УДК 004.056

Разработка DLP-модуля для защиты персональных данных в распределенной автоматизированной системе

Ковтуненко Аркадий Алексеевич — специалист по защите информации, выпускник кафедры «Информатика и информационная безопасность». Область научных интересов: разработка программного обеспечения, информационная безопасность. E-mail: bernadod2000@mail.ru

Петербургский государственный университет путей сообщения Императора Александра I, Россия, 190031, Санкт-Петербург, Московский пр., 9

Для цитирования: Ковтуненко А. А. Разработка DLP-модуля для защиты персональных данных в распределенной автоматизированной системе // Интеллектуальные технологии на транспорте. 2024. № 3 (39). С. 65–72. DOI: 10.20295/2413-2527-2024-339-65-72

Аннотация. *Рассматривается важность защиты персональных данных в автоматизированных системах. Представлено исследование о разработке программного модуля для защиты конфиденциальной информации, в том числе персональных данных, в распределенной автоматизированной системе.* ***Цель исследования:*** *создание программного средства, блокирующего утечки конфиденциальной информации. Разработанный программный модуль предназначен для блокировки утечки конфиденциальной информации по сетевым каналам и съемным носителям с помощью модели машинного обучения с централизованным управлением администратором безопасности.* ***Практическая значимость:*** *использование данного модуля в организации включает обеспечение централизованного управления безопасностью информации, снижение риска утечек конфиденциальных данных, а также поддержку в расследованиях инцидентов безопасности. Исследование может быть применимо как временная мера в период обновления и адаптации системы под новые реалии до внедрения полноценной системы защиты.*

Ключевые слова: *персональные данные, система управления информационной безопасностью, утечка конфиденциальной информации, машинное обучение, DLP (Data Leaks Protection)*

Введение

В последние годы задача обеспечения безопасности конфиденциальной информации и в частности персональных данных (ПДн) стала как никогда актуальна. По данным экспертно-аналитического центра группы компаний InfoWatch [1], количество скомпрометированных записей ПДн в России за вторую половину 2022 года в 1,5 раза превысило их объем за первую половину того же года, а только в первой половине 2023 года было скомпрометировано почти столько же единиц ПДн, сколько за весь 2022 год — около 705 млн записей. И пусть количество утечек за последние годы несколько снизилось, их объемы неуклонно растут в поражающих масштабах. Именно поэтому защита персональных данных в наше время — актуальная и крайне важная задача.

Огромные объемы персональных данных обрабатываются и хранятся в распределенных автоматизированных системах (АС). Распределенные АС используются в различных отраслях, включая здравоохранение, государственное управление и чрезвычайные службы. Именно это, а также их структура, в которой данные распределены между различными подразделениями и регионами, повышает сложность

защиты информации в таких системах и делает их уязвимыми для различных угроз информационной безопасности (ИБ). При этом утрата или компрометация ПДн может не только подорвать доверие общества к организации, но и привести к серьезным финансовым последствиям.

В период глобальных изменений в АС, таких как, например, переход на отечественную операционную систему Astra Linux, система уязвима и многие ее модули защиты могут находиться в нерабочем состоянии. В данный период вероятность утечек ПДн крайне высока, и, чтобы этого избежать, стоит использовать временные решения в период внедрения полноценной системы защиты, если ускорить данный процесс не получается. Такие решения характеризуются быстротой развертывания и малым потреблением ресурсов, но в то же время охватывают лишь основные направления защиты.

В данной работе будет предложен программный модуль на основе DLP-технологий, оптимизированный под систему Astra Linux, предотвращающий утечки ПДн, повышающий эффективность работы администратора безопасности и обеспечивающий централизованное управление безопасностью.

Выбор метода распознавания конфиденциальной информации в тексте

Программный алгоритм, отвечающий за обнаружение и категоризацию защищаемой информации, является основой функционирования любой DLP-системы. Данные алгоритмы чаще всего базируются на одной из двух групп методов анализа: лингвистических или статистических.

Лингвистические методы анализа текста основаны на знаниях о языке и его структуре. Эти методы учитывают языковые особенности и специфику текста, благодаря чему они могут распознавать термины и другие элементы, характерные для конкретного языка. Лингвистический анализ хоть и позволяет более точно выявлять конфиденциальную информацию благодаря распознаванию контекста и смысловой нагрузки, но в то же время требует больших объемов языковых данных для обучения моделей и правил. Другой недостаток связан со сложностью четкой категоризации при использовании вероятностного подхода в подобных методах, из-за чего снижается точность срабатывания системы.

Статистические методы анализа, в свою очередь, демонстрируют точность, близкую к абсолютной. По сути, статистический анализ использует вероятностные методы для идентификации конфиденциальной информации на основе ее статистических характеристик. То есть статистические технологии относятся к текстам не как к связной последовательности слов, а как к произвольной последовательности символов, поэтому одинаково хорошо работают с текстами на любых языках. Данные методы зачастую основаны на анализе частоты встречаемости определенных слов или выражений в тексте, определении закономерностей и выявлении отклонений от них, которые могут указывать на конфиденциальность информации. Хотя статистический анализ не всегда учитывает контекст и семантику текста, он все же является более эффективным для обработки больших объемов данных — требует меньше ресурсов для обучения и работы системы, а также быстрее обрабатывает текст в реальном времени.

По причине вышеперечисленных преимуществ было решено опираться на статистический анализ и использовать метод машинного обучения. Машинное обучение в контексте DLP используется для создания моделей, обнаруживающих конфиденциальную информацию на основе статистических закономерностей. Эти модели обучаются на больших объемах текстовых данных, по которым автоматически выявляют сигнатуры конфиденциальной информации. Также важным достоинством моделей машинного обучения является их доступность для улучшения в будущем благодаря модификации и добавлению новых данных, используемых для обучения.

Создание модели машинного обучения для обнаружения конфиденциальной информации в тексте

Первым этапом реализации машинного обучения является сбор данных, которые будут

использованы для обучения модели. Поскольку модель будет служить для распознавания конфиденциальной информации в тексте, то требуются данные, содержащие как конфиденциальную, так и неконфиденциальную информацию.

Объем текста отдельного примера влияет на качество обучения модели, поэтому стоит подбирать оптимальный объем текста для примеров — от нескольких десятков до двух сотен слов.

Но гораздо сильнее на качество обучения модели влияет количество самих примеров информации. Для модели будет использовано около тысячи обучающих примеров — такое количество подойдет для начального прототипа и хорошо сочетается с простыми алгоритмами, используемыми далее.

Каждый пример был помечен на предмет содержания им конфиденциальной информации и сохранен в табличном файле.

После сбора обучающие данные нужно обработать. Этот процесс включает несколько этапов:

- приведение текста к нижнему регистру;
- удаление ненужных символов, таких как знаки препинания;
- токенизация — разделение текста на слова;
- удаление стоп-слов — часто встречающихся, но малоинформативных слов;
- лемматизация — приведение слов к их леммам, базовым словарным формам.

Однако даже обработанные тексты все еще являются слишком сложными и многомерными для алгоритмов машинного обучения и не могут быть обработаны ими напрямую. Именно для этого перед обучением самой модели нужно извлечь из текстов признаки, то есть преобразовать их в числовые векторы, доступные для использования алгоритмами машинного обучения.

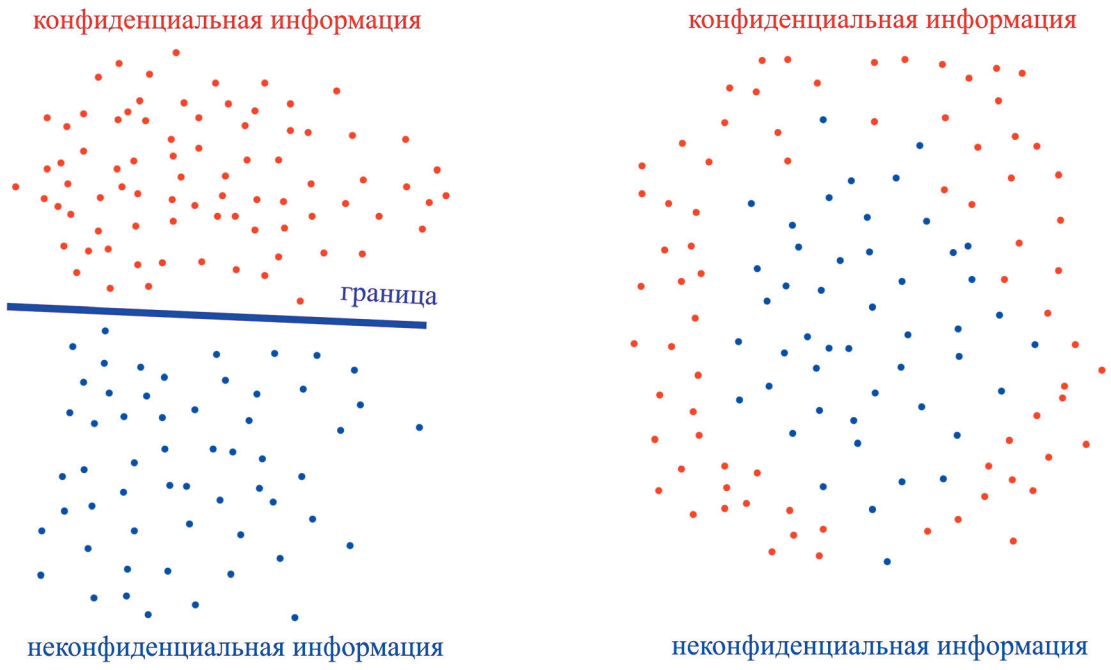
Для этой задачи был выбран метод TF-IDF. Данный метод проводит оценку важности отдельного слова в тексте относительно всего набора текстов, выделяет ключевые слова и определяет, какие из них наиболее точно характеризуют определенные тексты из набора. Если слово часто встречается в конкретном тексте, но не во многих, значит, оно наиболее точно описывает его содержание.

Последним шагом является выбор непосредственно алгоритма машинного обучения. Для разработки модуля был выбран метод опорных векторов (SVM).

Данный алгоритм является достаточно универсальным и используется для задач классификации. Метод опорных векторов относится к задаче обучения, то есть он основан на том, что имеется некоторая выборка значений, отношения которых к определенным классам заранее известны. Такой выборкой и являются обучающие примеры [2].

Работа SVM продемонстрирована рис. 1, *а*. Пусть обучающие примеры представляют собой набор точек на плоскости и разбиты на заранее известные два класса: конфиденциальная и неконфиденциальная информация. Задача метода заключается в построении границы между этими двумя множествами. Причем все новые значения будут классифицироваться в зависимости от положения относительно этой границы: выше прямой — информация конфиденциальна, ниже — неконфиденциальна. Однако этот пример достаточно тривиален. В реальных задачах речь идет о пространствах высоких размерностей и граница в виде прямой не сможет в них существовать, так же как и значения в виде точек, которые в реальных вычислениях являются числовыми векторами. Поэтому в методе SVM границей является не просто прямая, а гиперплоскость — подпространство с размерностью, на единицу меньшей, чем исходное пространство.

Однако прямых, разделяющих два множества точек, может быть много. Возникает вопрос: какая же именно должна являться границей? В методе опорных векторов выбирается граница, расстояние от которой до каждого класса максимально. Такая граница называется оптимальной разделяющей гиперплоскостью. Чем больше расстояние между гиперплоскостью и ближайшими векторами обоих классов, тем лучше будет обобщающая способность у обучаемой модели. А сами ближайшие к границе векторы каждого класса, расстояние от которых до границы является величиной зазора между классами и разделяющей гиперплоскостью, называются опорными векторами.



а) *б)*

Рис. 1. Работа метода опорных векторов
(*а* — линейная разделимость данных, *б* — линейная неразделимость данных)

Таким образом, основная задача метода SVM заключается в построении оптимальной разделяющей гиперплоскости. Однако в реальных задачах, особенно таких сложных, как распознавание конфиденциальной информации, данные очень редко бывают линейно разделимыми (рис. 1, *б*). Поэтому линейное ядро метода SVM хоть и можно использовать для простого и наглядного тестирования ввиду его большой скорости, но для реальных задач оно покажет себя не лучшим образом. Для таких задач использование нелинейных ядер поможет значительно улучшить результаты обучения модели. Среди таких ядер стоит выделить радиальную базисную функцию RBF. Ядро RBF обеспечивает лучшую производительность за счет своей гибкости и способности работать с нелинейными границами, хотя обучение модели с таким ядром занимает несколько больше времени.

Распознавание и блокирование утечек конфиденциальной информации

Требуется определить, по каким каналам могут возникать утечки конфиденциальной информации. В первую очередь стоит обратить внимание на сетевые каналы утечки. Около половины всех утечек конфиденциальной информации за последние годы происходит именно с использованием сетевых технологий, таких как интернет, локальные сети, электронная почта, облачные сервисы и прочее. На втором месте среди каналов утечек, которые можно контролировать на программном уровне, находятся утечки конфиденциальной информации, связанные с носителями данных [3]. К ним относятся кража, потеря или намеренная передача устройств и съемных носителей в руки злоумышленников.

Для того чтобы блокировать утечки конфиденциальной информации по Сети, потребуется анализировать сетевой трафик на рабочем месте. Будет осуществляться проверка на то, содержит ли анализируемый пакет слои TCP и IP, то есть является ли IP-пакетом с TCP-содержимым. Если пакет подходит под требования, то из него будет извлекаться полезная нагрузка в переменную строки. Анализ полезной нагрузки на конфиденциальность будет производиться с использованием TF-IDF векторизатора и SVM-модели. После определения конфиденциальности информации в пакете

нужно блокировать его передачу. Для этого будет использоваться подсистема ядра Linux NFQUEUE, позволяющая перехватывать сетевые пакеты и передавать их в пользовательское пространство для будущей обработки.

Для блокирования утечек конфиденциальной информации с помощью съемных носителей потребуется отслеживать все записываемые на носители файлы, анализировать их содержимое и в случае обнаружения конфиденциальных данных блокировать запись, производя откат изменений или удаление файла. Данное решение программа будет принимать в зависимости от ответа администратора на запрос о содержании конфиденциальной информации на носителе.

Однако данная программа все еще представляет собой единый скрипт, выполняемый локально на одном устройстве. Одной из главных целей разработки данного модуля является повышение эффективности работы администратора безопасности и централизованное управление безопасностью. Для достижения этой цели модуль необходимо разделить на клиентскую и серверную части.

Серверная часть будет располагаться на АРМ администратора безопасности. На ней будут находиться файлы модели машинного обучения и ее векторизатора, а также скрипт для повторного обучения модели в случае обновления ее базы данных. Данная часть программы будет отвечать за обработку сообщений и запросов на удаление конфиденциальной информации от клиентских частей.

Клиентская часть располагается на АРМ сотрудника и реализует основные функции по мониторингу сетевого трафика и информации на съемных носителях. Скрипт начинает свою работу одновременно с запуском системы — запрашивает у серверной части файлы SVM-модели и векторизатора, после чего запускает мониторинг сетевого трафика и цикл анализа файлов в каталогах съемных носителей. В процессе работы клиентская часть модуля обращается к серверной с запросами на удаление или откат изменений файлов на съемных носителях, передает сообщения о перехваченных и заблокированных пакетах, а также о завершении работы модуля.

Дополнительные механизмы безопасности модуля

Чтобы модуль в случае отказа сервера или соединения мог продолжать выполнять свои основные задачи, функция отправки серверу сообщения об отброшенном пакете в случае разрыва соединения будет сохранять сообщение в специальный локальный файл. После восстановления соединения все содержимое файла будет отправлено на сервер, после чего файл будет очищен. Функция отправки запроса на удаление или откат изменения файла на съемном носителе в случае разрыва соединения будет автоматически принимать решения, будто запрос одобрен администратором. Это усложняет работу сотрудника на АРМ, но зато наверняка предотвращает утечки, блокируя любую передачу подозрительной информации на съемный носитель, пока сетевое соединение не будет восстановлено. Таким образом, модуль сможет выполнять свои основные функции даже без соединения с сервером.

На данном этапе проектирования сообщения и запросы, передаваемые на серверную часть к администратору от клиентских модулей программы, представляют собой просто текстовые сообщения, выводимые на экран АРМ администратора. Соответственно, после перезапуска программы, например на следующий рабочий день, все предыдущие сообщения и запросы будут отсутствовать. Для DLP-технологий возможность ведения аудита крайне важна для расследований и анализа произошедших или предотвращенных утечек данных и прочих инцидентов безопасности. Поэтому дополним разрабатываемый модуль функцией журналирования событий, которая будет сохранять все сообщения и запросы от клиентов в базе данных (БД) на сервере.

Разработка будет направлена на библиотеки и инструменты для системы управления базами данных (СУБД) PostgreSQL, так как на ней базируются основные отечественные СУБД [4]. В БД будут использоваться две основные таблицы: для сообщений сетевого мониторинга и для запросов администратору. Каждая таблица имеет столбец с первичным ключом в виде автоматически увеличивающегося идентификатора, столбец

с идентификатором клиентского устройства, столбец с временем произошедшего события и столбец с текстом сообщения или запроса. Помимо этого, таблица запросов имеет столбец, содержащий ответ администратора на запрос, — разрешение или запрет на действие клиентского модуля.

Сообщения от клиентских частей включают данные о пакетах или файлах, вероятно содержащих конфиденциальные данные. На данном этапе сообщения передаются практически в открытом виде — в виде байтовой последовательности, при перехвате которой можно восстановить из нее изначальный объект. Для обеспечения безопасности необходимо шифровать сообщения перед передачей.

Шифрование будет производиться на основе криптографии на эллиптических кривых ECC. Безопасность шифрования ECC обусловлена сложностью решения задачи дискретного логарифмирования на основе эллиптических кривых [5]. Для обмена ключами будем использовать протокол Диффи — Хеллмана, основанный на криптографии эллиптических кривых. Обе стороны генерируют пары ключей, где закрытый ключ — это случайное целое число, а открытый ключ — произведение закрытого ключа на базовую точку на эллиптической кривой, являющуюся общей для обеих сторон. По открытому каналу стороны обмениваются открытыми ключами. Каждая сторона, используя полученный открытый ключ, вычисляет из него общий секрет — произведение открытого ключа другой стороны и своего закрытого ключа. Общий секрет получается одинаковым у обоих сторон, несмотря на разные множители. На основе общего секрета уже можно сгенерировать криптографический ключ нужной длины и использовать его для шифрования сообщений. Сделать это можно с помощью функции HKDF, которая работает в два этапа: сначала создает из общего секрета псевдослучайный ключ, а после на его основе создает конечный ключ нужной длины.

Для самого шифрования будем использовать симметричный алгоритм блочного шифрования AES. Данный алгоритм блочного шифрования разбивает исходный текст на блоки по 16 байт и, представляя их в виде квадратной матрицы текста, шифрует каждый по очереди. Шифрование включает замену исходных байтов на другие по специальной таблице S-box, циклический сдвиг строк матрицы текста влево, умножение каждого столбца матрицы текста на соответствующий столбец матрицы байтов одного блока, поэлементное добавление к матрице текста раундовых ключей [6]. А дешифрование представляет собой последовательность инвертированных операций шифрования, выполняемых в обратном порядке. AES будет реализован в режиме обратной связи по зашифрованному тексту. В этом варианте перед шифрованием каждого следующего блока текста он складывается по модулю два с зашифрованным результатом шифрования предыдущего блока. Предыдущим блоком для первого блока данных считается вектор инициализации — случайно сгенерированный блок данных, и благодаря этому шифрование одного и того же текста по тому же ключу каждый раз будет давать уникальный результат, поскольку вектор инициализации генерируется снова для каждой операции шифрования.

Разработка программного модуля

Программа была написана на языке Python, поскольку операционная система Astra Linux, на работу с которой она направлена, полностью поддерживает его. Использовались как стандартные механизмы языка, так и сторонние библиотеки, такие как re, NLTK, pymorphy2, sklearn, Scapy, netfilterqueue, Watchdog, socket, pickle, cryptography.

Разработанное клиент-серверное приложение прошло тестирование на испытательном стенде, представляющем собой две виртуальные машины на ОС Astra Linux Common Edition 2.12 с установленным между ними локальным соединением. На виртуальной машине администратора также дополнительно была установлена СУБД Postgres Pro Standard 11 и создана БД с двумя таблицами. После определения переменных окружения с адресом сервера, клиента и БД были развернуты модули. Клиентский модуль был установлен на автозапуск с системой и правами суперпользователя с помощью файла systemd.

По результатам тестирования можно сделать вывод, что разработанный программный модуль полностью выполняет требуемые функции по блокированию утечек конфиденциальной информации и централизованному контролю работы со стороны администратора безопасности.

Заключение

В результате разработки был создан программный модуль для защиты конфиденциальной информации от утечек с АРМ сотрудников. Данный модуль представляет собой клиент-серверное приложение с ведением аудита и криптографической защитой. Данный модуль может быть использован в организациях с распределенными системами для повышения уровня централизации управления безопасностью и блокирования утечек в период перевода организаций на иные серийные средства защиты. Модуль является базовой версией, которая может дополняться со временем и по требованиям конкретной организации к фильтруемой информации и структуре программы.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Утечки информации ограниченного доступа в мире и России, первое полугодие 2023 года. Аналитический отчет // Экспертно-аналитический центр InfoWatch, 2023. 17 с. URL: http://www.infowatch.ru/sites/default/files/analytics/files/utechki-informatsii-ogranichennogo-dostupa-v-mire-i-rossii-za-pervoe-polugodie-2023-goda.pdf (дата обращения: 01.04.2024)

2. Полетаева Н. Г. Классификация систем машинного обучения // Вестник Балтийского федерального университета им. И. Канта. Серия: Физико-математические и технические науки. 2020. № 1. С. 5–22.

3. Утечки информации // Anti-Malware.ru. URL: http://www.anti-malware.ru/threats/leaks (дата обращения: 01.04.2024).

4. Обзор отечественных СУБД // 1C-MSSoft.Ru. 2023. 4 декабря. URL: http://www.mssoft.ru/news/40506 (дата обращения: 09.04.2024).

5. Обухов В. А. Криптография на основе эллиптических кривых (ECC) // Потомки Аль-Фаргани. 2023. Т. 1, вып. 4. С. 182–188. DOI: 10.5281/zenodo.10337673

6. Батыркаев А. Г., Глотина И. М. Что такое шифрование AES и как оно работает // Агротехнологии XXI века: стратегия развития, технологии и инновации: материалы Всероссийской научно-практической конференции (Пермь, 16–18 ноября 2021 года). Пермь: Прокростъ, 2021. С. 254–258.

Дата поступления: 24.08.2024
Решение о публикации: 20.09.2024

Development of a DLP Module to Protect Personal Data in a Distributed Automated System

Arkady A. Kovtunenko — information security specialist, graduate of the Department “Informatics and Information Security”. Research interests: software development, information security. E-mail: bernadod2000@mail.ru

Emperor Alexander I Petersburg State Transport University, 9, Moskovsky pr., Saint Petersburg, 190031, Russia

For citation: Kovtunenko A. A. Development of a DLP module to protect personal data in a distributed automated system // Intellectual Technologies on Transport. 2024. No. 3 (39). P. 65–72. DOI:10.20295/2413-2527-2024-339-65-72 (In Russian)

Abstract. *The importance of personal data protection in automated systems is being considered. A study is provided on the development of a software module for protecting confidential information, including personal data, in a distributed automated system.* ***The primary goal of the study****: to create is to create software that prevents leaks of confidential information. The developed software module is designed to block the leakage of confidential information through network channels and removable media using a machine learning model with centralized security administrator management.* ***The practical significance:*** *the use of this module in organizations includes ensuring centralized information security management, reducing the risk of confidential data leaks, and supporting the investigation of security incidents. The research can be used as a temporary measure during the period of updating and adapting the system to new realities until a full-fledged protection system is achieved.*

Keywords: *personal data, information security management system, confidential information leaks, machine learning, DLP (Data Leaks Protection)*

REFERENCES

1. Utechki informacii ogranichennogo dostupa v mire i Rossii, pervoe polugodie 2023 goda. Analiticheskij otchet // Ekspertno-analiticheskij centr InfoWatch, 2023. 17 c. URL: http://www.infowatch.ru/sites/default/files/analytics/files/utechki-informatsii-ogranichennogo-dostupa-v-mire-i-rossii-za-pervoe-polugodie-2023-goda.pdf (data obrashcheniya: 01.04.2024). (In Russian)

2. Poletaeva N. G. Klassifikaciya sistem mashinnogo obucheniya // Vestnik Baltijskogo federal'nogo universiteta im. I. Kanta. Seriya: Fiziko-matematicheskie i tekhnicheskie nauki. 2020. № 1. S. 5–22. (In Russian)

3. Utechki informacii // Anti-Malware.ru. URL: http://www.anti-malware.ru/threats/leaks (data obrashcheniya: 01.04.2024). (In Russian)

4. Obzor otechestvennyh SUBD // 1C-MSSoft.Ru. 2023. 4 dekabrya. URL: http://www.mssoft.ru/news/40506 (data obrashcheniya: 09.04.2024). (In Russian)

5. Obuhov V. A. Kriptografiya na osnove ellipticheskih krivyh (ECC) // Potomki Al'-Fargani. 2023. T. 1, vyp. 4. S. 182–188. DOI: 10.5281/zenodo.10337673 (In Russian)

6. Batyrkaev A. G., Glotina I. M. CHto takoe shifrovanie AES i kak ono rabotaet // Agrotekhnologii XXI veka: strategiya razvitiya, tekhnologii i innovacii: materialy Vserossijskoj nauchno-prakticheskoj konferencii (Perm', 16–18 noyabrya 2021 goda). Perm': Prokrost", 2021. S. 254–258. (In Russian)

Received: 24.08.2024
Accepted: 20.09.2024