

Залежно від аргументів командного рядка (sys) програма формує JSON файл з протоколом натискання як новий профіль, чи як результат перевірки відповідності певному профілю.

Підсистема працює у консолі за алгоритмом, показаним на рис. 2.

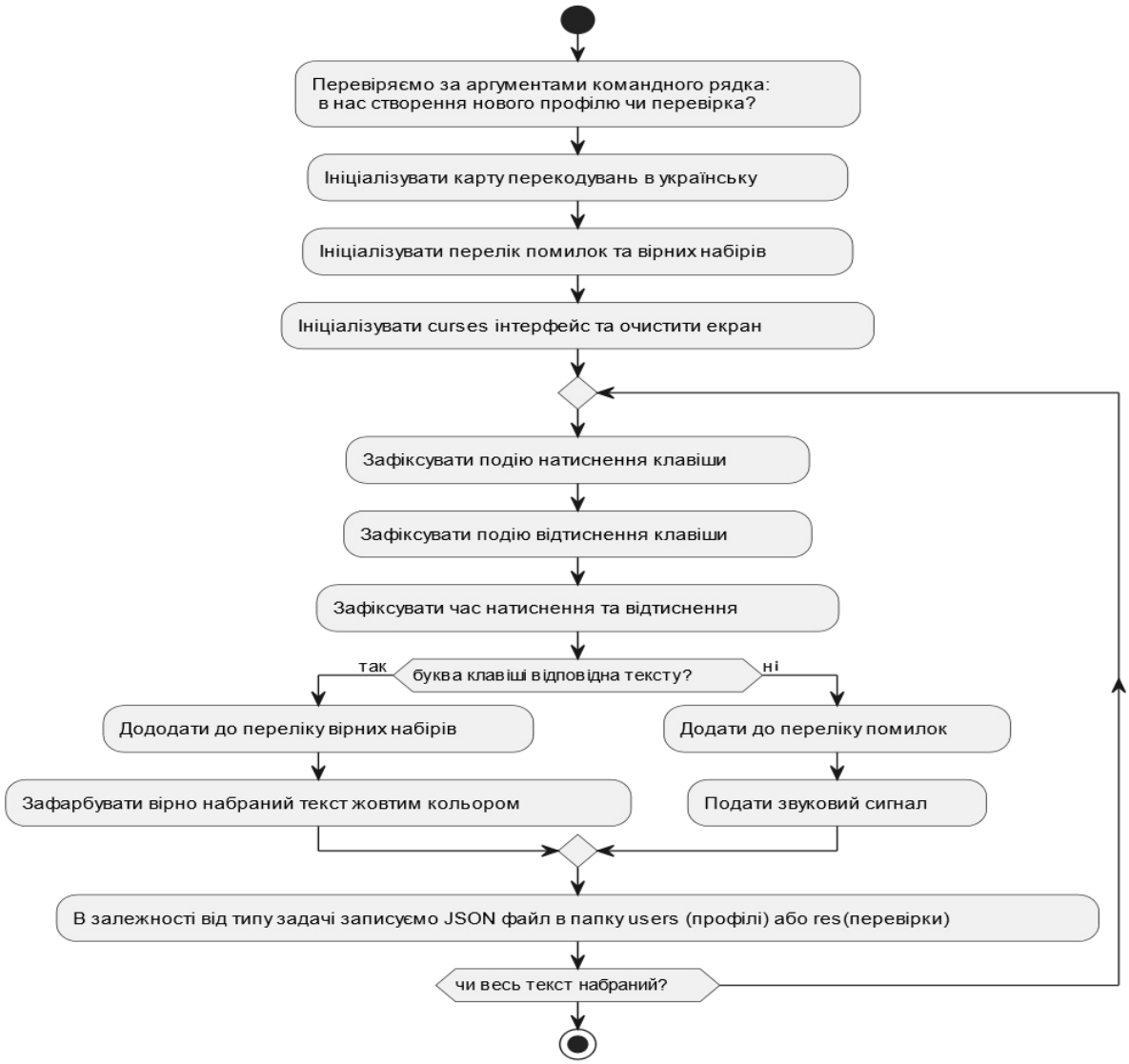

Рис. 2. Алгоритм роботи підсистеми для ідентифікації користувача

Підсистема приймає два аргументи командного рядка. Перший аргумент: new (новий профіль) чи check (перевірити профіль). Другий аргумент – ім'я користувача, яке має відповідати регулярному виразу

Зовнішній вигляд консольної програми, викликаної як python get_input.py new user3 в процесі набору тексту, показаний на рис. 3, 4.

```
d:\Program Files\python_keyboard_biometrics>python get_input.py new user3
Система підготовлена. Встановіть англійську розкладку якщо не стоїть.
Підготовтесь до набору та натисніть <enter>.
Набирайте зразу кібербезпека і далі.
```

Рис. 3. Попереджувальний текст – підготовка до набора фрази

```
Администратор: C:\Windows\System32\cmd.exe - python  get_input.py new user3
Кібербезпека це захищеність життєво важливих інтересів людини і громадянина суспільства та держави
під час використання кіберпростору за якої забезпечуються сталий розвиток інформаційного суспільства та
цифрового комунікативного середовища своєчасне виявлення запобігання і нейтралізація реальних і
потенційних загроз національній безпеці України у кіберпросторі.
```

Рис. 4. Процес набору фрази (жовтим відображена набрана частина)

Результативний JSON файл профілю user3 має наступний вигляд

{"text":
"\u041a\u0456\u0431\u0435\u0440\u0431\u0435\u0437\u043f\u0435\u043a\u0430
\u0446\u0435 …. ", "dt": "2024_05_03_19_19_14", "good": {"1": [0.4580399999395013,
0.08465239987708628, 19, "\u043a"], "2": [0.2832812999840826, 0.08598900004290044, 31,

"\u0456"], **….** "362": [0.20009749988093972, 0.08343700016848743, 31, "\u0456"], "363": [0.4003336001187563, 0.15026960009709, 53, "."]}, "fail": {"6": "\u0435", "27": "", "131": "\u0440", "150": "\u0441 \u0443", "154": "\u0441", "300": "\u043a \u043e", "360": "\u0440"}}

Друга програма має реалізовувати графічний інтерфейс для користувача. Цей інтерфейс має передбачати введення бази еталонних наборів користувачів (профілів), а також перевірку відповідності конкретного набору профілю заданого користувача.

Додаткова статистика може бути викликана при натисненні спеціальної клавіші. Ця статистика являє собою звіт у форматі веб сторінки (HTML), який містить детальну інформацію про набір користувача. Цей звіт може бути використаний як для аналізу причин того, чому процедура біометричної ідентифікації не вдалась, так і для подальшого удосконалення алгоритму ідентифікації. Крім інформації, що перевіряється, звіт містить також певну додаткову інформацію, яка використовується для ідентифікації інших параметрів біометричного набору, перелічених вище.

Для реалізації програми потрібні наступні програмні бібліотеки: tkinter, os, re, json, subprocess, ctypes, webbrowser, statistics.

Дослідження роботи застосунку у цілому. План досліджень наступний.

1. При створенні профілю Користувач №1 пише тестову фразу і вона буде збережена як профіль User1 (рис.5). Паралельно набір Користувача №1 запишемо програмою Keyboard Recorder.
2. При створенні профілю Користувач №2 пише тестову фразу й вона буде збережена як профіль User2 (рис.6).
3. Записаний набір Користувача №1 перевіримо на відповідність профілю User1.
4. Записаний набір Користувача №1 перевіримо на відповідність профілю User2.
5. Для перевірки сталості роботи програми Користувач №1 ще раз, через кілька днів, ідентифікується, написавши тестову фразу знов, й вона буде перевірена відносно профілю User1.

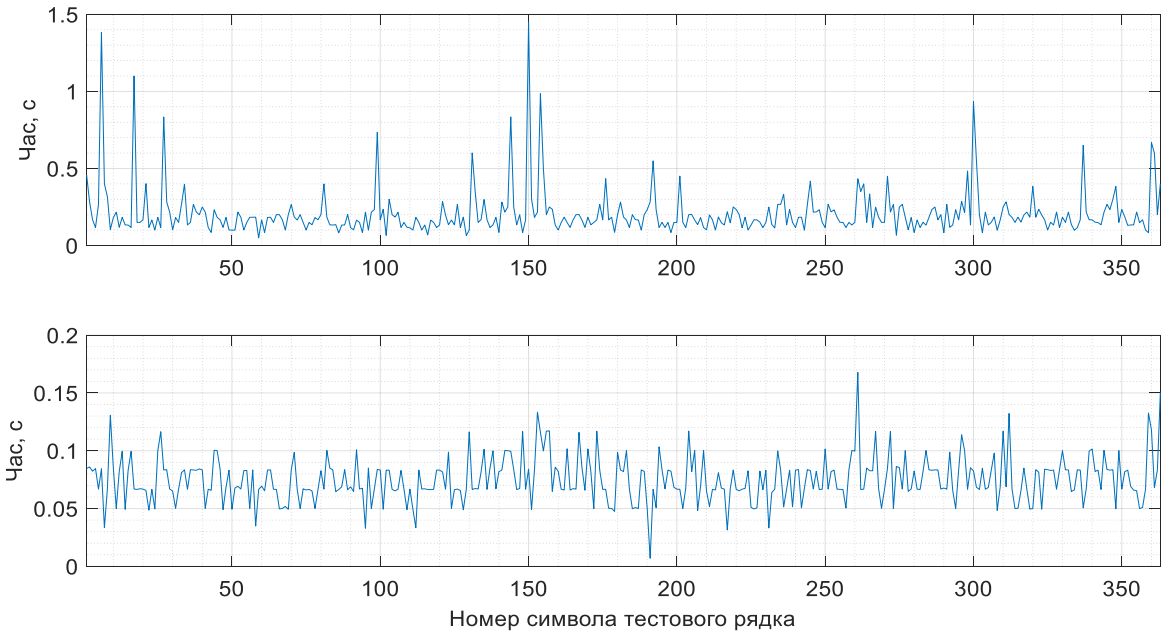

Рис. 5. Профіль користувача User1 (час пошуку та натискання)

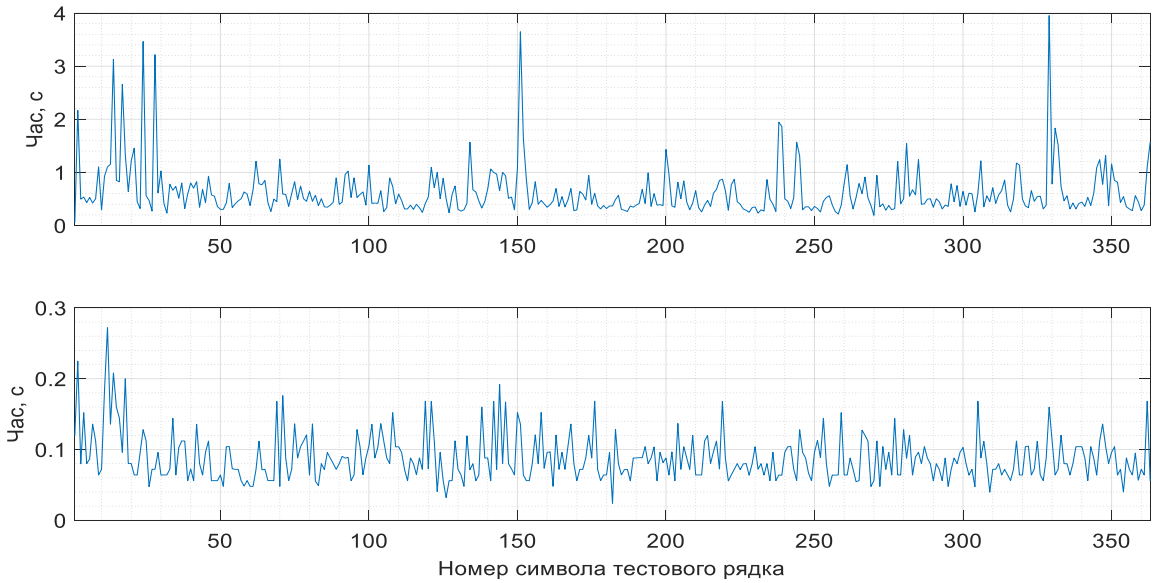

Рис. 6. Профіль користувача User2

Перевірка запису Keyboard Recorder за профілем користувача User1 показало майже повну відповідність за двома критеріями перевірки.

Перевірка запису Keyboard Recorder з профілем користувача User2 показало майже повну невідповідність за двома критеріями перевірки. За часом пошуку в інтервал потрапило лише 13 символів з 360, час часом натискання – 256 символів і 107 не попали. Таким чином, чітко бачимо, що це набирав інший користувач, ніж користувач User2.

Таким чином користувач був ідентифікований. Відсоток не співпадінь становив 8% для пошуку й 12% для натискання, що в цілому є допустимим відхиленням.

Проаналізуємо основні та додаткові параметри одного користувача, які можна отримати в програмі за допомогою звіту.

Закладений в критерії програми час натискання та інтервал його зміни дуже значимо залежить від літери. Це ми бачимо за фрагментом звіту, показаного на рис. 7. Наприклад, для букви «й» мінімальний час натискання – 0.09, а для «е» – 0.03. При чому різниця для букви «й» між мінімумом та максимумом не значна, а для буки «е» складає близько трьох раз.

Літера	Мінімум	Середнє	Максимум
	0.06625129998428747	0.08546933500256274	0.11693970000487752
а	0.0499716000049375	0.07146240356814815	0.10023859998909757
б	0.0833188000251539	0.08583101429394446	0.10027019999688491
в	0.049667099985526875	0.07210820666320311	0.0999491999973543
г	0.04892969998763874	0.06372225000814069	0.11619270002120174
д	0.06681400001980364	0.08627914000535383	0.09983479999937117
е	0.03331880000769161	0.06657056190167732	0.08352289997856133
ж	0.03340759998536669	0.06119966666058948	0.0834364999900572
з	0.06644220001180656	0.09562904545137743	0.16661909999675117
и	0.0499775999924168	0.0682624526272871	0.083427000005031
й	0.09987679999903776	0.10000601999927312	0.10007049998966977
к	0.05002150000655092	0.07000078999553808	0.08367729999008588
л	0.04999189998488873	0.07969094444221507	0.10020119999535382

Рис.7. Фрагмент звіту з мінімальним, середнім та максимальним значенням часу натискання літери при наборі тексту користувачем №1

Закладений в критерії програми час пошуку та інтервал його зміни теж дуже значимо залежить від літери. Це ми бачимо за фрагментом звіту, показаного на рис. 8. Наприклад, для букви «ж» мінімальний час пошуку – 0.2, а для «а» – 0.06. При чому, які й з часом натискання, різниця для букви «й» між мінімумом та максимумом не значна, а для літери «е» близько п’яти раз. Різниця для букви «з» більша за 10 раз. Між часом набору та часом пошуку присутня кореляція, однак вона має узагальнений характер. Наприклад найменший мінімальний час пошуку має літера «д», найбільший мінімальний час пошуку – літера «ж». Проте найменший мінімальний час натискання має літера «е», а найбільший мінімальний час натискання – літера «й».

Літера	Мінімум	Середнє	Максимум
	0.06661710000480525	0.17842948249672191	0.7331793999765068
а	0.06852640002034605	0.1918333607162432	0.8340992000012193
б	0.13348479999694973	0.3716093571399272	1.3676700000069104
в	0.08337020000908524	0.18473089333662454	0.40020110001205467
г	0.15020519998506643	0.1888071833285115	0.25035219997516833
д	0.06625430000713095	0.19037059999536723	0.2688532999891322
е	0.08349099999759346	0.15745305238219554	0.40035700000589713
ж	0.2001550999993924	0.21123473334591836	0.21681830001762137
з	0.09970639998209663	0.3137950181791728	1.1002237999928184
и	0.08341690001543611	0.15978699474362656	0.23417880001943558
й	0.15003119999892078	0.17343606000067666	0.18357929997728206
к	0.11687870000605471	0.22079115000087768	0.4589095000119414
л	0.11668909998843446	0.16308754444212858	0.2502237999869976

Рис.8. Фрагмент звіту з мінімальним, середнім та максимальним значенням часу пошуку літери при наборі тексту користувачем №1

На рис. 9 показані додаткові статистичні результати набору тексту користувачем №1.

Параметр	1 рядок	2 рядок	3 рядок	ліва частина	права частина
Мінімальний час пошуку	0.083	0.066	0.067	0.067	0.066
Середній час пошуку	0.207	0.208	0.211	0.203	0.229
Максимальний час пошуку	1.100	0.934	1.451	1.451	1.368
Мінімальний час натискання	0.033	0.033	0.007	0.007	0.033
Середній час натискання	0.073	0.071	0.081	0.072	0.077
Максимальний час натискання	0.167	0.133	0.150	0.133	0.167

а) перше тестування користувача №1

Параметр	1 рядок	2 рядок	3 рядок	ліва частина	права частина
Мінімальний час пошуку	0.088	0.080	0.080	0.080	0.088
Середній час пошуку	0.239	0.225	0.228	0.219	0.252
Максимальний час пошуку	1.584	1.504	1.200	1.584	1.504
Мінімальний час натискання	0.048	0.056	0.056	0.048	0.048
Середній час натискання	0.079	0.083	0.090	0.081	0.088
Максимальний час натискання	0.136	0.176	0.152	0.136	0.176

б) тестування користувача №1 через 5 днів

Рис. 9. Узагальнена статистика користувача №1 за рядками та частинами клавіатури

До додаткових результатів включено аналіз впливу такого фактору, як залежність параметрів набору від розташування клавіши у відповідному рядку клавіатури, її лівій та правій частині. З аналізу рис. 9 бачимо, що додаткові статистичні показники не дуже відтворились. Користувач – правша і користується десятипальцевим методом. Оскільки користувач правша, то, згідно з цим, показники мали б бути меншими на правій частині клавіатури. Тим не менш, бачимо, що виявлені показники таке правило порушують, більш стійкими до змін є максимальні показники. Мінімальний час пошуку в першому випадку дуже трохи менший на правій стороні, в другому випадку – навпаки. Середній час пошуку менший в лівій частині в обох випадках. Максимальний час пошуку трохи менше в правій частині в обох випадках. Мінімальний час натискання в першому випадку дуже відрізняються й менший в лівій стороні, в другому випадку – він однаковий. Середній час натискання в обох випадках трохи менше в лівій стороні. Максимальний час натискання у двох випадках менший в лівій стороні клавіатури. Таким чином, фактор «правша-лівша» не є значимим для використання при ідентифікації.

На рис. 10 досліджена залежність між кількістю символів в слові та часом набору слова (сума часу натиснення й пошуку всіх клавіш, які треба набрати для слова).

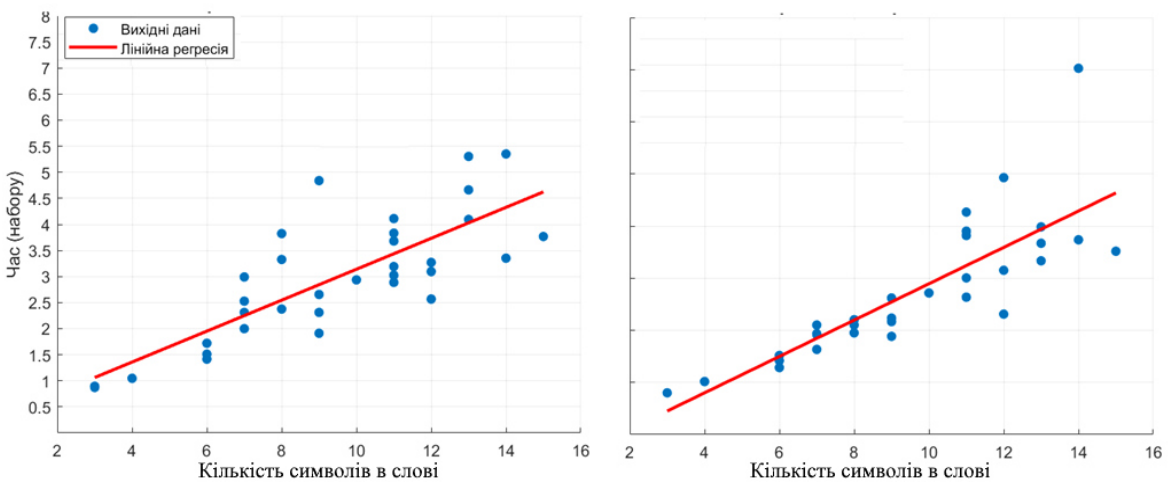

Рис. 10. Узагальнена статистика користувача №1 за залежністю часу набору слово від кількості символів в слові

Бачимо, що закономірність має майже лінійний характер: більше символів – більший час. У випадку, коли присутнє в тестовій фразі декілька різних слів однієї довжини, то вони набираються з різною швидкістю. І хоча в цілому нерівномірність набору слів однакової довжини за часом зберігається, конкретне відхилення може відрізнятись значимо.

У цілому, приходимо до висновку, що час пошуку та час натискання, якщо розглядати їх в інтервалі за різними літерами, дозволяють з високою точністю ідентифікувати стиль набору користувача.

Висновки. Розробка системи біометричної ідентифікації на основі набору тексту, введеного користувачем на клавіатурі, є привабливим рішенням, оскільки не лише потенційно забезпечує надійну ідентифікацію, а й не потребує витрат на спеціальне обладнання. Програмний застосунок, який був розроблений у рамках цієї роботи, дозволяє ефективно проводити процедуру біометричної ідентифікації на основі стилю набору на клавіатурі. Успішне проведення обмежених експериментів з використанням розробленого застосунку підтверджує його ефективність та надійність. Результати експериментів показали ефективність запропонованого критерію, пов’язаного з попаданням при наборі літери в інтервал пошуку та натиснення клавіш еталонного профілю.

Список літератури

1. Giot R, Dorizzi B, Rosenberger C. Analysis of template update strategies for keystroke dynamics. *IEEE Workshop on Computational Intelligence in Biometrics and Identity Management (CIBIM '11)*. 2011. P. 21–28.
2. Maisuria L. K, Soon O. C, Kin L. W. Comparison of artificial neural networks and cluster analysis for typing biometrics authentication. *International Joint Conference on Neural Networks.* 1999; P. 3295–3299.
3. Kang P, Hwang SS, Cho S. Continual retraining of keystroke dynamics based authenticator. *Advances in Biometrics, Proceedings.* 2007. V. 4642. P. 1203–1211.
4. Giot R, Dorizzi B, Rosenberger C. Analysis of template update strategies for keystroke dynamics. *IEEE Workshop on Computational Intelligence in Biometrics and Identity Management*. 2011. P. 21–28.
5. Keystroke Deanonymization. URL: https://www.whonix.org/wiki/Keystroke_ Deanonymization
6. Nonaka H, Kurihara M. Sensing pressure for authentication system using keystroke dynamics. *International Conference on Computational Intelligence*. *Istanbul, Turkey*. 2004. P. 19–22.
7. Loy C.C, Lai W.K, Lim C.P. The development of a pressure-based typing biometrics user authentication system. *ASEAN Virtual Instrumentation Applications Contest Submission. National Instruments. Austin., Tex. USA*. 2005.
8. Alshehri A., Coenen F., Bollegala D. Keyboard Usage Authentication Using Time Series Analysis. *18th International Conference on Big Data Analytics and Knowledge Discovery*. 2016. DOI: 10.1007/978-3-319-43946-4
9. Shadman R., Wahab A. A., Manno M., Lukaszewski M., Daqing H. F. H. Keystroke Dynamics: Concepts, Techniques, and Applications. 2303.04605. URL: https://arxiv.org/abs/2303.04605

DEVELOPMENT OF A SOFTWARE APPLICATION FOR BIOMETRIC IDENTIFICATION BY KEYBOARD TYPING STYLE

D.M.Slabenko[1], O.A. Stopakevych[1], A.A.Stopakevych[2]

[1]National Odesa Polytechnic University,
1, Shevchenko Ave, Odesa, 65044, Ukraine
email: stopakevich@op.edu.ua
[2]State University of Intellectual Technologies and Telecommunications
1, Kuznechna, Odesa, 65029, Ukraine
email: stopakevich@gmail.com

This article describes the development of software to identify a user based on his or her keystrokes. Biometric authentication, particularly methods based on behavioral characteristics, is becoming increasingly popular. This is due to its ability to provide security without the need to remember passwords. The development of a biometric identification system based on the typing of a text entered by the user on the keyboard is an attractive solution, as it not only has the potential to provide a fairly reliable identification, but also does not require significant costs for special equipment. This paper discusses parameters that affect typing, including keystroke time, typing speed, error rates, and others that depend on individual user characteristics. The paper also discusses the drawbacks of biometric identification, particularly the impact of external factors such as fatigue or distraction on authentication accuracy. Based on the analysis of parameters, known methods and approaches to biometric identification by keystroke style, a new verification algorithm is proposed. It is based on the analysis of the time intervals between keystrokes and key searches, which allows to determine the correspondence between the stored user profile and the current keystroke. In order to reduce the influence of external factors, the use of a fixed text of at least 300 characters is recommended. The effectiveness of the developed software based on the proposed identification algorithm is confirmed by the results of the experiments conducted with the developed software application. The results show sufficient reliability and accuracy in the identification process. The work is of practical importance for the development of new security methods in information technology and opens new opportunities for the implementation of biometric systems in various fields.

Keywords*:* biometrics, behavioral, identification, authorization, keyboard, typing, user, software, profile, method, algorithm.