Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Высшая школа программной инженерии

Практическая работа №2

«Первичный поиск научных источников и выбор критериев»

Выполнил
студент гр.5130903/20302 Н. Ю. Каралюс
<подпись>

Руководитель
ст. преподаватель А. О. Жаранова
<подпись>
«___» _____ 2025 г.

Санкт-Петербург
2025

Задание

Задание: Провести первичный поиск научных источников для выбранной темы.

1. В качестве входных данных нужно использовать ключевые слова из Практического задания № 1;

2. Провести первичный поиск по ключевым словам в наукометрических базах Google Scholar (сравнить результаты);

3. Провести первичный поиск по ключевым словам в eLibrary и Киберленинка (возможны другие источники) (сравнить результаты);

4. Оценить найденные статьи, отобрать наиболее подходящие (2-3 статьи);

5. Провести поиск связанных источников:

    5.1. Выбрать сильную статью в своем списке;

    5.2. Посмотреть статьи, на которые она ссылается, используя раздел Библиография (References);

    5.3. Проверить статьи, которые ссылаются на неё, используя перечисленные выше наукометрические базы;

6. Составить таблицу и указать в ней критерии для сравнения и/или ограничения, которые Вы нашли.
   Важно: нужно отобрать 15-20 разных источников (публикации в интернете, статьи на русском и английском, учебники, патенты, стандарты).

Желательно соблюдать баланс между новыми и "старыми" (авторитетными) публикациями. Не менее 50% источников должно быть не старше 5 лет.

Сравнительная таблица

Сравнение источников сделано в виде списка: таблица не вмещает нужное количество информации и ухудшает читаемость.

1. Authentication and Authorization in Microservices Architecture: A Systematic Literature Review

Запрос: auth microservice review.

Тип источника: статья.

Цитирование по ГОСТ: Аlmeida M. G. de, Canedo E. D. Authentication and Authorization in Microservices Architecture: A Systematic Literature Review // Applied Sciences. — 2022. — Т. 12, № 6. — С. 3023. — DOI 10.3390/app12063023Almeida M. G. de, Canedo E. D. Authentication and Authorization in Microservices Architecture: A Systematic Literature Review // Applied Sciences. — 2022. — Т. 12, № 6. — С. 3023. — DOI 10.3390/app12063023.

Полезные заметки: По сути просто статка механизмов безопасности. oauth 2.0, jwt, sso на первом месте. Есть TLS, но mTLS вообще нет. Очень много ссылок на другие статьи.

Критерии сравнения: архитектурные подходы, глубина анализ, упоминание AAA в

распределенных системах, пользовательская аутентификация и авторизация.

Ограничения: обзорная статья по другим источникам. Хорошо подходит как стартовая точка для поиска других источников.

2. Research Trends and Recommendations for Future Microservices Research

Запрос: microservice research.

Тип источника: статья.

Цитирование по ГОСТ: Research Trends and Recommendations for Future Microservices Research / Z. Stojanov [и др.] // Trudy ISP RAN/Proc. ISP RAS. — 2024. — Т. 36, № 1. — С. 105—130. — DOI 10.15514/ISPRAS-2024-36(1)-7.

Полезные заметки: Большое исследование с трендами микросервисной архитектуры. Много ссылок на другие исследования. Безопасность – топ 1 статей, посвященных сложностям микросервисной архитектуры. Приведено 13 источников, затрагивающих безопасность, может быть полезно. Источники записаны как secondary studies (SS14, SS15, SS 16, SS18, SS23, SS27, SS28, SS29, SS33, SS34, SS35, SS36, SS42)

Критерии сравнения: архитектурные подходы, глубина анализ, упоминание AAA в распределенных системах, пользовательская аутентификация и авторизация.

Ограничения: обзорная статья по другим источникам. Хорошо подходит как стартовая точка для поиска других источников.

3. Безопасность микросервисов: управление секретами и безопасная аутентификация

Запрос: безопасность аутентификация микросервисы.

Тип источника: статья.

Цитирование по ГОСТ: Glumov K. S. Безопасность микросервисов: управление секретами и безопасная аутентификация // Актуальные исследования. — 2024. — 45 (227). — С. 23—29. — DOI 10.5281/zenodo.14029864.

Полезные заметки: Максимально по верхам.Описал какие есть проблемы и какие есть инструменты, чтобы их решать. Хорошее вступление с понятным объяснением. Единственный их всех отметил про ротацию ключей, но не более. Все-таки важная тема, не стоит про нее забывать. Ссылки на другие статьи, много ссылок на habr.

Критерии сравнения: полнота рассмотрения AAA, практическая применимость.

Ограничения: небольшой объем, обзорный стиль.

4. Механизмы межсервисной аутентификации в приложениях с микросервисной архитектурой

Запрос: из источников статьи 3 (Безопасность микросервисов: управление секретами и безопасная аутентификация).

Тип источника: статья.

Цитирование по ГОСТ: Zimina K. I., Laponina O. R. Механизмы межсервисной аутентификации в приложениях с микросервисной архитектурой // International Journal of Open Information Technologies. — 2023. — Т. 11, № 5. — С. 146—154. — URL: https://cyberleninka.ru/article/n/mehanizmy-mezhservisnoy-autentifikatsii-v-prilozheniyah-s-mikroservisnoy-arhitekturoy (дата обращения: 25.10.2025).

Полезные заметки: Приходит к использованию mTLS. Хорошо изложено про основы микросервисной архитектуры. Много про способы коммуникации микросервисов и их форматы, кажется излишним. С асинхронным взаимодействием вообще дрянь, код какой-то пошел ненужный, модель publisher-subscriber как-то вяло объяснена (да и зачем в этой теме?), сразу к коду. Хороший пункт про доверенные сети. В mTLS интересный пункт, что async взаимодействие через Microservice D. Много ошибок. Некоторые источники не открываются. Утверждение, если сервис с ПД, то лучше всего mTLS. А почему так особо не пояснили. Есть упоминание, что закрытый ключ должен храниться где-то в секрете. Есть про Certification Authority Server. Рассказывает про сложности реализации mTLS. Про Service Mesh упоминаний при этом нет. mTLS здесь как бы в вакууме.

Критерии сравнения: фокус на протоколах, микросервисная архитектура, практичность схем.

Ограничения: узкая направленность.

5. Использование управления доступом на основе атрибутов в протоколе OAuth 2.0

Запрос: oauth2 микросервисы.

Тип источника: статья (киберленинка).

Цитирование по ГОСТ: Беловодов А. В., Лапонина О. Р. Использование управления доступом на основе атрибутов в протоколе OAuth 2.0 // International Journal of Open Information Technologies. — 2023. — 10 (63). — С. 204—214.

Полезные заметки: Предлагается модель сочетания RBAC/ABAC в рамках oauth2. ABAC-сервис работает совместно с гейтвеем, обеспечивая гибкую авторизацию. Показано, что такое решение вводит сквозную безопасность для междоменных вызовов.

Критерии сравнения: комбинация RBAC/ABAC в микросервисах, oauth2.

Ограничения: изложен только концепт, нет практических примеров, готовых решений.

6. Использование управления доступом на основе атрибутов в протоколе OAuth 2.0

Запрос: jwt microservices gateway.

Тип источника: статья.

Цитирование по ГОСТ: Ferreira D. M., Matias M. J. Enhancing Effectiveness and Security in Microservices Architecture // Procedia Computer Science. — 2023. — Т. 239. — С. 2260—2269. — DOI 10.1016/j.procs.2024.06.417.

Полезные заметки: Предлагает гибридную схему: gateway выпускает JWT-токены и использует HTTPS только при необходимости, чтобы снизить накладные расходы. Показано, что избыточное шифрование ухудшает производительность; предлагается баланс безопасности и скорости. Я не считаю, что за выпуск JWT должен отвечать gateway, но это не первая статья с подобным решением.

Критерии сравнения: JWT, Gateway, производительность систем аутентификации и авторизации.

Ограничения: производительность не приоритетная тема исследования для меня.

7. Использование управления доступом на основе атрибутов в протоколе OAuth 2.0

Запрос: service mesh sber.

Тип источника: пост в блоге.

Цитирование по ГОСТ: Blog P. S. Как мы строили безопасную микросервисную архитектуру с Service Mesh: взгляд изнутри. — URL: https://platformv.sbertech.ru/blog/kak-my-stroili-bezopasnuyu-mikroservisnuyu-arhitekturu-s-service-mesh-vzglyad-iznutri (дата обращения: 25.10.2025).

Полезные заметки: Явно проговорено, что в k8s нет встроенных механизмов аутентификации, авторизации, управления доступом и шифрования трафика. Статья про очередной форк сбера, на этот раз форкнули Istio. Большое количество практических примеров по конфигурации Istio. Описаны разные ресурсы и их применение. Разбор mTLS и JWT аутентификации. При описании JWT аутентификации появляется IdP. Текст как будто не проверяли. Автор с одной темы перескакивает на другую (в разделе аутентификации). Дублируются предложения. Видны признаки использования LLM для написания статьи. Описываются политики авторизации Istio.

Критерии сравнения: примеры внедрения, практическое описание.

Ограничения: не научный источник, но полезнее некоторых научных.

8. Использование управления доступом на основе атрибутов в протоколе OAuth 2.0

Запрос: service mesh.

Тип источника: статья.

Цитирование по ГОСТ: Sikha G., Dorai V. End-to-End Security and Operations for Kubernetes– Based Microservices // International Research Journal of Modernization in Engineering Technology and Science. — 2025. — Т. 7, № 8. — С. 2582—5208. — DOI 10.56726/IRJMETS82304.

Полезные заметки: Ссылки плохие – только на документацию инструментов и известные манифесты. Практики имплементации с использованием Helm, Prometheus, Grafana, Fluent Bit, and Spinnaker. Рекомендуют zero trust architecture, mTLS for s2s encryption, IdP, secure vaults for secrets. Хорошее замечание, что сначала безопасность систем (etcd, k8s API, ноды) кластера – потом все остальное. Политики для inter-pod коммуникации. Секреты,

хранимые внутри k8s должны быть зашифрованы. Упоминание Istio для s2s mTLS. Уделено внимание защите CI/CD. Есть пример конечного сервиса. Статья поверхностно обозревает технологиями, которые нужны для обеспечения безопасности в k8s кластере.

Критерии сравнения: практическое описание Istio и RBAC.

Ограничения: не лучшая репутация журнала, больше практическая статья.

9. A Container-Native IAM Framework for Secure Green Mobility: A Case Study with Keycloak and Kubernetes

Запрос: keycloak kubernetes.

Тип источника: статья.

Цитирование по ГОСТ: A Container-Native IAM Framework for Secure Green Mobility: A Case Study with Keycloak and Kubernetes / A. Sousa [и др.] // Information. — 2025. — Т. 16, № 9. — С. 802. — DOI 10.3390/info16090802.

Полезные заметки: Предложена архитектура IAM на основе Keycloak и K8s для IoT. Подтверждена экспериментом: контейнерный вариант показывает в 3–4 раза лучшую производительность и меньшую задержку по сравнению с традиционными решениями mdpi.com. Демонстрирует интеграцию IdP (Keycloak) и принципы Zero Trust в Kubernetes. Также есть сравнение различных IdP, может быть полезным.

Критерии сравнения: практическое использование IdP в kubernetes.

Ограничения: специализация на интернете вещей.

10. Role-Based Access Control (RBAC) in Modern Cloud Security Governance: An In-depth Analysis

Запрос: kubernetes rbac.

Тип источника: статья.

Цитирование по ГОСТ: Akuthota A. K. Role-Based Access Control (RBAC) in Modern Cloud Security Governance: An In-depth Analysis // International Journal of Scientific Research in Computer Science, Engineering and Information Technology. — 2025. — Т. 11, № 2. — С. 3297—3311. — DOI 10.32628/CSEIT25112793.

Полезные заметки: Разбирает модель RBAC в контексте современных облаков и микросервисов. Показывается как RBAC интегрируется с Zero Trust и механизмами мониторинга поведения пользователей. Даются реальные метрики крупной платформы. Уделено внимание применению ИИ для оптимизации доступа, предиктивной аналитики и поведенческих моделей. Показано, как RBAC связывается с федеративной аутентификацией.

Критерии сравнения: RBAC, Zero trust, Idp, Federation.

Ограничения: про RBAC в целом, хотя искал kubernetes.

11. Implementing Zero Trust Architecture for Microservices

Запрос: zero trust microservices.

Тип источника: технический отчет.

Цитирование по ГОСТ: Madupati B. Observability in Microservices Architectures: Leveraging Logging, Metrics, and Distributed Tracing in Large-Scale Systems: тех. отч. / SSRN (Elsevier). — 2023. — DOI 10.5281/zenodo.13951033.

Полезные заметки: Все не читал, документ огромный. Представляет собой руководство NIST по архитектуре Zero Trust в микросервисах. Полезно для формализации принципов AAA и сетевой изоляции. Хорошо подходит для теоретической части и описания требований безопасности.

Критерии сравнения: принципы zero trust, формализация.

Ограничения: объем материала.

12. Towards Concurrent Audit Logging in Microservices

Запрос: audit microservices.

Тип источника: статья.

Цитирование по ГОСТ: Amir-Mohammadian S., Zowj A. Y. Towards Concurrent Audit Logging in Microservices // Proceedings of the 2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC). — 2021. — С. 1357—1361. — DOI 10.1109/COMPSAC51774.2021.00191.

Полезные заметки: Инструмент на Java Spring для автоматического внедрения спецификаций аудита в код микросервисов. Обеспечивает правильное логирование в условиях конкурентности: события из разных сервисов корректно объединяются в общие логи. Подчеркивает важность формализации требований к логам. Много сложных непонятных математических выкладок.

Критерии сравнения: формализм и корректность генерации логов.

Ограничения: фокус на Java стек.

13. Observability in Microservices Architectures: Leveraging Logging, Metrics, and Distributed Tracing in Large-Scale Systems

Запрос: audit microservices.

Тип источника: статья.

Цитирование по ГОСТ: Madupati B. Observability in Microservices Architectures: Leveraging Logging, Metrics, and Distributed Tracing in Large-Scale Systems: тех. отч. / SSRN (Elsevier). — 2023. — DOI 10.5281/zenodo.13951033.

Полезные заметки: Рассматривает взаимосвязь логирования, метрик и трассировок. Обозначает основные проблемы observability: избыток данных, сложность настройки инструментов и лучшие практики. Помогает оценить, как организовать сбор данных для аудита и мониторинга.

Критерии сравнения: инструменты и принципы observability.

Ограничения: нет фокуса на AAA, технический обзор.

14. Listening to what the system tells us: Innovative auditing for distributed systems

Запрос: audit microservices.

Тип источника: статья.

Цитирование по ГОСТ: Listening to what the system tells us: Innovative auditing for distributed systems /P. Di Pilla [и др.] // Frontiers in Computer Science. — 2023. — Т. 4. — С. 1020946. — DOI 10.3389/fcomp.2022.1020946.

Полезные заметки: Во вступлении упоминается AI для организации безопасности и анализа входящих сигналов. Статья про автоматизированный анализ логов на основе ELK. Разрабатывают инструмент анализа логов блокчейн систем. То есть распределенный здесь не кластер, а сама система. Хотят использовать NLP для поиска по логам. Обработка логов моделями происходит в реальном времени. Система умеет работать с неструктурированными логами. Система предоставляет сервис мониторинга для обнаружения аномалий. В системе одновременно используются модели для непересекающихся задач. Модели системы учатся понимать слова предметной области, которые другие классификаторы могли бы пропустить. Зачем-то есть анализ тональности логов (позитивный, негативный, нейтральный). Дают пояснение, что это хороший индикатор происходящего в системе. На синтетических тестах показали, что могут автоматически обнаружить DDoS атаки. Много ссылок на похожие решения, так что разработанное решение интересно, но не уникально.

Критерии сравнения: методы обработки логов.

Ограничения: нет фокуса на AAA, только аспекты аудита.

15. Security Audit Logging in Microservice-Based Systems: Survey of Architecture Patterns

Запрос: audit microservices.

Тип источника: статья.

Цитирование по ГОСТ: Barabanov A. I., Makrushin D. Security Audit Logging in Microservice-Based Systems: Survey of Architecture Patterns // Вопросы кибербезопасности. — 2021. — № 2. — С. 71—80. — DOI 10.21681/2311-3456-2021-2-71-80.

Полезные заметки: Знакомое вступление, но теперь средство решение не M2M auth, а про логирование. Делали анализ академических источников, стандартов, документации, конференций. Затрагивают 3 основные темы: Threat modeling, Security Design, Implementation. Статья содержит набор статей по безопасности микросервисных архитектур. Много пишут про методы исследования, может быть полезно. Нашли 8 угроз для модели, когда приложение напрямую отправляет логи в сервис логирования (в основном, связанные с сетью). Крупные игроки используют паттерн с агентом-сборщиком логов из файла (МКСы пишут в этот файл), публикующим логи в message broker в HA кластере. Брокер доставляет логи в сервис логов. Агент деплоится демоном или в k8s

сайдкаром. Крутая таблица №3 с определением категорий событий, которые надо логировать. Рекомендация по использованию структурированных логов. Крутой поинт, что за логирование и его инструментарий должна отвечать инфраструктурная команда. It is preferable to ask development teams to use standard output to write log messages. МКС и агент должны использовать mTLS, чтобы устранить все сетевые уязвимости. Фильтрация и санитизация логов – ответственность агента CorrelationID и хелсчеки агента как лучшие практики. Указаны категории контекстов для включения в логи.

Критерии сравнения: роли логирования в безопасной архитектуре.

Ограничения: обзорный характер статьи.