Раздел II. Безопасность информационных систем и сетей

УДК 004.056.57

А.Т. Алиев

ПРОАКТИВНЫЕ СИСТЕМЫ ЗАЩИТЫ ОТ ВРЕДОНОСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Защита компьютерных систем от вредоносных программ в настоящее время является одной из наиболее актуальных задач в области защиты информации. Ежегодные потери от компьютерных вирусов оцениваются в десятки и сотни миллиардов долларов. В работе проводится анализ существующих решений по обнаружению вредоносных программ. Показана перспективность и актуальность разработки антивирусных средств, основанных на проактивных методах защиты. Данные методы отличаются возможностью борьбы с новыми еще незарегистрированными вирусами, для которых не выделены сигнатуры и не определен алгоритм работы. Выявлены недостатки известных существующих решений в этой области. Предложена новая трехуровневая схема системы проактивной защиты и метод анализа на основе экспертных оценок, которые позволяют значительно снизить вероятность ложных срабатываний и уровень требований к квалификации обслуживающего персонала. Использование трехуровневой схемы позволяет снизить нагрузку на модуль перехвата потенциально опасных действий и дает возможность для реализации более сложных алгоритмов анализа. Метод анализа на основе экспертных оценок, в свою очередь, делает возможной реализацию автоматизированной системы, способной блокировать действия вредоносного программного обеспечения без непосредственного участия пользователя. Все предложенные решения проверены экспериментально в реальной среде и на практике показали свою высокую эффективность.

Компьютерные вирусы; вредоносные программы; антивирусы; принятие решения.

A.T. Aliev

PROACTIVE MALWARE PROTECTION SYSTEMS

Malicious software, or malware, is the most widespread problem in the field of information security. Annual losses from computer viruses are estimated in the hundreds of billions of dollars. In this article, we analyze the existing solutions to detect malicious software and show the perspective and relevance of the development and implementation of anti-virus tools based on proactive detection methods. These methods allow protecting the computers from new unknown and yet unregistered viruses even if there signatures and algorithm are not defined. The shortcomings of existing solutions based on proactive detection methods are highlighted. We propose new three-tier system of proactive protection scheme and a new method of analysis based on expert judgment. The three-level architecture reduces the load on the intercept module and allows us to realize the more complex analysis algorithms. The analysis method based on expert judgment makes it possible to implement an automated system which is capable to block the dangerous actions of malicious software without a user involved. All new solutions proposed in the work were verified experimentally in a real environment and on the practices have shown to be highly effective.

Computer viruses; malware; antivirus; decision making.

Введение. Антивирусные программы являются одними из наиболее востребованных программ на современном рынке программного обеспечения и представлены на нем самым широким спектром решений. Среди поставщиков средств защиты от

вредоносных программ можно отметить как ряд крупных производителей, так и множество небольших компаний. На настоящее время насчитываются сотни различных антивирусов и специальных утилит, предназначенных для восстановления работоспособности компьютерных систем (КС) после заражения вредоносным программным обеспечением (ВПО). Обладание таким арсеналом средств защиты должно было уже давно положить конец всевозможным компьютерным вирусам. Но количество фактов заражения КС вирусами с каждым готом только растет.

Многие связывают рост количества фактов заражения ВПО с высокой активностью «хакеров», которые постоянно создают все новые и новые вредоносные программы. Но более детальный анализ позволяет сделать иные выводы. Да, каждый год появляются десятки тысяч новых вирусов, но большинство из них основаны на общих алгоритмах, используют одни и те же уязвимости КС. Более того, многие вирусы идентичны и по исходным кодам. Последнее связано с тем, что исходные коды большинства вредоносных программ продаются или свободно распространяются в хакерской среде. В результате они многократно используются различными разработчиками для реализации собственных вредоносных программ, которые зачастую лишь незначительно отличаются от исходного образца. Как следствие и наблюдается такое множество компьютерных вирусов.

Производители антивирусных средств, в свою очередь, рекламируя свои продукты, не забывают упомянуть, сколько миллионов сигнатур содержится в их антивирусной базе и как быстро они выпускают обновления при появлении новых вирусов. Проигравшими в этом случае остаются в основном рядовые пользователи персональных компьютеров. Год от года им приходится сталкиваться с сообщениями от антивирусной программы об обнаружении на их компьютере нового вируса, который уже успел уничтожить или повредить несколько нужных файлов. В данной статье мы рассмотрим один из возможных способов решения данной проблемы.

Анализ существующих решений. Самым популярным на настоящий момент способом борьбы с вредоносными программами был и остается метод сигнатурного анализа [1–3]. Суть метода заключается в обнаружении вирусов по сигнатуре – уникальной последовательности байт, которая позволяет однозначно идентифицировать ту или иную вредоносную программу. Имея в своем распоряжении базу сигнатур ВПО, антивирусная программа сканирует файлы на предмет совпадения байтовых цепочек с одной из сигнатур вредоносных программ. В случае выявления совпадения выдается сообщение о присутствии соответствующего вируса. Очевидным плюсом такого подхода является высокая точность выявления известных вирусов. Если антивирусная программа узнает вирус по его сигнатуре то, в большинстве случаев она сможет «вылечить» зараженный файл. Следует отметить, что данный метод используется уже более 25 лет без каких-либо существенных изменений. В основном изменения затрагивали лишь алгоритмы поиска сигнатур.

Несмотря на свою популярность, метод сигнатурного анализа не лишен целого ряда существенных недостатков. Одним из таких недостатков является так называемая уязвимость «нулевого дня». Если сигнатура вируса еще не попала в базу сигнатур (вирус только появился), то антивирусная программа совершенно спокойно пропустит любые действия такого вируса. Разумеется, с очередным обновлением базы, сигнатура вируса может быть в нее добавлена, и антивирус его обнаружит. Ряд файлов к этому времени может быть уже безвозвратно потерян. Методы сигнатурного анализа также не эффективны против вирусов, которые специально написаны для заражения компьютеров какой-либо отдельно взятой компании. В этом случае вирус не выходит за пределы локальной сети жертвы и не может быть проанализирован ни в одной из антивирусных лабораторий. Также методы сигнатурного анализа не позволяют бороться с целым рядом полиморфных и стелс вирусов.

Очевидно, что без регулярных обновлений антивирусная программа, основанная на сигнатурном методе анализа, со временем становится совершенно бесполезной. По этой причине большинство разработчиков антивирусов стараются включить в свои программы дополнительные методы и средства защиты от вирусов: эвристический метод анализа, песочницу, технологию эмуляции окружения, контроль целостности системных файлов, поведенческий анализ.

Особый интерес представляют методы поведенческого анализа или как их еще называют – методы проактивной защиты. Благодаря возможности перехвата потенциально опасных действий программного обеспечения в реальном режиме времени методы проактивной защиты обладают целым рядом существенных преимуществ:

- возможность обнаружения неизвестных и полиморфных вирусов;
- блокирование опасных и потенциально опасных действий;
- активное препятствие проникновению вирусов в систему;
- отсутствие необходимости в постоянных обновлениях;
- возможность автоматического «отката» опасных действий;
- защита системы от ошибок оператора.

Реализация механизмов проактивной защиты в современных антивирусных программах не позволяет полноценно использовать их потенциал. Основным недостатком, с которым постоянно приходится сталкиваться рядовым пользователям ЭВМ, является большое количество ложных срабатываний. Это приводит к тому, что большинство пользователей либо просто игнорируют сообщения о ВПО, либо вовсе отключают данный механизм. При этом следует отметить, что в рамках рассматриваемых реализаций данные методы весьма требовательны к квалификации пользователей. Довольно часто антивирусные программы выводят сообщения о подозрительных действиях тех или иных программ, сопровождая их лишь краткой пометкой без какого-либо пояснения. Далеко не каждый опытный пользователь КС может понять, что означает то или иное сообщение антивирусной программы о попытке записи данных в реестр по такому-то адресу. Оставляя решение за пользователем по каждому такому действию антивирусная программа, по сути, перекладывает на пользователя решение и всю полноту ответственности за возможное заражение КС вирусами.

Еще одним недостатком программ реализующих механизмы проактивной защиты зачастую оказывается их работа и перехват вызовов основных API функций на уровне пользователя. В результате, часть вредоносных программ, работающих на более низком уровне, может обойти механизмы защиты и успешно проникнуть в систему.

Отмеченные недостатки привели к тому, что в современных антивирусных программных продуктах использование проактивных методов защиты сведено к минимуму и зачастую они используются лишь в качестве дополнительного элемента защиты. В тоже время, программы, реализующие методы проактивной защиты от вредоносных программ [4], могут занять лидирующие положение на рынке антивирусных программных продуктов.

Перехват потенциально опасных действий. Для реализации механизма перехвата потенциально опасных действий предлагается использовать технологию перехвата таблицы системных сервисов SSDT с помощью дополнительного драйвера, работающего на уровне ядра [5]. Данная технология обладает рядом существенных преимуществ. Работая на уровне ядра, модуль перехвата позволяет осуществлять перехват всех обращений к файловой подсистеме и реестру даже на ранних стадиях загрузки операционной системы. При этом все перехваченные действия могут быть своевременно заблокированы. Работа модуля перехвата на наиболее низ-

ком уровне позволяет отслеживать действия стелс-вирусов, которые реализуют свою маскировку на более высоких уровнях. Также реализация модуля в виде драйвера ОС значительно усложняет задачу удаления и обхода модуля перехвата для ВПО.

Схема системы проактивной защиты. Реализация системы проактивной защиты на базе двухуровневой архитектуры, когда за перехват действий отвечает драйвер уровня ядра, а за вывод информации о происходящих событиях пользовательский процесс, в реальных условиях неприменима. В случае если на драйвер переложить функции перехвата действий и функции принятия решения об их блокировке, то это либо приведет к большому числу ложных срабатываний, либо значительно увеличит общее время прохождения вызова функции и как следствие замедлит работу всей системы. Кроме того непосредственное взаимодействие драйвера с пользовательским процессом может привести к нарушению стабильной работы операционной системы. Такой подход не позволяет построить более надежную и соответственно сложную систему принятия решений и сводит её к самому простому варианту.

Общая схема предлагаемой системы проактивной защиты представлена на рис. 1. Предлагаемая схема основана на трехуровневой архитектуре [6]. На самом низком уровне работает модуль перехвата потенциально опасных действий, реализованный в виде драйвера операционной системы. Он отвечает за перехват обращений к файловой подсистеме и реестру, а также отслеживает сетевое взаимодействие.

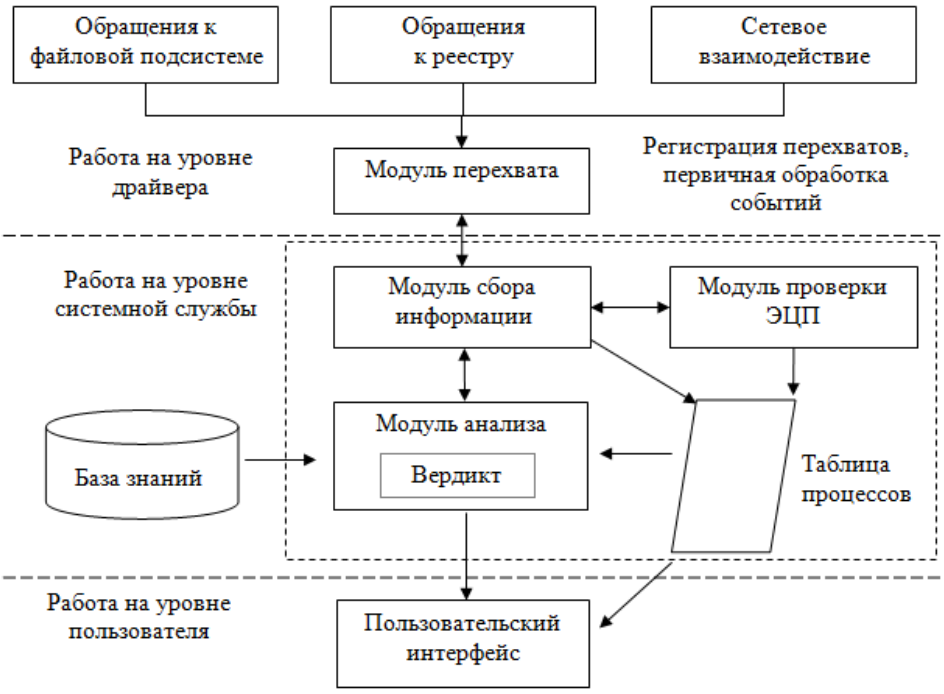

Рис. 1. Общая схема системы проактивной защиты от ВПО

Модуль перехвата потенциально опасных действий может также обеспечивать блокировку в реальном режиме времени однозначно опасных действий, предварительный сложный анализ которых не требуется. К таким действиям могут быть отнесены, например, попытка замены или изменения основных системных файлов.

Второй уровень системы представлен системной службой (процессом), работающим с правами операционной системы. На данном уровне осуществляется накопление и анализ всех потенциально опасных действий и принимается решение о блокировке работы того или иного процесса. Модуль анализа может быть легко модифицирован, может использовать любое необходимое количество системных ресурсов, обладает возможность обращения к локальной и сетевой базе знаний, а также возможностью проверки электронно-цифровой подписи всех запущенных в системе процессов. Работа модуля анализа отдельно от механизма перехвата потенциально опасных действий позволяет реализовать алгоритмы анализа, основанные на накоплении информации о работе программ в течение определенного времени.

За непосредственное взаимодействие с пользователем отвечает пользовательский интерфейс, который работает на самом высоком уровне системы. Основное его назначение заключается в уведомлении пользователя о наступлении опасных событий, которые были выявлены алгоритмами анализа и/или блокированы автоматически модулем перехвата. Заметим, что непосредственного участия в принятии решения о блокировании программ в рамках предлагаемой системы пользователь не принимает. Вынесение пользователя за рамки системы защиты позволяет значительно повысить скорость реакции системы и снизить влияние человеческого фактора.

Классификация действий по степени опасности. Любая система проактивной защиты от вредоносных программ основана на перехвате потенциально опасных действий и их последующем анализе. От того, насколько качественно будет проведен анализ потенциально опасных действий, зависит эффективность работы всей системы. Очевидно, что не все потенциально опасные действия равнозначны. Какие-то из них более опасны, какие-то менее. Все действия программного обеспечения можно разделить на классы по степени опасности: безопасные, низкого уровня опасности, среднего уровня опасности, высокого уровня опасности и особо опасные. Если система защиты будет одинаково реагировать на все небезопасные действия, то это приведет к большому числу ложных срабатываний. Для того чтобы снизить их количество в большинстве современных систем проактивной защиты действия, относящиеся к первым трем классам, просто игнорируются. При этом действия, которые относятся к последним двум классам, обрабатываются одинаково, что также повышает вероятность ошибки второго рода и увеличивает нагрузку на модуль перехвата потенциально опасных действий.

Отсутствие контроля над действиями, относящимися к безопасным, низкого и среднего уровня опасности, не позволят проследить развитие атаки ВПО с целью заражения КС во времени. Зачастую безопасные действия могут быть предвестниками более опасных действий вредоносных программ. Так, попав в систему, вирус в первую очередь начинает анализировать новую для него среду обитания. Эти действия не относятся к опасным и игнорируются. В тоже время, если одна из полезных рабочих программ выполнит действие, относящиеся к группе среднего или высокого уровня опасности, она будет блокирована антивирусом. Например, запись данных в исполнимый файл можно отнести к действию с самым высоким уровнем опасности. Большинство вирусов стараются записать себя в исполнимые файлы или создать их на диске. Но это может быть и всего лишь копирование файла из одного каталога в другой.

Таким образом, одного деления действий программного обеспечения по классам опасности недостаточно. Более эффективным будет использование метода анализа учитывающего всю историю действий программного обеспечения. При этом должны учитываться не только опасные и особо опасные действия, но и даже такие безопасные действия, как чтение каталога и проверка атрибутов файлов.

Большое количество информации о работе той или иной программы позволяет повысить надежность подсистемы принятия решений и реализовать возможность создания со временем поведенческих профилей для каждой из программ.

Накопление информации о работе программного обеспечения. На обработку в модуль анализа данных может поступать огромный поток разрозненной информации о работе всех запущенных в системе процессов. При этом одним из ключевых требований к модулю анализа является требование минимизации использования системных ресурсов. Данное требование может быть выполнено только за счет сокращения объемов хранимой оперативной информации и количества необходимых вычислительных операций. По этой причине модуль анализа не может хранить информацию обо всех действиях всех запущенных в КС программ. Вместо этого предлагается использовать несколько численных показателей, которые бы могли характеризовать все предыдущие действия той или иной программы. При использовании численных показателей объем хранимой информации по каждой программе сокращается всего до десятка байтов, а анализ каждого нового действия требует лишь пересчета показателей с учетом этого действия. В результате становится возможным сократить объем хранимой информации и количество вычислений.

Применение метода экспертных оценок. Как было отмечено выше, в современных системах проактивной защиты от вредоносных программ при перехвате потенциально опасного действия пользователю программы выдается сообщение с запросом на разрешение или блокирование данного действия. Принятие грамотного решения требует от пользователя достаточно высокой квалификации и уровня знаний, порой на уровне системного программиста или администратора. Очевидно, что подобный подход неприемлем, когда вопрос касается массового использования антивируса.

В системе проактивной защиты должен использоваться модуль принятия решения, который позволит переложить принятие решения на сам программный продукт. При этом, исходными данными для его работы должны быть не только события, поступающие от модуля перехвата действий потенциально опасного программного обеспечения, но и заранее подготовленная база знаний, основанная на мнении экспертов.

Для формирования базы знаний предлагается использовать метод экспертных оценок, который позволит упорядочить знания и мнения отдельных специалистов. Предполагается, что в формировании базы знаний будет принимать участие несколько высококвалифицированных специалистов, которые смогут дать объективную оценку тем или иным действиям, происходящим в системе. При этом, вместо классификации потенциально опасных действий по степени опасности, предлагается использовать классификацию по тому насколько те или иные действия характерны для вредоносных программ.

Для упрощения работы экспертов при выставлении оценок потенциально опасным действиям все действия предлагается разделить на группы в соответствии с этапами жизненного цикла вредоносной программы. Всего можно выделить шесть основных этапов: первый запуск, разведка, обеспечение повторного запуска, сокрытие присутствия, размножение и деструктивное воздействие.

Каждый из экспертов заполняет таблицу, самостоятельно проставляя баллы для каждого действия по десяти бальной шкале. Сначала проставляется независимая оценка для каждого из действий так, будто все предыдущие и последующие действия не важны. После заполнения первого столбца эксперт переходит к заполнению оставшихся столбцов. В них заносится оценка того, насколько однозначно рассматриваемое действие может классифицироваться как действие вредоносной программы при условии выполненных ранее действий из других групп.

Так как оценка проводилась в баллах, то итоговые результаты работы группы экспертов усредняются и отражаются на расширенной шкале (например, на 100 бальной). Полученные таблицы заносятся в базу знаний программы и являются исходными данными, на которые программа будет опираться при принятии решений.

Модуль принятия решения. Общая схема модуля принятия решения представлена на рис. 2. Как видно из рисунка в качестве хранимых данных используются только три таблицы ограниченного объема. В каждой из таблиц на один процесс приходится не более сотни байтов. Списки всех накопленных действий не используются. В результате удается свести к минимуму использование ресурсов оперативной памяти.

Все операции с данными осуществляются двумя сумматорами, что также значительно снижает объем требуемых вычислительных ресурсов. Первый сумматор используется для пересчета данных таблицы накопленных оценок. Второй для расчета текущей оценки вредоносности процесса, выполнившего пришедшее на вход модуля принятия решения действия. Накопленные оценки $s_{i,j}$ пересчитываются по каждому действию согласно формуле:

$$s_{i,j} = S_1, S_1(s_{i,j}, t, p, action) = g(s_{i,j}, t) + a_i \cdot q_p \cdot \sum_j v_{i,j} \cdot action_j \cdot g(s_{i,j}, t),$$

где $g(s_{i,j}, t)$ – функция изменения оценки от времени t; q – корректирующий коэффициент для процесса; p – индекс процесса в таблице корректирующих коэффициентов; $action$ – действие; $action_j$ – оценка действия по соответствующему столбцу таблицы экспертных оценок.

Таблица корректирующих коэффициентов отражает степень изначального доверия к тому или иному процессу. Например, меньший коэффициент может быть установлен для процессов, исполнимый модуль которых снабжен ЭЦП известного производителя ПО.

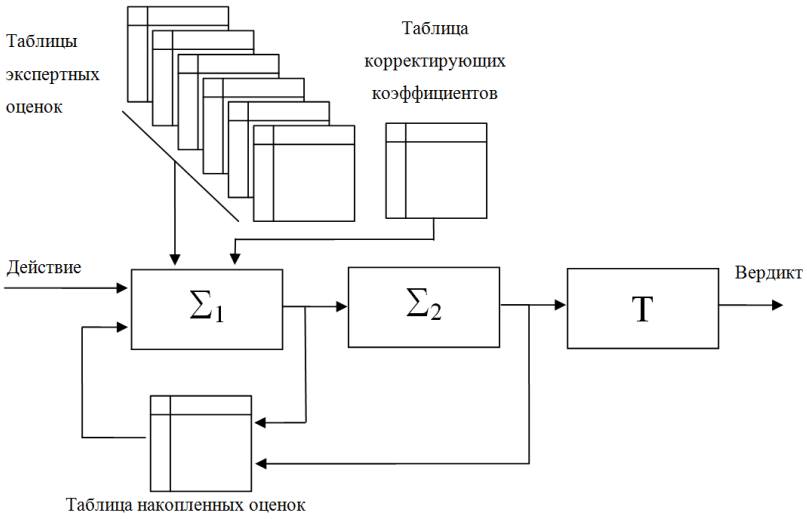

Рис. 2. Схема модуля принятия решения

Вторая сумма рассчитывается как сумма накопленных оценок умноженных на соответствующие коэффициенты:

$$s_i = S_2, S_2(s_i, t) = g(s_i, t) + \sum_j v_{i,j} \cdot s_{i,j},$$

где s_i – предыдущее значение.

Результат вычисления S_2 используется для вынесения вердикта об отнесении программы соответствующей процессу *p* к вредоносным. Решение принимается по превышению некоторого заранее заданного порога принятия решения T, простым сравнением с заданной величиной. Величина T, при которой программа может быть признана вредоносной, подбирается экспериментально и сильно зависит от исходной таблицы экспертных оценок.

Выводы. Предложенный метод проактивной защиты опробован экспериментально. Модуль принятия решения, использованный в ходе экспериментов, показал достаточно хорошие результаты. Как и ожидалось, удалось значительно снизить количество ложных срабатываний. В тоже время, зависимость величины порога принятия решения от таблиц экспертных оценок можно отнести к недостаткам, для устранения которых требуется проведение дальнейших исследований. Вместе с тем, удалось показать, что система проактивной защиты, построенная на описанных в статье принципах, в подавляющем большинстве случаев способна самостоятельно без обращения к пользователю, принимать верные решения. Результаты работы уже на настоящем этапе исследований могут быть использованы для реализации методов проактивной защиты компьютерных систем от вредоносных программ.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. РД ФСТЭК «Базовая модель угроз безопасности персональных данных их обработке в информационных системах персональных данных», 15 февраля 2008 г.
2. *Касперский К.* Записки исследователя компьютерных вирусов. – СПб.: Питер, 2002. – 316 с.
3. *Aycock J.* Computer Viruses and Malware. Advances in information security. – Calgary: Springer, 2006. – 227 p.
4. *Алиев А.Т., Морозов А.П.* Защита информационных систем от вредоносного программного обеспечения // XIX науч. конф. “Современные информационные технологии: тенденции и перспективы развития”. – Ростов-на-Дону, 2012. – С. 26-27.
5. *Шрайбер С.* Недокументированные возможности Windows 2000. Библиотека программиста. – СПб.: Питер, 2002. – 544 с.
6. *Алиев А.Т.* Построение проактивной системы защиты от вредоносных программ // XII науч.-практ. конф. «Информационная безопасность – 2012». – Таганрог: ТТИ ЮФУ, 2012. Ч. 2. – С. 45-50.

Статью рекомендовал к опубликованию д.т.н., профессор О.Б. Макаревич.

Алиев Александр Тофикович – ООО НПО «Редут»; e-mail: aliev@nporedut.ru; 344002, г. Ростов-на-Дону, ул. Темерницкая, 44; генеральный директор; к.т.н.

Aliev Alexander Tofikovich – RPA “Redut”, Llc; e-mail: aliev@nporedut.ru; 44, Temernitskaya street, Rostov-on-Don, 344002, Russia; general director; cand. of eng. sc.

УДК 681.324

Ю.А. Брюхомицкий

ИММУНОЛОГИЧЕСКИЙ ПОДХОД К ОРГАНИЗАЦИИ КЛАВИАТУРНОГО МОНИТОРИНГА*

Клавиатурный мониторинг (КМ) позволяет вести непрерывную скрытную аутентификацию пользователей компьютерных систем и решать ряд других задач компьютерной безопасности. Известные системы КМ обладают недостаточной точностью и скоростью верификации работающего пользователя. С целью повышения точности и скорости

* Работа выполнена при поддержке гранта РФФИ 12-07-00081-а.