Архитектура вычислительного комплекса с многоуровневым контролем доступа к веб-сервисам по общедоступным сетям

Ш. Г. Магомедов

***Аннотация*—Информационные технологии приобретают глобальный трансграничный характер и стали неотъемлемой частью всех сфер деятельности личности, общества и государства. Их эффективное применение является фактором ускорения экономического развития государства и формирования информационного общества. При этом, бурный научно-технический прогресс и постоянное совершенствование программного и аппаратного обеспечения, требует с одной стороны такого же постоянного совершенствования и повышения эффективности функционирования всех компонент, с точки зрения снижения расходов, повышения качества, надежности, безопасности, но при этом обеспечивая высокий уровень защищенности доступа к вычислительным ресурсам, не нарушая требований существующих законов к защите персональных и конфиденциальных данных. С учетом все возрастающих требований к информационной инфраструктуре разработка архитектуры вычислительного комплекса с различными механизмами контроля доступа, способными расширять возможности инструментов информационной безопасности весьма актуальная задача. Несмотря на то, что современные системы обладают широким спектром механизмов и средств защиты информации их явно недостаточно с учетом постоянных взломов и утечек информации, поэтому необходимо определить их роль и место в каждом уровне вычислительной архитектуры. В работе предлагается архитектура системы обработки данных вычислительного комплекса, с описанными компонентами служащими для обеспечения контроля доступа. Также представлены уровни доступа к данным при обращении компонентов к вычислительному комплексу, с подробным описанием функционирования и наполнения модулей. Приведены результаты экспериментальных исследований для оценки ресурсных затрат при ведении учета действий пользователя.**

***Ключевые слова*—архитектура вычислительного комплекса, контроль доступа, веб-сервисы, SIEM-технологии.**

I. Введение

Для современных условий цифровизации характерно перенесение государственных [1], медицинских [2], образовательных [3], банковских и проч. услуг и экономических отношений [4] в форму взаимодействия с вычислительными сервисами по компьютерным сетям. Основным инструментом цифрового взаимодействия поставщиков и потребителей услуг становится веб-сервисы или цифровые порталы (далее ЦП) через общедоступные компьютерные сети.

Разработчики ЦП стоят перед разрешением противоречия: с одной стороны — вычислительный сервис должен быть максимально доступен, платформонезависим, удобен и прост в использования, если услуга, например, предоставляется в «один клик», доступ не должен занимать огромное время, с другой — предоставляемые услуги связаны с большим количество персональных, медицинских, банковских и прочих данных конфиденциальных данных [5], защита которых требует значительных усилий. Для обеспечения защищенного контроля доступа используются технологии информационной безопасности [6], которые, как правило, являются внешними по отношению к используемой программно-аппаратной инфраструктуре информационного обеспечения ЦП. Влючение систем контроля доступа (КД) на поздних стадиях проектирвоания и эксплуатации ЦП часто негативно влияет на характеристики системы: сервисы замедляются при работе с большими данными [7], ограничивается количество пользователей, оставляет для разработчиков и администраторов ЦП возможности программно несанкционированного доступа. В этих условиях разработка методов, информационных моделей и архитектур ВК со встроенными механизмами многоуровневого КД является актуальной задачей, способной повысить эффективность функционирования ЦП, дополняя существующие разноуровневые инструменты информационной безопасности.

В общедоступных сервисах ЦП, имеется ряд особенностей, которые не могут быть учтены при КД к данным на основе использования внешних инструментов [8] информационной безопасности. Это передача паролей сторонним лицам, перехват паролей программными средствами, или условий, в которых пользователь, прошедший все мыслимые и немыслимые способы верификации, сознательно или случайно оставил систему активной (не закрыл веб-страницу или аккаунт) и другой пользователь стал взаимодействовать с ЦП, тем самым имея доступ к конфиденциальной информации. Для разработки КД требуется использовать встроенные в ЦП инструменты подтверждения личности пользователя в процессе взаимодействия [9].

Для веб-сервисов одним из способов является использование role-based access control RBAC [10], где каждая точка входа связана с набором ролей пользователя. Различные исследовательские группы разработали

Статья получена 15 февраля 2021.

Ш. Г. Магомедов, к. т. н., доцент, МИРЭА — Российский технологический университет (magomedov_sh@mirea.ru)

контекстно-зависимые подходы и структуры управления доступом, которые различаются своими контекстными моделями и моделями политик. Были предложены модели управления доступом на основе ролей [11], включающие в политики динамически изменяющиеся контекстные условия [12–14] (например, информацию, ориентированную на местоположение пользователя и ресурсы).

Современным направлением в области архитектур ВК является создание SIEM-систем [15], которые возникли в результате слияния систем SEM и SIM. SEM (Security Event Management) — системы действуют в времени, приближенном к реальному, включают мониторинг событий и генерацию предупреждающих сообщений. SIM (Security Information Management) — системы, анализируют накопленную статистическую информацию и фиксируют различные отклонения. Перед системой SIEM ставятся следующие задачи: консолидация и хранение журналов событий от различных источников; предоставление инструментов для анализа событий и разбора инцидентов; корреляция и обработка событий по правилам; инцидент-менеджмент. Важно, что SIEM-системы в качестве самостоятельного решения не предназначены для обеспечения информационной безопасности, их задача заключается в анализе информации, поступающей из различных источников и дальнейшее выявление отклонений значений от норм.

Разработанные решение требуют создания и совершенствования существующих архитектур вычислительных комплексов (ВК), учитывающих обеспечения контроля доступа к сервисам.

II. Архитектура вычислительного комплекса с многоуровневым контролем доступа

Современная система доступа в веб-сервисам включает большое количество средств контроля, идентификации и верификации пользователей (рис. 1).

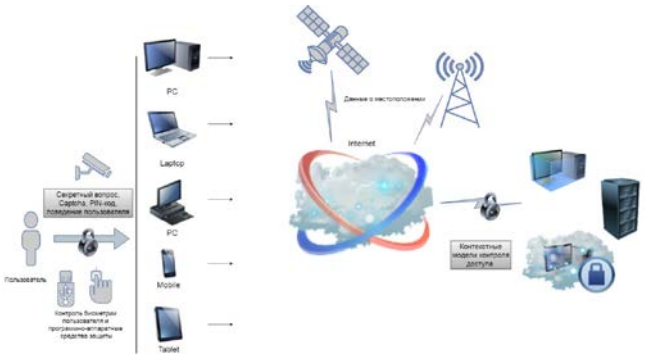

Рис. 1. Архитектура вычислительного комплекса с многоуровневым контролем доступа

ВК с системой с защищенного контроля доступа в веб-сервисам по компьютерным сетям обеспечивает возможность доступа с любого устройства, используемого пользователем для осуществления доступа. При этом происходит идентификация пользователя с использованием различных средств биометрии (например, отпечаток пальца, встроенные или внешние камеры), в процессе доступа и/или используются программные средства верификации (секретный вопрос, PIN-коды, Captcha), в настоящее время развиваются системы, связанные с анализом поведения пользователей (например, реакции пользователей на вопросы доступа или поведенческие модели). Элементом КД является учет локации пользователя, эти данные могут быть сравнены с данными в профиле пользователя. Для доступа к сервисам существует уровень контроля доступа, обеспечиваемым контекстными моделями. Такими моделями могут быть политики, построение моделей по журналам событий, соответствующим группам пользователей, а также специализированные методики, относящиеся к предметной области сервиса – медицинские, банковские, образовательные, имеют свои специализированные протоколы и особенности доступа. На уровне вычислительных ресурсов – серверов, виртуальных машин, баз данных, осуществляется аппаратный контроль доступа.

Анализ конфигурации вычислительных комплексов показывает, что повышение эффективности функционирования ВК может быть обеспечена за счет встраивания в архитектуру систем контроля доступа на всех уровнях передачи данных. При использовании внешних систем и внешних контуров контроля доступа возможен обход данных вне встроенного контура на программном или аппаратном уровне, а также требует больших ресурсов за счет дополнительных программных модулей и их стыковок с программно-аппаратным обеспечением вычислительных сервисов.

В архитектуре ВК, реализующий многоуровневый контроль доступа на основе концепции SIEM (Security information and event management) [16], необходимо реализовать сбор и хранение событий. Информации о состоянии различных элементов доступа к сервисам и управление событиями фиксируются обработчиками событий, распределенными по сети на всех уровнях. Каждый из обработчиков событий напрямую подключен к одному или нескольким источникам событий. Каждый из обработчиков событий имеет набор правил. При обработке событий от локально подключенных источников событий применяется набор правил. При обнаружении нарушения контроля доступа безопасности обработчик событий выдает предупреждение системы безопасности.

В общем виде архитектуру технологии SIEM можно представить в следующем виде. Метод реализуется ВК, предоставляющем веб-сервисы и подключенного к глобальной сети. Реализуемый метод контроля доступа включает преобразование правил SIEM в формальные представления. С точки зрения программной реализации необходимо обеспечить преобразования информации о доступе и правил управления событиями (SIEM) и развертывания правил SIEM в сети обработчиков событий. Программно может быть реализовано развертывание оптимизированных правил SIEM в сети обработчиков событий.

Аппаратное обеспечение ВК содержит одну или несколько виртуальных машин, серверов, систему хранения данных, сетевые устройства для доступа

клиентов к предоставляемым сервисам. Программное обеспечение, интегрированное в состав информационного обеспечения сервисов, управления доступом и обработке данных реализует следующие функции: фиксирование, передачу и хранение данных о доступе пользователей к вычислительным сервисам; генерация правил и прогноз значений (оценка вероятных значений) маркеров поведения пользователей в соответствии с заданными политиками защищенного доступа; сравнение полученных показателей с шаблонными персонифицированными значениями.

На рис. 2 изображена архитектура серверной системы обработки данных. Система предоставляет доступ клиентским устройствам к данным и услугам через веб-сервисы. Система обработки данных включает в себя структуру связи, которая обеспечивает связь между процессорным блоком, памятью, постоянным хранилищем, блоком коммуникационного оборудования.

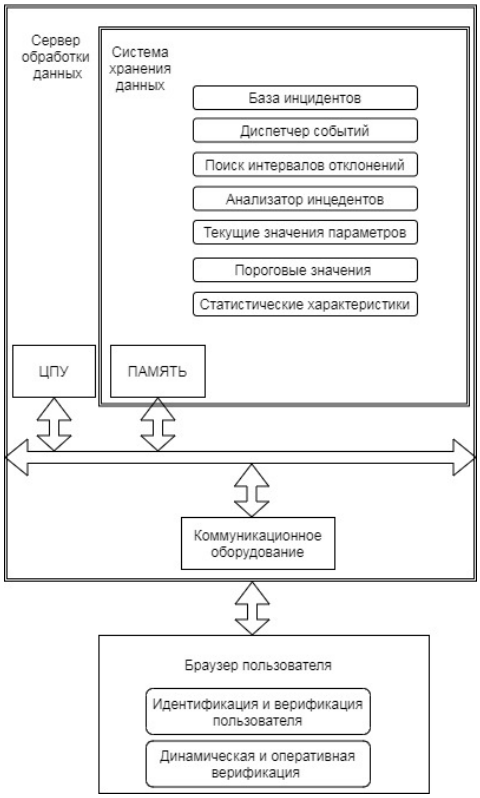

Рис. 2. Пример архитектуры системы обработки данных ВК

В архитектуре ВК, представляющих собой совокупность серверного оборудования и системы хранения данных (СХД), содержатся компоненты, реализующие контроль доступа к ВК. В архитектуру включены следующие компоненты:

- база инцидентов, содержащая информацию о возможных вариантах поведения системы и/или пользователя при несанкционированном доступе или угрозам целостности данных или компонентам ВК;
- диспетчер событий — может быть отдельным аппаратным модулем, или программно-аппаратным модулем, отвечающим за фиксирование событий безопасности и контроле доступа через сети;
- анализатор инцидентов — программный модуль, обеспечивающие оперативный контроль доступа на основе анализа событий;
- модуль сравнения полученных значений параметров доступа и характеристик событий с заданными моделями или пороговыми значениями;
- текущие значения параметров, характеризующие контроль доступа пользователей, сбор статистических данных;
- пороговые значения — база значений, характеризующие событие, при которых нужно обеспечивать программные или аппаратные сценарии защиты, блокировки пользователя, повторной верификации и т.д., а так же возможные варианты перерасчета пороговых значений;
- статистические характеристики — модуль, хранящий и вычисляющий значения статистических параметров конкретных пользователей, групп пользователей и других параметров, используемых при анализе инцидентов.

На рис. 3. изображены уровни обращения к данным при использовании веб-сервисов для доступа к ВК, включающие: уровень аппаратного и программного обеспечения; уровень виртуализации; уровень управления; уровень рабочих нагрузок.

Уровень оборудования и программного обеспечения включает в себя аппаратные и программные компоненты среды облачных вычислений. Компоненты оборудования могут включать, например, мэйнфреймы, серверы, блейд-серверы, устройства хранения, сетевые компоненты, а также программное обеспечение.

Уровень виртуализации обеспечивает взаимодействие виртуальных серверов, виртуальной памяти; виртуальные сети; виртуальные приложения и операционные системы; и виртуальные клиенты.

Уровень управления предоставляет следующие функции: распределение вычислительных ресурсов, которые используются для выполнения задач в облачной среде; модуль безопасности обеспечивает проверку потребителей и задач облака, а также защиту данных и других ресурсов; пользовательский портал обеспечивает доступ к среде облачных вычислений для потребителей и системных администраторов.

Уровень рабочих нагрузок включает реализацию функциональных возможностей, для которых может использоваться облачная среда ВК. Например, разработка программного обеспечения и управление жизненным циклом, предоставление обучения, обработка и анализа данных, обработку транзакций и анализ инцидентов безопасности.

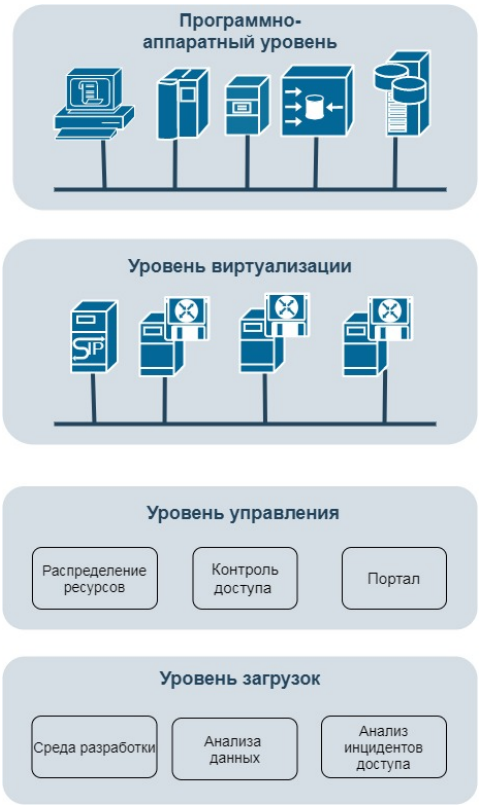

Рис. 3. Абстрактная 4-х уровневая архитектура облачного ВК

Разработана архитектура многоуровневого контроля доступа к вычислительным сервисам ВК с использованием сетей, что позволяет разрабатывать технологические решения и варианты реализации.

III. Методика анализа затрат вычислительных ресурсов для реализации систем контроля доступа

Включение в архитектуры ВК элементов, реализующих КД, будет увеличивать нагрузку на ресурсы и каналы связи. Поэтому необходимо на этапе проектирования ВК определить ресурсные затраты на компоненты КД, при этом, при выборе того или иного решения в качестве основного критерия оценки качества доступа к веб-ресурсов использовать QoS (гарантированную доставку данных, Quality-of-Service) [17], то есть обеспечивать интеграцию компонентов управления контролем доступа без уменьшения качества предоставляемых сервисов. Это может быть достигнуто за счет адекватных оценок затраченных ресурсов, чтобы в архитектуре ВК было зарезервировано необходимое количество памяти, ресурсов процессора, ширина каналов, время обработки данных и проч. Также, в случае возможных альтернатив технических решений, выбирается то решение, которое требует меньшее количество ресурсов.

модулей, обеспечивающий контроль доступа к данным по компьютерным сетям, очевидно, требует вычислительных ресурсов. Методику предлагается разработать на основе подхода [18-19] с использованием виртуальных стендов, обеспечивающих имитационную среду использования ВК.

Методика оценивания ресурсных затрат компонентов КД в ВК состоит из следующих шагов:

1. Создание виртуального экспериментального стенда, имитирующего среду использования системы контроля доступа.
2. Реализации программных моделей, реализующих обмен данных с контролем доступа и без данного модуля.
3. Планирование эксперимента и формирование набора данных для проведения достоверных исследований.
4. Измерение и обработка результатов экспериментов.
5. Получение значений ресурсов, требуемых для использования системы контроля доступа.
6. Формирования архитектуры вычислительного комплекса с учетом полученных значений затрат вычислительных ресурсов.

Рассмотрим, например, задачу оценки вычислительных затрат при записи событий в БД при работе с веб-сервисами по компьютерным сетям, с логированием действий пользователей.

Предполагается, что запись действий пользователей будет осуществляться в журнал событий. Для каждого устройства предлагается вести свою запись журнала действий. По причине разного вида интерфейса, который может меняться в зависимости от типа устройства. Пусть каждое действие в журнале событий включает следующие данные:

- идентификатор пользователя;
- устройство (идентификатор конкретного устройства);
- действие (уникальное название или идентификатор, чтобы не путать действия; если одно и то же действие может запустить разные кнопки, то и идентификатор действий тоже должен быть разным);
- время совершенного действия.

Для проведения экспериментальных исследований будем использовать следующие данные: объём исходного файла с данными составляет 450 МБ; файл содержит 12000 записей ResearchSubject и 55458 записей ResearchResult; данные представлены в слабоструктурированном формате JSON.

Перед началом эксперимента создаются 3 виртуальные машины (ВМ) (Client, Server, Database) с заданными характеристиками. Для повторных экспериментов, Если они были созданы ранее, то существующие ВМ предварительно удаляются, чтобы обеспечить чистоту эксперимента между повторениями. Структура экспериментального стенда приведена на рис. 4 и в табл. 1.

Рис. 4. Схема эксперимента

Таблица 1. Параметры виртуальных машин

	Количество ядер ЦПУ (шт)	Объём ОЗУ (МБ)	Максимально разрешённая загрузка ядер ЦПУ (%)	Пропускная способность подсистемы ввода-вывода (МБ)
Client	4	8192	100	-
Server	2	2048	100	-
Database (MongoDB)	2	2048	50	25

После создания ВМ, установки и запуска серверного ПО и СУБД, начинается сам эксперимент. Исходные данные загружаются в оперативную память ВМ Client. После их полной загрузки начинается отправка данных с заданными параметрами.

На первом этапе эксперимента производится отправка POST-запросов к серверу на сохранение записей ResearchSubject. На втором этапе эксперимента производится отправка POST-запросов к серверу на сохранение записей ResearchResult. В обоих случаях каждый POST-запрос содержит информацию только об одной записи. Таким образом, число запросов соответствует количеству исходных записей данных. Для логирования к запросу добавляется соответствующий программный код.

Эксперимент производится с двумя различными конфигурациями серверного ПО. В первом случае производится только сохранение данных. Во второй конфигурации добавлен программный код (рис. 5), осуществляющий логирование каждого запроса, полученного серверным ПО. При этом на каждый поступивший запрос в базе данных создаётся дополнительная запись. Эти записи создаются после выполнения каждого POST-запроса.

```
Model.afterRemote('**', (ctx, modelInstance) => {
  const params = ctx.req.body.data || ctx.req.body;
  const { id, researchSubjectId } = params;
  const userId = researchSubjectId || id;
  if (!userId) {
    return Promise.resolve();
  }
  return Model.app.models.ResearchSubject.findById(userId)
    .then((researchSubject) => {
      Model.app.models.ActionLog.logEvent(modelInstance,      ctx,      userId,
!!researchSubject);
    });
});
```

Рис. 5. Исходный код алгоритма логирования запросов к серверному ПО

Сбор данных об используемых ресурсах осуществляется с помощью утилиты Atop с интервалом в 1 секунду.

Отправка запросов осуществляется в 4 потока, до 10 одновременных запросов. Задержка между отправками пакетов по 10 запросов равна 300мс, а задержка между этапами эксперимента – 60 секунд. Максимальное ожидание ответа от сервера – 10 секунд.

Код исполняется с использованием Node.JS версии 12.x.

Серверное ПО запущено под управлением менеджера процессов PM2. Это означает, что объем выделенной для процесса ОЗУ не превышает 1024 МБ, количество параллельно работающих процессов равно числу ядер ЦПУ (т.е. 2), задержка перед повторным запуском процесса в случае отказа – 5 секунд, а максимально число перезапуска процесса – 1000 раз.

Код исполняется с использованием Node.JS версии 12.x.

Установлена MongoDB версии 4.2.

Результаты вычислительного эксперимента приведены на рис. 6–11 и табл. 2.

Рис. 6. Используемые ресурсы ЦПУ ВМ Client, %

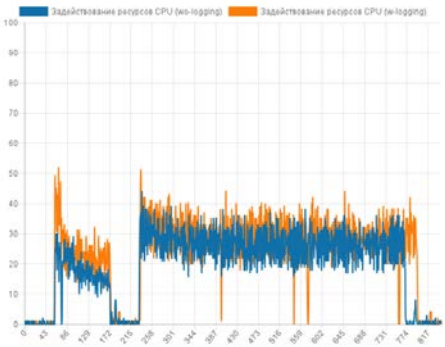

Рис. 7. Используемые ресурсы ЦПУ ВМ Server, %

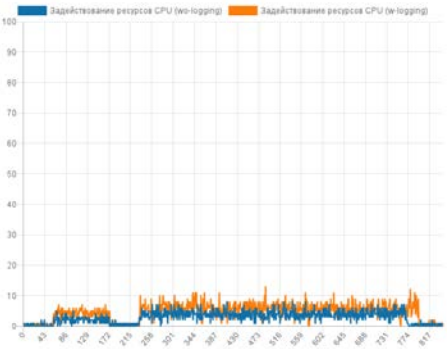

Рис. 8. Используемые ресурсы ЦПУ ВМ Mongodb, %

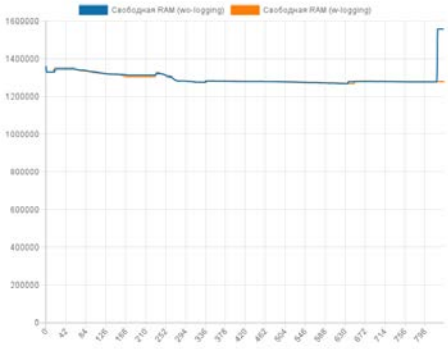

Рис. 9. Используемые ресурсы памяти ВМ Client, Mb

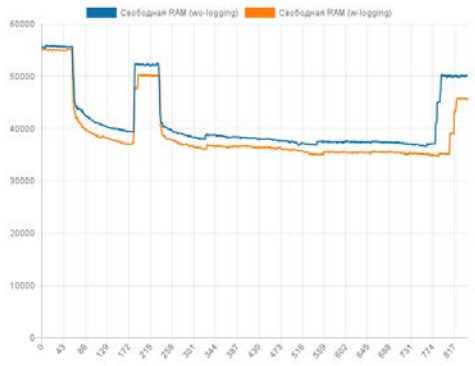

Рис. 10. Используемые ресурсы памяти ВМ Server, Mb

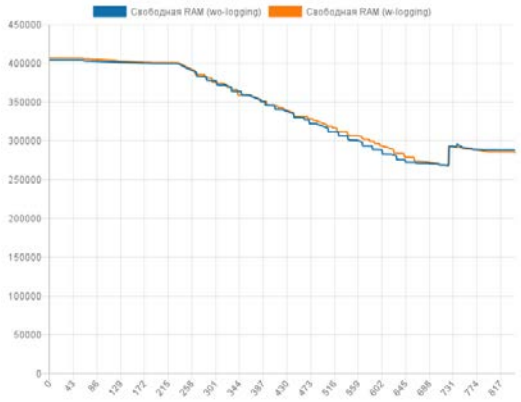

Рис. 11. Используемые ресурсы памяти ВМ Mongodb, Mb

Таблица 2. Показатели ресурсных затрат на логирование действий пользователей

Ресурсный показатель	Значение без использования логирование	Значение с использованием логирования	Разница в %
ЦПУ ВМ client	8.498212157330155	8.2170361726954	3.3
ЦПУ ВМ server	19.71090047393365	22.230149597238	12.7
ЦПУ ВМ mongodb	2.8590047393364	4.3716915995397	52.9
Память ВМ client	1296828.1859356	1295666.4060676	0.08
Память ВМ server	41345.797393365	39147.056386651	5.32
Память ВМ mongodb	340911.11492890	341359.78826237	0,13

Для рассматриваемого примера экспериментально установлено, что использование логирования действий пользователей существенно влияет только на загрузку процессора сервера и базы данных, незначительно увеличивает загрузку памяти сервера, что требует при внедрении КД внесение соответствующих запасов ресурсов в ВК для логирования.

IV. Заключение

Рассмотрены вопросы построения архитектуры вычислительного комплекса с многоуровневым контролем доступа к веб-сервисам по общедоступным сетям. Рассмотрена типовая архитектура многоуровневого защищенного контроля доступа, сформированы принципы технологической программно-аппаратной реализации, выделены логические уровни архитектуры ВК.

В работе предложена методика экспериментальной оценки ресурсных затрат на реализацию технологии контроля доступа. Оценка проводится на основе имитационного стенда на виртуальных машинах с использованием реализации среды использования разрабатываемой системы. Приведен пример экспериментальной оценки ресурсов для логирования действий пользователя в слабоструктурированном формате данных.

Библиография

[1] J. R. Gil-Garcia, M. Á. Flores-Zúñiga , “Towards a comprehensive understanding of digital government success: Integrating implementation and adoption factors,” *Government Information Quarterly*, vol. 37, no. 4, p. 101518, 2020.

[2] K. E. Lewinter, S. M. Hudson, L. Kysh, M. Lara, C. L. Betz, J. Espinoza, “Reconsidering reviews: the role of scoping reviews in digital medicine and pediatrics,” *NPJ Digital Medicine*, vol. 3, no. 1, p. 1-4, 2020.

[3] A. Emejulu, C. McGregor, “Towards a radical digital citizenship in digital education,” *Critical Studies in Education*, vol. 60, no. 1, pp. 131-147, 2019.

[4] G. Elia, A. Margherita, G. Passiante, “Digital entrepreneurship ecosystem: How digital technologies and collective intelligence are reshaping the entrepreneurial process,” *Technological Forecasting and Social Change*, vol. 150, p. 119791, 2020.

[5] P. De Hert, V. Papakonstantinou, G. Malgieri, L. Beslay, I. Sanchez, “The right to data portability in the GDPR: Towards user-centric interoperability of digital services,” *Computer Law & Security Review*, vol. 34, no. 2, p. 193-203, 2018.

[6] R. El Sibai, N. Gemayel, J. Bou Abdo, J. Demerjian, “A survey on access control mechanisms for cloud computing,” *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 2, p. e3720, 2020.

[7] J. Sheng, J. Amankwah-Amoah, X. Wang, “Technology in the 21st century: New challenges and opportunities,” *Technological Forecasting and Social Change*, 2019, 143, 321-335.

[8] F. Cai, J. He, Z. Ali Zardari, S. Han, “Distributed management of permission for access control model,” *Journal of Intelligent & Fuzzy Systems*, vol. 38, no. 2, p. 1539-1548, 2020.

[9] M. H. Yarmand, K. Sartipi, D. G. Down, “Behavior-based access control for distributed healthcare systems,” *Journal of Computer Security*, vol. 21, no. 1, p. 1-39, 2013.

[10] A. Walker, J. Svacina, J. Simmons, T. Cerny, “On automated role-based access control assessment in enterprise systems,” In *Information Science and Applications*, Springer, Singapore, pp. 375-385, 2020.

[11] S. Kirrane, A. Mileo, S. Decker, “Access control and the resource description framework: A survey,” *Semantic Web*, vol. 8, no. 2, p. 311-352, 2017.

[12] X. C. Yin, Z. G. Liu, B. Ndibanje, L. Nkenyereye, S. M. Riazul Islam, “An IoT-based anonymous function for security and privacy in healthcare sensor networks,” *Sensors*, vol. 19, no. 14, p. 3146, 2019.

[13] A. S. M. Kayes, R. Kalaria, I. H. Sarker, M. Islam et al., “A survey of context-aware access control mechanisms for cloud and fog networks: Taxonomy and open research issues,” *Sensors*, vol. 20, no. 9, p. 2464, 2020.

[14] J. Qiu, Z. Tian, C. Du, Q. Zuo, S. Su, B. Fang, “A survey on access control in the age of internet of things,” *IEEE Internet of Things Journal*, vol. 7, no. 6, p. 4682-4696, 2020.

[15] F. Menges, T. Latzo, M. Vielberth et al. “Towards GDPR-compliant data processing in modern SIEM systems,” *Computers & Security*, vol. 103, p. 102165, 2021.

[16] Ш.Г. Магомедов, П.В. Колясников, Е.В. Никульчев, “Разработка технологии контроля доступа к цифровым порталам и платформам на основе встроенных в интерфейс оценок времени реакций пользователей,” *Российский технологический журнал*, т. 8, № 6, с. 34-46, 2020.

[17] A. Gusev, D. Ilin, P. Kolyasnikov, E. Nikulchev, “Effective selection of software components based on experimental evaluations of quality of operation,” *Engineering Letters*, vol. 28, no. 2, p. 420–427, 2020.

Об авторе:
Шамиль Гасангусейнович Магомедов — кандидат технических наук, доцент, заведующий кафедрой «Интеллектуальные системы информационной безопасности» Института комплексной безопасности и специального приборостроения МИРЭА — Российский технологический университет

Architecture of a computing complex for web services and portals with multilevel access control over public networks

S. Magomedov

***Abstract*— The article is devoted to the development of the architecture of computing systems with multilevel access control to web services over public networks. Information technologies are becoming global and cross-border which leads them to become an integral part of all spheres of activity of an individual, society and state. Their effective application is a factor in accelerating the economic development of the state and the formation of an information society. At the same time, the rapid scientific and technological progress and the constant improvement of software and hardware require, on the one hand, the same constant improvement and increase in the efficiency of the functioning of all components, in terms of reducing costs, improving quality, reliability, safety, but at the same time providing a high level security of access to computing resources, without violating the requirements of existing laws to protect personal and confidential data. Taking into account the ever-increasing requirements for information infrastructure, the development of a computing complex architecture with various access control mechanisms that can expand the capabilities of information security tools is a very urgent task. Despite the fact that modern systems have a wide range of mechanisms and means of protecting information, they are clearly not enough, taking into account constant hacks and information leaks, therefore it is necessary to determine their role and place in each level of computing architecture. The paper proposes the architecture of a data processing system of a computing complex, with the described components serving to ensure access control. Also presented are the levels of data access when components access the computing complex, with a detailed description of the functioning and filling of modules. The results of experimental studies for assessing resource costs when tracking all the user actions are presented.**

***Keywords*— architecture of a computing complex, access control, web-service, SIEM.**

Библиография

[1] J. R. Gil-Garcia, M. Á. Flores-Zúñiga , “Towards a comprehensive understanding of digital government success: Integrating implementation and adoption factors,” *Government Information Quarterly*, vol. 37, no. 4, p. 101518, 2020.

[2] K. E. Lewinter, S. M. Hudson, L. Kysh, M. Lara, C. L. Betz, J. Espinoza, “Reconsidering reviews: the role of scoping reviews in digital medicine and pediatrics,” *NPJ Digital Medicine*, vol. 3, no. 1, p. 1-4, 2020.

[3] A. Emejulu, C. McGregor, “Towards a radical digital citizenship in digital education,” Critical Studies in Education, vol. 60, no. 1, pp. 131-147, 2019.

[4] G. Elia, A. Margherita, G. Passiante, “Digital entrepreneurship ecosystem: How digital technologies and collective intelligence are reshaping the entrepreneurial process,” *Technological Forecasting and Social Change*, vol. 150, p. 119791, 2020.

[5] P. De Hert, V. Papakonstantinou, G. Malgieri, L. Beslay, I. Sanchez, “The right to data portability in the GDPR: Towards user-centric interoperability of digital services,” *Computer Law & Security Review*, vol. 34, no. 2, p. 193-203, 2018.

[6] R. El Sibai, N. Gemayel, J. Bou Abdo, J. Demerjian, “A survey on access control mechanisms for cloud computing,” *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 2, p. e3720, 2020.

[7] J. Sheng, J. Amankwah-Amoah, X. Wang, “Technology in the 21st century: New challenges and opportunities,” *Technological Forecasting and Social Change*, 2019, 143, 321-335.

[8] F. Cai, J. He, Z. Ali Zardari, S. Han, “Distributed management of permission for access control model,” *Journal of Intelligent & Fuzzy Systems*, vol. 38, no. 2, p. 1539-1548, 2020.

[9] M. H. Yarmand, K. Sartipi, D. G. Down, “Behavior-based access control for distributed healthcare systems,” *Journal of Computer Security*, vol. 21, no. 1, p. 1-39, 2013.

[10] A. Walker, J. Svacina, J. Simmons, T. Cerny, “On automated role-based access control assessment in enterprise systems,” In *Information Science and Applications*, Springer, Singapore, pp. 375-385, 2020.

[11] S. Kirrane, A. Mileo, S. Decker, “Access control and the resource description framework: A survey,” *Semantic Web*, vol. 8, no. 2, p. 311-352, 2017.

[12] X. C. Yin, Z. G. Liu, B. Ndibanje, L. Nkenyereye, S. M. Riazul Islam, “An IoT-based anonymous function for security and privacy in healthcare sensor networks,” *Sensors*, vol. 19, no. 14, p. 3146, 2019.

[13] A. S. M. Kayes, R. Kalaria, I. H. Sarker, M. Islam et al., “A survey of context-aware access control mechanisms for cloud and fog networks: Taxonomy and open research issues,” *Sensors*, vol. 20, no. 9, p. 2464, 2020.

[14] J. Qiu, Z. Tian, C. Du, Q. Zuo, S. Su, B. Fang, “A survey on access control in the age of internet of things,” *IEEE Internet of Things Journal*, vol. 7, no. 6, p. 4682-4696, 2020.

[15] F. Menges, T. Latzo, M. Vielberth et al. “Towards GDPR-compliant data processing in modern SIEM systems,” *Computers & Security*, vol. 103, p. 102165, 2021.

[16] S. G. Magomedov, P.V. Kolyasnikov, E.V. Nikulchev, “Development of technology for controlling access to digital portals and platforms based on estimates of user reaction time built into the interface,” *Russian Technological Journal*, vol. 8, no. 6, p. 34-46, 2020. [In RUS]

[17] A. Gusev, D. Ilin, P. Kolyasnikov, E. Nikulchev, “Effective selection of software components based on experimental evaluations of quality of operation,” *Engineering Letters*, vol. 28, no. 2, p. 420–427, 2020.

Manuscript received Febr., 15, 2021.

S. Magomedov, PhD, associate professor, MIREA – Russian Technological University (magomedov_sh@mirea.ru)