УДК 004.048 Вестник УрФО № 1(27) / 2018, с. 19–23

Власенко А. В., Бердник М. В., Швырев Б. А.

ИССЛЕДОВАНИЕ ПРОГРАММНЫХ РЕШЕНИЙ РЕГИСТРАЦИИ И ЗАПИСИ СОБЫТИЙ КЛАВИАТУРЫ

В данной статье рассматривается вопрос идентификации пользователя по его уникальному клавиатурному подчерку. В частности, рассматриваются различные механизмы идентификации, такие как скорость письма, сила нажатия на клавиши и другие способы. Представлен алгоритм процесса регистрации времени нажатия и отпускания клавиши. Изложены основные методы регистрации состояния клавиш.

Ключевые слова*: клавиатурный почерк, идентификация, события клавиатуры, методы регистрации.*

Vlasenko A. V., Berdnik M. V., Shvyrev B. A.

RESEARCH OF SOFTWARE SOLUTIONS FOR REGISTRATION AND RECORDING OF KEYBOARD EVENTS

This article discusses the problem of identifying a user using a unique key combination. In particular, various identification mechanisms are considered, such as letter speed, keystrokes and other methods. The algorithm of the process of recording the time of pressing and releasing the key is presented. The main methods for registering the state of keys are described.

Keywords: *keyboard handwriting, identification, keyboard events, registration methods.*

Использование интернет-мессенджеров в личной и деловой переписке является реалиями сегодняшней жизни. Не смотря на все плюсы информационного обмена в виртуальной среде, одной из основных проблем является идентификация пользователя, находящегося по ту сторону экрана. На сегодняшний момент существует достаточно много технологий позволяющих как определить, так и обойти получение информации о месте нахождения, конечном адресе устройства, с которого передается информация.

Наиболее интересным направлением, с нашей точки зрения, в области идентификации пользователя, является анализ клавиатурного почерка. Важность и точность этого метода в нецифровой среде подтверждена наличием такой науки как графология, суще-

ствованием графологической экспертизы, исследований, связанных с определением по почерку образа жизни, пола, возраста, профессии человека.

Большинство исследований в области анализа клавиатурного почерка были связаны с оценкой особенности работы профессиональных наборщиков текста. Сейчас, когда социальные сети становятся достаточно существенным инструментом, в том числе и информационного противоборства, важно разработать механизм, позволяющий идентифицировать лицо, отправляющее короткие сообщения (размером до 300 знаков).

Одним из достоинств идентификации пользователя по клавиатурному подчерку является использование стандартной клавиатуры, подключенной к персональной ЭВМ стандартным интерфейсом PS\2 или USB. Устройства регистрации биометрических данных пользователей обладают высокой стоимость ограничивающей их применимость в системах безопасности. К тому же эти устройства используют специальные интерфейсы и контроллеры для передачи данных в компьютер, использование которых сопряжено с трудностями установки, настройки и калибровки, а также ограничивает мобильность устройства идентификации. Клавиатура как устройство биометрической идентификации лишено этих недостатков. Для ввода информации не требуется дополнительных аппаратных преобразователей.

С целью повышения информативности биометрической информации пользователя предложено использовать дополнительный контроллер, отслеживающий параметры надавливания клавиши. При надавливании происходит изменение расстояния между контактами электрической группы конечной площади. Контактная пара рассматривается как параметрическая емкость, изменяемая при нажатии на клавишу. Скорость изменения емкости клавиш при нажатии имеет индивидуальную составляющую. Мы предлагаем использовать этот параметр для повышения достоверности идентификации пользователей.

Целесообразность использования предложенного устройства вызвана тем, что контроллер стандартной клавиатуры выполняет функцию компаратора и обрезает временную форму длительности импульса нажатия клавиши. Для борьбы с дребезгом контактов и снижения ошибок интерпретации контроллер вводит временную задержку.

Создание дополнительного контроллера приведет к увеличению стоимости устройства, что может негативно сказать на востребованности метода.

Анализ источников показал, что большинство исследований клавиатурного подчерка посвящено изучению индивидуальных особенностей пользователей на основе интервалов времени между нажатиями клавиш и длительности удержания клавиш. По измеренным значениям интервалов находились производные характеристики, такие как скорость нажатия одной или группы клавиш, среднее значение интервала между нажатиями, длительности перекрытий клавиш, длительность биграмм и триграмм и т. д. Исследователи в меньше степени уделяли внимание анализу длительности удержания клавиши и построению модели этого процесса, при этом рядом исследователей отмечается высокая информативность этих параметров для идентификации пользователей.

Все события клавиатуры, подключенной к компьютеру, регистрируются программными средствами. Процесс формирования массива экспериментальных данных связан с определением точности предполагаемых измерений. Для этого проанализируем исследуемый процесс набора текста на клавиатуре. Набор могут осуществлять две основные категории пользователей, обладающие навыком слепого набора или десятипальцевым способом и пользователи, печатающие одним пальцем. Считается что пользователь, обладающий десяти пальцевым набором текста, обладает большей вероятностью обнаружения, чем не имеющий такого навыка. Такие суждения справедливы в рамках распространенных моделей идентификации пользователей по средним значениям интервалов времени между нажатиями клавиш. Исходя из физиологических, анатомических и психических особенностей человека, мелкая моторика и динамика каждого индивида уникальна. Эти особенности обладают большим порядком малости, и для их описания и регистрации требуется использование максимально возможной точности измерения временных интервалов нажатия и отпускания клавиш. Скорость набора профессиональной машинистки составляет порядка 500–600 символов в минуту, в предположении последовательного набора символов интервал между нажатиями должен составлять 8,3 мс, без перекрытий, с минимальным физиологически объяснимым

временем удержания клавиши 50 мс. Для регистрации этих событий необходимо обеспечить регистрацию событий клавиатуры с интервалом порядка 4мс. Обычные пользователи, как и опытные наиболее вероятно нажимают клавиши с интервалом от 30 до 400 мс.

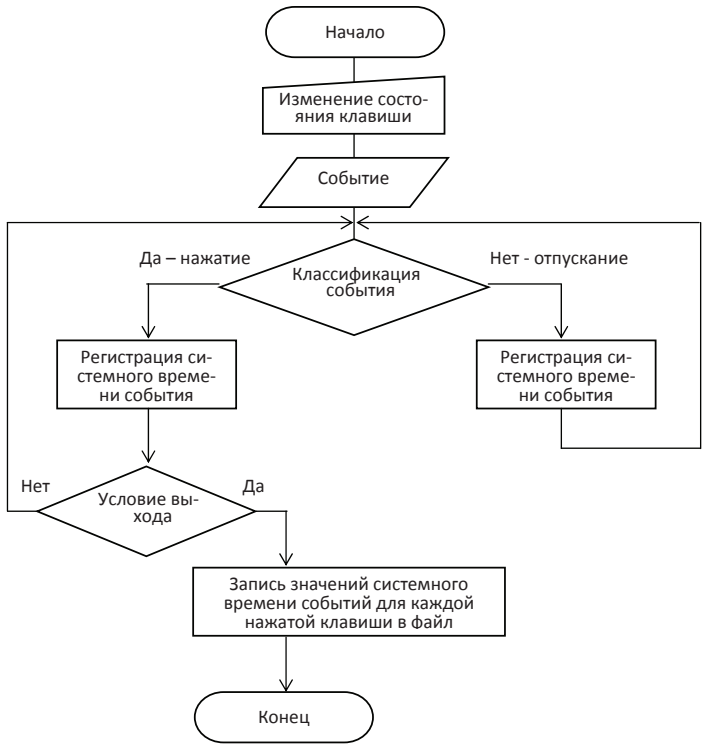
Рассмотрим программные решения для регистрации и записи событий клавиатуры, написанные на высокоуровневых языках программирования.

Клавиатура по своему назначению разрабатывалась как интерфейс с пользователем для ввода информации в компьютер с низкой скоростью. Это условие отразилось во всех языках программирования. От клавиатуры в стандартных и офисных программных продуктах не требуется высокая скорость передачи данных. Низкая скорость опроса клавиатуры компенсировалась наличием кольцевого буфера. Ситуация изменилась с развитием компьютерных игр, когда пользователь должен адекватно реагировать на очень динамичные изменения в виртуальном пространстве. Так игровая индустрия способствовала развитию программных средств DirectInput.

Методы регистрации временных характеристики ввода с клавиатуры, допускают реализации на разных языках программирования с вариантами внутренних алгоритмов и функций отсчета времени, фиксации измерений в файле. Для анализа возможности фиксации событий клавиатуры и оценки погрешности отсчетов разработаны три программы на разных языках высокоуровневого программирования реализующие основные методы сбора событий клавиатуры.

Общий алгоритм процесса регистрации времени нажатия и отпускания клавиши представлен на рисунке. Основные процессы алгоритма отследить нажатую клавишу, определить значение текущего времени и записать данные в файл. Алгоритм заключается в ожидании либо нажатия клавиши, либо её отпускания. При наступлении каждого из этих событий фиксируется системное время.

Для регистрации состояния клавиш используют три основных метода. Каждый метод использует обработку процессов на одном из трех основных этапах получения данных от клавиатуры приложением Windows. Каждый раз при нажатии на клавиатуру формируется событие Windows и передается запрашиваемому приложению, которое находится в фокусе. Для обслуживания клавиатуры в Windows существует специальный драйвер в виде файла динамической библиотеки с расширением dll. Он определяет скан код нажатой и отпущенной клавиши и преобразует его в ANSI код. Затем используется для форми-



Алгоритм сбора экспериментальных значений времени нажатия и отпускания клавиш

рования событий wm_keydown и wm_keyup, которые становятся в очередь событий для транспортировки их приложению в фокусе.

Основной способ регистрации, используемый большинством приложений, является обработка готовых сообщений Windows, таких как wm_keydown и wm_keyup находящихся в очереди событий. Для реализации такого приема написана программа на языке высоко уровня программирования C++. Она содержит обработку событий Windows и запись результатов в файл в формате txt.

Следующим подходом является использование setWinHook или программ перехвата событий Windows и минуя очередь событий запись в выходной файл. Для реализации такого приема написана программа на языке высоко уровня программирования Delphi 7.

Описанные выше приемы используют стандартный драйвер клавиатуры, предназначенный для своевременного нажатия клавиши обработки кольцевого буфера клавиатуры на случай невозможности своевременной обработки, а также отображения удержания служебных и специальных клавиш. Начиная с первых клавиатур персональных компьютеров драйвер, осуществлял регистрацию удержания клавиши, отсчет этого времени и запуск автоповтора скан кода соответствующей клавиши.

Большинство приложений современных операционных систем используют только время нажатия клавиши, время удержания не является информативным за исключением функции автоповтора символа на уровне операционной системы.

Время удержания клавиши так же является важным информационным параметром. Для исследования влияния временных характеристик отображения событий клавиатуры написана программа, реализующая функции драйвера. Для этого применены функции DirectInput из пакета Direct X 9.0. Разработанная программа на языке высоко уровня программирования C++ позволяла на прямую обращаться к устройству ввода клавиатуре и опрашивать её состояние с интервалов в единицы микросекунды. Работа программы использовала весь ресурс вычислительной машины, что приводило к «зависанию» компьютера. «Зависание» компьютера при выполнении этой программы наблюдалось на системах, обладающих самыми последними разработками бытовой вычислительной техники. Высокая точность регистрации отмечалась при полной передачи центральному процессору функции обработки событий клавиатуры. Уменьшение доли участия процессора в обработки состояния клавиатуры приводит к росту погрешности времени фиксации нажатия и времени удержания. Интервал дискретизации составляет порядка 15 625 мкс.

Анализ полученных результатов показал, что первый подход дает самую высокую погрешность регистрируемого времени удержания клавиши и составляет порядка 47 мс. Такое состояние объясняется использованием очереди событий Windows.

Использование Hook позволяет сократить погрешность до порядка 15 мс. Результат достигается за счет того, что при регистрации событий Windows программа минует очередь событий.

Все программные реализации обеспечили отличающиеся числовые значения между нажатиями клавиш и интервалов удержания для одной и той же контрольной группы пользователей. При этом отмечается схожесть частотной структуры распределения интервалов, удержания клавиш. Для каждого результата характерно увеличение числа интервалов удержания клавиши при наборе произвольного текста на кратных значениях.

Анализируемые программные решения использовали различные способы регистрации событий. Каждая программа имела некоторый интервал дискретизации по времени, или минимально регистрируемый интервал времени, эта величина в общем случае является погрешностью отображения. Программные особенности выполнения процедуры регистрации событий клавиатуры в полной мере не объясняют выявленные интервалы дискретизации. Задержки, сопутствующие этапам обработки событий и представления операционной системе имеют не постоянные значения порядка единиц мкс. Выявленная дискретизация отображения времени удержания клавиши от части определяется особенностью интерфейса соединения с компьютером. В п. 1.3 описывались возможные задержки во времени отображения, обусловленные интерфейсом передачи данных, но они носят системный характер и относятся ко всем передаваемым событиям. Одинаковая ошибка добавляется как к времени нажатия клавиши, так и ко времени ее отпускания.

Другой причиной дискретного отображения временных параметров является особенности регистрации текущего времени. Для определения времени нажатия клавиши необходи-

мо точно знать системное время или запускать дополнительный таймер, который также будет привязываться к системному времени.

Для синхронизации высокой точности на ОС Windows обычно использовался (Time Stamp Counter – счетчик отметок времени) TSC центрального процессора. Счетчик появился в x86 процессорах начиная с Pentium и является 64 разрядным. Он считывает тактовые импульсы центрального процессора. Значения счетчика TSC обычно запрашивается через инструкцию RDTSC пользователя. Эта операция легко и быстро выполняется и гарантирует высокую точность времени на современных компьютерах, порядка единиц микросекунд.

С развитием вычислительной техники и появления много ядерных систем, мобильных устройств частота процессора не остается постоянной в течение работы. Для экономии электроэнергии частота процессора мобильных вычислительных устройств уменьшается. При передачи задач между процессорами в многоядерной системе значение счетчика TSC изменяется и даже может показывать обратное время. TSC не всегда синхронизируется на двухядерных системах или SMP системах.

Для двухядерных систем Microsoft рекомендует использовать Query Performance Counter для синхронизации высокой точности порядка микросекунд. В двухядерной системе особенно при ее загрузке, TSC предоставляет программе использующей QPC значения частоты процессора не соответствующее текущему.

Вызов QueryPerformanceCounter и Time Get Tim приводит к изменению точности с микросекунд до миллисекунд, что более надежно. Большей надежностью и быстродействием, но низкой точностью обладает функция GetTickCount показавшая на Windows 9x минимальны интервал времени 55 мс.

Функция GetSystemTimeAdjustment возвращает значение приращения времени, возвращаемого GetTickCount. Как показали наблюдения (на компьютере с ОС Windows) эта функция возвращает 15625 мкс. Следовательно, GetTickCount возвращает время в миллисекундах, но с дискретностью в 15.625 мс. Системный таймер работает с этим периодом.

Проведенный анализ позволяет выделить ошибки в регистрации времени в самостоятельный класс погрешностей. Уменьшение величины ошибки приводит к нестабильности работы или «зависанию» вычислительной системы. Погрешность связана с архитектурными особенностями современных вычислительных систем. Для практических измерений выбирают компромисс между точностью и стабильностью работы вычислительной системы. Как показали измерения, дискретизация временных значений событий клавиатуры с интервалов ≈ 15 мс является оптимальным вариантом, при котором сохраняется работоспособность вычислительной системы.

ВЛАСЕНКО Александра Владимировна, доцент, кандидат технических наук, заведующий кафедрой компьютерных технологий и информационной безопасности Кубанский государственный технологический университет. 350000, г. Краснодар, ул. Московская, 2. E-mail: alex_vlasenko@list.ru

ШВЫРЕВ Борис Анатольевич, кандидат физико-математических наук, доцент кафедры компьютерных технологий и информационной безопасности. Кубанский государственный технологический университет. 350000, г. Краснодар, ул. Московская, 2. E-mail: bor2275@yandex.ru

БЕРДНИК Мария Викторовна, доцент кафедры компьютерных технологий и информационной безопасности. Кубанский государственный технологический университет. 350000, г. Краснодар, ул. Московская,2. E-mail: marviktr@mail.ru

VLASENKO Alexandra, Associate Professor, Candidate of Technical Sciences, Head of the Department of Computer Technologies and Information Security. Kuban State Technological University. 350000 Krasnodar, Bld. 2, Moskovskaya street. E-mail: alex_vlasenko@list.ru

SHVYREV Boris, Candidate of Physical and Mathematical Sciences, Associate Professor of the Department of Computer Technologies and Information Security. Kuban State Technological University. 350000 Krasnodar, Bld. 2, Moskovskaya street. E-mail: bor2275@yandex.ru

BERDNIK Maria, Associate Professor of the Department of Computer Technologies and Information Security. Kuban State Technological University. 350000 Krasnodar, Bld. 2, Moskovskaya street. E-mail: marviktr@mail.ru