Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)
Комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр
(назва освітнього ступеня)

на тему: **Методи та засоби динамічної аутентифікації користувачів на основі моделі клавіатурного почерку користувача**

Виконав(ла): студент(ка) 6 курсу, групи САм-61
спеціальності 124 «Системний аналіз»
(шифр і назва спеціальності)

(підпис) Багрій О.О. (прізвище та ініціали)

Керівник (підпис) Матійчук Л.П. (прізвище та ініціали)
Нормоконтроль (підпис) Мацюк О. В. (прізвище та ініціали)
Завідувач кафедри (підпис) Боднарчук І. О. (прізвище та ініціали)
Рецензент (підпис) Жаровський Р.О. (прізвище та ініціали)

Тернопіль
2022

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет Комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра Комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Боднарчук І.О.
(підпис) (прізвище та ініціали)
« » 20___ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр
(назва освітнього ступеня)

за спеціальністю 124 «Системний аналіз»
(шифр і назва спеціальності)

студенту Багрію Олександру Олеговичу
(прізвище, ім'я, по батькові)

1. Тема роботи Методи та засоби динамічної аутентифікації користувачів на основі моделі клавіатурного почерку користувача

Керівник роботи Матійчук Любомир Павлович, к.е.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «*22*» *листопада* 2022 року № *4/7-947*.

2. Термін подання студентом завершеної роботи

3. Вихідні дані до роботи Наукові публікації, електронні ресурси, підручники , посібники з тематики дослідження, щодо методів та засобів динамічної аутентифікації користувачів на основі моделі клавіатурного почерку користувача

4. Зміст роботи (перелік питань, які потрібно розробити) Вступ. 1. Аналіз методів та засобів динамічної аутентифікації користувачів. 2. Математичне забезпечення динамічної аутентифікації користувачів на основі моделі клавіатурного почерку користувача. 3. Програмне забезпечення динамічної аутентифікації користувачів на основі моделі клавіатурного почерку користувача . 4. Охорона праці та безпека в надзвичайних ситуаціях. Висновки.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Актуальність. 2. Аналіз відимих рішень. 3. Підготовка даних. 4. Модель клавіатурного почерку користувача. 5. Графік log(1+) що демонструє якість розпізнавання користувача за динамікою його роботи з клавіатурою. 6. Архітектура системи. 7. Вміст .csv-файлу та json-файлу, в який записується зібрана інформація про взаємодію користувача з клавіатурою комп'ютера. 8. Реалізація системи. 9. Графік схожості даних тестованого користувача з легітимним. 10. Висновки. 14. Завершальний слайд.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Мацюк Олександр Васильович доцент кафедри КН		
Безпека в надзвичайних ситуаціях	Клепчик Василь Михайлович		

7. Дата видачі завдання 14 листопада 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	14.11.2022-15.11.2022	*Виконано*
2.	Підбір наукових джерел щодо виявлення динамічної	16.11.2022-20.11.2022	*Виконано*
	аутентифікації користувачів на основі моделі		
	клавіатурного почерку користувача		
3.	Переклад та опрацювання наукових джерел щодо	21.11.2022-23.11.2022	*Виконано*
	виявлення динамічної аутентифікації користувачів		
4.	Виконання дослідження щодо виявлення	24.11.2022-27.11.2022	*Виконано*
	клавіатурного почерку користувача		
	аутентифікації користувачів на основі моделі		
5.	Оформлення розділу «Аналіз методів та засобів	28.11.2022-30.11.2022	*Виконано*
	динамічної аутентифікації користувачів»		
6.	Оформлення розділу « Математичне забезпечення	01.12.2022-04.12.2022	*Виконано*
	аутентифікації користувачів на основі моделі		
	клавіатурного почерку користувача»		
7.	Оформлення розділу « Програмне забезпечення	05.12.2022-07.12.2022	*Виконано*
	аутентифікації користувачів на основі моделі		
	клавіатурного почерку користувача»		
8.	Виконання завдання до підрозділу «Охорона праці»	08.12.2022-09.12.2022	*Виконано*
9.	Виконання завдання до підрозділу «Безпека в	10.12.2022-11.12.2022	*Виконано*
	надзвичайних ситуаціях»		
10.	Оформлення кваліфікаційної роботи	12.12.2022-13.12.2022	*Виконано*
11.	Нормоконтроль	14.12.2022-15.12.2022	*Виконано*
12.	Перевірка на плагіат	15.12.2022	*Виконано*
13.	Попередній захист кваліфікаційної роботи	16.12.2022	*Виконано*
14.	Захист кваліфікаційної роботи	20.12.2022	

Студент _____ Багрій О.О.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Матійчук Л.П.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Методи та засоби динамічної аутентифікації користувачів на основі моделі клавіатурного почерку користувача // Кваліфікаційна робота освітнього рівня «Магістр» // Багрій Олександр Олегович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група САм-61 // Тернопіль, 2022 // с. 82, рис. – 25, табл. – 1,бібліогр. – 52, додат. – 6.

Ключові слова: аутентифікація користувачів, моделі клавіатурного почерку користувача, метод ESFC.

У кваліфікаційній роботі запропоновано підхід до підготовки даних, що описують клавіатурний почерк користувача, що включає спосіб побудови ознакового простору та підхід до подальшої обробки ознак на основі дискретизації їх за квантилями. Розроблено нечіткий метод виявлення аномалій у даних на основі еліптичної кластеризації (ESFC) у RKHS, що будує у просторі високої розмірності еліптичні області з оптимальним центром для виявлення аномалій.

За результатами експериментів метод ESFC перевершив якість розпізнавання існуючих алгоритмів при класифікації як окремих векторів ознак, і цілих сесій роботи користувачів за комп'ютером.

Розроблено архітектуру, реалізовано та апробовано експериментальний зразок мультиагентного програмного комплексу, який використовує запропонований комплекс алгоритмів для виявлення аномального поведінки користувачів щодо особливостей роботи з клавіатурою комп'ютера.

ANNOTATION

Methods and Means of Users Dynamic Authentication Based on a User Keyboard Handwriting Model// Bagriy Oleksandr Olegovich // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Sciences, group SAM-61 // Ternopil, 2022 // p. 82, Fig. - 25, table. - 1, bibliogr. - 52, add. - 6.

Keywords: user authentication, user handwriting models, ESFC method.

In the qualification work, an approach to the preparation of data describing the user's keyboard handwriting is proposed, which includes a method of constructing a feature space and an approach to further processing of features based on their discretization by quantiles. A fuzzy data anomaly detection method based on elliptic clustering (ESFC) is developed in RKHS, which constructs elliptic regions with an optimal center for anomaly detection in a high-dimensional space.

According to the results of the experiments, the ESFC method surpassed the recognition quality of the existing algorithms when classifying both individual feature vectors and entire sessions of user work at the computer.

An experimental model of a multi-agent software complex was developed, implemented and tested, which uses the proposed set of algorithms to detect abnormal user behavior regarding the features of working with a computer keyboard.

ЗМІСТ

ВСТУП

Актуальність теми. Останні кілька десятиліть ознаменували період стрімкого розвитку інформаційних технологій та їх впровадження у житті людей. Вся важлива інформація тепер зберігається в комп'ютерах користувачів, і питання, пов'язані із забезпеченням безпеки даної інформації стає найбільш критичним. Під інформаційною безпекою розуміється комплекс заходів, спрямований на забезпечення конфіденційності, цілісності та доступності інформації. При цьому, найбільш важливим аспектом є забезпечення конфіденційності – запобігання витоку інформації, оскільки у разі втрати ризику піддаються й інші фактори. Сценаріїв внутрішніх та зовнішніх вторгнень на комп’ютери користувачів розвиваються з кожним днем і необхідно оперативно вміти з ними боротися – розробляти системи,здатні вчасно запобігати спробам порушення конфіденційності інформації.

Одним із ключових завдань при забезпеченні конфіденційності інформації є задача аутентифікації, або перевірки власності суб’єкту доступу пред'явленого ним ідентифікатора [1,2]. Як суб'єкт доступу розглядається особа або одиниця ресурсу автоматизованої інформаційної системи, дії якої за доступом до ресурсів системи регламентуються правилами розмежування доступу, а його ідентифікатором називається унікальна ознака об'єкта, що дозволяє відрізняти його з інших об'єктів.

Залежновідпринципуздійсненняпроцедуриперевіркиідентифікатора,под іляютьстатичнутадинамічнуавтентифікацію. Статична автентифікація полягає в епізодичній перевірці ідентифікатора користувача. Системи статичної автентифікації мають досить просту реалізацію. Проте, найчастіше особистість користувача перевіряється лише при вході в систему та

подальшу зміну користувачеві стежити неможливо. Динамічна автентифікація вирішує цю проблему: особистість користувача перевіряється постійно протягом всієї сесії роботи користувача за комп’ютером. Тим самим зловмисник не зможе заволодіти комп'ютером після того, як легітимний користувач (користувач, для якого була побудована модель) відійде від свого робочого місця, забувши вийти з системи. Проте, системи динамічної автентифікації споживають більше кількість програмно-апаратних ресурсів комп'ютера.

Таким чином, існуючі системи аутентифікації користувачів досить вразливі і схильні до високого ризику вторгнень. Їх основними недоліками є неможливість проконтролювати факт зміни того, хто увійшов до систем користувача, а також ненадійність використання паролів та магнітних карток для входу до системи. Тому необхідно розробити систему, що дозволяє безперервно перевіряти особу працюючого за комп'ютером користувача (тим самим здатну своєчасно виявляти різні спроби вторгнень) та використовуючи у своїй роботі надійний ідентифікатор. Можливим шляхом вирішення цієї проблеми є запровадження динамічної аутентифікації користувача під час його взаємодії зі стандартними пристроями введення, зокрема на основі динаміки його роботи із комп'ютерною клавіатурою.

Зв'язок роботи з науковими програмами, планами, темами

Напрямок виконаних досліджень безпосередньо пов'язаний з науково-дослідним напрямком кафедри комп’ютерних наук.

Мета і задачі дослідження

Мета даної роботи є розробка математичного та програмного забезпечення динамічної аутентифікації користувачів комп’ютерів на основі аналізу їхнього клавіатурного почерку.

Для досягнення цілей в дослідженні поставлені та вирішені наступні завдання:

1) проаналізувати відомі методи та засоби аутентифікації

користувачів комп’ютерів

2) розробити алгоритми попередньої обробки даних, що характеризують динаміку роботи користувачів з клавіатурою комп'ютера.

3) розробити методи побудови моделі користувача, що дозволяють досягти високої якості розпізнавання;

4) реалізувати програмне забезпечення, яке виконує збір поведінкової інформації, побудову та застосування індивідуальних моделей поведінки користувачів на основі розробленого комплексу алгоритмів для виявлення аномалій у поведінці користувачів.

Об’єкт дослідження – процеси динамічної аутентифікації користувачів.

Предмет дослідження – виступають методи та програмні засоби динамічної аутентифікації користувачів.

Методи дослідження.

Дослідження проводилися на базі методів теорії ймовірностей, математичної статистики та теорії машинного навчання.

Наукова новизна одержаних результатів. Запропоновано нечіткий метод виявлення аномалій у даних на основі еліптичної кластеризації, що будує у просторі високої розмірності еліптичні області з оптимальним центром для виявлення аномалій.

Практичне значення одержаних результатів полягає у розробці та реалізації програмного забезпечення виявлення аномальної поведінки користувачів на основі аналізу їхнього клавіатурного почерку.

РОЗДІЛ 1
АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ДИНАМІЧНОЇ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ

1.1. Аналіз засобів динамічної аутентифікації користувачів на основі клавіатурного почерку

На сьогоднішній день проблема динамічної аутентифікації користувачів та динаміки їх роботи з клавіатурою персонального комп'ютера (ноутбука) є актуальною. Як і для будь-якої задачі одно класової класифікації, вирішення даної задачі складається з наступних етапів:

1) підготовка даних;

2) визначення та розрахунок набору ознак, які будуть використовуватися при аутентифікації користувача;

3) застосування методів обробки ознак;

4) застосування методів зменшення розмірності даних;

5) застосування методів побудови моделі користувача.

Розглянемо сучасні технічні рішення, методи та досягнуті результати з динамічної аутентифікації користувачів на основі динаміки їхньої роботи з клавіатурою персонального комп'ютера (ноутбука) більш детально.

На сьогоднішній день кількість рішень щодо аутентифікації користувачів на основі динаміки їхньої роботи з клавіатурою персонального комп'ютера постійно збільшується. І якщо раніше дані системи обмежувалися лише аналізом введення пари логін/пароль (розглядалася виключно статична аутентифікація), то зараз активно розвиваються системи, здатні аналізувати поведінку користувача за комп'ютером безперервно.

BehavioWeb (рис. 1.1). Одним з найбільш відомих комерційних рішень у галузі безперервної фонової аутентифікації користувачів за клавіатурним почерком є продукт BehavioWeb компанії BehavioSec [3]. Для аналізу поведінки користувача у ньому використовуються ритм та швидкість набору тексту, а також сила натискання на клавіші.

Програмне забезпечення вбудовується у веб-сайт або додаток. Для цього використовуються JavaScript-бібліотека, що поставляється, а також J2EE-модуль, що вбудовується bweb-сервер для здійснення процедури автентифікації. Розробники звертають увагу на те, що їхнє рішення аналізує зміну характеристик введення користувача з часом і періодично оновлює модель користувача. Проте, алгоритми, що використовуються для цього, не називаються.

Рисунок 1.1 - Система BehavioWeb

KeyTrac (рисунок 1.2). Не менш популярним рішенням є продукт KeyTrac [4, 5], що дозволяє здійснювати у фоновому режимі як

автентифікацію, так і ідентифікацію користувачів комп'ютера, ґрунтуючись на динаміці їх клавіатурного введення. Дані користувачів (тривалість натискання на клавіші клавіатури, а також тривалості перескоку між клавішами) записуються за допомогою компонента KeyTrac Recorder і відправляються на сервер компанії, де відбувається їх порівняння із побудованою раніше моделлю. При цьому побудова моделі користувача здатна здійснюватися на будь-якому довільному тексті, а не тільки при багаторазовому введенні тих самих фраз. Для передачі даних використовується наданий KeyTrac API. Далі сервер повертає свій вердикт у вигляді булевої величини true/false – чи відповідають надіслані тестові дані розглянутому легітимного профілю чи ні. Для вбудовування цього рішення на веб-сайт також пропонується використовувати JavaScript-бібліотеку, що надається. Розробники системи стверджують, що їх рішення нечутливе до зміни мови, що використовується, а також до зміни обладнання, що використовується. Використовувані для цього алгоритми, а також методи побудови моделі та їх подальшої класифікації не називаються.

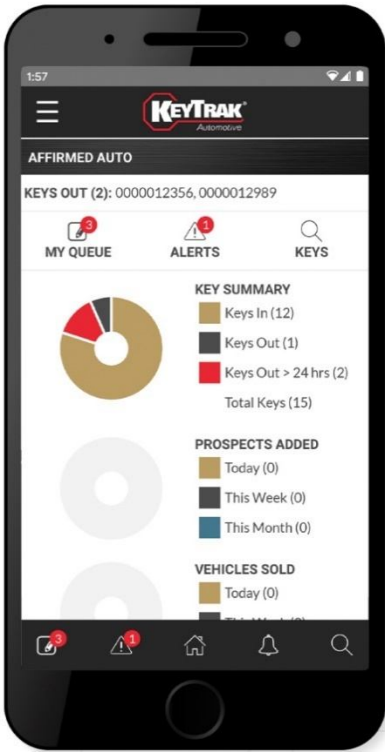


Рисунок 1.2 - Система KeyTrac

Оскільки всі дані динаміки роботи користувачів з клавіатурою анонімуються, розробники рекомендують використовувати свій продукт у тому числі й у додатках електронної комерції.

KeystrokeID (рис. 1.3). Іншим відомим рішенням, що здійснює безперервний аналіз динаміки роботи користувачів з клавіатурою комп'ютера, є продукт KeystrokeID компанії ID Control [5]. Для подальшого аналізу тут використовуються такі характеристики набору, як проміжки часу між натисканням та відпусканням однієї клавіші та проміжки часу між двома послідовними натисканнями різних кнопок. Збір даних, що характеризують динаміку роботи користувачів із клавіатурою, реалізується за допомогою Java-аплету. Спочатку модель навчається при введенні користувачем логіну та пароля, далі відбувається до навчання на даних фонової роботи користувача.

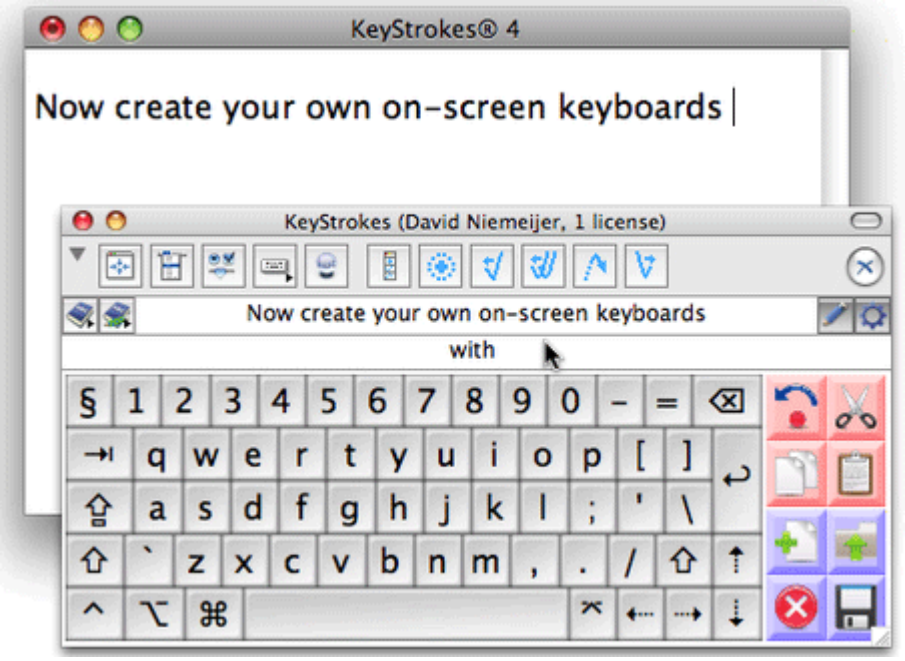


Рисунок 1.3 - Система KeystrokeID

Використовувані у своїй програмі алгоритми не розкриваються. Цей продукт простий у використані та використовується для захисту комп'ютерів користувачів від мережевих атак.

Scout Analytics (рис. 1.4). Цікавим рішенням є продукт компанії Scout Analytics [6,7], що використовується для коректної автентифікації користувачів системи зберігання та перегляду електронних публікацій. Він дозволяє запобігти написанню відгуків по публікаціях різними людьми з одного облікового запису, тим самим допомагаючи скласти більш чесну та об'єктивну оцінку публікацій. Алгоритми, що використовуються в даному рішенні, запатентовані. Для аналізу використовуються код натиснутої клавіші, тип події (натискання / відтискання), а також час події, що відбулася.

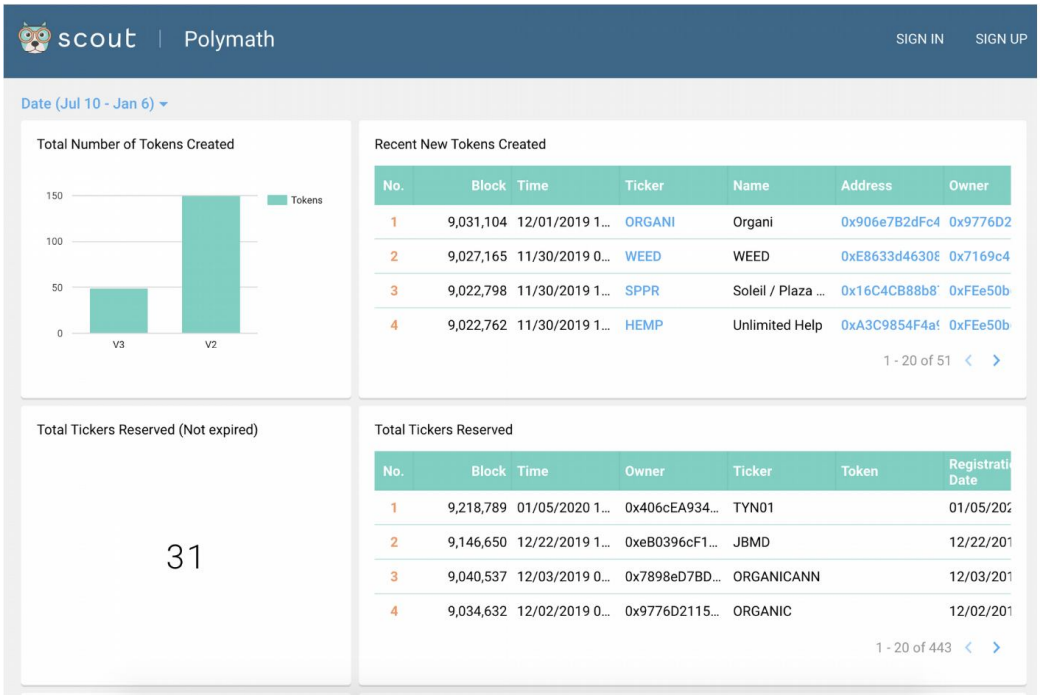


Рисунок 1.4 - Система Scout Analytics

Далі, як розрахункові ознаки виступають тривалості натискань на окремі клавіші клавіатури, тривалості перескоків при натисканні двох або трьох клавіш, а також різні статистики від даних величин (середнє значення, медіана, середньоквадратичне відхилення, максимум, мінімум тощо). Як класифікатор використовується нейронна мережа. Модель поведінки користувача будується виходячи з його поведінки за останні 60 днів (тим самим, щодня відбувається оновлення моделі). Також для додаткового

статистичного аналізу використовується інформація про пристрій, з якого було здійснено вхід на сайт, і IP-адресу, що використовується при цьому.

TypingDNA(рис. 1.5). Одним з останніх рішень, що з'явилися на ринку, в області безперервної аутентифікації користувачів на основі клавіатурного почерку. Розробники пропонують використовувати дане рішення у навчальному процесі: протягом семестру студенти працюють за комп'ютерами під час занять час як у фоновому режимі відбувається збір даних динаміки їхньої роботи з клавіатурою комп'ютера та навчання персональних моделей на цих даних. Під час контрольних заходів також проводяться збір подій, що прийшли від клавіатури і зіставлення цих даних із побудованими раніше моделями. Таким чином, вдається виявити несумлінних студентів, які виконують контрольні роботи не самостійно.

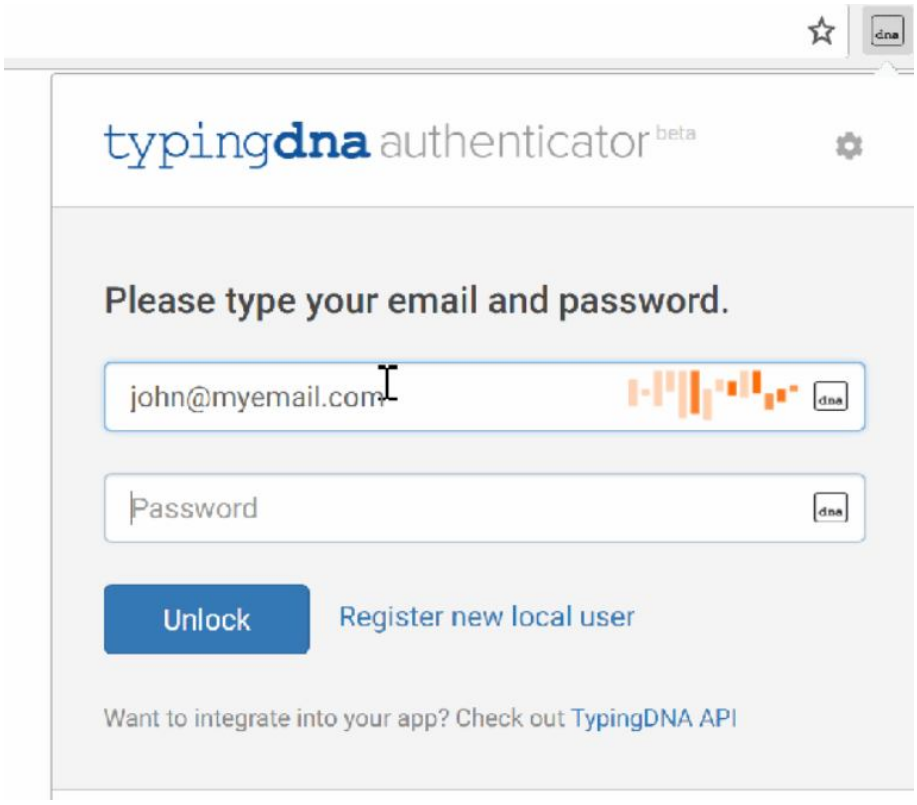


Рисунок 1.5 - Система TypingDNA

Друга сфера застосування даного продукту – захист клієнтів он-лайн банківських систем від несанкціонованого доступу сторонніх осіб до їх облікових записів. Як характеристичні ознаки виступають тривалості

натискань, а також тривалості перескоків для найбільш часто використовуваних користувачем клавіш. Розробники гарантують високу якість роботи системи за умови, що для побудови моделі користувачем було надруковано на клавіатурі без тривалих пауз щонайменше 100 символів. Збір даних здійснюється за допомогою Java-аплету. Розробники надають TypingDNA API та заявляють, що це рішення сумісне з більшістю сучасних мов програмування. Алгоритми машинного навчання, що використовуються в даному продукті не називаються.

KeystrokeDNA(рис. 1.6). Також набирає популярності комерційне рішення KeystrokeDNA, розробники якого дозволяють вбудовувати його в будь-який web-додаток і радять використовувати цей продукт для захисту комп'ютерів користувачів від мережевих вторгнень. При цьому, для аналізу поведінки користувачів використовуються ритми та швидкість набору тексту. Розробники цього рішення заявляють про стабільність його роботи як при зміні мови введення, так і при зміні клавіатури, що використовується. Для установки та вбудовування KeystrokeDNA користувачам надається KeystrokeDNAAPI, а також власна JavaScript-бібліотека та докладна інструкція щодо їх використання. Більше ніякої інформації про цей продукт у вільному доступі не надається.

Зазначимо, що крім рішень, заснованих лише на безперервній аутентифікації користувачів по динаміці їхньої роботи з клавіатурою персонального комп'ютера (ноутбука), на ринку також поширені комерційні продукти, що здійснюють багатофакторну динамічну автентифікацію користувачів.

Дані рішення, крім клавіатурного почерку, також аналізують динаміку роботи користувачів з мишею, з файловою системою та інші різні біометричні показники. Розробники даних продуктів (які аналізують лише клавіатурний почерк, так і рішень, заснованих на багатофакторній

аутентифікації) заявляють, що точність аутентифікації їх рішень досягає 90%.

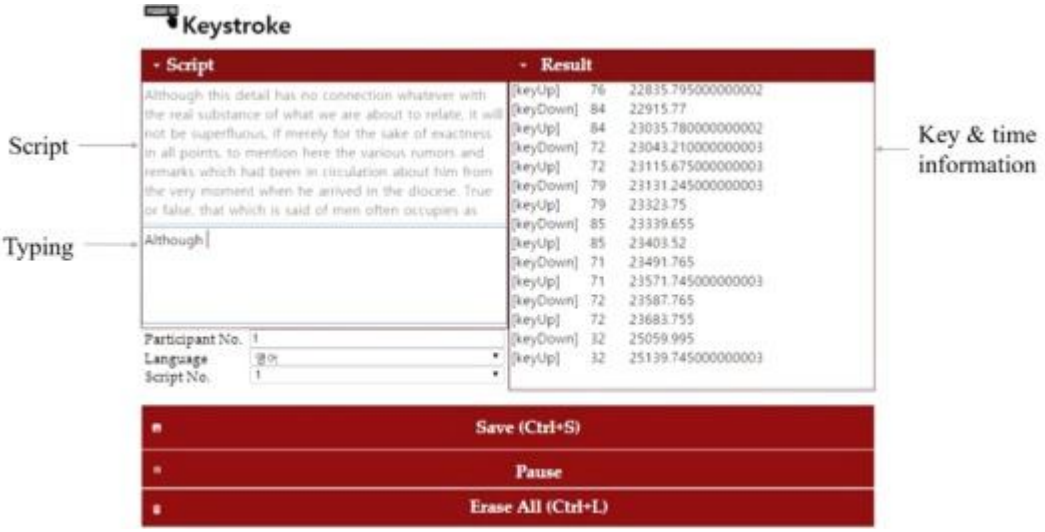


Рисунок 1.6 - Система KeystrokeDNA

Розглянемо найбільш відомі рішення щодо безперервної багатофакторної автентифікації користувачів, які використовують як одну зі своїх аналізованих характеристик клавіатурний почерк користувачів.

BioCatch (рис. 1.7). Одним із рішень, що здійснюють динамічну багатофакторну аутентифікацію користувачів, є широко відомий продукт Biocatch, аналізуючий у режимі он-лайн динаміку роботи користувачів з клавіатурою, мишею та web-ресурсами. При аналізі клавіатурного почерку, також це рішення дозволяє визначати, правшою або лівшою є розглянутий користувач, обчислює розмір його руки та також враховує ці характеристики для визначення легітимності користувача. Усього це рішення обчислює близько 2000 характеристичних ознак, у тому числі алгоритмами машинного навчання відбираються 20 найзначніших. Але які саме це ознаки та що за алгоритми використовуються для їхнього відбору – розробники замовчують. Цей продукт пропонується використовувати для захисту веб-програм від несанкціонованого доступу, а також для захисту користувачів он-лайн банківських систем від доступу зловмисників до їхніх рахунків.

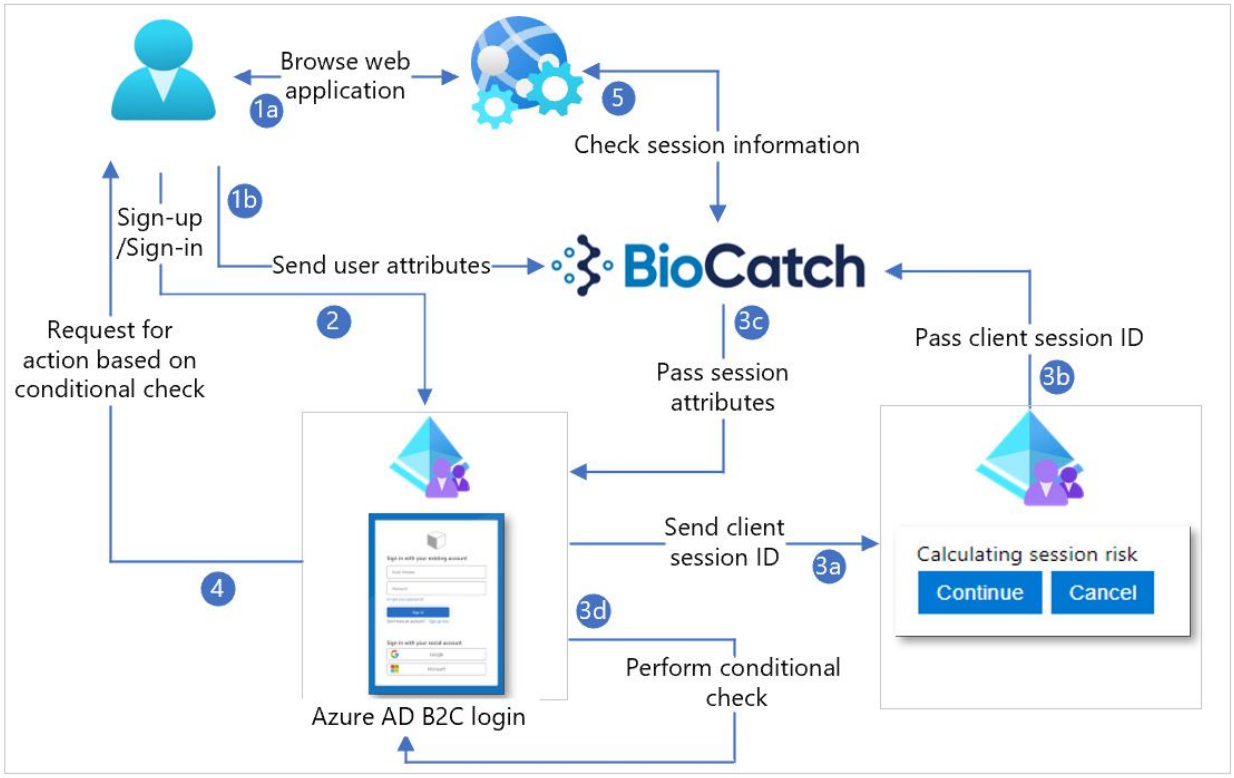


Рисунок 1.7 - Система BioCatch

Аналітикам надається широкий інструмент моніторингу, що відображає різні графіки можливі ризики та аномалії, а також інформативні звіти щодо роботи системи.

BioTracker (рис.1.8). Одним з найбільш відомих рішень щодо безперервної багатофакторної аутентифікації користувачів є продукт BioTracker компанії Plurilock [8,9]. Для того, щоб з високою точністю автентифікувати користувача, програмі необхідно навчитися близько 30 хвилинах його безперервної роботи з клавіатурою персонального комп'ютера. Додатково для покращення якості збудованої моделі також аналізується робота користувача з мишею. Слід зазначити, що BioTracker періодично перебудовує модель користувача, що дозволяє враховувати зміну динаміки його роботи з клавіатурою та мишею протягом тривалого часу.

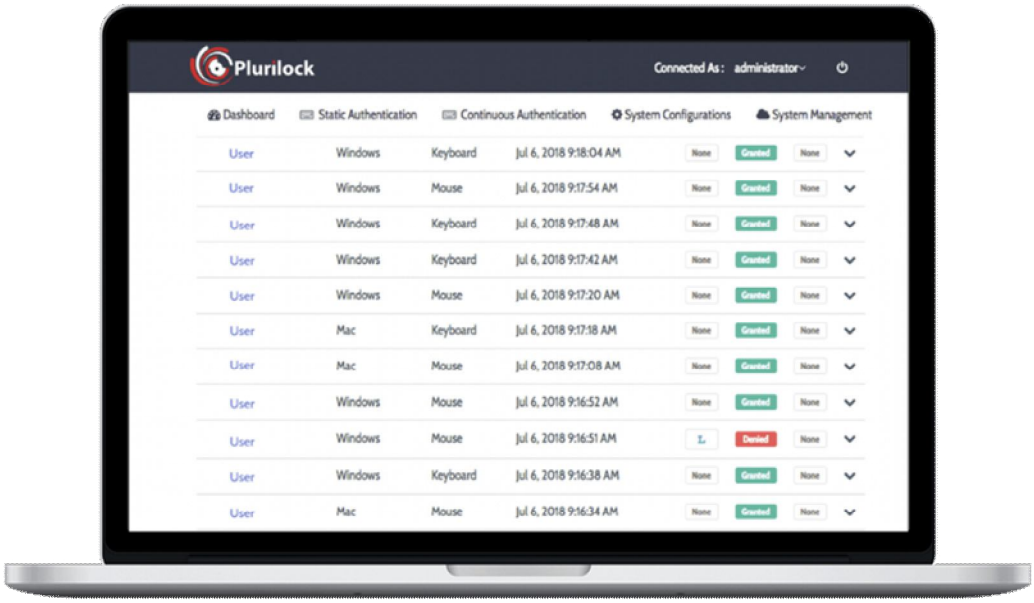


Рисунок 1.8 - Система BioTracker

Розробники рекомендують використовувати цей продукт як для захисту персональних комп'ютерів від мережевих атак, так і в додатках електронної комерції та державних структурах – BioTracker може працювати і як самостійне рішення, так і вбудовуватися в готові програми користувачів. Продукт підтримується усіма найпопулярнішими сьогодні операційними системами (Windows, Linux, Mac OS). Алгоритми, використовувані у цьому рішенні, не називаються.

CVMetrics, Tickstream (Intensity Analytics) (рис. 1.9). Продукти CVMetrics та Tickstream компанії Intensity Analytics також широко використовуються для безперервної автентифікації користувачів по динаміці їх роботи з клавіатурою та мишею. Дані рішення знайшли своє застосування у сфері електронної комерції, енергетики, судово-медичної експертизи, державних структурах – місцях, де захист комп'ютерів від сторонніх вторгнень є одним із найважливіших завдань.

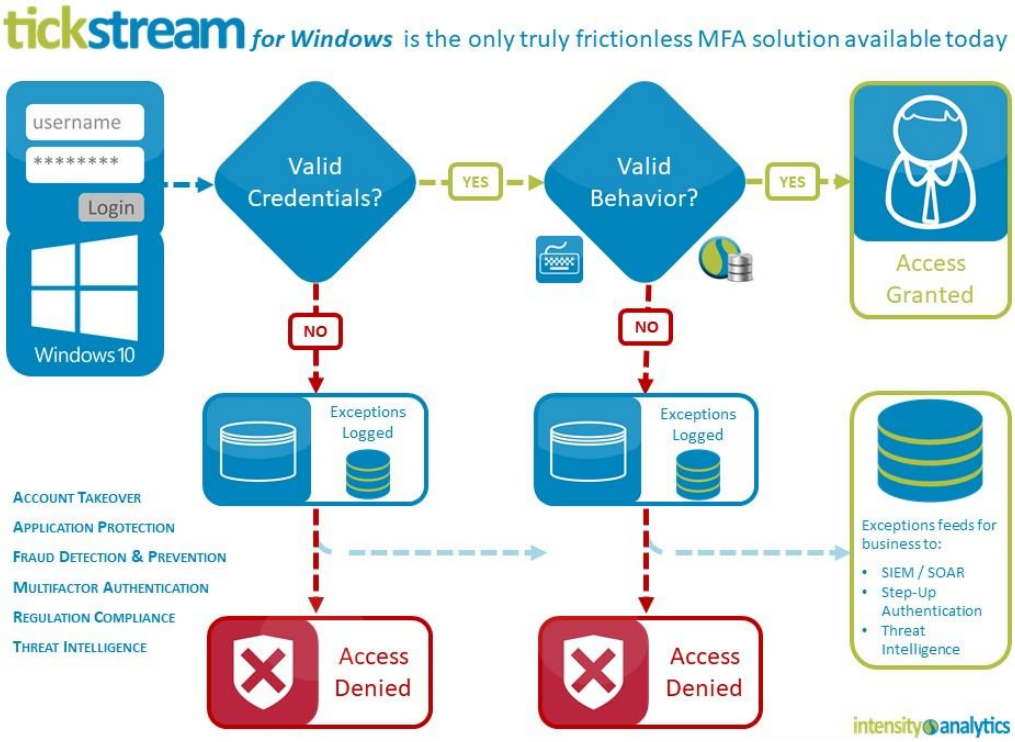


Рисунок 1.9 - Система Tickstream

Крім аналізу взаємодії користувачів з мишею та клавіатурою, продукт Tickstream також додатково аналізує текстові документи, з якими працює користувач. Для аналізу клавіатурного почерку, і CVMetrics, і Tickstream для кожної натиснутої клавіші записують її номер, відповідну подію (натискання / відтискання), а також тимчасову мітку, коли ця подія відбулася.

Розробники зауважують, що основним у їхній методиці є саме аналіз динаміки роботи користувачів з клавіатурою, тому що, на відміну від динаміки роботи з мишею, клавіатурний почерк є стабільнішою характеристикою. Механізм аутентифікації заснований на обчисленні різних статистик, що визначають рівень схожості тестованих даних із даними легітимного профілю. Проте, які сааме статистики обчислюються при цьому у цих продуктах – розробники замовчують.

Symantec VIP (рис. 1.10). Висока якість роботи також демонструє продукт Symantec VIP [12], що пропонує свої рішення як для захисту домашніх комп'ютерів користувачів від несанкціонованого доступу, так і для

захисту систем електронної комерції від мережевих вторгнень. Динамічна багатофакторна автентифікація користувача здійснюється на основі аналізу динаміки його роботи з клавіатурою та мишею.

Процедура аутентифікації проводиться через кожні 125 клавіатурних натискань, при цьому характеристики взаємодії користувача з мишею використовуються для уточнення побудованої моделі. Для ухвалення рішення про легітимність розглянутого користувача використовується статистичний підхід. У разі виявлення зловмисника відбувається блокування системи, також система може сфотографувати зловмисника. Symantec VIP підтримується операційними системами Windows та Mac OS. У разі вбудовування Symantec VIP у web-сайт, він буде коректно працювати у всіх популярних сьогодні браузерах: Internet Explorer, Firefox,Chrome, Safari. Конкретні алгоритми, які використовуються в даному рішенні, розробниками не називаються.

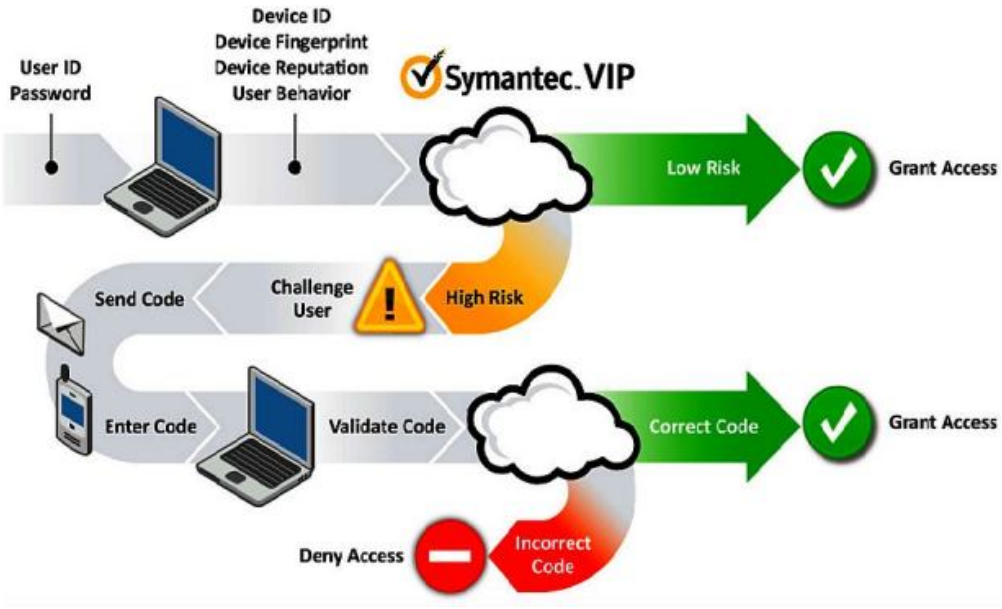


Рисунок 1.10 - Система Symantec VIP

Механізми динамічної аутентифікації користувачів використовуватись для запобігання несанкціонованому доступу до web-сайтів (уданому випадку

буде здійснюватися web-збір даних), так і вбудовуватися в додатки користувачів або використовуватися як самостійні рішення (при цьому дані будуть збиратися за допомогою локального збирача). Для цього розробниками надаються зручні у використанні JavaScript-бібліотеки, J2EE-модулі, Java аплети, а також власне розроблені компаніями API та SDK. В якості збираних характеристик виступають коди натиснутих клавіш, подій (натискання / відтискання), а також відповідні тимчасові мітки. Для подальшого аналізу у комерційних рішеннях обчислюються тривалості натискання на ці клавіші, а також тривалості перескоків між клавішами. Вряд рішень дані характеристики обчислюються не для всіх, а тільки для найбільш часто використовуваних користувачем клавіш. У деяких продуктах додатково аналізується швидкість набору тексту. Використані алгоритми зменшення розмірності ознакового простору, а також алгоритми машинного навчання розробниками комерційних систем не розкриваються. У середньому для побудови високо якісної моделі, комерційним рішенням потрібно навчитися приблизно на 15–20 хвилин безперервної роботи користувача з клавіатурою комп'ютера.

Розробники зазначають, що згодом модель користувача застаріває, тому її періодично необхідно перебудовувати. Також варто звернути увагу на те, що при зміні мови введення, що використовується, а також при зміні використовуваної апаратури якість розпізнавання може падати, що також необхідно враховувати. У деяких продуктах системним адміністраторам надаються багатофункціональні інструменти моніторингу активності користувачів у режимі он-лайн, що включають перегляд різних графіків ризиків та аномалій, автоматичне складання звітів, а також надається можливість гнучкого налаштування та швидкого коригування політик доступу. Розглянуті комерційні рішення підтримуються усіма найбільш поширеними сьогодні браузерами та операційними системами.

1.2. Особливості реалізації збору та опрацювання даних

Оскільки динамічна автентифікація користувачів за клавіатурним почерком може широко застосовуватися і на web-сайтах, і в самостійних локальних додатках, у відомих рішеннях, розглядається як здійснюваний web-браузером web-збір даних, що характеризують динаміку роботи користувачів з клавіатурою, і локальний збір даних засобами операційної системи комп'ютера.

Технології, що використовуються для web-збирання даних, що характеризують динаміку роботи користувача з клавіатурою, можна поділити на дві категорії залежно від використання плагінів – незалежних програмних модулів, динамічно підключаються до web-браузера і призначені для розширення його функціональних можливостей. До програмних засобів, що не використовують плагіни при web-зборі, відносяться вбудовуванні в web-сторінки JavaScript-програми, а також розширення веб-браузера. До програмних засобів, робота яких ґрунтується на використанні у системі спеціальних плагінів, відносяться такі широко відомі технології як Flashта Java-аплети. Взаємодія web-браузера з плагіном здійснюється через API інтерфейс. Серед найпопулярніших API інтерфейсів виділяють NPAPI (Netscape Plugin Application Programming Interface), PPAPI (Pepper Plugin Application Programming Interface) та ActiveX. Проте, поступово Internet Explorer відмовляється від використання NPAPI та ActiveX, а інтерфейс PPAPI підтримується лише web-браузерами Google Chrome та Opera, що свідчить у тому, що немає універсального рішення. Відмітимо, що найчастіше використовується для web-збору та перспективними сьогодні є технологія JavaScript.

JavaScript-програми. Вбудовуванні на web-сторінки JavaScript-програми виконуються на стороні клієнта і тим самим можуть взаємодіяти із

зовнішніми ресурсами, збираючи інформацію про їхнє використання без використання додаткового ПЗ.

Однак, варто мати на увазі, що у різних web-браузерах можуть використовуватись різні версії JavaScript-інтерпретатора, що необхідно враховувати під час розробки. Також, автори говорять про великі затримки при обробці подій натискання на клавіші клавіатури при використанні технологій web-збору, тривалість яких істотно залежить від ступеня завантаженості комп'ютера. Час, що минає від моменту натискання на клавішу до виклику обробника цієї події, може становити від десятків до сотень мілісекунд, що значно нижче швидкості обробки збирачів даних засобами операційної системи – близько двохсотмс. Також варто зазначити, що при використанні JavaScript-технологій в більшості web-браузерів немає можливості визначити, ліві чи праві функціональні клавіші (Shift, Ctrl, Alt і т.д.) були натиснуті, що є серйозним недоліком, оскільки дана інформація може суттєво допомогти автентифікувати користувача.

Розширення для веб-браузера. Розширення для web-браузера є програмами, що розширюють його функціональні можливості. На відміну від розглянутих вище JavaScript програм, що вбудовуються в коди web-сторінок, розширення для web-браузера надають можливість збору даних, що характеризують динаміку роботи користувача з клавіатурою, під час перегляду будь-яких web-сторінок, а не тільки тих, у коди яких заздалегідь вшиті JavaScript дані. Це обґрунтовується здатністю розширень модифікувати код веб-сторінок, що переглядаються. В цьому полягає їхня головна відмінність від плагінів. Однак, варто мати на увазі, що різні браузери надають різні програмні інтерфейси для написання розширень (у тому числі і вимагають реалізації різними мовами програмування), що значно ускладнює створення універсальних рішень.

Flash – це мультимедійна платформа компанії Adobe Systems,призначена для створення інтерактивних web-додатків з багатою

векторною,растровою, тривимірною комп'ютерною графікою та мультимедіа, що працюють як усередині,так і поза веб-браузером. Для створення програм використовується власне розроблена мова ActionScript. Adobe Flash використовується у web-браузерах Opera таGoogle Chrome (за допомогою інтерфейсу PPAPI), а також у веб-браузері Firefox(за допомогою інтерфейсу NPAPI). Варто відзначити, що використання Flash сильно уповільнює роботу браузера, внаслідок чого попит на цей продукт поступово знижується.

Зауважимо, що при використанні технології Flash зібрати дані динаміки роботи користувача з клавіатурою вийде тільки всередині Flash-об'єкта.

Java-аплетє прикладною програмою, найчастіше написаною намові Java у форматі байт-коду та виконуваній у віртуальній машині JVM (JavaVirtual Machine) як усередині web-браузера, так і локально в операційній системі[8]. Дане рішення є крос платформним та підтримується більшістю сучасних браузерів та операційних систем, його основне призначення - надання інтерактивних можливостей web-додатків.

Для вбудовування Java-аплетів у web-браузер використовуються інтерфейси NPAPI та ActiveX. Однак, при використані Java-аплетів не надається можливості збору даних, що прийшли від клавіатури, в рамках усієї перегляданої користувачем web-сторінки: дані можна отримати лише всередині того її об'єкта, в якому даний Java-аплет розташовується.

1.3. Особливості застосування методів, заснованих на аналізі використовуваних користувачем клавіш

Методи, засновані на аналізі використовуваних користувачем клавіш проводять відбір клавіш, що часто використовуються користувачем, і N-грам,для яких далі розраховуються описані вище ознаки.

Кількість клавіш, що відбираються для подальшого аналізу клавіатури, є параметром системи, що підбирається експериментально. Для успішного розпізнавання користувачів достатньо використати характеристичні ознаки для 50 найчастіше використовуваних користувачем диграфів. Оскільки кожен користувач має свій власний набір найчастіше використовуваних клавіш, даний метод дозволяє врахувати більше індивідуальних особливостей користувачів і тим самим підвищити якість аутентифікації.

Метод головних компонентів (Principal Component Analysis, PCA) є одним з найбільш поширених методів скорочення розмірності вхідних даних. Завдання даного методу – пошук підпросторів меншої розмірності,ортогональної проекції на які розкид даних буде максимальним. Таким чином, PCA проектує дані на нову координатну систему меншої розмірності,що визначається власними векторами та власними числами матриці коваріації (зауважимо, що коваріація двох випадкових величин є мірою їхньої лінійної залежності). Після того, як власні вектори та власні числа знайдені,власні числа сортуються в порядку спадання. Завдяки цьому можна отримати компоненти як зменшення їх значимості. Власний вектор,відповідний найбільшому власному числу – найголовніша компонента набору даних. Він висловлює найважливіші відносини між координатами. Тому основні компоненти виходять множенням рядків зі своїх векторів на відсортовані власні значення матриці коваріації. Кількість відібраних основних компонент є параметром алгоритму. У існуючих роботах метод основних компонентів зазвичай застосовується в комбінації з методом машинного навчання SVM.

Демонстраційний приклад, який відбиває принцип роботи даного методу,представленого на рисунку 1.11.

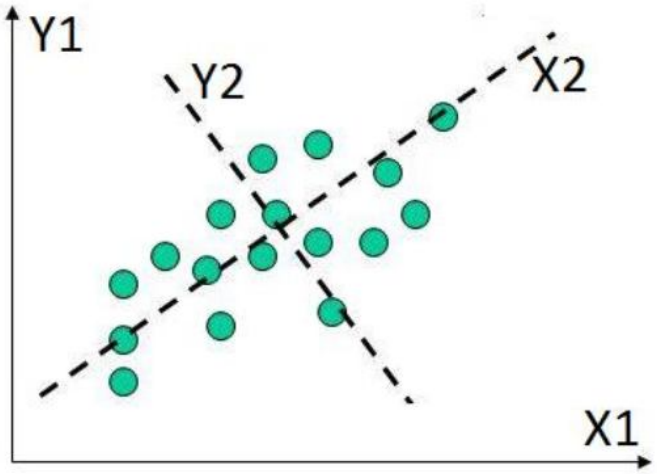


Рисунок 1.11 - Демонстрація принципу роботи методу головних компонент

Зазначимо, що одним з основних недоліків даного методу є прямо пропорційна залежність розміру коваріаційної матриці та розмірності вхідних даних, внаслідок чого пошук власних векторів може бути достатньо скрутним для даних високої розмірності. Варто пам’ятати, що цей метод не завжди ефективно знижує розмірність вхідних даних при заданих обмеженнях точність. Також зауважимо, що прямі та площини не завжди забезпечують хорошу апроксимацію.

Евристичні методи скорочення розмірності ознакового простору. Найбільш відомими евристичними методами зменшення розмірності даних,застосовуються в задачі аутентифікації користувачів за клавіатурним почерком,є генетичний алгоритм (Genetic Algorithm, GA), метод рою частинок (ParticleSwarm Optimization, PSO), а також мурашиний алгоритм (Ant Colony Optimization,ACO).

Генетичний алгоритм - це евристичний алгоритм пошуку,використовується для вирішення задач оптимізації та моделювання шляхом послідовного підбору, комбінування та варіації шуканих параметрів при вивченні впливу на підсумковий відгук з використанням механізмів, аналогічних природного відбору у природі. У разі вирішення задачі відбору ознак кожна хромосома в даному алгоритмі є набором ознак, де кожен ген

характеризує наявність ознаки: 0 – відсутня, 1 – присутня. Головним елементом даного алгоритму є функція адаптації гена (фітнес-функція).

Існує кілька способів вибору фітнес-функції. Наприклад, один із них заснований на алгоритмі класифікації найближчих сусідів. Фітнес-функція формується в припущенні, що після відкидання неінформативної ознаки набір найближчих сусідів змінюється мінімально.

Інший спосіб завдання фітнес-функції – обчислення її як точності класифікації. Слід враховувати, що деякі параметри цього алгоритму вибираються довільним чином, тому його необхідно запускати багато раз на перебування оптимального рішення.

Метод рою частинок є методом чисельної оптимізації,використання якого не потрібно знати точного градієнта функції, що оптимізується. Цей метод оптимізує вирішальну функцію, підтримуючи популяцію можливих рішень, званих частинками, і переміщує ці частинки у просторі згідно простій формулі. При цьому запам'ятовуються найкращі місця вже відвіданих.

Найкраще місце безпосередньо визначається якістю класифікації при даному наборі ознак. Пошук найкращих положень здійснюється всією колонією бджіл (часток)вихідному просторі ознак. Параметрами даного алгоритму є кількість бджіл-агентів, максимальна кількість ітерацій, загальна кількість ознак та кількість ознак, які необхідно залишити.

Перевагами даного методу є відсутність схильності до за циклювання в локальних оптимумах (оскільки алгоритм заснований на випадковому пошуку), а також той факт, що пошук найкращого рішення ґрунтується на рішеннях агентів усієї колонії бджіл.

Мурашиний алгоритм. Суть мурашиного алгоритму полягає у моделюванні поведінки мурашиною колонії. У реальному світі мурахи ходять у випадковому порядку і по знаходження продовольства

повертаються до своєї колонії, прокладаючи феромонами стежки. Якщо інші мурахи знаходять такі стежки, то вони, найімовірніше, підуть ними.

Вони також відкладають феромони, внаслідок чого на коротких стежках концентрація феромонів буде більшою і всі мурахи вибиратимуть цей шлях. У разі вирішення завдання вибору інформативних ознак, набір ознак подається у вигляді графа,у якому кожен вузол – це ознака. Параметром алгоритму, що задається на початку роботи, є кількість інформативних ознак, які потрібно знайти.

Мураха зупиняється тоді, коли пройдено необхідну кількість ознак. На на кожному кроці відбувається випаровування феромону. На кожній ітерації алгоритму вибирається набір ознак (шлях мурашки) із мінімальною помилкою. Критеріями закінчення роботи алгоритму є пройдена необхідна кількість ітерацій, а також досягнення порога мінімальної помилки. Таким чином, кількість феромону на кожної грані обернено пропорційно кількості помилок, отриманих при класифікації об'єктів за цією ознакою.

Дані евристичні алгоритми дозволяють виявляти як окремі інформативні ознаки, так і групи ознак, спільне використання яких дає зменшення помилки класифікації.

Основними недоліками даних алгоритмів є їх обчислювальна трудомісткість і низька масштабованість, оскільки вони ґрунтуються на методах випадкового пошуку. На практиці перераховані вище евристичні алгоритми зазвичай застосовують у комбінації з методом опорних векторів (SVM).

Нейронна мережа є математичною моделлю біологічних нейронних мереж, що складаються з набору, що взаємодіють між собою нейронів, кожен із яких отримує на вхід певний набір сигналів (функції виходу попередніх нейронів) та виробляє на виході результуючий сигнал.

Першим нейронам мережі на вхід подаються елементи векторів ознак аналізованих об’єктів. В результаті роботи останнього нейрона ми отримуємо число,

яке є результатом класифікації. За його значенням можемо визначити, чи належить аналізований об'єкт легітимного класу чи ні.

Нейронні мережі дозволяють ефективно будувати нелінійні залежності точно описують набори даних, а також дозволяють ефективніше стискати дані.

Також однією з головних переваг нейронних мереж є їх здатність додонавчання. Існує велика кількість видів нейронних мереж та їх модифікацій.

Найбільш відомими та часто використовуються з них є обмежена машина Больцмана, RBF-нейронна мережа, FF-MLP нейронна мережа, а також рекурентна нейронна мережа .

Порівняння якості роботи підходів, що використовуються для оцінки якості розпізнавання користувачів за динамікою їх роботи клавіатурою в існуючих роботах використовуються такі метрики:

• точність класифікації (Accuracy) – частка вірних спрацьовувань класифікатор;

• FAR (False Acceptance Rate) – відношення кількості випадків, коли зловмисник був розпізнаний системою як зареєстрований користувач, до загального числа спостережень, що розглядаються (величина помилки другого роду);

• FRR (False Rejection Rate) – частка ситуацій, коли зареєстрований користувач був розпізнаний системою як зловмисник (величина помилки першого роду);

• EER (Equal Error Rate) – відсоткова кількість помилок, за яких FAR збігається з FRR (при варіюванні порога для ухвалення рішення класифікатором);

• ROC-крива (Receiver Operating Characteristic) – графічна інтерпретація якості роботи бінарного класифікатора. Ця характеристика відображає залежність величин TPR (частки вірно розпізнаних легітимних користувачів) і FPR (частки відкинутих зловмисників);

• AUC (Area Under Curve) – кількісна інтерпретація ROC-кривої,що позначає площу під нею.

1.4. Постановка задачі дослідження

Серед основних проблем, що виявляються при аутентифікації користувачів в динаміці їх роботи з клавіатурою комп'ютера та сильно впливають на підсумковий результат розпізнавання виділяють такі:

• сучасні рішення не пропонують способу побудови стабільного часу ознакового простору, внаслідок чого спостерігається падіння якості розпізнавання користувачів з часом, у тому числі і при зміні використовуваної апаратури (оскільки клавіатури різних виробників мають різне розташування клавіш, а також різні технічні характеристики, відзначають різке падіння якості розпізнавання (на 20-25%) при зміні використовуваної апаратури). Для часткового вирішення цієї проблеми пропонують періодично оновлювати збудовану модель клавіатурного почерку користувача;

• Оскільки в реальних ситуаціях нам доступні дані лише легітимного класу, а приклади цільового нелегітимного класу або відсутні, або не зазначені в навчальній вибірці, звичайні методи підбору мета параметрів алгоритмів побудови одно класової моделі користувача з використанням валідаційного набору, що містить розмічені приклади обох класів (як для методів навчання з учителем)використовувати неможливо. Для часткового вирішення цієї проблеми пропонують використовувати штучну генерацію об’єктів нелегітимного класу.

Сформовані припущення є обґрунтуванням напрямів подальших досліджень:

• вибір ознакового простору, дослідження та розробка методів пост обробки ознак та скорочення розмірності ознакового простору з метою виділення найбільш інформативних та стабільних за часом ознак.

• Дослідження та розробка методів побудови моделі користувача,що дозволяють досягти високої якості розпізнавання (більше 0.90 ROCAUC) у

цій задачі, а також розробка методу підбору значень мета параметрів алгоритмів одно класової класифікації, здатного виявляти оптимальні значення параметрів алгоритмів, що використовуються.

• Дослідження та розробка методів та програмних засобів оцінки аномальності поведінки користувача на основі аналізу цілих сесій роботи за комп'ютером.

Висновки до першого розділу

З проведеного огляду відомих підходів до аналізу клавіатурного почерку користувачів, що застосовуються у сучасних програмних системах,можна зробити такі висновки:

• Динамічна автентифікація користувачів на основі аналізу динаміки їх роботи з клавіатурою комп'ютера є досить перспективним напрямом досліджень та широко застосовується для забезпечення безпеки як домашніх комп'ютерів користувачів, так і комп’ютерів найбільших корпорацій, а також для запобігання несанкціонованого доступу зловмисників до веб-сайтів.

• Для аналізу клавіатурного почерку користувача,необхідно фіксувати наступні характеристики його введення: код використовуваної клавіші, тип події (натискання / відтискання), а також тимчасову мітку, що відповідає події, що відбулася. Крім цього додатково можна фіксувати ім'я процесу, в рамках якого здійснюється взаємодія користувача з клавіатурою, а також адресу веб-сторінки, з якою працює користувач.

Сучасні підходи не здатні виділяти найбільш стабільні по часу ознаки, внаслідок чого спостерігається проблема падіння якості розпізнавання користувачів з часом (у тому числі і при зміні використовуваного обладнання).

РОЗДІЛ 2
МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ ДИНАМІЧНОЇ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ МОДЕЛІ КЛАВІАТУРНОГО ПОЧЕРКУ КОРИСТУВАЧА

2.1. Опис використовуваних для дослідження наборів даних

Оскільки динамічна аутентифікація користувачів за динамікою їх роботи з клавіатурою комп'ютера широко застосовується як у web-додатках, так і локальних програмах користувачів, необхідно розробити алгоритми, що показують високу якість роботи в обох даних областях. Тому, для експериментальної перевірки пропонованих підходів до вирішення поставленого завдання, необхідно використовувати як набори даних, що характеризують динаміку фонової роботи користувача з локальними програмами, так і набори даних, характеризуючи клавіатурний почерк користувача підчас роботи у web-браузері.

В рамках даного дослідження використовувалися чотири набори тестових даних:

Набір даних 1 (локальні дані).

Даний набір являє собою дані динаміки роботи користувачів з клавіатурою, зібрані у фоновому режимі підчас повсякденної роботи за комп'ютером (додатково, користувачі писали у вільній формі задані теми). У середньому кожен користувач пропрацював за комп'ютером порядку одного робочого дня. Крім інформації про динаміку клавіатурного введення, по кожному користувачу також збиралася наступна інформація: тип використовуваного комп'ютера (персональний комп'ютер / ноутбук), стать, вік користувача та наявність поінформовано стіп розбирання даних.

Інформація за типом використовуваних клавіатур власниками набору даних не надається. Побожному користувачеві було зібрано близько 27000 подій клавіатурного введення та побудовано в середньому по 50 векторів характеристичних ознак (кожен вектор ознак описував динаміку роботи користувача з клавіатурою в межах 300 послідовних подій клавіатурного введення).

Набір даних 2 (локальні дані). Даний набір є даними динаміки роботи 20 користувачів з клавіатурою, зібрані у фоновому режимі підчас повсякденної роботи за комп'ютер. У середньому, кожен користувач пропрацював за комп'ютером близько 10 годин: по одній годині протягом 10 днів. Використовувалися мембранні клавіатури з цифровим блоком та циліндричними клавішами. По кожному користувачеві було зібрано близько 65500 подій клавіатурного введення та побудовано в середньому векторів характеристичних ознак (кожен вектор ознак описував динаміку роботи користувача з клавіатурою в межах 300 послідовних подій клавіатурного введення).

Набір даних 3 (web-дані). Даний набір є даними динаміки роботи 20 користувачів з клавіатурою, зібрані за допомогою технології JavaScript у фоновому режимі роботи користувачів на форумі у веб-браузері. У середньому, кожен користувач пропрацював за комп'ютером близько 10 годин: по одній годин і протягом 10 днів. Використовувалися мембранні клавіатури з цифровим блоком та циліндричними клавішами.

Кожному користувачеві було зібрано близько 65000 подій клавіатурного введення та побудовано в середньому по 95 векторів характеристичних ознак (кожен вектор ознак описував динаміку роботи користувача з клавіатурою у межах порядку 300 послідовних подій клавіатурного введення)

Набір даних 4 (локальні дані). Даний набір даних є дані динаміки роботи 10 користувачів з клавіатурою комп'ютера двох видів (в обох

випадках дані збиралися у фоновому режимі підчас повсякденної роботи користувачів за комп'ютером):

• Дані динаміки роботи користувачів за загальним комп'ютером протягом трьох днів. Використовувалася мембранна клавіатура з цифровим блоком та плоскими кнопками. По кожному користувачеві було зібрано порядку 43000 подій клавіатурного введення та побудовано в середньому по 78 векторів характеристичних ознак (кожен вектор ознак описував динаміку роботи користувача з клавіатурою в межах порядку 300 послідовних подій клавіатурного введення);

• Дані динаміки роботи користувачів за різними комп'ютерами (кожен користувач працював за своїм власним комп'ютером) протягом трьох днів. Використовувалися мембранні клавіатури з цифровим блоком та циліндричними клавішами. По кожному користувачеві було зібрано близько 37000 подій клавіатурного введення та побудовано в середньому по 70 векторів характеристичних ознак (кожний вектор ознак описував динаміку роботи користувача з клавіатурою у межах порядку 300 послідовних подій клавіатурного введення).

Всі ці набори містять такі характеристики введення користувача, як код натиснутої клавіші, тип події (натискання/відтискання), а також тимчасову мітку, відповідну події, що відбулася. Також до даткові міститься інформація про назву процесу або адресу web-сторінки, в рамках яких відбувалася взаємодія користувача і з клавіатурою комп'ютера. У середньому, за кожному користувачеві було зібрано близько 47 500 подій клавіатурного введення.

Зазначимо, що всі тестові набори, що використовуються, містять введений користувачами текст як українською, і англійською мовами. З метою перевірки стабільності по часу розробленого рішення, збір даних відбувався у різний час доби (додатково, для перевірки роботи системи при зміні обладнання, що використовується використовувався набір даних 4). При використанні першого, другого та третього тестових наборів модель за

кожним користувачем будувалася на першій половині його зібраних даних. Для тестування використовувалася друга половина даних динаміки його роботи з клавіатурою та дані всіх інших користувачів. При використанні четвертого тестового набору побудова моделі користувача здійснювалася на першій частині набору (дані, зібрані на загальному комп'ютері) та тестувалося на другій частині набору (дані, зібрані за власними комп'ютерами користувачів).

2.2. Фільтрація даних

У ході аналізу наявних наборів даних, що характеризують динаміку роботи користувачів із клавіатурою комп'ютера, а також за результатами проведеного огляду існуючих рішень, було виявлено низку проблем, що виникають на стадії збору даних при використанні різних операційних систем, браузерів, локального програмного забезпечення та різного обладнання. Зокрема, було встановлено,що використання клавіатурних тренажерів (зокрема, клавіатурного тренажера Stamina), негативно впливає якість розпізнавання, оскільки змушує працювати користувача в нехарактерному темпі клавіатурного введення. Тому, всі події,зареєстровані в рамках даного процесу Stamina та аналогічних процесів,повинні вилучатися з подальшого розгляду.

Також було встановлено, що у зібраних даних можуть бути непарні події натискання або відтискання клавіш (для однієї клавіші може бути подія натискання, але не бути парною йому подія відтискання і навпаки). Дані події є шумовими і можуть негативно впливати на підсумковий результат розпізнавання. Тому всі непарні події для кожної клавіші також потрібні видаляти з подальшого розгляду.

Окремо варто розглянути ситуацію тривалого утримування клавіш клавіатури. У такій ситуації система послідовно генерує події натискання клавіші, встановлюючи спеціальний прапор події про повторення натискання. Час утримування клавіші, після якого фіксується подія повторення, залежить від налаштувань операційної системи і зазвичай варіюється від 250 до 1000 мілісекунд, при цьому частота генерації події повторення зазвичай варіюється від 2.5 повторень секунду до 30. Такий ланцюжок подій варто виключати з розгляду, оскільки він не дає жодної інформації про особливості роботи користувача з клавіатурою, а може лише дати дані про конкретні налаштування операційної системи. Видалення даних ланцюжків також дозволяє вирішити проблему фіксування в системі дублікатів, що прийшли від клавіатури подій, яка може спостерігатися з рідкісною періодичністю через ряд особливостей деяких операційних систем.

Наступним етапом вирішення поставленого завдання є поділ потоку, що надходять від клавіатури подій на часові вікна (на послідовні фрагменти наступних один за одним подій) та розрахунок характеристичних ознак для кожного часового вікна.

Однією з основних проблем, що виникають на етапі розбиття потоку даних на часові вікна, є вибір оптимального розміру часового вікна. Оскільки вікна великого розміру повноцінно характеризують динаміку роботи користувача з клавіатурою комп'ютера протягом тривалого проміжку часу, вони більш інформативними. Проте, чим більший розмір часового вікна, тим менше часових вікон ми можемо одержати з обмеженого потоку подій.

З метою збільшення кількості часових вікон, що використовуються для аналізу, необхідно використовувати вікна з перекриттями (див. рис. 2.2). Важливими параметрами, які сильно впливають на підсумковий результат класифікації, є розмір часового вікна, а також відсоток перекриття між часовими вікнами. Зазначимо, що при використанні великого відсотка перекриття вікон ми отримаємо велику кількість практично однакових

векторів ознак, що підвищить складність, але не дозволить підвищити інформативність моделі, що будується. З іншого боку, при використанні низького відсотка перекриття ми можемо втратити більшу частину корисної інформації.

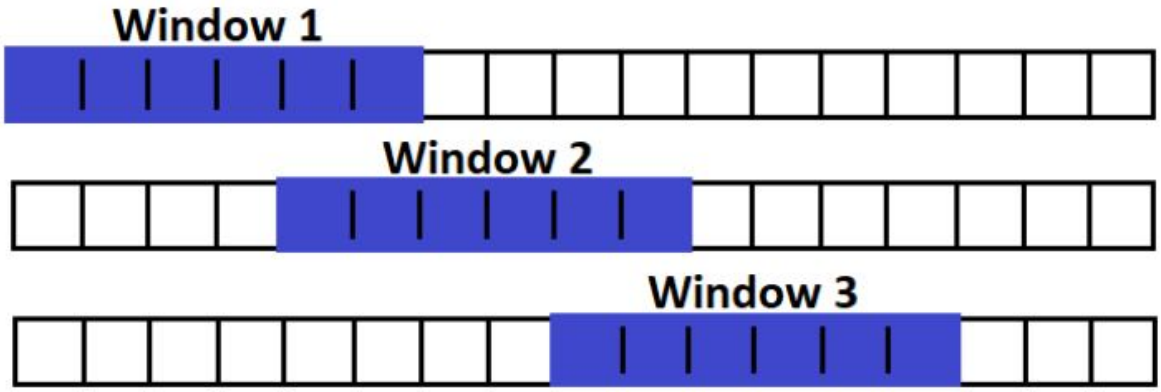


Рисунок 2.1 - Часові вікна з перекриттями

Оскільки існують методики поділу потоку подій на часові вікна як за кількістю подій у вікні, так і за тривалістю роботи користувача в рамках даного часового вікна, був проведений аналіз активності користувачів в рамках наданих тестових наборів даних. Розбиття за кількістю подій у вікні краще використовувати при помірній роботі користувача з клавіатурою, розбиття за тривалістю роботи користувача в рамках часового вікна ефективніше використовувати за активного темпу роботи користувача з клавіатурою. Проведений аналіз показав, що в середньому користувачі мають помірний темп клавіатурного введення, у зв'язку з чим, ефективніше розділяти потік зібраних даних на часові вікна, ґрунтуючись на кількості подій у часовому вікні. Тим самим гарантується, що кожне часове вікно міститиме достатню інформацію про клавіатурний рукописний текст почерком користувача.

Такожбулопроведенодослідженняможливостівикористанняпримусовогорозбиттяпотокуклавіатурнихподійначасовівікнавиникненнятривалихпауз у роботі користувача за клавіатурою, а також при зміні користувачем активного процесу або веб-сторінки, що переглядається. Розбиття поток у

клавіатурному введенні користувача (і віднесення подій, що прийшли до системи до і після паузи, до різних часових вікон) дозволило значно покращити якість аутентифікації. Обробка ж зміни діяльності при переході між програмами або web-сайтами (віднесення до різних часовим вікон подій, що прийшли в систему до та після переходу між будь-якими двома до датками або web-сайтами), як показали експерименти, не дає покращення точності розпізнавання. Це може бути пов'язано з тим, що під час переходу між додатками або web-сторінками користувач робить характерні дії, що дозволяють класифікатору фіксувати певну специфіку його поведінки.

Отримано, що ефективніше розділяти потік зібраних даних на вікна розміром у 500 подій для кожного вікна, та здійснювати примусове розбиття на вікна у разі виникнення тривалих пауз у діяльності користувача. При цьому, якщо при розбитті виходять вікна меншого розміру, їх можна залишати для подальшого розгляду за умови, що вони містять що найменше 300 подій. Якщо після примусового розбиття на часові вікна виникли вікна меншого розміру, їх слід прибирати з подальшого аналізу з метою покращення якості розпізнавання.

На підставі вищенаведеного аналітичного огляду , а також проведеної серії експериментальних досліджень, було прийнято рішення використовувати комбінований набір характеристичних ознак, що характеризують динаміку роботи користувачів з клавіатурою комп'ютера, оскільки він дозволяє врахувати більше індивідуальних особливостей користувача підчас роботи з клавіатурою, аналізуючи як його роботу з окремими клавішами, так і їх комбінаціями (N-грамами, де N – число клавіш, послідовно натиснутих у межах аналізованої комбінації). Оскільки у існуючих наукових працях розрахунок ознак не для всіх, а тільки для най частіше використовуваних легітимним користувачем клавіш значно покращує якість класифікації, а також збільшує швидкість роботи програми, сформований комбінований набір ознак розраховувався для 50 найбільш

часто використовуваних користувачем одиночки клавіш клавіатури та 100 найбільш часто використовуваних користувачем диграфів. Дані порогові значення було підібрано експериментально. Зазначимо, що при виборі порогових значень, що розглядаються, варто не забувати, що перед нами ставиться завдання побудови точної моделі, що швидко будується і швидко застосовується, оскільки дані алгоритми повинні працювати у режимі, близькому до режиму реального часу. N- грами при $N > 3$ у запропонованому підході не розглядаються, оскільки розрахунок характеристичних ознак для них є обчислювальна важким і не дає значного покращення підсумкової якості аутентифікації.

Таким чином, для кожного користувача використовуватиметься власний ознаковий простір, що також дозволить покращити якість аутентифікації, оскільки дозволить врахувати більше його індивідуальних особливостей під час роботи з клавіатурою комп'ютера. Найчастіше використовувані легітимним користувачем одиночні клавіші та диграфи визначаються на навчальній вибірці. Відмітимо, що оскільки ми розглядаємо безпосередньо самі клавіші клавіатури, без використання відомостей про поточну мовну розкладку, моделі, що будуються клавіатурним почерком користувачів не будуть чутливі до мови, що використовує ведення, що також підвищить якість розпізнавання.

Додатково, бувпроведенийаналізфізичногорозташуваннятафункціональногопризначення клавішнаклавіатурі, внаслідок якого були виділено наступні 17 груп клавіш (різні групи клавіш виділено різними кольорами, див. рисунок 2.2), характеристики роботи з кожною з яких також увійшли до результуючий набір ознак:

- Esc, F1, F2, F3, F4, F5, F6;
- F7, F8, F9, F10, F11, F12;
- Клавіша Windows;

• Стрілки;

• Клавіша "~", A, C, D, E, F, G, Q, R, S, T, V, W, X, Y, Z, Клавіша "Пробіл";

• Клавіша '/", Клавіша "[{", Клавіша \\|", Клавіша "]}", Клавіша ";:", Клавіша

",", Клавіша ".", Клавіша "/?", B, H, I, J, K, L, M, N, O, P, U;

• Shift, Ctrl, Alt;

• Лівий Shift, Лівий Ctrl, Ліва клавіша Windows, Ліва клавіша Контекстне меню»;

• Правий Shift, Правий Ctrl, Права клавіша Windows, Права клавіша "Контекстне меню";

• Tab, Caps Lock;

• Backspace, Enter;

• Scroll Lock, Select, Print, Execute, Print Screen, Insert, Delete, Help, Page Up, PageDown, Home, End, Pause;

• 1, 2, 3, 4, 5, 6;

• 7, 8, 9, 0, Клавіша "-", Клавіша "+";

• Усі клавіші Num Pad;

• Клавіша Cancel (зупинка процесу);

• Медіа-клавіші (сон, вперед-назад, браузер, звук, керування музичними треками, пошта тощо).

Рисунок 2.2 - Виділені групи клавіш клавіатури

2.3. Модель клавіатурного почерку користувача

Для вирішення задачі побудови моделі клавіатурного почерку користувача даної роботи пропонується розглянути методи пошуку винятків уданих.

Методи пошуку винятків відіграють важливу роль при вирішенні багатьох прикладних завдань, насамперед пов'язаних із безпекою [16]. У таких завданнях як правило доступні дані лише одного, легітимного класу. А прикладів нелегітимного цільового класу мало і часто їх важко виділити шляхом ручної розмітки. Дані завдання називаються завданнями одно класової класифікації. Вони легітимна модель будується без використання зразків інших класів, хоча може робитися припущення, що у навчальній вибірці може бути певний невеликий відсоток спостережень із цільового нелегітимного класу. Такі завдання виникають у галузях комп'ютерної, фінансової та суспільної безпеки. Розглянуте завдання динамічної аутентифікації користувачів по клавіатурному почерку також відноситься до даного класу завдань. Неформально під аномалією (або винятком) розуміється об'єкт чи подія у вибірці, чиї ознаки або їх комбінації не відповідають залежностям, характерним для інших об’єкті вабо подій у цій вибірці. Для пошуку аномалій традиційно використовуються статистичні (імовірнісні) та метричні підходи, а також методи, засновані на аналізі відхилень. У статистичному підході під винятком розуміється спостереження (використовується імовірнісна інтерпретація поняття винятку як малоймовірної події). У метричному підході використовується геометрична інтерпретація поняття виключення: винятком є спостереження,віддалений від більшості спостережень у вибірці. Методи, що не використовують ні ймовірнісну, ні геометричну інтерпретацію поняття виключення, відносяться до методів, що ґрунтуються на аналізі відхилень. Слід зазначити, що дані в задачах,пов'язаних з безпекою, як правило мають простір ознак великої

розмірності. Внаслідок чого, багато ознак виявляються не релевантними з точки зору виділення цільового класу, а багато ознак є взаємозалежними. Це ускладнює використання традиційних імовірнісних та метричних підходів,які не стійкі до прокляття розмірності. Перспективним підходом у цій ситуації є використання kernel методів (методів, заснованих на переході з вихідного простору ознак (Input space) у простір ознак більшої розмірності (RKHS) з використанням потенційної (kernel) функції та пошуку залежностей у новому результуючому просторі). Найбільш популярними серед них є одно класовий SVM (або SVC) і kernel-версія методу головних компонент - Kernel PCA. Але ці методи мають ряд недоліків. Зокрема, SVC шукаєоптимальний центр множини образів спостережень у просторі характеристик високої розмірності (RKHS) і обмежує їх гіперсферою мінімального радіусу,рахуючи спостереження, чиї образи виходять за межі гіперсфери, винятками.

Очевидним недоліком тут є сферичність області, оскільки залежність між вихідними ознаками також можуть призвести до залежностей між індукованими ознаками в RKHS, а значить, логічніше використовувати не сферичні області, а еліптичні. З іншого боку, Kernel PCA будує якраз еліптичні області в RKHS, що містять образи допустимих спостережень, що дозволяє ефективно працювати з сильно залежними ознаками. Однак, Kernel фіксує центр розподілу та не перераховує його з урахуванням знайдених викидів.

Для подолання цих недоліків у цій роботі пропонується новий метод виявлення аномалій, що базується на аналізі відхилень. Основною ідеєю запропонованого методу є перехід з вихідного простору ознак у простір характеристик високої розмірності та подальша нечітка кластеризація образів спостережень у результуючому просторі з використанням метрики Махаланобіса для розрахунку відстаней між об'єктами та центром кластера. В просторі більшої розмірності будується один загальний нечіткий кластер еліпсоїдальної форми, де кожен образ спостереження має свій ступінь

приналежності (типовості). Центр кластера також знаходиться в індукованому просторі та ітераційно перераховується. Налаштування алгоритму кластеризації(параметри регуляризації та ступінь нечіткості) задаються так, щоб ступінь приналежності «основної частини» образів спостережень кластера в RKHS була досить висока (вище заданого порога, наприклад, вище 0.5). Винятками вважаються спостереження зі ступенем типовості, меншою за заданий поріг. Відстань Махаланобіса обчислюється шляхом проектування даних на простір, заданий власними векторами матриці коваріації RKHS. Завдяки цьому вдається врахувати дисперсії та кореляції між ознаками в RKHS.

Метою переходу в простір характеристик більшої розмірності є ефективне використання більш простих геометричних структур для опису залежностей у вхідних даних. Демонстраційний приклад, що відображає принцип переходу в простір ознак високої розмірності, представлений на рисунку 2.3.

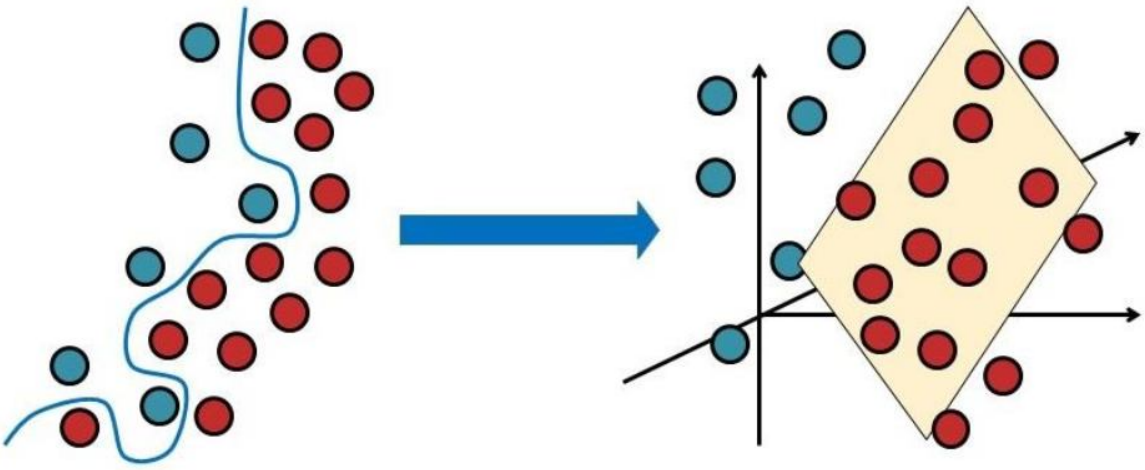


Рисунок 2.3 - Принцип переходу у простір ознак більшої розмірності (RKHS)

Приклад дискретизації ознак за кванти лями представлений на рисунку 2.4.

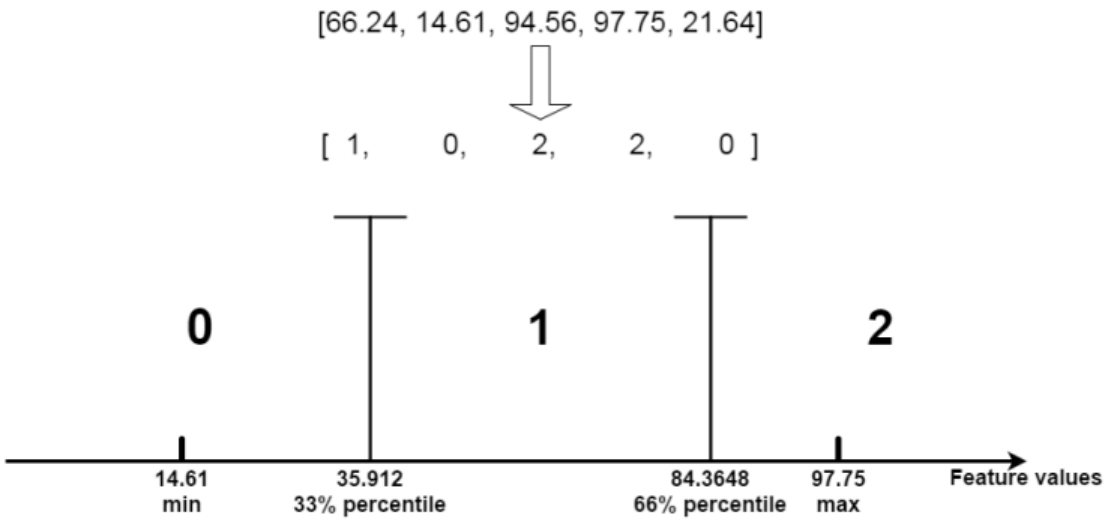


Рисунок 2.4 - Приклад дискретизації ознак за кванти лями

Розглянемо завдання нечіткої кластеризації у просторі характеристик високої розмірності (RKHS) у разі побудови одного кластеру. Є кінцева множина аналізованих об'єктів

$$\{x_i\}_{1 \le i \le N} \subset X \tag{2.1}$$

Також, на $X \cdot X$ визначено потенційну функцію

$$K = X \cdot X \to R_0 \tag{2.2}$$

яка визначає відображення вихідного ознакового простору у простір ознак більшої розмірності. Необхідно в RKHS (уданому випадку – вH) побудувати єдиний кластер еліпсоїдальної форми, що включав себе всі образи аналізованих об'єктів таким чином, що ступінь належності(типовості) u_iкожного образу даному кластеру обчислюється шляхом вирішення наступної задачі оптимізації:

$$\min_{U,a,\eta} E(U,a,\eta) = \sum_{i=1}^{N} u_i^m \left\| a - \varphi(x_i) \right\|_C^2 + \eta \sum_{i=1}^{N} (1-u_i)^m$$

де a – центр нечіткого кластера у просторі параметрів; N - число аналізованих об’єктів; U - вектор значень.

Для кожного вхідного аналізованого об'єкта відстань Махаланобіса $D_k(a)$ між його образом $\varphi(x_k)$ і центром нечіткого кластера у просторі характеристик обчислюється так:

$$D_k(a) = \|a - \varphi(x_k)\|_C^{\,2} = (a - \varphi(x_k))^T M(a - \varphi(x_k)) = \sum_{j=1}^{N}\left(\sum_{i=1}^{N}(a - \varphi(x_k))\, M_{ij}\right)$$

$$\varphi(x_k)) = \sum_{j=1}^{N}\sum_{i=1}^{N} M_{ij}aa - M_{ij}a\,\varphi(x_k) - M_{ij}\varphi(x_k)a + M_{ij}\varphi(x_k)\varphi(x_k).$$

$$D_k(a) = \|a - \varphi(x_k)\|_C^{\,2} = (a - \varphi(x_k))^T M(a - \varphi(x_k)) =$$

$$= \sum_{j=1}^{N}\left(\sum_{i=1}^{N}(a - \varphi(x_k))\, M_{ij}\right)(a - \varphi(x_k)) =$$

$$= \sum_{j=1}^{N}\sum_{i=1}^{N} M_{ij}aa - M_{ij}a\,\varphi(x_k) - M_{ij}\varphi(x_k)a + M_{ij}\varphi(x_k)\varphi(x_k) =$$

$$= \sum_{j=1}^{N}\sum_{i=1}^{N} M_{ij}(< a, a >_C - 2 < \varphi(x_k), a >_C + K(x_k, x_k)) =$$

$$= \sum_{j=1}^{N}\sum_{i=1}^{N} M_{ij}(< a^* + a^{**}, a^* + a^{**} >_C - 2 < \varphi(x_k), a^* + a^{**} >_C + K(x_k, x_k)) =$$

$$= \sum_{j=1}^{N}\sum_{i=1}^{N} M_{ij}(< a^*, a^* >_C + < a^{**}, a^{**} >_C - 2 < \varphi(x_k), a^* >_C + K(x_k, x_k)) =$$

$$= \|a^* - \varphi(x_k)\|_C^{\,2} + < a^{**}, a^{**} >_C .$$

Продемонструємо роботу алгоритму на прикладі. Зібравши holdout вибірку і порахувавши середнє значення всіх її елементів, ми починаємо подавати на вхід алгоритм тестових векторів. З приходом кожного нового тестового вектора, ми оновлюємо величину середнього значення елементів, що накопичилися до цього моменту тестової вибірки. Далі ми обчислюємо t-статистику Уелша та відповідний їй $p - value$.

Таким чином, ми можемо побудувати графік – для легітимного та нелегітимних користувачів. Також можна побудувати графік $log(1+)$.

Тут чим ближче графік користувача до нуля, тим вища ймовірність, що він є зловмисником.

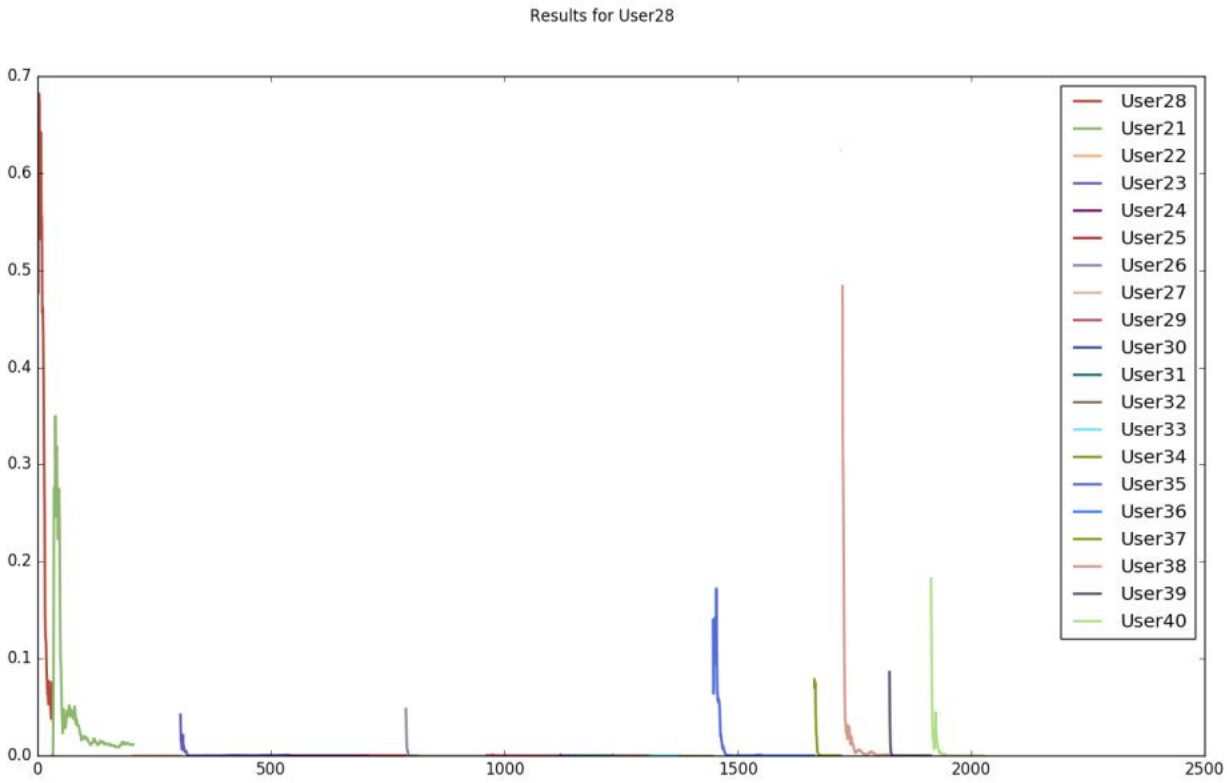


Рисунок 2.5 - Графік $log(1+)$, що демонструє якість розпізнавання користувача за динамікою його роботи з клавіатурою

Даний графік демонструє якість розпізнавання вибіркового користувача по динаміці його роботи із клавіатурою. Червоним кольором позначено динаміку роботи легітимного користувача з клавіатурою. Іншими кольорами позначена динаміка роботи інших користувачів із клавіатурою (зловмисників). Як ми бачимо, криві графіків зловмисників набагато нижче кривих графіка легітимного користувача, що свідчить про високу якість його розпізнавання.

Висновки до другого розділу

Розроблено метод оцінки аномальності поведінки користувачів на основі аналізу цілих сесій роботи за комп'ютером з використанням статистики Уелша, що узагальнює всі відгуки класифікатора для даних тестового користувача за аналізований період (цілу сесію) його роботи за комп'ютером і дозволяє досягти високої якості автентифікації.

РОЗДІЛ 3
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДИНАМІЧНОЇ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ МОДЕЛІ КЛАВІАТУРНОГО ПОЧЕРКУ КОРИСТУВАЧА

3.1. Особливості реалізації системи

Програмне забезпечення повинне підтримувати три базові сценарії функціонування:

• Збір поведінкової інформації про взаємодію користувачів з клавіатурою комп'ютера;

• Побудова індивідуальних моделей поведінки користувачів;

• Застосування індивідуальних моделей поведінки користувачів.

Збір поведінкової інформації щодо взаємодії користувачів з клавіатурою комп'ютера. Завданням даного етапу є збирання та збереження даних, що описують взаємодія користувачів із клавіатурою комп'ютера. Мається на увазі, що існує кілька робочих місць користувачів (персональних комп’ютерів або ноутбуків), на кожному з яких за допомогою спеціалізованих агентів моніторингу збирається користувальницька поведінкова інформація,зберігається локально на робочому місці користувача та за допомогою агента консолідації передається в єдине мережеве сховище даних (на сервер консолідації) для подальшого аналізу.

Як користувальницька поведінкова інформація в даному завданні виступає набір записів про роботу користувачів з клавіатурою комп'ютера наступного виду:

• Ім'я користувача;

• Код клавіші;

• Тип події (натискання/віджимання);

• Тимчасова мітка, що відповідає події, що відбулася.

Також, для додаткового аналізу може використовуватись ім'я процесу, в рамках якого виконується взаємодія користувача з клавіатурою комп'ютера. Проте,як показала попередня серія експериментів, використання даної характеристики не дозволяє покращити якість розпізнавання користувачів.

Збирати інформацію пропонується локально у фоновому режимі з використанням спеціального hook-а (перехоплювача), що здійснює перехоплення що прийшли в систему подій від клавіатури (за допомогою WinAPI) і зберігає отриману інформацію локально на робочому місці користувача.

Далі здійснюється передача зібраної інформації до єдиного мережевого сховища для подальшої обробки. Варіант безпосереднього запису даних у загальне мережеве сховище є найбільш перспективним у даному завданні порівняно зі стратегіями ручного копіювання даних або використання клієнт серверної архітектури, оскільки вимагає мінімальної кількості трудовитрат і має високу ефективність. Підтримка проміжного локального зберігання зібраної інформації дозволяє оптимізувати навантаження на мережу передачі даних, а також дозволяє не втратити зібрані дані у разі відсутності з'єднання з сервером консолідації.

Оскільки раніше було встановлено, що у зібраних даних можуть бути присутніми деякі шумові події (події, отримані під час роботи користувачів з клавіатурними тренажерами, а також можливі непарні події натискання / віджимання клавіш або ситуації тривалого утримування клавіш), зібрані в єдиному мережевому сховищі дані необхідно відфільтровувати агентом консолідації перед процедурою побудови індивідуальних поведінкових моделей.

Таким чином, розв'язання задачі складається з наступних етапів:

• Збір та збереження поведінкової інформації на робочих місцях користувачів;

• Передача зібраної інформації в єдине мережеве сховище (сервер консолідації), збереження отриманої інформації на сервері консолідації;

• Фільтрування отриманої інформації (видалення шумових подій) агентом консолідації.

Далі здійснюється передача зібраної інформації до єдиного мережевого сховища для подальшої обробки. Варіант безпосереднього запису даних у загальне мережеве сховище є найбільш перспективним у даному завданні порівняно зі стратегіями ручного копіювання даних або використання клієнт-серверної архітектури, оскільки вимагає мінімальної кількості трудовитрат і має високу ефективність. Підтримка проміжного локального зберігання зібраної інформації дозволяє оптимізувати навантаження на мережу передачі даних, а також дозволяє не втратити зібрані дані у разі відсутності з'єднання з сервером консолідації.

Оскільки раніше було встановлено, що у зібраних даних можуть бути присутніми деякі шумові події (події, отримані під час роботи користувачів з клавіатурними тренажерами, а також можливі непарні події натискання / віджимання клавіш або ситуації тривалого утримування клавіш), зібрані в єдиному мережному сховищі дані необхідно відфільтровувати агентом консолідації перед процедурою побудови індивідуальних поведінкових моделей

Розглянемо запропоновану програмну реалізацію експериментального зразка програмного комплексу. Вимогами до системи, що розробляється - функціонування на ОС Windows версії 8 та вище, а також оперативність роботи,ефективне використання ресурсів та стійкість до можливих помилок.

Пропоноване архітектурне рішення є мультиагентною системою, що складається з наступних модулів:

1. Агент моніторингу. Цей агент встановлюється безпосередньо на робоче місце користувача і складається з двох паралельно працюючих модулів:модуля збору інформації та модуля класифікації. Модуль збору інформації у фоновому режимі здійснює збір інформації про взаємодії користувача з клавіатурою комп'ютера, зберігає її локально та періодично (відповідно до заданого режиму передачі)здійснює передачу збереженої інформації консолідації агенту.

Модуль класифікації працює на робочому місці користувача та здійснює застосування індивідуальної поведінкової моделі користувача, отриманого зі сховища моделей, у режимі, близькому до режиму реального часу, до даних клавіатурного почерку, зібраних локально на даному робочому місці. У разі виявлення аномальної поведінки користувача, доступ до системи блокується. Отримані результати класифікації також передаються агенту консолідації.

2. Агент консолідації. Цей агент встановлюється на сервері консолідації та включає модуль консолідації. Модуль консолідації здійснює прийом даних клавіатурного почерку та локальної класифікації користувачів від агентів моніторингу, що забезпечує їх зберігання в єдиному мережевому сховищі (на сервері консолідації), фільтрації зібраної інформації та надає доступ до даних, що зберігаються, для їх подальшого аналізу та побудови індивідуальних моделей поведінки користувачів.

Також даний модуль здійснює передачу копій збудованих поведінкових моделей із сховища моделей на робочі місця відповідних користувачів з метою прискорення процедури автентифікації, а також щоб уникнути проблем, пов'язаних з можливим обривом з'єднання між робочими місцями користувачів та сервером консолідації.

3. Модуль побудови індивідуальних моделей поведінки користувачів. Цей модуль здійснює побудову моделей поведінки користувачів на основі даних, що зберігаються в єдиному мережевому сховищі, та зберігає

побудовані моделі у сховищі моделей (що знаходиться на сервері консолідації).

4. Модуль аналізу клавіатурного почерку користувачів у відстроченому режимі. Цей модуль здійснює застосування побудованих моделей поведінки користувачів, отриманих зі сховища моделей, до даних клавіатурного рукописного тексту почерк користувачів, що зберігаються в єдиному мережевому сховищі.

5. Автоматизоване робоче місце (АРМ) аналітика. АРМ аналітика є графічним інтерфейсом, що дозволяє керувати модулем консолідації, будувати та перебудовувати (оновлювати) індивідуальні моделі поведінки користувачів (з використанням модуля побудови індивідуальних моделей поведінки користувачів), застосовувати побудовані моделі у відкладеному режимі (з використанням модуля аналізу клавіатурного почерку користувачів у відкладеному режимі), спостерігати за результатами застосування моделей на робочих місцях користувачів у режимі, близькому до режиму реального часу (отримуючи цю інформацію з єдиного мережевого сховища), оцінювати рівень аномальності поведінки користувача на основі аналізу цілих сесій роботи за комп'ютером, а також будувати докладні звіти щодо роботи системи.

Загальна схема архітектури пропонованого програмного забезпечення представлена на рисунку 3.1.Кожен із розглянутих модулів складається з компонентів, що виконують дії над вихідними або проміжними даними. Компонент – це програмний під модуль певного типу, виконання якого визначається конфігураційним файлом.

Компоненти поділяються на два основні види: завдання та процесори і можуть складатися з декількох під компонентів. Взаємодія між компонентами здійснюється за допомогою текстових та конфігураційних файлів.

Рисунок 3.1 - Архітектура системи

Текстові файли містять вхідні для компонентів дані чи результати роботи компонентів. Конфігураційні файли містять параметри алгоритмів, реалізованих у компонентах. Відповідність між програмними модулями та вхідними в них програмними компонентами наведено в Таблиці 3.1

Таблиця 3.1- Відповідність між програмними модулями та програмними компонентами

Програмний модуль	Програмні компоненти
Модуль збору інформації агента моніторингу	Програмний компонент збору даних клавіатурного почерку користувачів
Модуль класифікації агента моніторингу	Програмний компонент обробки даних клавіатурного почерку користувачів
Модуль аналізу клавіатурного почерку користувачів у відстроченому режимі	Програмний компонент побудови векторів ознак користувача
	Програмний компонент обробки векторів ознак користувача
	Програмний компонент класифікації нових даних

Завдання – це вид компонента, завданнями якого є: читання необхідних вихідних даних; обробка та можлива фільтрація прочитаних даних; виклик алгоритму обробки даних (процесора) та запис вихідних даних у результуючий файл

Процесор – це вид компонента, основним завданням якого є виконання алгоритму обробки даних, включаючи обробку векторів ознак, побудова моделі,класифікацію тощо. Параметри алгоритму, що використовується процесором, задаються при створення процесора і зберігаються у конфігураційному файлі.

Послідовність завдань може виконуватися як в автоматичному режимі, так поетапно. Поетапний режим виконання послідовностей завдань використовується нарам аналітика, що виробляє глибоке дослідження процесів побудови та застосування індивідуальних моделей користувачів.

Розбиття всього процесу аутентифікації на послідовність завдань дозволяє відкотитися на крок назад у разі виникнення можливих помилок (при цьому,аналізовані дані та проміжні результати роботи не втратяться). Тим самим,підвищується стійкість розробленої системи до можливих помилок.

Додатково відзначимо, що система, що розробляється, підтримує багатозадачність: одночасно можуть виконуватись завдання для здійснення різних етапів аутентифікації декількох користувачів (наприклад, у модулі побудови індивідуальних моделей поведінки користувачів може здійснюватися одночасна побудова векторів ознак та моделей для різних профілів, а не тільки для одного), що також підвищує оперативність роботи запропонованої системи, а також дозволяє ефективніше використовувати наявні ресурси. Зауважимо, що розроблена архітектура підтримує аутентифікацію користувача при зміні обладнання, що використовується.

3.2. Опис програмних компонентів

Цей програмний компонент входить до складу модуля збору інформації агента моніторингу та у фоновому режимі здійснює збір поведінкової інформації про динаміку роботи користувачів з клавіатурою комп'ютера.

Програмний компонент складається з двох файлів: динамічної бібліотеки перехоплювача (.dll), призначеної для вбудовування у процеси Windows для перехоплення цільових подій, та виконуваного файлу-інжектора (.exe), призначеного для управління цією бібліотекою. Підключення перехоплювача локальних подій від клавіатури здійснюється засобами WinApi (SetWindowsHookEx/UnhookWindowsHookEx). Перехоплювач та інжектор використовують паралельну обробку і взаємодіють за допомогою процесів (threads) та стандартних засобів синхронізації Windows (критичні секції, пам'ять, що розділяється). При запуску програми в цільовому каталозі створюються (у разі відсутності) логів програми, які можуть використовуватися надалі в інформаційно-налагоджувальних цілях.

Для кожної сесії роботи користувача за комп'ютером створюється директорія,що містить у своїй назві ім'я користувача, дату та час початку сесії.

Дана директорія включає .csv-файл, що містить записані дані про динаміку роботи користувачів з клавіатурою комп'ютера в порядку їх приходу в систему, а також файл у форматі JSON, що містить інформацію про апаратно-програмну конфігурацію клієнтської машини, що відповідає часу початку сесії (базові відомості про використовувану апаратуру, IP-адресу робочої машини, ОС,версії програми). Приклади вмісту даних файлів наведено на рисунках 3.2–3.3.

time	username	session_id	hwnd_ori	processna	PID	hookmod	provocati	virtualcod	scancode	keyup	prev_keys	repeat-co	alt_down	extended	langID	keybdID	raw_data_debug
2021-04-2	afilippov	0x1	0x100cc	FlashMon	2916	0	0	13	28	1	1	1	0	0	0x409	0x409	0xc01c0001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	67	46	0	0	1	0	0	0x409	0x409	0x2e0001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	67	46	1	1	1	0	0	0x409	0x409	0xc02e0001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	72	35	0	0	1	0	0	0x409	0x409	0x230001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	72	35	1	1	1	0	0	0x409	0x409	0xc0230001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	8	14	0	0	1	0	0	0x409	0x409	0xe0001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	8	14	1	1	1	0	0	0x409	0x409	0xc00e0001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	8	14	0	0	1	0	0	0x409	0x409	0xe0001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	8	14	1	1	1	0	0	0x409	0x409	0xc00e0001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	89	21	0	0	1	0	0	0x409	0x409	0x150001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	89	21	1	1	1	0	0	0x409	0x409	0xc0150001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	79	24	0	0	1	0	0	0x409	0x409	0x180001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	85	22	0	0	1	0	0	0x409	0x409	0x160001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	79	24	1	1	1	0	0	0x409	0x409	0xc0180001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	85	22	1	1	1	0	0	0x409	0x409	0xc0160001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	84	20	0	0	1	0	0	0x409	0x409	0x140001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	84	20	1	1	1	0	0	0x409	0x409	0xc0140001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	85	22	0	0	1	0	0	0x409	0x409	0x160001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	85	22	1	1	1	0	0	0x409	0x409	0xc0160001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	66	48	0	0	1	0	0	0x409	0x409	0x300001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	66	48	1	1	1	0	0	0x409	0x409	0xc0300001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	69	18	0	0	1	0	0	0x409	0x409	0x120001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	69	18	1	1	1	0	0	0x409	0x409	0xc0120001
2021-04-2	afilippov	0x1	0x10196	chrome.e:	3220	0	0	40	80	0	0	1	0	1	0x409	0x409	0x1500001

Рисунок 3.2 - Вміст .csv-файлу, в який записується зібрана інформація про взаємодію користувача з клавіатурою комп'ютера

Файл, який містить запис активності користувача під час роботи з клавіатурою(.csv), містить такі основні поля:

• Час приходу події у систему;

• Ім'я (ідентифікатор) користувача;

• Ідентифікатор поточної сесії роботи за комп'ютером;

• Ім'я процесу, з яким ведеться робота;

• PID процесу, з яким ведеться робота;

• Код клавіші, що розглядається (натиснутою/віджатою);

• Мітка: натискання або відтискання клавіші;

• Поточна мова клавіатури.

Цей програмний компонент реалізований на мові C++.

Цей програмний компонент входить до складу модуля консолідації агента консолідації, а також модуль класифікації агента моніторингу. Його основними завданнями є:

• Завдання об'єднання даних (Data Task), що є об’єднання вихідних даних для конкретного користувача з різних сесій (отриманих з відповідних .csv-файлів) та подальшу фільтрацію даних, що не потрапляють у тимчасовий

інтервал (дані за який необхідно проаналізувати), що задається в конфігураційному файлі даного завдання;


```
{
"Version": "3.2.4.0",
"Bitness": 32,
"Options": {
    "provocationMode": 0,
    "restartTimer": 0,
    "internalSessionID": 0,
    "timeUTC": 0
    },
"OS": "Windows Server: 6.1.7601 Service Pack 1",
"ComputerName": "JINN",
"IPAddress": "158.250.19.10",
"UserID": "kazachuk",
"StartTime": "2021-09-26T11:28:42.441",
"SessionID": 1,
"Processor": {
    "frequencyMhz": 2999,
    "architectureType": 9,
    "oemID": 9,
    "count": 2,
    "type": 8664,
    "activemask": 3
    },
"Display": {
    "width": 1280,
    "height": 1024,
    "hmmSize": 452,
    "vmmSize": 361,
    "numMonitors": 1
    },
"Keyboard": {
    "type": 7,
    "subType": 0,
    "functionKeysType": 12,
    "repeatDelay": 1,
    "repeatSpeed": 31
    },
"InputDevices": {
    "Keyboard0": "acpi#pnp0303#4&226211b3&0#{884b96c3-56ef-11d1-bc8c-00a0c91405dd}",
    "Keyboard1": "root#rdp_kbd#0000#{884b96c3-56ef-11d1-bc8c-00a0c91405dd}",
    }
}
```


Рисунок 3.3 - Вміст JSON-файлу, що зберігає інформацію про апаратно-програмну конфігурацію клієнтської машини

• Завдання обробки даних (Data Process Task). Основне завдання завдання обробки об'єднаних даних є читання результатів виконання завдання об'єднання даних, виклик процесора обробки даних (Data Processor) та запис оброблених даних у результуючий .csv-файл.

Процесор обробки об'єднаних даних виконує фільтрацію даних: прибирає непарні події натискання/відтискання клавіш,тривалу серію подій

утримування клавіш, а також події,відповідні роботі користувача із клавіатурними тренажерами. Цей програмний компонент реалізований на Python 3.5 спільно з модулями numpy, pandas, user-agents, statsmodels, що дозволяють суттєво спростити код програми та прискорити її роботу за рахунок вбудованих методів розпаралелювання.

Програмний компонент побудови структури векторів ознак. Цей програмний компонент входить до складу модуля побудови індивідуальних моделей поведінки користувачів та складається із завдання побудови структури векторів ознак (Structure Task). Основне завдання даного завдання є читання вихідних оброблених даних, виклик процесора побудови структури векторів ознак (Structure Processor), а також запис результуючого файлу структури у відповідний .csv-файл. Процесор побудови структури векторів ознак виявляє найчастіше використовувані користувачем поодинокі клавіші клавіатури та комбінації клавіш. Кількість клавіш, що відбираються, задається при створення компонента і зберігається у конфігураційному файлі.

Результуючий .csv-файл має таку структуру (рисунок 4.3):

• У першому рядку впорядковані за частотою коди найбільш поширені часто зустрічаються під час роботи розглянутого користувача клавіш(параметр – їх максимальна кількість вказується під час створення процесора побудови структури ознакового простору). На першому місці стоїть код найбільш часто зустрічається клавіші.

• У другому рядку впорядковані за частотою коди найбільш часто зустрічаються при роботі розглянутого користувача диграфів(параметр – їх максимальна кількість вказується під час створення процесора побудови структури ознакового простору). На першому місці стоїть код найчастіше зустрічається диграфа. Код кожного диграфа являє собою розділену нулем («0») пару кодів двох клавіш,складових цей диграф. На першому місці в цій парі стоїть перша клавіша цього диграфа.

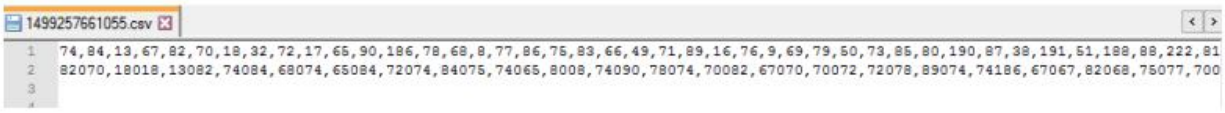


Рисунок 3.4 - Побудована структура ознакового простору

Цей програмний компонент реалізований на Python 3.5 спільно з модулями numpy, pandas, user-agents, statsmodels, що дозволяють суттєво спростити код програми та прискорити її роботу за рахунок вбудованих методів розпаралелювання.

Програмний компонент побудови векторів ознак користувача. Цей програмний компонент входить до складу модуля побудови індивідуальних моделей поведінки користувачів, а також модуля класифікації агента моніторингу та модуля аналізу клавіатурного почерку користувачів у відкладеному режимі, і складається із завдання побудови векторів ознак (Features Task). Основне завданням даного завдання є читання вихідних оброблених даних, файлу побудованої структури векторів ознак, виклик процесора побудови векторів ознак користувача (Features Extractor) та запис побудованих векторів ознак у результуючий .csv-файл. Процесор побудови векторів ознак виконує розбиття вхідних даних на тимчасові вікна та розрахунок ознак для кожного тимчасового вікна. Параметрами даного процесора є налаштування розбиття вихідних оброблених даних на часові вікна. Підтримуються можливості розрахунку тимчасових вікон, що перекриваються між собою, а також обліку пауз у діях користувача та обліку зміни активних процесів при здійсненні розбиття на тимчасові вікна. Розбиття на тимчасові вікна може здійснюватися як за перевищення ліміту на кількість подій у вікні, так і при перевищенні ліміту на тимчасову тривалість вікна. Для кожного тимчасового вікна здійснюється розрахунок наступних характеристичних ознак:

• Середній час утримання клавіш у часовому вікні;

• Середній час між відпусткою першої та відпусканням другої клавіші у часовому вікні;

• Середній час між натисканням першої та відпусканням другої клавіші тимчасового вікна;

• Середній час утримання групи клавіш у часовому вікні;

• Частота набору тексту користувачем у часовому вікні.

Програмний компонент обробки векторів ознак користувача. Цей програмний компонент входить до складу модуля побудови індивідуальних моделей поведінки користувачів, а також модуля класифікації агента моніторингу та модуля аналізу клавіатурного почерку користувачів у відкладеному режимі, і складається із завдання обробки векторів ознак (Features Process Task).

Основним завданням цього завдання є читання побудованих векторів ознак,виклик необхідного процесора обробки векторів ознак користувача (FeaturesProcessor) та запис оброблених векторів ознак у результуючий .csv-файл. Існують шість реалізацій даного процесора: усереднення складових ознак(Mean Features Processor), вибір найбільш чи найменш стабільних ознак на основі критерію Колмогорова-Смирнова (Stability Features Processor), відбір ознак методом головних компонентів (PCA Features Processor), а також дискретизація за квантилями (Quantiles Discretization Features Processor), стандартизація значень ознак (StandardScaler Features Processor) та конвеєр обробки векторів ознак (Features ChainProcessor). Основним завданням процесора Features Chain Processor є об’єднання кількох процесорів обробки векторів ознак в один конвеєр. Конвеєр складається із стадій, які виконуються послідовно. Кожна стадія являє собою набір окремих трансформацій, кожна з яких представлена процесором обробки ознак, а також переліком імен ознак, для яких має бути застосовано вказане перетворення. Для відібраних (процесорами Stability FeaturesProcessor або PCA Features Processor) ознак проводиться усереднення їх значень межах

кожного тимчасового вікна (процесором Mean Features Processor). Далі проводиться обробка набутих значень ознак процесорами дискретизації чи стандартизації. Отримані вектори ознак записуються в результуючий. csv-файл. Ідентифікатори використовуваних відібраних ознак, а також використовувані параметри алгоритмів обробки ознак записуються у відповідний конфігураційний файл і використовуються для подальшої класифікації нових тестових даних.

Цей програмний компонент реалізований на Python 3.5 спільно з модулями numpy, pandas, user-agents, scipy, scikit-learn, statsmodels, що дозволяють суттєво спростити код програми, прискорити її роботу за рахунок вбудованих методів розпаралелювання та надають широкі можливості для розрахунку статистики.

Програмний компонент побудови моделі користувача. Цей програмний компонент входить до складу модуля побудови індивідуальних моделей поведінки користувачів та складається із завдання побудови моделі користувача (Model Task). Основним завданням цього завдання є читання оброблених векторів ознак навчального набору даних, навчання моделі користувача за допомогою процесора побудови моделі та класифікації (Classifier), а також запис побудованої моделі в результуючий .pkl-файл. Основним завданням даного процесора є використання оброблених векторів ознак навчальної вибірки для навчання моделі (та наступної класифікації). Існують різні реалізації даного процесора: SVM Classifier (відповідний методу машинного навчання SVM (SVC)), Kernel PCA Classifier (відповідний методу машинного навчання Kernel PCA), Fuzzy Classifier (відповідний методу машинного навчання Fuzzy), RNN Classifier (відповідний методу машинного навчання RNN) та ESFC Classifier (відповідний методу машинного навчання ESFC).

Додатково, підтримуються перебудова та оновлення побудованої раніше(застарілої) моделі розглянутого користувача. Цей програмний

компонент реалізований на Python 3.5 спільно з модулями numpy, pandas, user-agents, scipy, scikit-learn, statsmodels, що дозволяють суттєво спростити код програми, прискорити її роботу за рахунок вбудованих методів розпаралелювання та надають широкі можливості для розрахунку статистик та аналізу даних.

3.3. Експериментальні дослідження

Продемонструємо роботу розробленого ПЗ на прикладі процесів поетапної побудови моделі та подальшої класифікації користувачів на основі аналізу їхнього клавіатурного почерку у відкладеному режимі.

Графічний інтерфейс додатку представлений на рисунку 3.5.

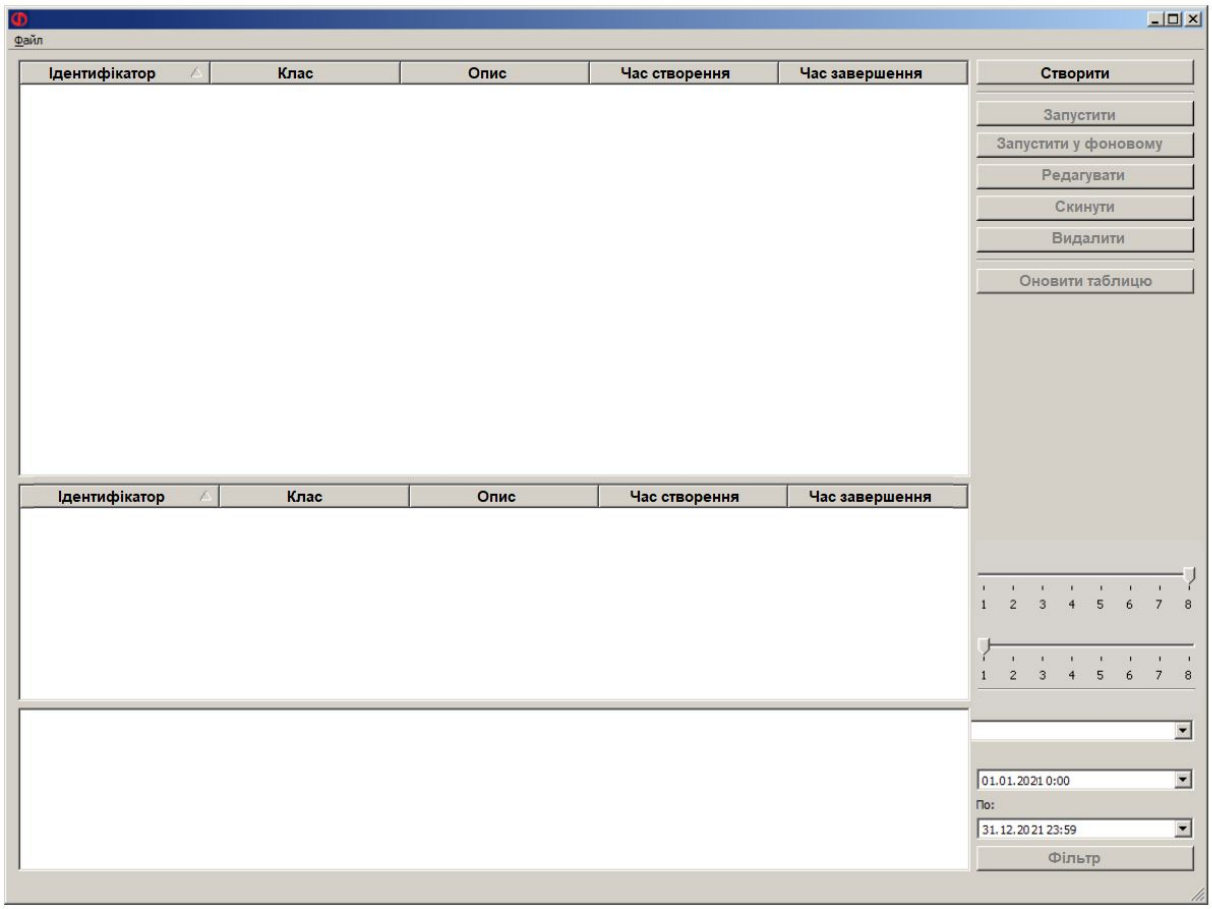


Рисунок 3.5 - Графічний інтерфейс

Тут бачимо такі області: область створених завдань і процесорів; область виконуваних на даний момент завдань; область виведення інформаційних повідомлень про роботу системи; панель керуючих кнопок для створення завдань, процесорів та черг завдань, а також панель для фільтрації відображуваних на даний момент завдань та процесорів.

Спочатку необхідно створити завдання поєднання даних динаміки роботи користувача із клавіатурою (Data Task). Для цього необхідно натиснути кнопку» створити», вибрати у вікні пункт «Data Task» і натиснути ОК. Далі, у новому вікно, що з'явиться, необхідно задати відповідні параметри даного завдання (див. рисунок3.6).

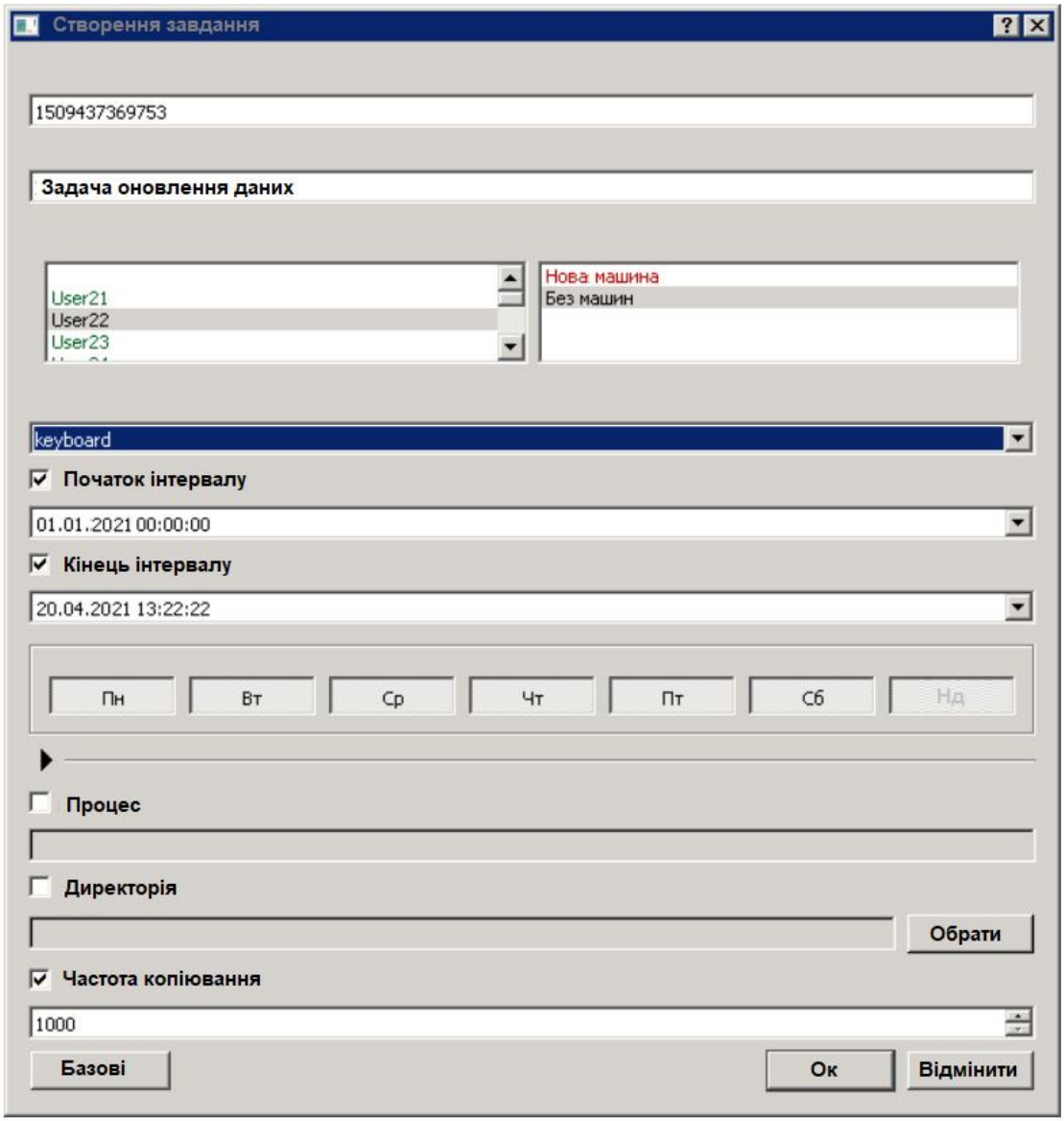


Рисунок 3.6 - Створення завдання об'єднання даних

Для запуску даного завдання необхідно його виділити, натиснувши на нього один раз лівою кнопкою миші, та натиснути на кнопку «Запустити». У ході роботи даного завдання робочу директорію запишуться відповідні даному завдання результуючі та конфігураційні файли. Після завершення роботи цього завдання, в інформаційне поле буде виведено відповідне повідомлення про результати його виконання.

Наступним етапом є створення процесора обробки даних (KeyboardData Processor), а також створення та запуск завдання обробки об'єднаних даних (DataProcess Task). При створенні процесора обробки об'єднаних даних необхідно незабути вказати опції перед обробки, що використовуються. Під час створення завдання Data ProcessTask необхідно вибрати відповідне завдання об'єднання даних, створене раніше, вибрати створений раніше процесор обробки даних та натиснути ОК (див. рисунок 3.7).

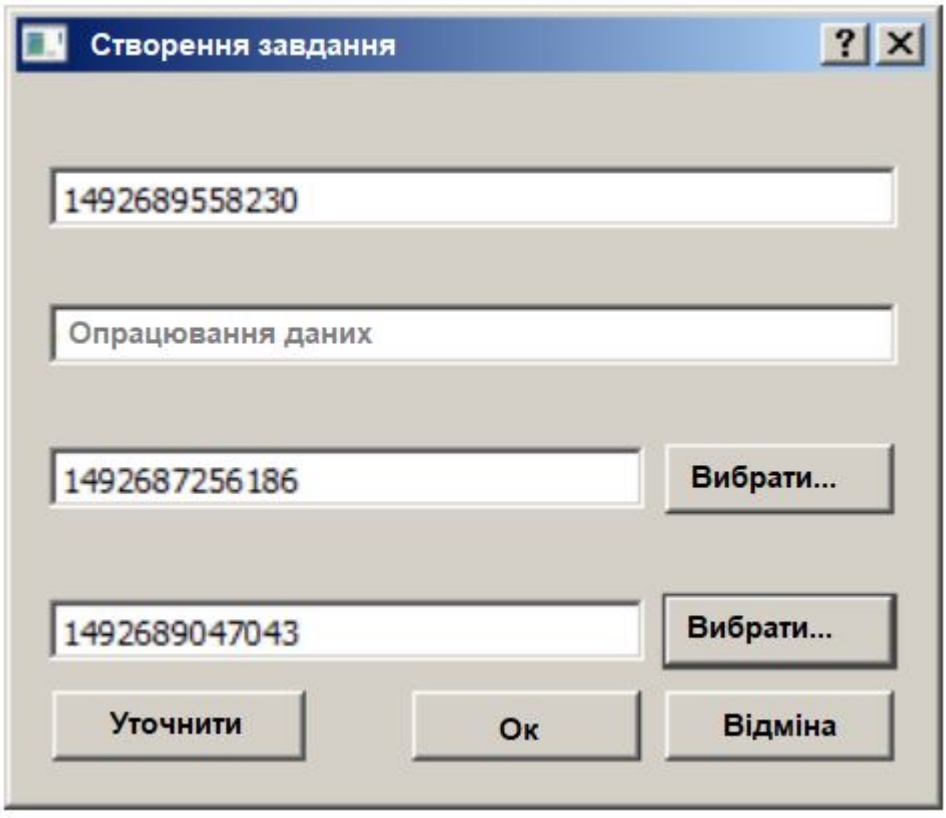
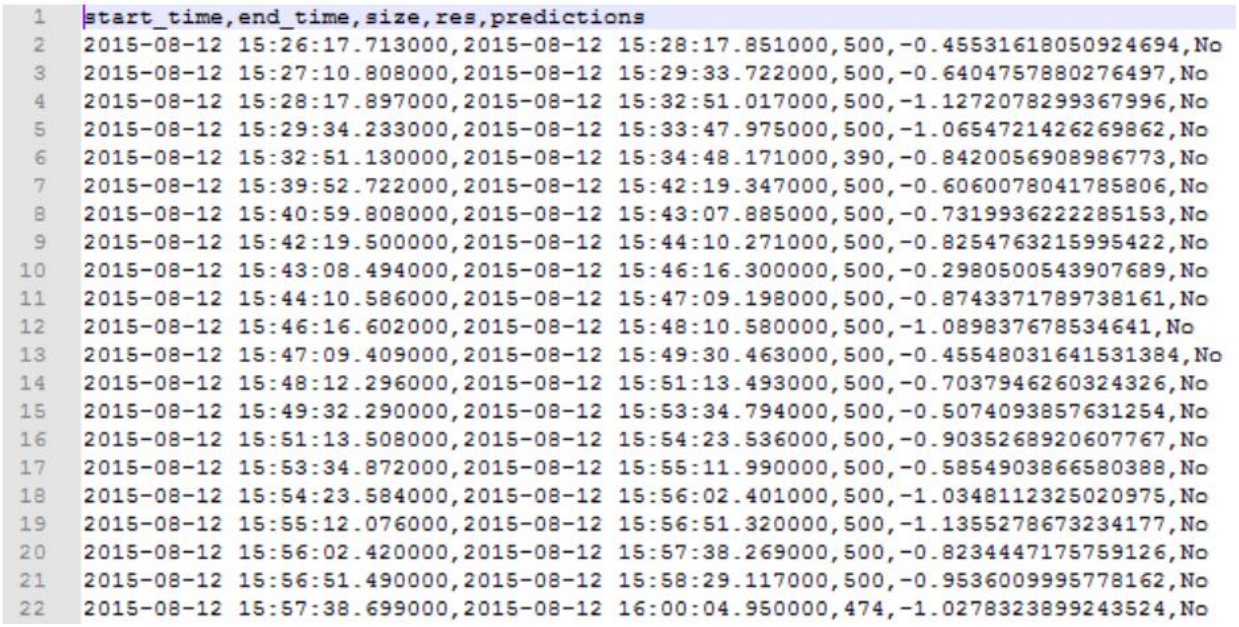


Рисунок 3.7 -. Створення завдання обробки об’єднаних даних

В результаті виконання завдання класифікації тестованого користувача ми отримаємо .csv-файл, що містить характеристики векторів, що тестуються тестованого користувача та результати їхньої класифікації (див. рисунок 3.8).



```
start_time,end_time,size,res,predictions
2015-08-12 15:26:17.713000,2015-08-12 15:28:17.851000,500,-0.4553161805092469 4,No
2015-08-12 15:27:10.808000,2015-08-12 15:29:33.722000,500,-0.640475788027649 7,No
2015-08-12 15:28:17.897000,2015-08-12 15:32:51.017000,500,-1.127207829936799 6,No
2015-08-12 15:29:34.233000,2015-08-12 15:33:47.975000,500,-1.065472142626986 2,No
2015-08-12 15:32:51.130000,2015-08-12 15:34:48.171000,390,-0.842005690898677 3,No
2015-08-12 15:39:52.722000,2015-08-12 15:42:19.347000,500,-0.606007804178580 6,No
2015-08-12 15:40:59.808000,2015-08-12 15:43:07.885000,500,-0.731993622228515 3,No
2015-08-12 15:42:19.500000,2015-08-12 15:44:10.271000,500,-0.825476321599542 2,No
2015-08-12 15:43:08.494000,2015-08-12 15:46:16.300000,500,-0.298050054390768 9,No
2015-08-12 15:44:10.586000,2015-08-12 15:47:09.198000,500,-0.874337178973816 1,No
2015-08-12 15:46:16.602000,2015-08-12 15:48:10.580000,500,-1.089837678534641,No
2015-08-12 15:47:09.409000,2015-08-12 15:49:30.463000,500,-0.455480316415313 84,No
2015-08-12 15:48:12.296000,2015-08-12 15:51:13.493000,500,-0.703794626032432 6,No
2015-08-12 15:49:32.290000,2015-08-12 15:53:34.794000,500,-0.507409385763125 4,No
2015-08-12 15:51:13.508000,2015-08-12 15:54:23.536000,500,-0.903526892060776 7,No
2015-08-12 15:53:34.872000,2015-08-12 15:55:11.990000,500,-0.585490386658038 8,No
2015-08-12 15:54:23.584000,2015-08-12 15:56:02.401000,500,-1.034811232502097 5,No
2015-08-12 15:55:12.076000,2015-08-12 15:56:51.320000,500,-1.135527867323417 7,No
2015-08-12 15:56:02.420000,2015-08-12 15:57:38.269000,500,-0.823444717575912 6,No
2015-08-12 15:56:51.490000,2015-08-12 15:58:29.117000,500,-0.953600999577816 2,No
2015-08-12 15:57:38.699000,2015-08-12 16:00:04.950000,474,-1.027832389924352 4,No
```

Рисунок 3.8 - Вміст файлу з результатами класифікації тестованого користувача

Структура файлу визначається наступними колонками:

1) Start_time – час приходу першої події у аналізованому векторі ознак;

2) End_Time – час приходу останньої події у аналізованому векторі ознак;

3) Size – розмір часового вікна, за яким був побудований аналізований вектор ознак;

4) Res – результат класифікації (величина відгуку класифікатора) для аналізованого вектора ознак;

5) Predictions – вердикт класифікатора (чи легітимний аналізований користувач).

Крім .csv-файлу з отриманими результатами класифікації,відповідному вікні ми також зможемо бачити графік схожості даних тестованого користувача з легітимним (рисунок 3.9).

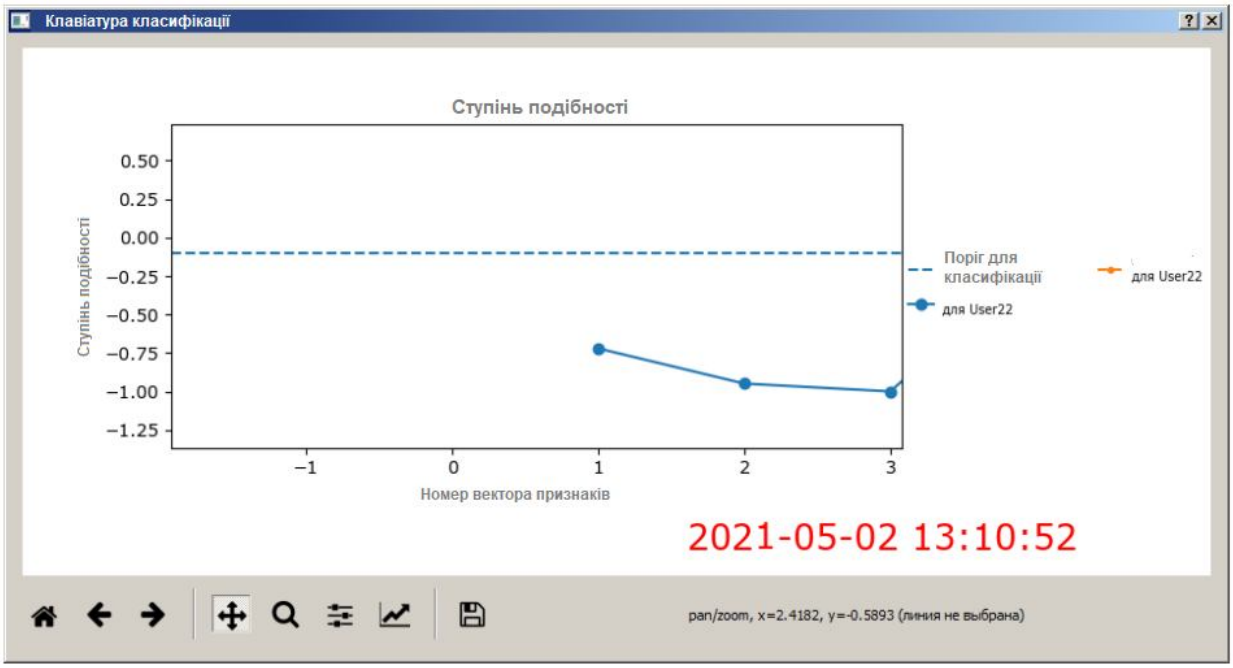


Рисунок 3.9 - Графік схожості даних тестованого користувача з легітимним

Зазначимо, що однією з головних особливостей динамічної системи аутентифікації є необхідність їх роботи в режимі, близькому до режиму реального часу. Збір даних та автентифікація користувачів повинні відбуватися в фоновому режимі непомітно для користувачів та не навантажувати систему: для користувача при роботі з комп'ютером не слід бачити жодних додаткових затримок.

Для перевірки непомітності роботи компонентів збору та аутентифікації користувачів,було залучено 10 незалежних експертів, на робочі комп'ютери яких було встановлено розроблене ПЗ. Кожен із експертів пропрацював за своїм комп’ютером порядку одного робочого дня і не помітив жодних додаткових затримок у роботі комп’ютера: реакція системи не відрізнялася від роботи при вимкнених компонентах ПК. Також додатковий аналіз показав, що всі клавіатурні події були оброблені своєчасно, і вся інформація про них була записана в необхідні файли в повному обсязі.

Висновки до третього розділу

Докладно описано програмну реалізацію розробленого програмного забезпечення (описано архітектуру системи, принцип роботи програмних компонентів, описаний графічний інтерфейс та наведено покроковий приклад використання системи). Оскільки розроблене ПЗ являє собою набір взаємопов'язаних між собою програмних агентів, кожен з яких виконує власне логічне завдання,дане ПЗ має властивості масштабованості (можливістю розподіляти програмні компоненти з різних фізичних машин) і розширюваності (можливістю додавати або замінювати окремі програмні модулі та компоненти).

РОЗДІЛ 4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Основні принципи конструювання робочого місця користувача ЕОМ.

Ергономіка (від грецьк. ἔργον наука про пристосування посадових– у традиційному розумінні –роботи») обов‘язків, робочих місць, обладнання та комп‘ютерних програм задля створення найбільш безпечних та ефективних умов праці для людини, виходячи з фізичних і психічних особливостей людського організму.

Більш широке визначення ергономіки, яке було прийняте в 2010 році Міжнародною асоціацією ергономіки (IEA) (Міжнародною ергономічною це наукова дисципліна, що вивчає–асоціацією), звучить так: «Ергономіка взаємодію людини та інших елементів системи, а також сфера діяльності щодо застосування теорії, принципів, даних і методів цієї науки для забезпечення благополуччя людини та оптимізації загальної продуктивності системи».

З цього визначення випливають такі головні завдання ергономіки:

1. Проведення досліджень, спрямованих на пристосування елементів системи "людина – трудовий процес" до природних фізичних і психічних можливостей працівника.

2. Прагнення до забезпечення таким шляхом умов для максимальної ефективності праці.

3. Прагнення запобігти всім можливим загрозам для здоров‘я працівника.

4. Прагнення до оптимальної витрати біологічних ресурсів у процесі праці.

Загальні ергономічні вимоги для організації робочого місця користувача ПЕОМ (ГОСТ 12.2.049-80, ГОСТ 122032-78, ГОСТ 22269-76). Ці вимоги встановлюють основні параметри робочого місця, оснащеного дисплеєм, і враховують особливість виконуваних робіт.

Параметри робочого місця повинні бути наступними.

Площа кабінету, в якому буде проходити робота повинна бути не менш 6 м², а об'єм не менш 24 м³. Для внутрішньої обробки приміщення повинні використовуватися дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі – 0,7-0,8; для стін – 0,5-0,6; для підлоги – 0,3-0,5.

Конструкція робочого столу повинна забезпечувати оптимальне розміщення на робочій поверхні використовуваного обладнання. Конструкція крісла повинна забезпечувати підтримку раціональної робочої пози під час роботи з відео-дисплейним терміналом (Далі ВДТ) і ПЕОМ, дозволяти змінювати позу з метою зниження статичного напруження м'язів шийно-плечової області і спини для попередження розвитку втоми працюючого (згідно з ГОСТ 12.2.032-78). Поверхня сидіння, спинки та інших елементів стільця (крісла) повинна бути напівм'якою, з покриттям, що не електризується, неслизьке та повітронепроникне, що забезпечує легке очищення від забруднення.

Висота робочої поверхні столу, за відсутності можливості її регулювання повинна складати 725 мм. Робочий стіл повинен мати простір для ніг висотою не менше 600 мм, шириною – не менше 500 мм, не менше 450 мм в глибину на рівні колін і на рівні простягнутої ноги – не менше 650 мм. Робоче місце має бути обладнане підставкою для ніг, має ширину не менше 300 мм, глибину не менше 400 мм, регулювання по висоті в межах 150 мм за кутом нахилу опорної поверхні підставки до 20 градусів.

Відстань від очей користувача до екрану дисплея має становити 500-700 мм. Кут зору 10-20°, але не більше 40°; кут між верхнім краєм дисплея і

рівнем очей користувача має становити не менше 10°. Кращим є розташування екрану перпендикулярно до лінії зору користувача.

Робочі місця по відношенню до світлових прорізів повинні розташовуватися не ближче 3 м так, щоб природне світло падало збоку, переважно зліва. Освітленість також впливає на стан здоров'я і працездатність людини. У відповідності зі СНіП 11-4-79 встановлені наступні вимоги до освітленості**:**

Для штучного освітлення**:**

- Комбіноване освітлення – освітленість 1500 лк;
- Загальне освітлення – освітленість 400 лк.

Для природного освітлення**:**

- Верхнє або комбіноване освітлення – коефіцієнт природної освітленості (далі КПО) 10%;
- Бічне освітлення – КПО 3.5%.

Для суміщеного освітлення**:**

- Верхнє або комбіноване освітлення – КПО 3-6%;
- Бічне освітлення – КПО 1.1-2%.

До основних показників, що визначають умови здорової роботи, належать: фон, контраст об'єкта з фоном, видимість, показник осліпленості, коефіцієнт пульсації освітленості.

Фон характеризується коефіцієнтом відбиття. Контраст об'єкта з фоном (К) характеризується співвідношенням яскравості розглянутого об'єкта (точки, лінії, знаки) і фону. Оскільки роботи користувача ПЕОМ відносяться до категорії 1а – легкі фізичні роботи (роботи проводяться сидячи і супроводжуються незначним фізичним напруженням, з енерговитратами до 120 ккал / годину), необхідно дотримуватися наступних норм: коефіцієнт відображення більше 0,4, тобто світлий фон; контраст об'єкта з фоном великий і середній при К більше 0,2 (згідно СНіП 11-4-79).

У полі зору користувача ПЕОМ має бути забезпечений відповідний розподіл яскравості. Відношення яскравості екрана до яскравості оточуючих його поверхонь не повинно перевищувати у робочій зоні 3:1 (СНіП 11-4-79). У зв'язку з цим дисплей ПЕОМ повинен відповідати наступним вимогам:

- Яскравість свічення екрану не менше 100 кд/м;
- Мінімальний розмір світної точки для кольорового дисплея не більше 0,6 мм ;
- Контрастність зображення знаку – не менше 0,8;
- Низькочастотне тремтіння зображення в діапазоні 0,05-1,0 Гц повинно знаходитися в межах 0,1 мм;
- Екран повинен мати покриття антивідблиску;
- Відеомонітор повинен бути обладнаний поворотним майданчиком, що дозволяє переміщати відеотермінал в горизонтальній і вертикальній площинах в межах 130-220 мм і змінювати кут нахилу на 10-15 мм.

Коефіцієнт відбиття світла матеріалами і обладнанням всередині приміщень має велике значення для освітлення: чим більше світла відбивається від поверхонь, тим вище освітленість. Коефіцієнт відображення відповідно повинен бути для: стелі 60-70%, стін 40-50%, підлоги 30%, для інших поверхонь 30-40%.

Результати досліджень показують, що найбільшою мірою негативний фізіологічний вплив на операторів ПК пов'язаний з дискомфортними зоровими умовами через неправильно спроектоване освітлення. Згідно СНіП II-4-79 освітленість на горизонтальній площині робочого місця оператора ЕОМ повинна складати 400 лк при висоті цій площині 0,8 м над підлогою.

4.2 Забезпечення захисту працівників суб’єкта господарювання від іонізуючих випромінювань

Іонізуюче випромінювання або радіоактивність є небезпечним явищем для людського організму. При взаємодії впливу іонізаційних випромінювань у навколишнє середовище можуть відбутись різні утворення зарядів . Існують два різновиди випромінювання – «альфа» та «бета».

В залежності від носія та енергії, вони мають різну проникаючу здатність. Альфа це випромінювання яке проявляється важкими частинами складеними з протонів і нейтронів.

В свою чергу бета випромінювання являє собою ланцюг електронів та позитронів які є більшу здатність проникати у середовище. Працюючи на таких територіях, де існує радіаційна атмосфера можуть виникнути різні випадки.

На підприємстві можуть виникнути інциденти при користуванні ядерними матеріалами , зберіганні радіоактивних відходів в наслідок чого працівники можуть отримати травму у вигляді дози опромінення, використання іонізуючих джерел випромінювання.

Також у випадку такої радіаційної аварії забруднюється навколишнє середовище, люди можуть отримати травму у вигляді потужної дози опромінення. Призвести аварію на підприємстві може також якщо активна реакційна речовина знаходиться у роботі та це відбувається незаконно.

Це може привезти до опромінення жителів та перевищити межу дози опромінення. Частинки з цього випромінювання можуть залишати сліди на дихальній системі на травній системі людського організму. Також ці елементи можуть бути у водних каналах, які постачають питну воду людям.

На підприємстві де проводяться роботи з радіаційними речовинами обов’язково мають вживатись заходи проти радіації. Протирадіаційні захисти це така система правових, організаційних норм та санітарної гігієни.

До переліку таких захистів можна включити медичні заходи для забезпечення радіаційної безпеки персоналу та проектно-конструкторські. Для організації заходів проти іонізації опромінювання підприємство має

ввести обов’язкові методи щоб подбати про безпеку працюючого персоналу. До таких методів можуть належати заходи які обмежують допуск працівників до джерел які випромінюють радіацію.

До таких працівників можемо віднести таких, які не підходять за віком, за статтю та працівники які вже отримали дозу випромінення. Підприємство мусить створити сприятливі умови що дотримуються встановлених норм та вимог для працівників та застосовувати індивідуальні засоби для захисту працівника цього підприємства.

Організація повинна контролювати рівні опромінювання та вести інформаційну систему про стан радіації на підприємстві та призначених місць для праці.

На підприємстві повинні бути проведені заходи щодо організації безпеки для робіт які проводяться у радіаційних ділянках а саме: -організація роботи нарядів та розпоряджень; -організація та перевірка пропусків до робочих місць; -оформлення контролю за процесом виконання роботи; -введення примусового часу на перерву та вчасне закінчення робочого процесу.

Реалізувати заходи проти радіації за певний відрізок часу можливо, тим що працівники , які працюють з іонізованими випромінюваннями можуть виконувати вчасно свою роботу ,відповідно керівництво може за якісну роботу зменшити кількість робочих днів у тижні.

Цим самим вони застереженням вони зменшать знаходження працівників у зоні випромінювання та відповідно буде менше контактування з радіаційними приладами. Захистити працівників за допомогою відстані підприємство може шляхом доцільного розміщення приміщення, правильно розставити та розрахувати робочі місця для працівників а також забезпечити приладами, які зможуть контактувати, керувати робочим процесом з технікою яка має радіаційний вплив на відстані.

Слугувати захистом може покриття свинцем меблів які присутні у приміщенні (двері, вікна, робочі столи), створення перекриття між поверхами та перегородки. Працівникам обов’язково має бути виданий спеціальний одяг ,такі як фартухи, шапочки та рукавиці зшиті з просвинцевої тканини.

Розміщення робочих місць повинно мати правильний розрахунок на загальну кімнату, не робити перенабір та забезпечити відповідним та необхідним обладнанням робочі кабінети. При користуванні відкритими приладами іонізованого опромінення провести герметизації цих систем, при можливості використовувати роботу техніки. Підприємство повинне вжити усіх санітарно-гігієнічних заходів та соціальних, а також важливо необхідний є медичний захист робочих на об’єкті.

Висновок до четвертого розділу

В даному розділі описано основні принципи конструювання робочого місця користувача ЕОМ, зазначено діючі вимоги щодо ергономіки робочого місця. А також визначені заходи та методи із забезпечення радіаційних впливів та іонізації опромінювання на підприємствах. Описані вимоги для керівництва та підлеглих працюючих на об’єктах щодо їхніх дій в разі виникнення радіації .

ВИСНОВКИ

1. Запропоновано підхід до підготовки даних, що описують клавіатурний почерк користувача, що включає спосіб побудови ознакового простору та підхід до подальшої обробки ознак на основі дискретизації їх за квантилями. Скорочення розмірності ознакового простору проводиться шляхом відбору найбільш значущих ознак з використанням критерію Колмогорова-Смирнова. Експериментально встановлено, що цей підхід дозволяє побудувати простір стабільних за часом о знакових показників;

2. Розроблено нечіткий метод виявлення аномалій у даних на основі еліптичної кластеризації (ESFC) у RKHS, що будує у просторі високої розмірності еліптичні області з оптимальним центром для виявлення аномалій.

Підбір оптимальних значень мета параметрів даного алгоритму здійснюється власне розробленим методом,тим, хто будує стабільні до зміни тестового набору даних, одно класові моделі без використання інформації про дані нелегітимного класу.

Оцінка аномальності поведінки користувача проводиться як закороткий, так і за тривалий період роботи – з використанням розробленого методу оцінки аномальності поведінки користувачів на основі аналізу цілих сесій роботи за комп'ютером з використанням статистики Уелша.

За результатами експериментів метод ESFC перевершив якість розпізнавання існуючих алгоритмів при класифікації як окремих векторів ознак, і цілих сесій роботи користувачів за комп’ютером;

3. Розроблено архітектуру, реалізовано та апробовано експериментальний зразок мультиагентного програмного комплексу, який використовує запропонований комплекс алгоритмів для виявлення

аномального поведінки користувачів щодо особливостей роботи з клавіатурою комп’ютера.

Проведені на його основі експериментальні дослідження підтвердили якість та обґрунтували достовірність отриманих результатів.

СПИСОК ВИКОРИСТАНИХДЖЕРЕЛ

1. Peltier, T. R. Information security risk analysis, Third Edition. / T. R. Peltier. – CRC Press, 2020. 456 p.

2. Olsson, T. Assessing security risk to a network using a statistical model ofattacker community competence / T. Olsson // Proceedings of the 11th internationalconference on Information and Communications Security. – 2019. – P. 308–324.

3. Peltier, T. R. Information security risk analysis, Third Edition. / T. R. Peltier. –CRC Press, 2020. 456 p.

4. Poolsappasit, N. Dynamic security risk management using Bayesian attackgraphs / N. Poolsappasit, R. Dewri, I. Ray // IEEE Transactions on Dependable andSecurity Computing. – 2012. – Vol.9, No.1 – P. 61–74.

5. Toth, T. Evaluating the impact of automated intrusion response mechanisms / T. Toth, C. Kruegel // Proceedings of the 18th Annual Computer Security Applications Conference (ACSAC). – 2020.– P. 301–310.

6. Gibellini, E.; Righetti, C. Unsupervised Learning for Detection of Leakage from the HFC Network. In Proceedings of the ITUKaleidoscope: Machine Learning for a 5G Future (ITU K), Santa Fe, Argentina, 26–28 November 2018; pp. 1–8. [CrossRef]

7. Baek, M.; Song, J.; Jung, J. Design and Performance Verification of Time-Domain Self-Interference Estimation Technique forDOCSIS 3.1 System with Full Duplex. In Proceedings of the IEEE International Symposium on Broadband Multimedia Systemsand Broadcasting, Valencia, Spain, 6–8 June 2018; pp. 1–4. [CrossRef]

8. Конспект лекцій з дисципліни «Програмування для мобільних пристроїв» для студентів денної форми навчання спеціальності 126

«Інформаційні системи та технології» / Укладачі: Готович В.А., Михайлович Т.В. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2020. – 216 с.

9. Коноваленко І. В. Платформа .NET та мова програмування C# 8.0 : навчальний посібник / І. В. Коноваленко, П. О. Марущак. – Тернопіль : ФОП Паляниця В. А., 2020. – 320 с.

10. Комп'ютерні мережі: [навчальний посібник] / А. Г. Микитишин, М. М. Митник, П. Д. Стухляк, В. В. Пасічник. — Львів: «Магнолія 2006», 2013. — 256 с. ISBN 978-617-574-087-3

11. Петрик М.Р. Моделювання програмного забезпечення : науково методичний посібник / М.Р. Петрик, О.Ю. Петрик – Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2015. – 200 с.

12. Baek, M.; Song, J.; Kwon, O.; Jung, J. Self-Interference Cancellation in Time-Domain for DOCSIS 3.1 Uplink System with FullDuplex. IEEE Trans. Broadcast. 2019, 65, 695–701.

13. Li, Z.; Zhao, Y.; Li, Y.; Rahman, S.; Yu, X.; Zhang, J. Demonstration of Fault Localization in Optical Networks Based on KnowledgeGraph and Graph Neural Network. In Proceedings of the Optical Fiber Communications Conference and Exposition (OFC 2020),San Diego, CA, USA, 8–12 March 2020; pp. 1–3. [CrossRef]

14. Gray, W.; Tsokanos, A.; Kirner, R. Multi-Link Failure Effects on MPLS Resilient Fast-Reroute Network Architectures. InProceedings of the International Symposium on Real-Time Distributed Computing (ISORC 2021), Daegu, Korea, 1–3 June 2021;pp. 29–33. [CrossRef]

15. Dusia, A.; Sethi, A.S. Recent Advances in Fault Localization in Computer Networks. IEEE Commun. Surv. Tutor. 2016, 18,3030–3051. [CrossRef]

16. Zych, P. Network failure detection based on correlation data analysis. Int. J.Electron. Commun. 2017, 77, 27–35. [CrossRef]

17. Ab-Rahman, M.S.; Chuan, N.B.; Safnal, M.H.G.; Jumari, K. The overview of fiber fault localization technology in TDM-PONnetwork. In Proceedings of the International Conference on Electronic Design, Penang, Malaysia, 1–3 December 2008; pp. 1–5.[CrossRef]

18. . Data and file storage overview [Електронний ресурс] // Android Developer – Режим доступу до ресурсу: https://developer.android.com/guide/topics/data/data-storage#filesInternal.

19. Man-in-the-Disk: Android Apps Exposed via External Storage [Електронний ресурс] // CHECK POINT RESEARCH. . – Режим доступу до ресурсу: https://research.checkpoint.com/androids-man-in-the-disk/.

20. Man-in-the-Disk: A New Attack Surface for Android Apps [Електронний ресурс] // Check Point Blog. – Режим доступу до ресурсу: 71 https://blog.checkpoint.com/2018/08/12/man-in-the-disk-a-new-attacksurface-for-android-apps/.

21. Hadi, M.S.; Lawey, A.Q.; El-Gorashi, T.E.H.; Elmirghani, J.M.H. Big data analytics for wireless and wired network design:A survey. Comput. Netw. 2018, 132, 180–199. [CrossRef]

22. Chih-Lin, I.; Liu, Y.; Han, S.; Wang, S.; Liu, G. On Big Data Analytics for Greener and Softer RAN. IEEE Access 2015, 3, 3068–3075.[CrossRef]

23. Simakovic, M.; Masnikosa, I.; Cica, Z. Performance monitoring challenges in HFC networks. In Proceedings of the TELSIKS 2017,Nis, Serbia, 18–20 October 2017; pp. 385–388

24. Installer downloads are vulnerable to hijacking [Електронний ресурс] // Google Issue Tracker. – 2018. – Режим доступу до ресурсу: https://issuetracker.google.com/issues/112630336

25. MacOS X GateKeeper Bypass [Електронний ресурс] // Filippo Cavallarin. – 2019. – Режим доступу до ресурсу: https://www.fcvl.net/vulnerabilities/macosx-gatekeeper-bypass.

26. Android Q privacy change: Scoped storage [Електронний ресурс] // Developers Android – Режим доступу до ресурсу: https://developer.android.com/preview/privacy/scoped-storage.

27. Створення месенджерів для навчального закладу: [Електронний ресурс] – Режим доступу до ресурсу: https://vrc.rv.ua/case_study/vetmarketing/

28. Пошук цільової аудиторії: [Електронний ресурс] – Режим доступу до ресурсу: https://creativesmm.com.ua/jak-znajjtu-svoju-cilovuauditoriju/

29. Application Sandbox [Електронний ресурс] // Source Android – Режим доступу до ресурсу: https://source.android.com/security/app-sandbox.

30. Data and file storage overview [Електронний ресурс] // Android Developer – Режим доступу до ресурсу: https://developer.android.com/guide/topics/data/data-storage#filesInternal.

31. Man-in-the-Disk: Android Apps Exposed via External Storage [Електронний ресурс] // CHECK POINT RESEARCH. . – Режим доступу до ресурсу: https://research.checkpoint.com/androids-man-in-the-disk/.

32. Man-in-the-Disk: A New Attack Surface for Android Apps [Електронний ресурс] // Check Point Blog. – Режим доступу до ресурсу: 71 https://blog.checkpoint.com/2018/08/12/man-in-the-disk-a-new-attacksurface-for-android-apps/.

33. DEF CON 26 - Slava Makkaveev - Man In The Disk [Електронний ресурс] // DEFCONConference YouTube channel. – Режим доступу до ресурсу: https://www.youtube.com/watch?v=vvfs0u1or3M.

34. DIKIDI [Електронний ресурс]. – Режим доступу до ресурсу: https://play.google.com/store/apps/details?id=ru.dikidi&hl=uk– 14.04.2020р.

35. DIKIDI [Електронний ресурс]. – Режим доступу до ресурсу: https://beauty.dikidi.ru/uk/ – 13.04.2020р. – Назва з титулу екрана.

36. Велюр [Електронний ресурс]. – Режим доступу до ресурсу: http://velurspa.com.ua/ – 21.04.2020р. – Назва з титулу екрана

37. MySQL [Електронний ресурс]. – Режим доступу до ресурсу: https://metanit.com/sql/mysql/1.1.php

38. Відгуки та презентації інструментів розробки програмного забезпечення [Електронний ресурс]. Режим доступу: http://www.methodsandtools.com/tools/tools.php. – Назва з екрану.

39. HTML [Електронний ресурс]. – Режим доступу до ресурсу: https://metanit.com/web/html5/1.1.php

40. Коли тестування повинно бути автоматизованим? [Електронний ресурс]. Режим доступу: https://www.stickyminds.com/article/when-should-testbe-automated– Назва з екрану.

41. Understanding and Working With Data in WordPress [Електронний ресурс]. – Режим доступу до ресурсу: https://code.tutsplus.com/tutorials/understanding-andworking-with-data-in-wordpress--cms-20567

42. APACHE [Електронний ресурс]. – Режим доступу до ресурсу: https://www.atlantic.net/what-is-an-apache-server/

43. « $_SERVER['HTTP_REFERER'] [Електронний ресурс]. – Режим доступу до ресурсу: https://www.php.net/manual/en/reserved.variables.server.php

44. mysqli_query [Електронний ресурс]. – Режим доступу до ресурсу: https://www.php.net/manual/en/mysqli.query.php

45. mysqli_fetch_array [Електронний ресурс]. – Режим доступу до ресурсу: https://www.php.net/manual/en/function.mysql-fetch-array.php

46. Стандарти мобільного зв’язку. [Електронний ресурс] – Режим доступу:https://www.itbox.ua/ua/blog/Standarti-mobilnogo-zvyazku-4G5G6G--podibnist-vidminnosti-perspektivi/(17.03.2022);

47. Технології захисту інформації.[Електронний ресурс] – Режим доступу:https://ela.kpi.ua/bitstream/123456789/23896/1/TZI_book.pdf(05.03.2022);

48. Технології захисту інформації. [Електронний ресурс] – Режим доступу: http://kist.ntu.edu.ua/textPhD/tzi.pdf(20.04.2022);

49. Маршутизатори. Методи маршутизації. [Електронний ресурс] – Режим доступу:https://studopedia.com.ua/1_13250_marshrutizatori.html (14.04.2022);

50. Мережі передачі даних. [Електронний ресурс] – Режим доступу: https://rci-c.com/technology/merezhi/(20.03.2022);

51. Клієнт серверна архітектура та ролі серверів. [Електронний ресурс] – Режим доступу: https://medium.com/@IvanZmerzlyi/клієнт-серверна-архітектура-та-ролі-серверів-9893d8048229

52. Клієнт серверні технології. [Електронний ресурс] – Режим доступу: https://crashbox.ru/solving-problems/klient-servernye-tehnologii-ispolzovanie-tehnologii-klient-server-v-tehnologii-klient-server-klie/(08.03.2022);

ДОДАТКИ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ

М А Т Е Р І А Л И

X НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

7–8 грудня 2022 року

ТЕРНОПІЛЬ
2022

ЗМІСТ

УДК 004.031.6
О. Багрій
(Тернопільський національний технічний університет імені Івана Пулюя)

ОСОБЛИВОСТІ РЕАЛІЗАЦІЇ ЗБОРУ ТА ОПРАЦЮВАННЯ ДАНИХ ДЛЯ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ КЛАВАТУРНОГО ПОЧЕРКУ

UDC 004.031.6
O. Bagriy

FEATURES OF IMPLEMENTATION OF DATA COLLECTION AND PROCESSING FOR KEYBOARD-BASED USER AUTHENTICATION

Оскільки динамічна автентифікація користувачів за клавіатурним почерком може широко застосовуватися і на web-сайтах, і в самостійних локальних додатках, у відомих рішеннях, розглядається як здійснюваний web-браузером web-збір даних, що характеризують динаміку роботи користувачів з клавіатурою, і локальний збір даних засобами операційної системи комп'ютера.

Технології, що використовуються для web-збирання даних, що характеризують динаміку роботи користувача з клавіатурою, можна поділити на дві категорії залежно від використання плагінів – незалежних програмних модулів, динамічно підключаються до web-браузера і призначені для розширення його функціональних можливостей. До програмних засобів, що не використовують плагіни при web-зборі,відносяться вбудовувані в web-сторінки JavaScript-програми, а також розширення веб-браузера. До програмних засобів, робота яких ґрунтується на використанні у системі спеціальних плагінів, відносяться такі широко відомі технології як Flashта Java-аплети. Взаємодія web-браузера з плагіном здійснюється через APIінтерфейс. Серед найпопулярніших APIінтерфейсів виділяють NPAPI (Netscape Plugin Application Programming Interface),PPAPI (Pepper Plugin Application Programming Interface) та ActiveX. Проте, поступово Internet Explorer відмовляється від використання NPAPI та ActiveX, а інтерфейсPPAPI підтримується лише web-браузерами Google Chrome та Opera, що свідчить у тому, що немає універсального рішення. Відмітимо, щонайчастіше використовується для web-збору та перспективними сьогодні є технологія JavaScript.

JavaScript-програми. Вбудовувані на web-сторінки JavaScript-програми виконуються на стороні клієнта і тим самим можуть взаємодіяти із зовнішніми ресурсами, збираючи інформацію про їхнє використання без використання додаткового ПЗ.

Однак, варто мати на увазі, що у різних web-браузерах можуть використовуватись різні версії JavaScript-інтерпретатора, що необхідно враховувати під час розробки. Також,автори говорять про великі затримки при обробці подій натискання на клавіші клавіатури при використанні технологій web-збору, тривалість яких істотно залежить від ступеня завантаженості комп'ютера. Час, що минає від моменту натискання на клавішу до виклику обробника цієї події, може становити від десятків до сотень мілісекунд, що значно нижче швидкості обробки подій локальних збирачів даних засобами операційної системи. Також варто зазначити, що при використанні JavaScript-технологій в більшості web-браузерів немає можливості визначити, ліві чи праві функціональні клавіші (Shift, Ctrl, Alt і т.д.) були натиснуті, що є серйозним недоліком, оскільки дана інформація може суттєво допомогти автентифікувати користувача.

Розширення для веб-браузера. Розширення для web-браузера є програмами, що розширюють його функціональні можливості. На відміну від розглянутих вище JavaScript програм, що вбудовуються в коди web-сторінок, розширення для web-браузеранадають можливість збору даних, що характеризують динаміку роботи користувача з клавіатурою, під час перегляду будь-яких web-сторінок, а не тільки тих, у коди яких заздалегідь вшиті JavaScript дані. Це обґрунтовується здатністю розширень модифікувати код веб-сторінок, що переглядаються. В

цьому полягає їхня головна відмінність від плагінів. Однак, варто мати на увазі, що різні браузери надають різні програмні інтерфейси для написання розширень (у тому числі і вимагають реалізації різними мовами програмування), що значно ускладнює створення універсальних рішень.

Flash – це мультимедійна платформа компанії Adobe Systems, призначена для створення інтерактивних web-додатків з багатою векторною,растровою, тривимірною комп'ютерною графікою та мультимедіа, що працюють як усередині,так і поза веб-браузером. Для створення програм використовується власне розроблена мова ActionScript. Adobe Flash використовується у web-браузерах Opera таGoogle Chrome (за допомогою інтерфейсу PPAPI), а також у веб-браузері Firefox(за допомогою інтерфейсу NPAPI). Варто відзначити, що використання Flash сильно уповільнює роботу браузера, внаслідок чого попит на цей продукт поступово знижується.

Зауважимо, що при використанні технології Flash зібрати дані динаміки роботи користувача з клавіатурою вийде тільки всередині Flash-об'єкта.

Література

1. N. Poolsappasit, R. Dewri, I. Ray Dynamic security risk management using Bayesian attackgraphs. IEEE Transactions on Dependable andSecurity Computing. 2012. Vol. 9. No.1. P. 61–74.
2. T. Toth, C. Kruegel Evaluating the impact of automated intrusion response mechanisms. Proceedings of the 18th Annual Computer Security Applications Conference (ACSAC). 2020. P. 301–310.
3. Gibellini, E.; Righetti, C. Unsupervised Learning for Detection of Leakage from the HFC Network. In Proceedings of the ITUKaleidoscope: Machine Learning for a 5G Future (ITU K), Santa Fe, Argentina, 26–28 November 2018; pp. 1–8. [CrossRef].

УДК 004.031.6
О. Багрій
(Тернопільський національний технічний університет імені Івана Пулюя)

АНАЛІЗ ЗАСОБІВ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ КЛАВАТУРНОГО ПОЧЕРКУ

UDC 004.031.6
O. Bagriy

ANALYSIS OF KEYBOARD-BASED USER AUTHENTICATION MEANS

На сьогоднішній день кількість рішень щодо аутентифікації користувачів на основі динаміки їхньої роботи з клавіатурою персонального комп'ютера постійно збільшується. І якщо раніше дані системи обмежувалися лише аналізом введення пари логін/пароль (розглядалася виключно статична аутентифікація), то зараз активно розвиваються системи,здатні аналізувати поведінку користувача за комп'ютером безперервно.

BehavioWeb (рисунок 1). Одним з найбільш відомих комерційних рішень у галузі безперервної фонової аутентифікації користувачів за клавіатурним почерком є продукт BehavioWeb компанії BehavioSec. Для аналізу поведінки користувача у ньому використовуються ритм та швидкість набору тексту, а також сила натискання на клавіші.

Програмне забезпечення вбудовується у веб-сайт або додаток. Для цього використовуються JavaScript-бібліотека, що поставляється, а також J2EE-модуль, що вбудовується bweb-сервер для здійснення процедури автентифікації. Розробники звертають увагу на те, що їхнє рішення аналізує зміну характеристик введення користувача з часом і періодично оновлює модель користувача. Проте,алгоритми, що використовуються для цього, не називаються.

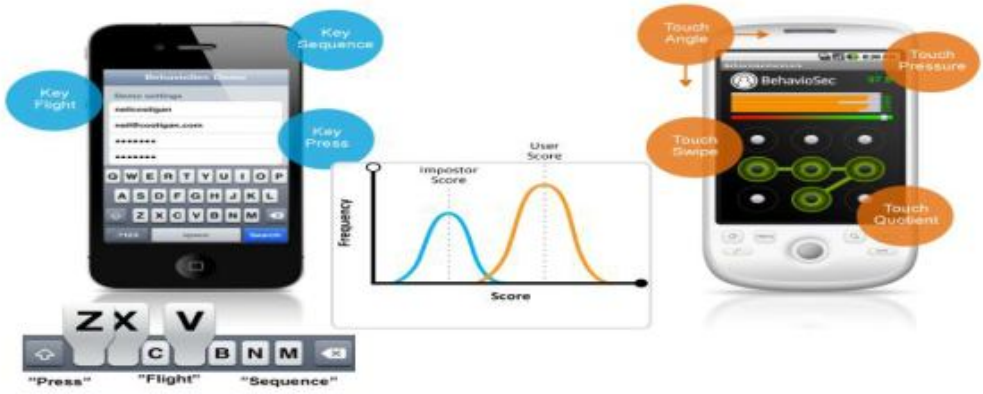


Рисунок 1. Система BehavioWeb

KeyTrac (рисунок 2). Не менш популярним рішенням є продукт KeyTrac [4, 5],що дозволяє здійснювати у фоновому режимі як автентифікацію, так ідентифікацію користувачів комп'ютера, ґрунтуючись на динаміці їх клавіатурного введення. Дані користувачів (тривалість натискання на клавіші клавіатури, а також тривалості перескоку між клавішами) записуються за допомогою компонента KeyTrac Recorder і відправляються на сервер компанії, де відбувається їх порівняння із побудованою раніше моделлю. При цьому побудова моделі користувача здатна здійснюватися на будь-якому довільному тексті, а не тільки при багаторазовому введенні тих самих фраз. Для передачі даних використовується наданий KeyTrac API. Далі сервер повертає свій вердикт у вигляді булевоївеличини true/false – чи

відповідають надіслані тестові дані розглянутому легітимного профілю чи ні. Для вбудовування цього рішення на веб-сайт також пропонується використовувати JavaScript-бібліотеку, що надається. Розробники системи стверджують, що їх рішення нечутливе до зміни мови, що використовується, а також до зміни обладнання, що використовується. Використовувані для цього алгоритми, а також методи побудови моделі та їх подальшої класифікації не називаються.

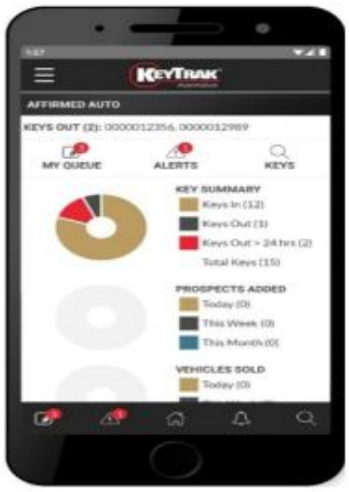


Рисунок 2. Система KeyTrac

Література

1. Peltier, T. R. Information security risk analysis, Third Edition. CRC Press, 2020. 456 p.
2. Olsson, T. Assessing security risk to a network using a statistical model ofattacker community competence. Proceedings of the 11th internationalconference on Information and Communications Security. 2019. P. 308–324.
3. Peltier, T. R. Information security risk analysis, Third Edition. CRC Press, 2020. 456 p