

# Какие риски несёт с собой заказная разработка?

30.06.2022 / Андрей Минаев

Classification: Public

PUBLIC  
—  
ОБЩЕДОСТУПНО

# Марки «ФОЛЬКСВАГЕН Груп Рус»



# Мы активно используем заказную разработку и Outsourcing



30+

ИТ подрядчиков



300+

Внешних  
сотрудников



150+

ИТ систем



50+

Бизнес  
систем



Вы - менеджер проекта.  
Ваша задача - внедрить приложение  
в соответствие с требованиями  
бизнеса. Вы привлекаете внешнего  
подрядчика для разработки  
приложения.  
О чём нужно знать, прежде чем  
преступить к проекту?

# Разработчик устанавливает разработанное приложение на Web/Application Server

**Для кого** наше приложение? внутренние или внешние пользователи, другие системы?

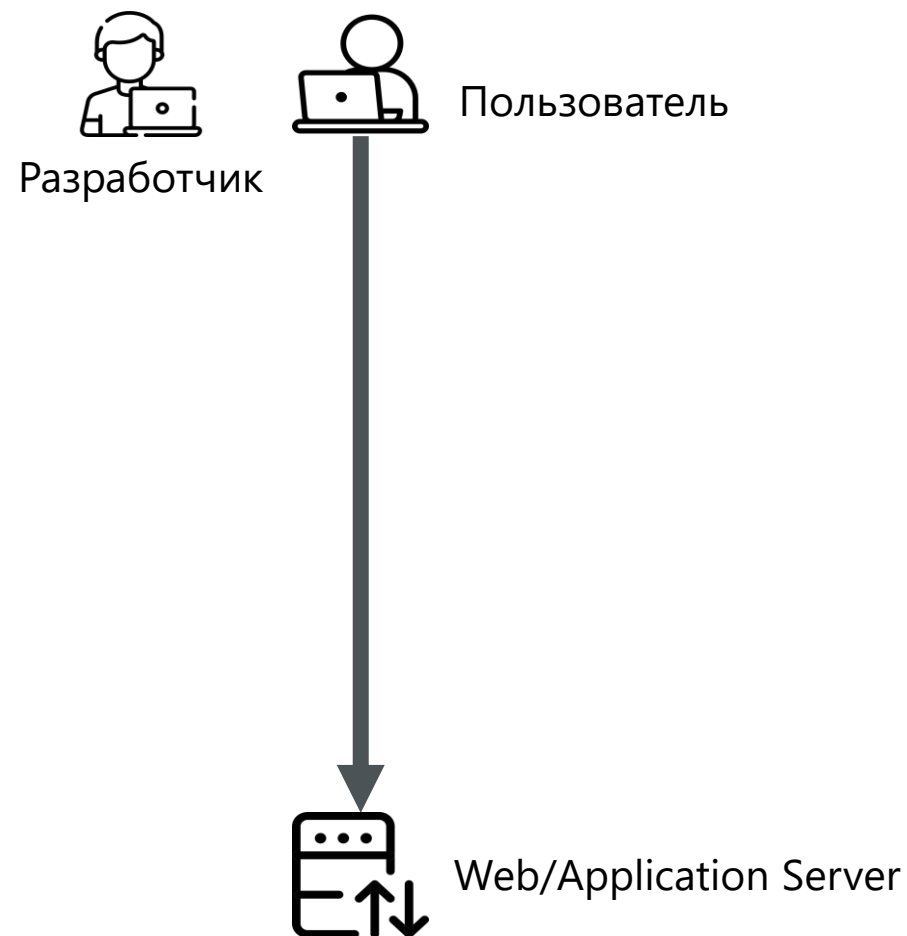
**Сколько** будет пользователей?

**Будут ли** персональные данные?

**Как** пользователи будут получать доступ? HTTPS?

**Будет ли** мобильная версия для телефона?

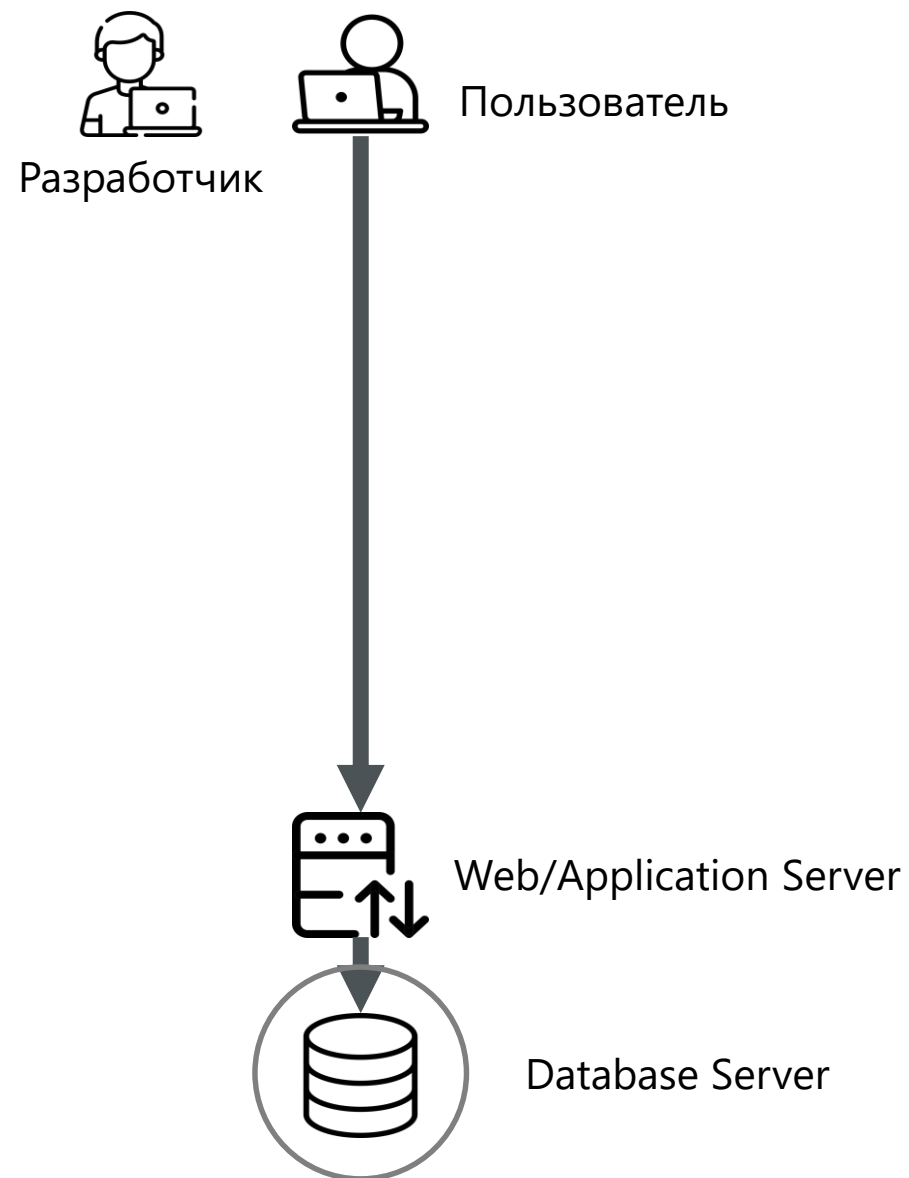
**Как** другие приложения будут получать доступ? HTTPS API, Kafka, SMB, SFTP?



# Данные приложения хранятся в базе данных

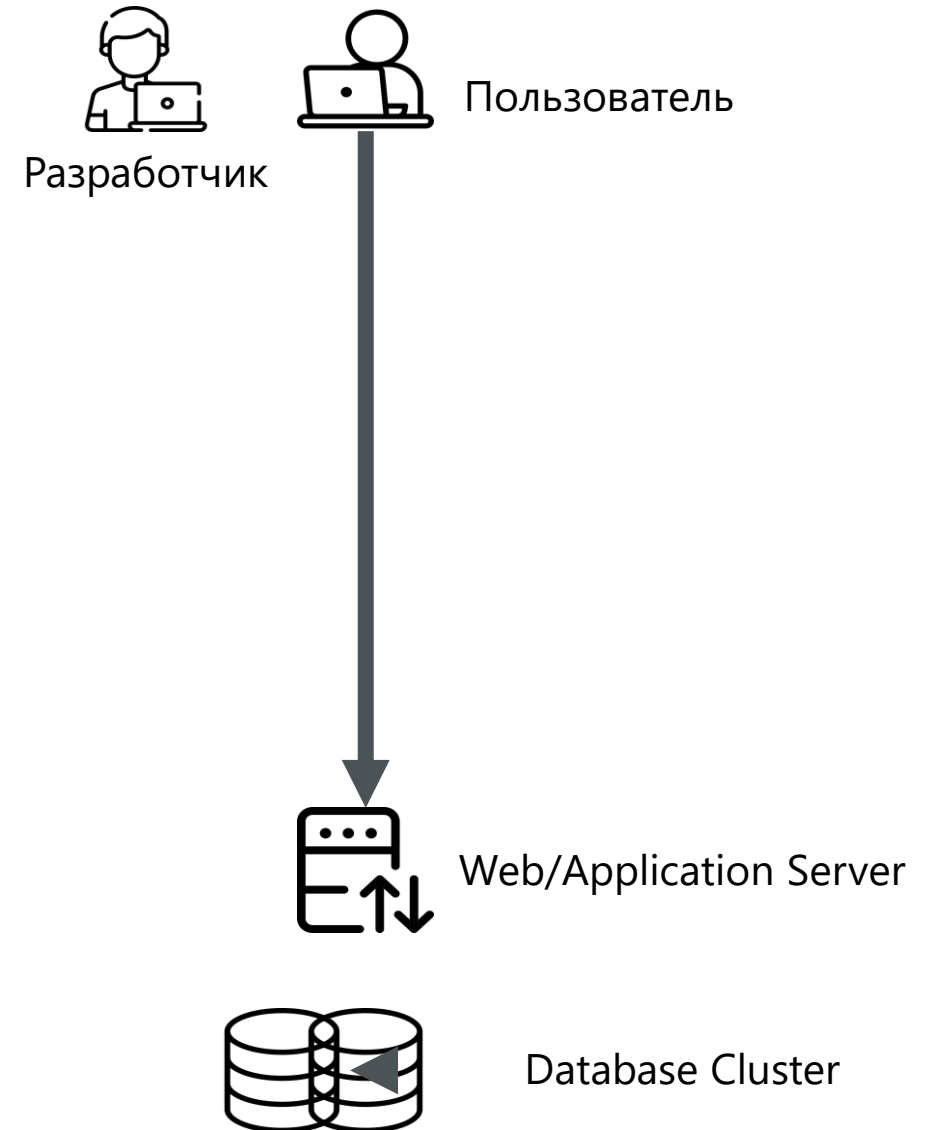
Какая СУБД будет использоваться?

Сколько места потребуется для данных?



# Для отказоустойчивости используем кластер баз данных

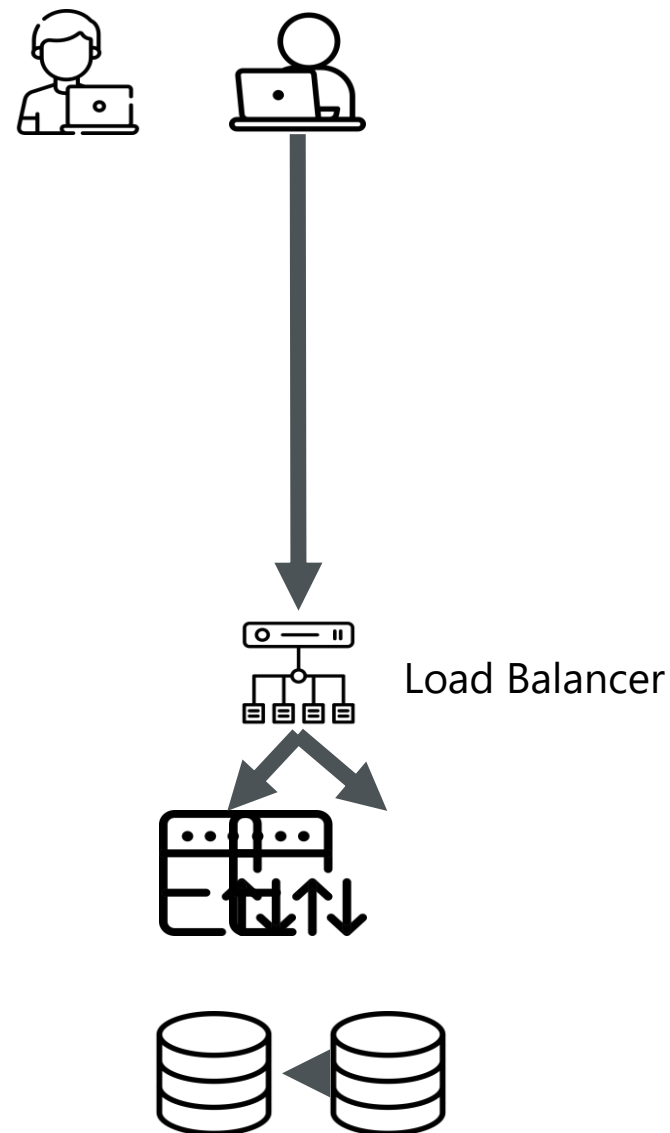
Какая нужна репликация СУБД синхронная или асинхронная?



## Web сервер также нужно задублировать для отказоустойчивости и балансировать нагрузку балансировщиком

Какой режим работы у приложения? 24x7, 5x8, 7x12?

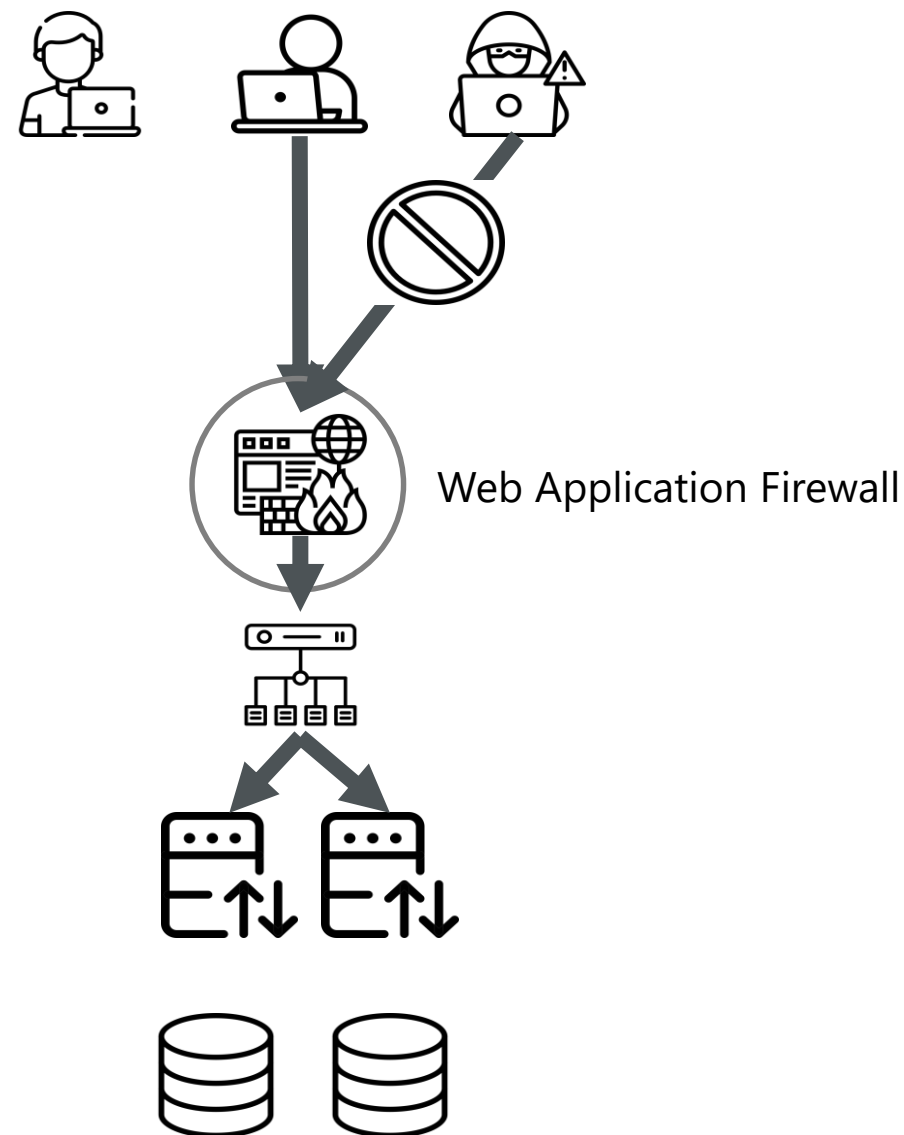
Какое окно обслуживания у приложения? ПН-ПТ 18:00-20:00, ПН-ПТ 20:00-22:00, СБ-ВС 00:00-24:00, любое



# Для защиты приложения от взлома понадобится Web Application Firewall

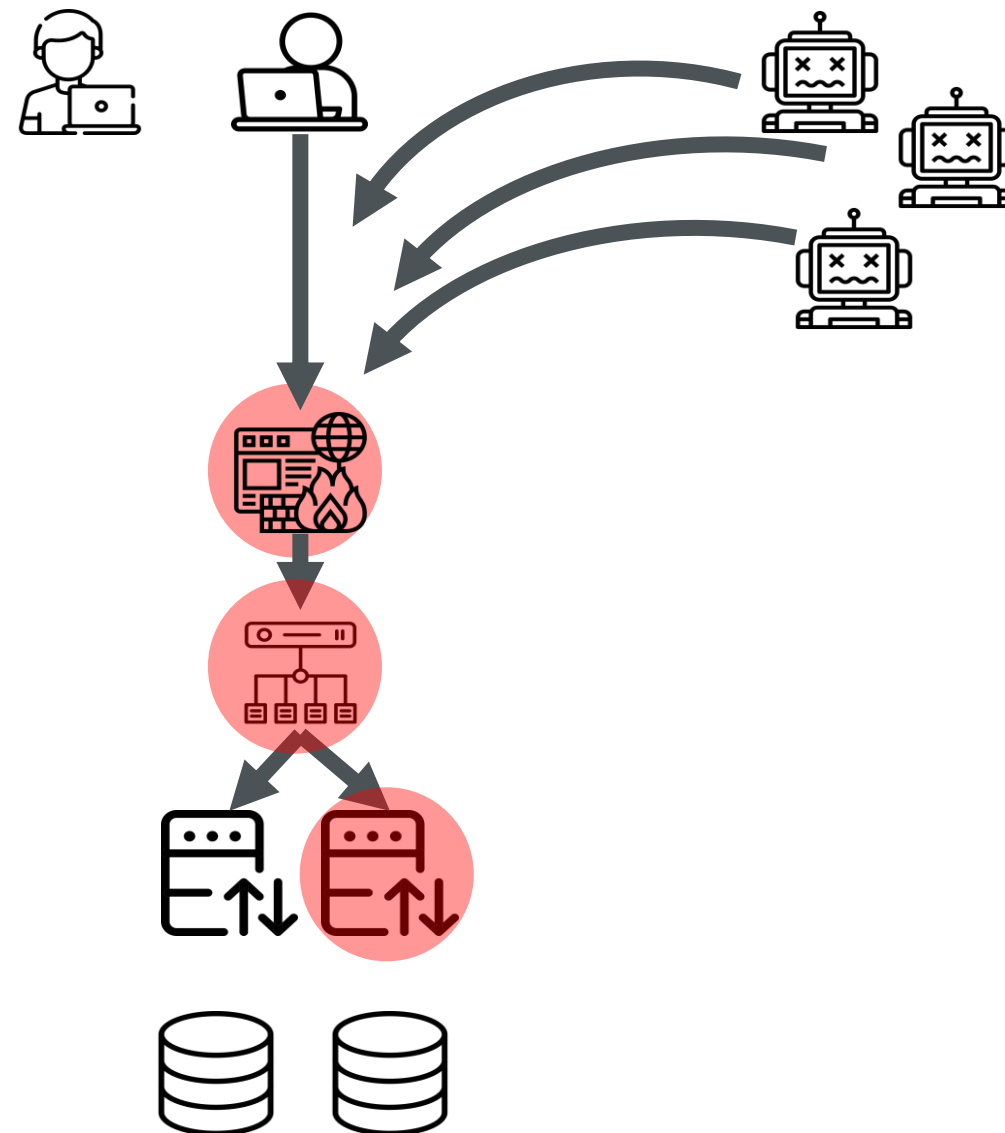
**Цель:** защита от взлома приложения

**Особенность:** на WAF необходимо терминировать TLS трафик



## Для защиты приложения от DDOS атак понадобится DDOS protection

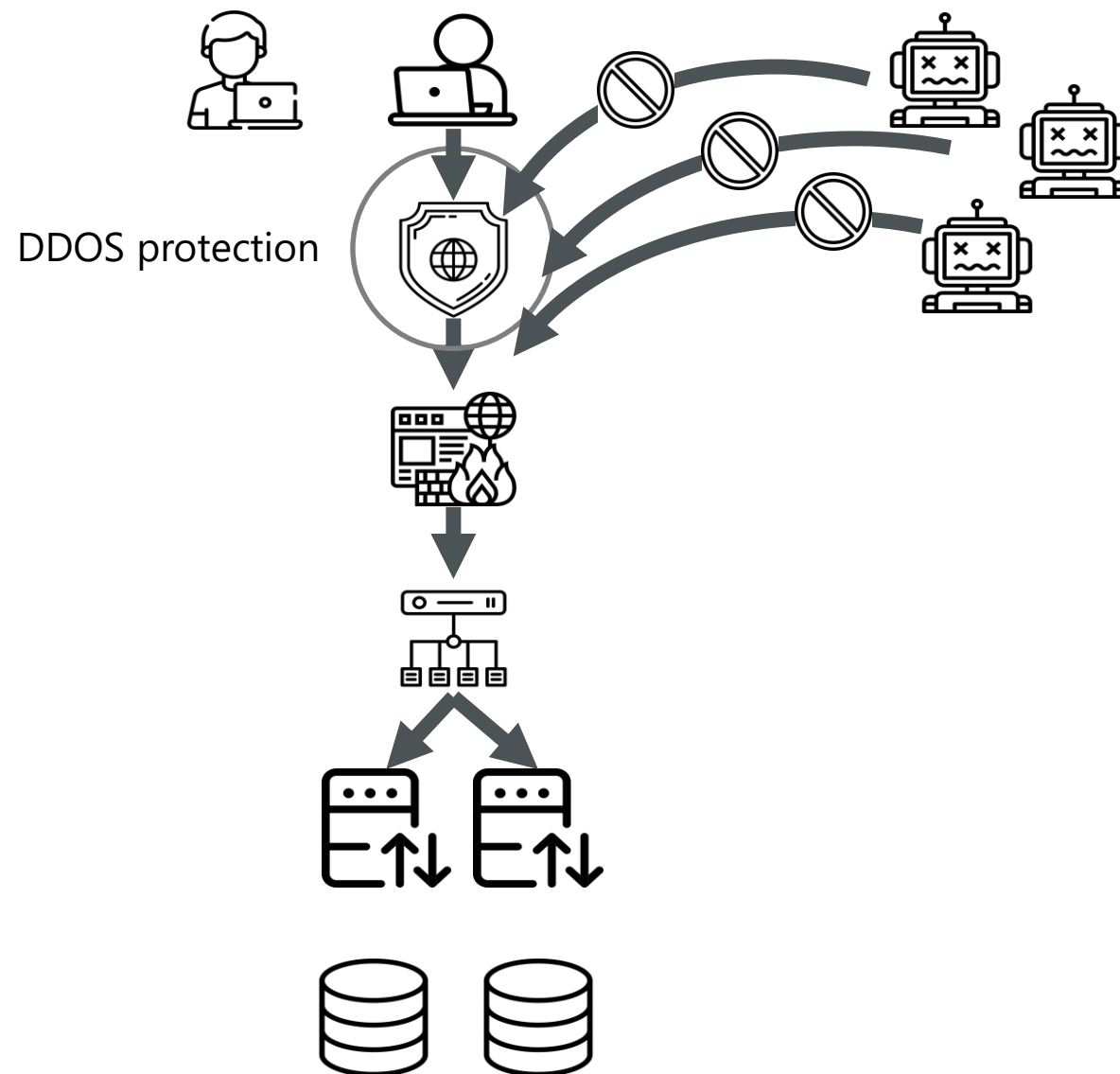
**Риск:** массированная DDOS атака может вызвать перегрузку компонентов приложения и отказ в обслуживании



## Для защиты приложения от DDOS атак понадобится DDOS protection

**Цель:** защита приложения от DDOS атак

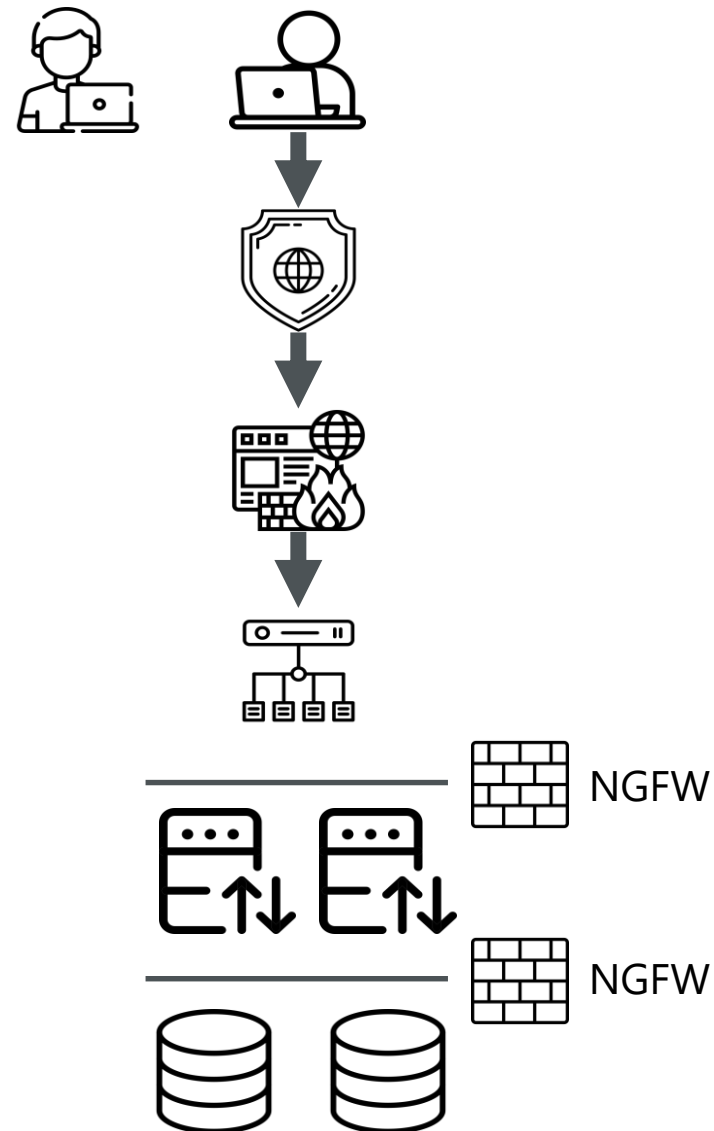
**Особенность:** эффективна только на стороне провайдера связи или как отдельная услуга



# Для защиты сетевого трафика понадобится Next Generation Firewall

**Цель:** ограничить площадь атаки на компоненты системы

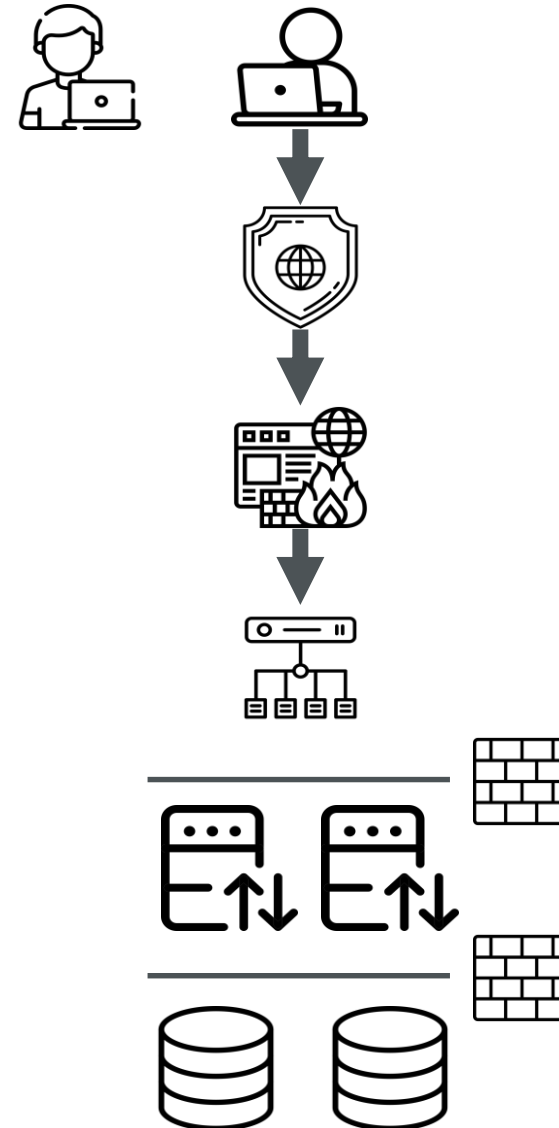
**Особенность:** необходимо понимать по каким портам и протоколам компоненты приложения общаются между собой



# Для аутентификации пользователей необходим Identity Provider. Желательно с поддержкой 2х факторной аутентификации

**Цель:** централизованный процесс по аутентификации пользователей

**Особенность:** для систем с персональными данными 2х факторная аутентификация обязательна

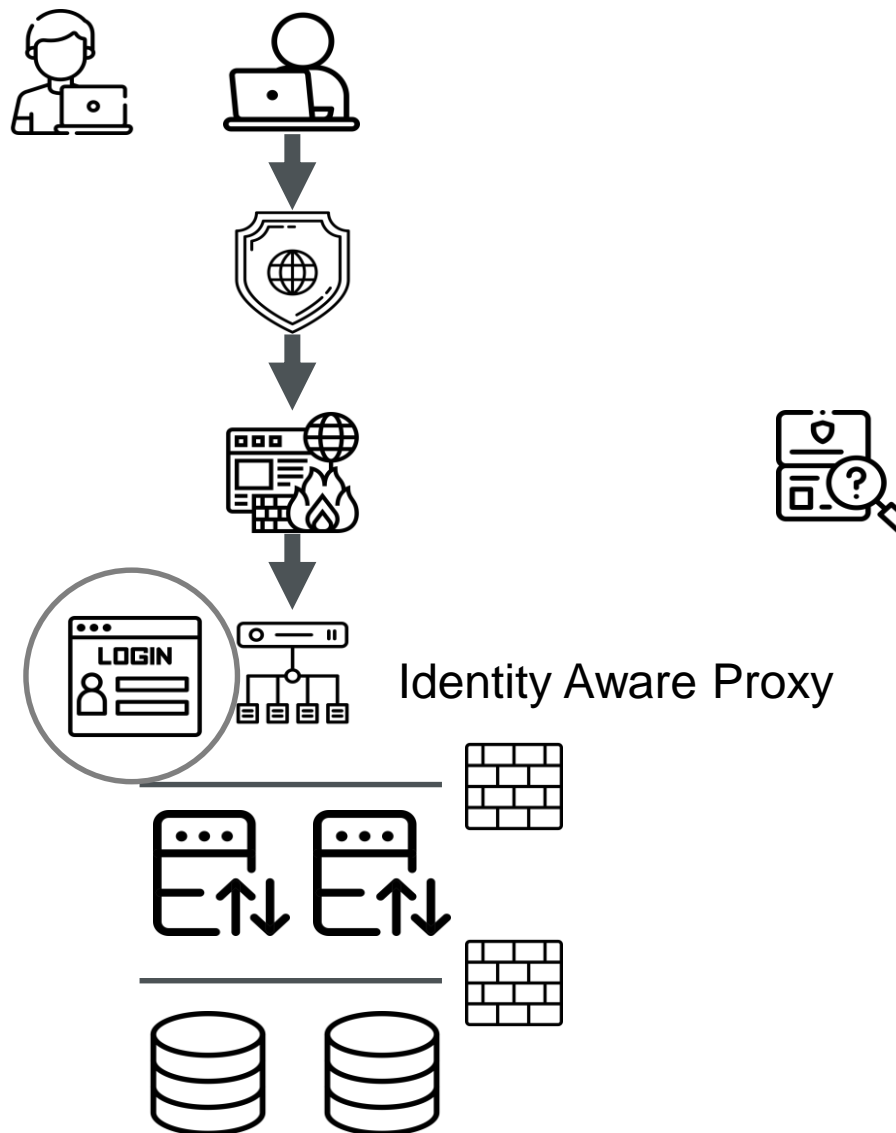


Identity Provider

## В идеале использовать балансировщик или Reverse Proxy с поддержкой Identity Provider (Identity Aware Proxy)

**Цель:** исключить доступ к приложению не аутентифицированных пользователей

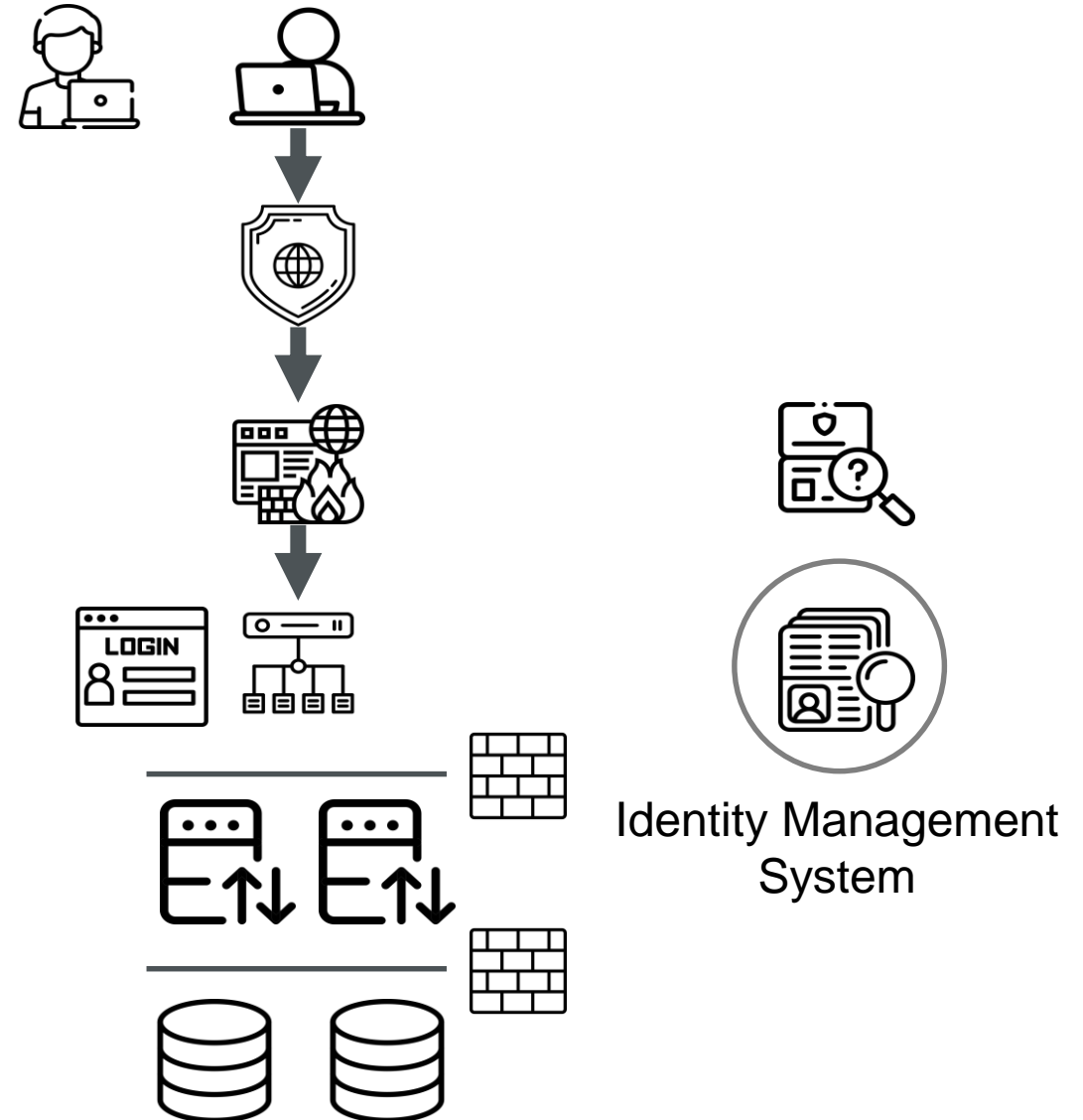
**Особенность:** особенно важно если для внутренних систем не используется WAF



# Учётными записями пользователей и администраторов необходимо управлять с помощью систем Identity Management

**Цель:** как минимум блокировать доступ к системе уволившимся сотрудникам

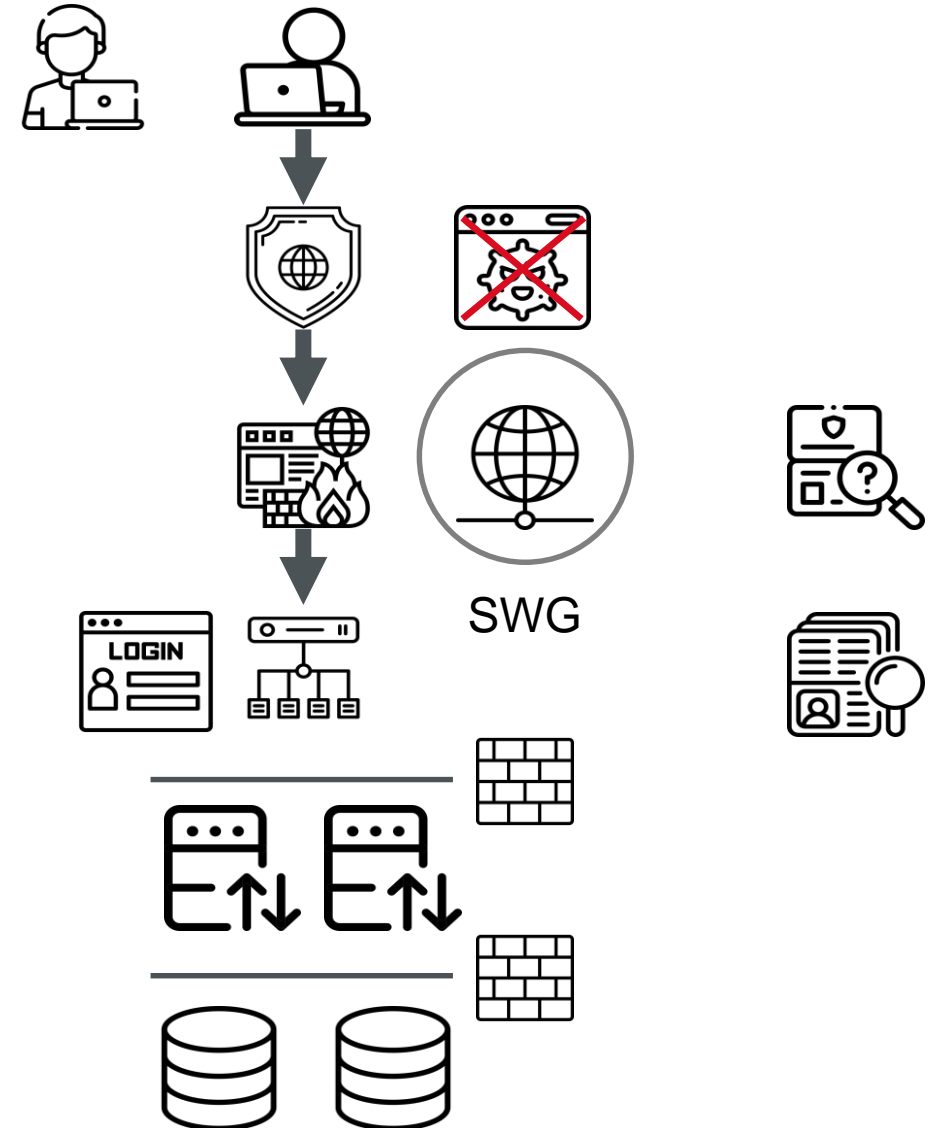
**Особенность:** система работает в связке с identity provider



# Для защиты трафика серверов в интернет понадобится Secure Web Gateway

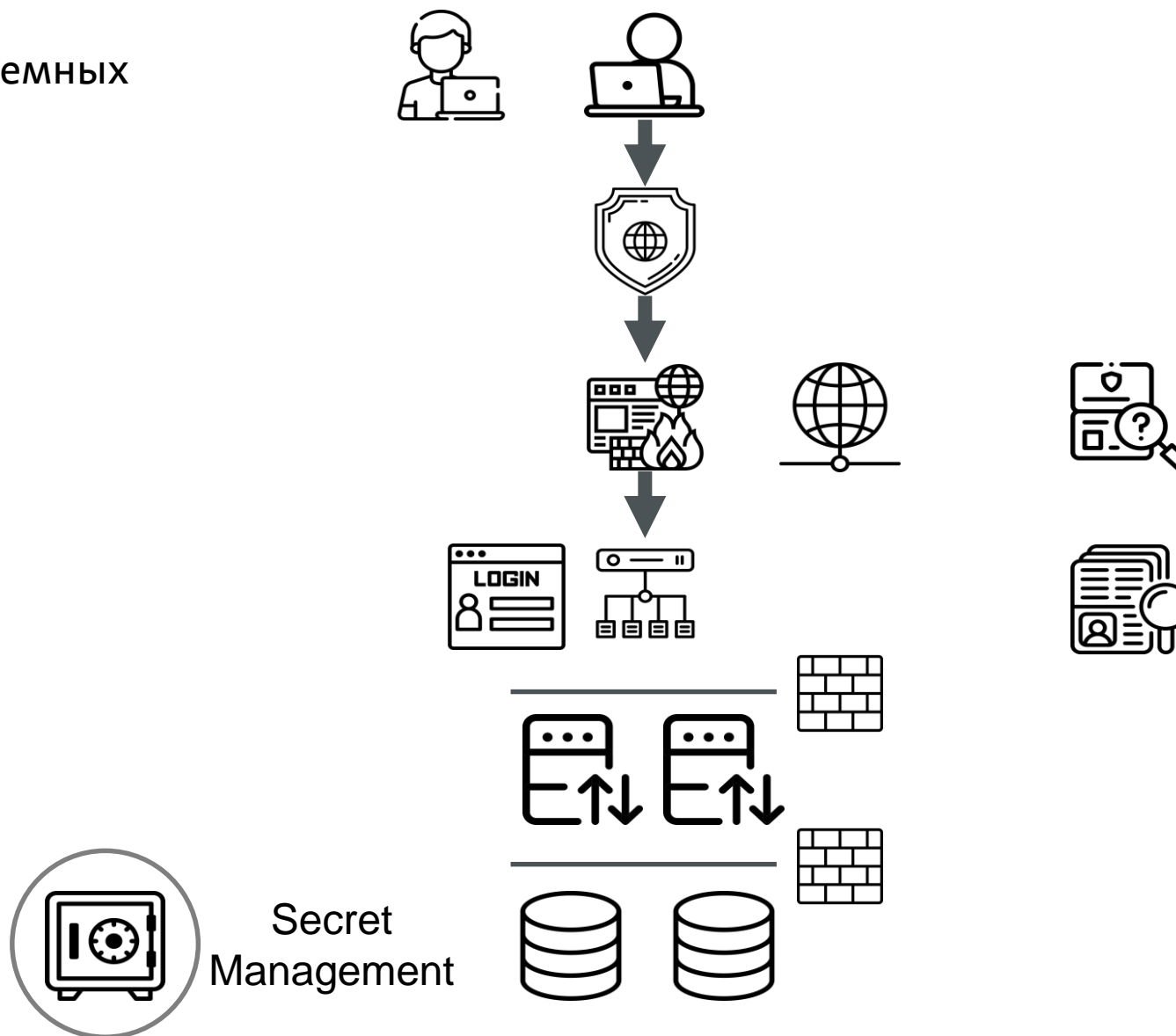
**Цель:** предотвращение доступа с серверов приложения к SnC серверам, а также блокирование эксплуатации уязвимостей вроде Log4Shell

**Особенность:** ограничивает доступ от серверов приложений в интернет только к нужным серверам по белым или репутационным спискам. Необходимо знать к каким сервисам (URL) в интернете приложению нужен доступ



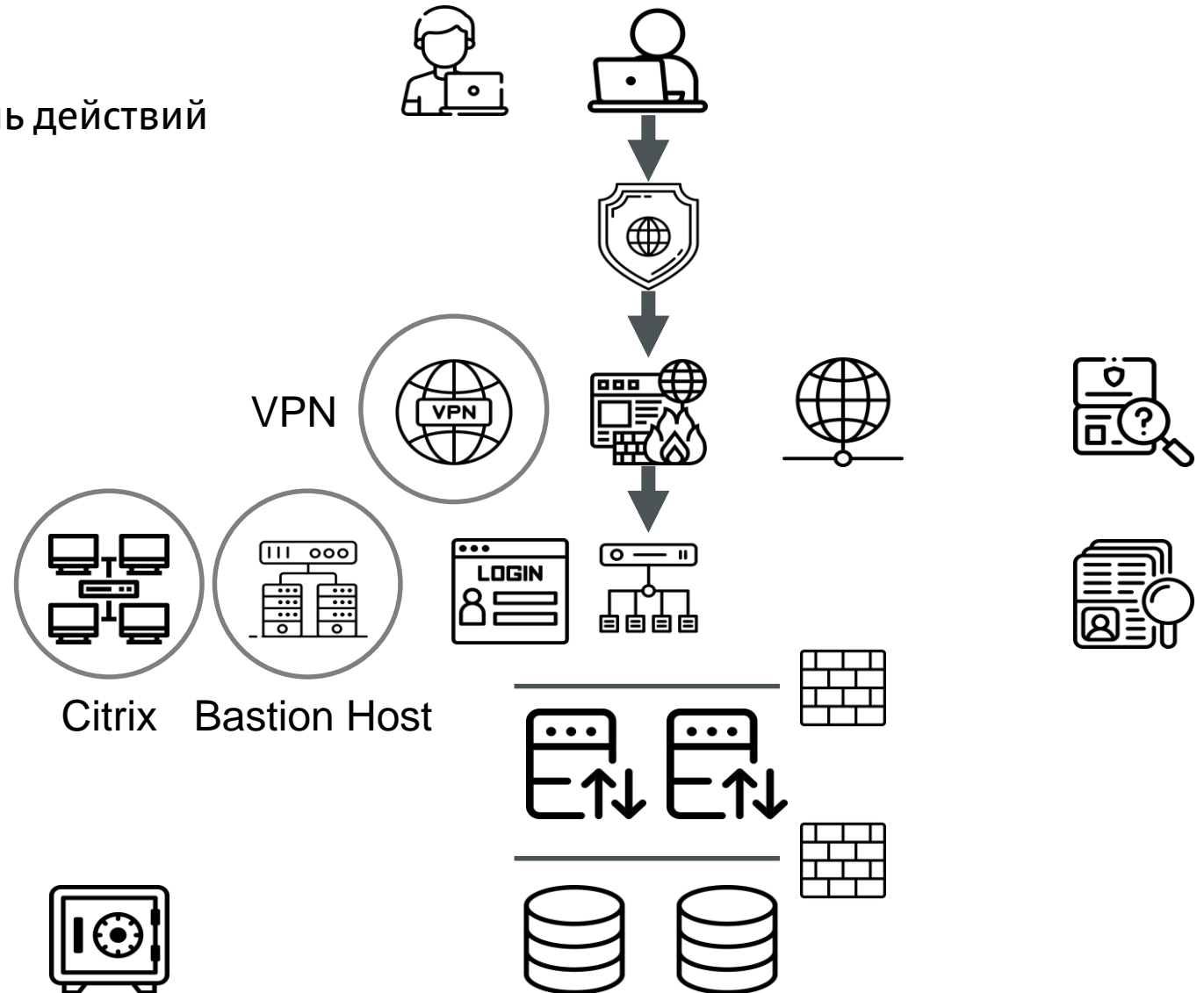
# Для защиты секретов понадобится система хранения секретов

**Цель:** защитить приложение от кражи системных учётки и других важных секретов



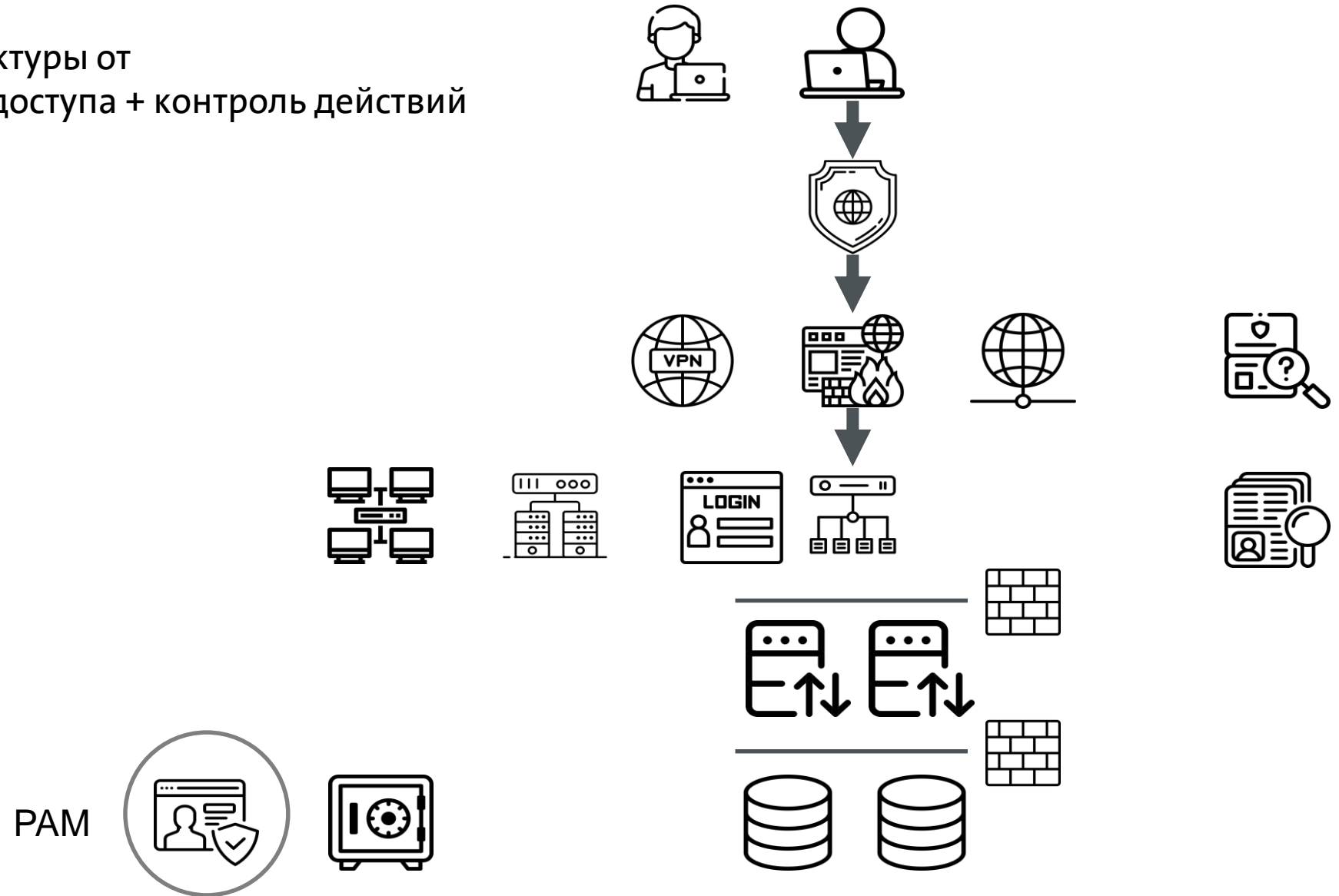
# Для защиты и контроля доступа администраторов к инфраструктуре понадобится VPN + Citrix / Bastion Host

**Цель:** защита инфраструктуры от несанкционированного доступа + контроль действий администраторов



# Привилегированными учётными записями пользователей и администраторов необходимо управлять через системы Privileged Account Management

**Цель:** защита инфраструктуры от несанкционированного доступа + контроль действий администраторов



# Для разрешения имён понадобится DNS сервис

**Цель:** разрешение имени системы

**Особенность:** желательно использовать публичные имена. Проще выписывать TLS сертификаты и решать проблемы с их доверием



# Для защиты Web трафика до приложения понадобится выписывать и управлять TLS сертификатами

**Цель:** защита соединения от пользователя до сервера

**Особенность:** нужен процесс. Кто будет выписывать/заказывать сертификаты, следить за окончанием срока действия и менять их на системах?



TLS Certificates

# Для защиты данных приложения понадобится Backup

**Цель:** защита данных от случайного или намеренного удаления

**Особенность:** для каждого сервера определить:

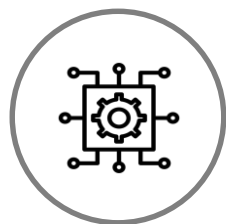
- Что бэкапить? VM целиком или отдельные папки?
- Надо ли бэкапить базу данных?
- Как часто бэкапить? (Weekly, Daily)
- Сколько хранить бэкапы?



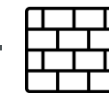
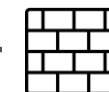
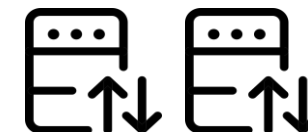
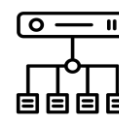
# Для оперативного реагирования на сбои в системе понадобится Мониторинг

**Цель:** сократить время реакции на инциденты

**Особенность:** поддержка приложения должна напрямую получать сообщения мониторинга и начинать на них реагировать



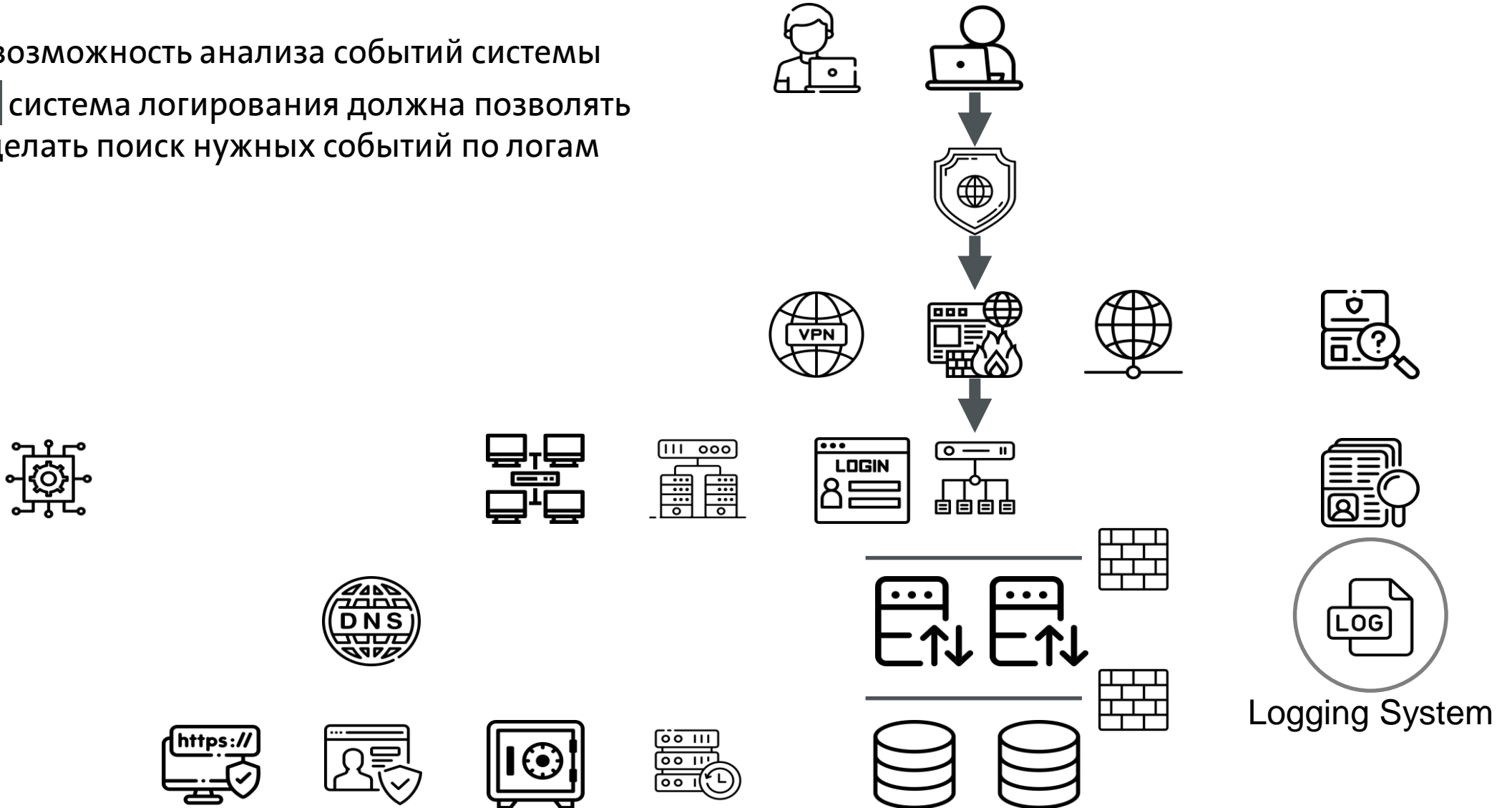
Monitoring



# Для качественной диагностики проблем, мониторинга безопасности и соответствия требованиям понадобится система логирования

**Цель:** иметь возможность анализа событий системы

**Особенность:** система логирования должна позволять оперативно делать поиск нужных событий по логам



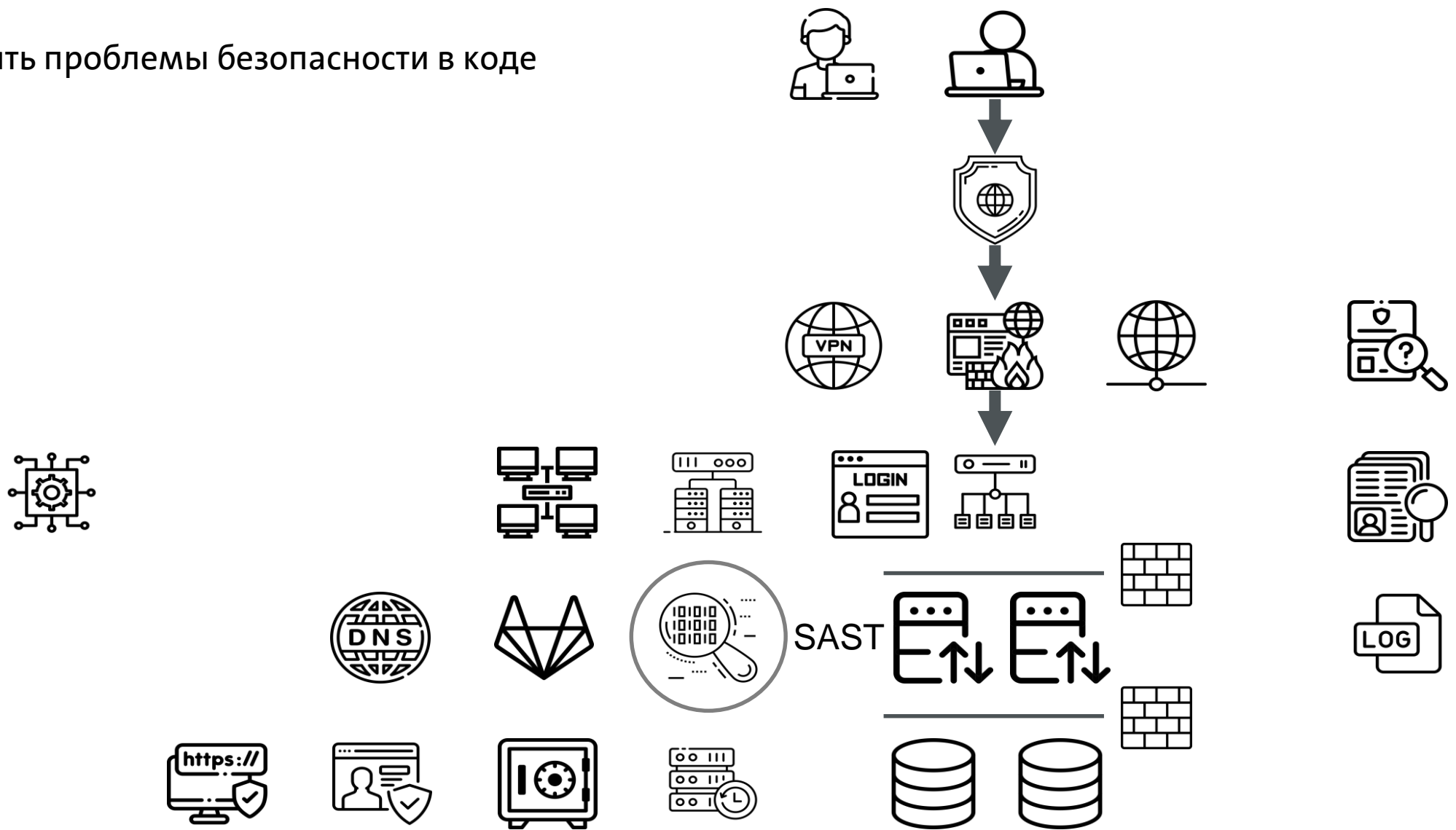
# Код приложения должен принадлежать заказчику и храниться у него

**Цель:** исключить риск потери кода приложения, ускорить развёртывание приложения



# Для проверки безопасности самого кода понадобятся специализированные системы

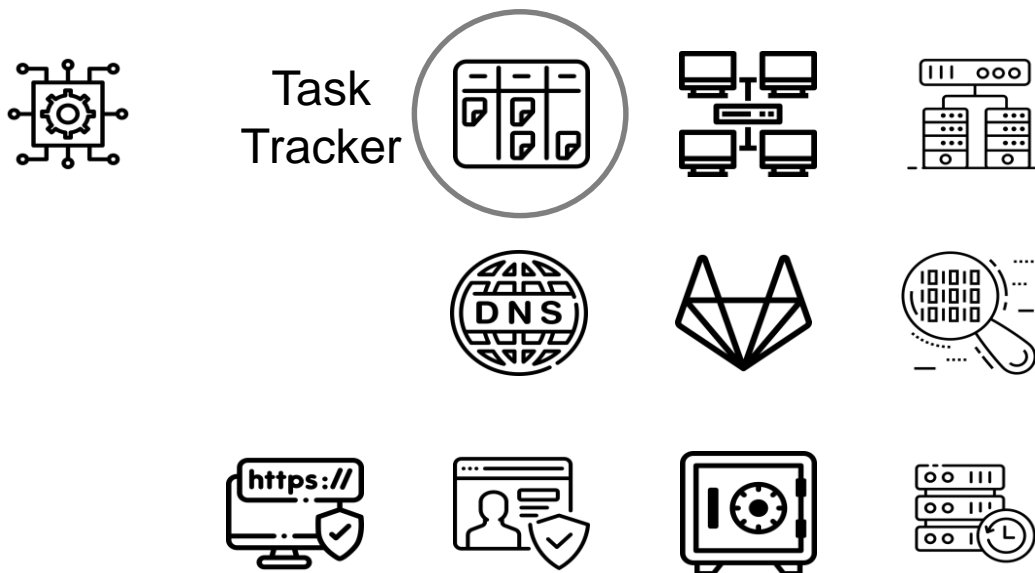
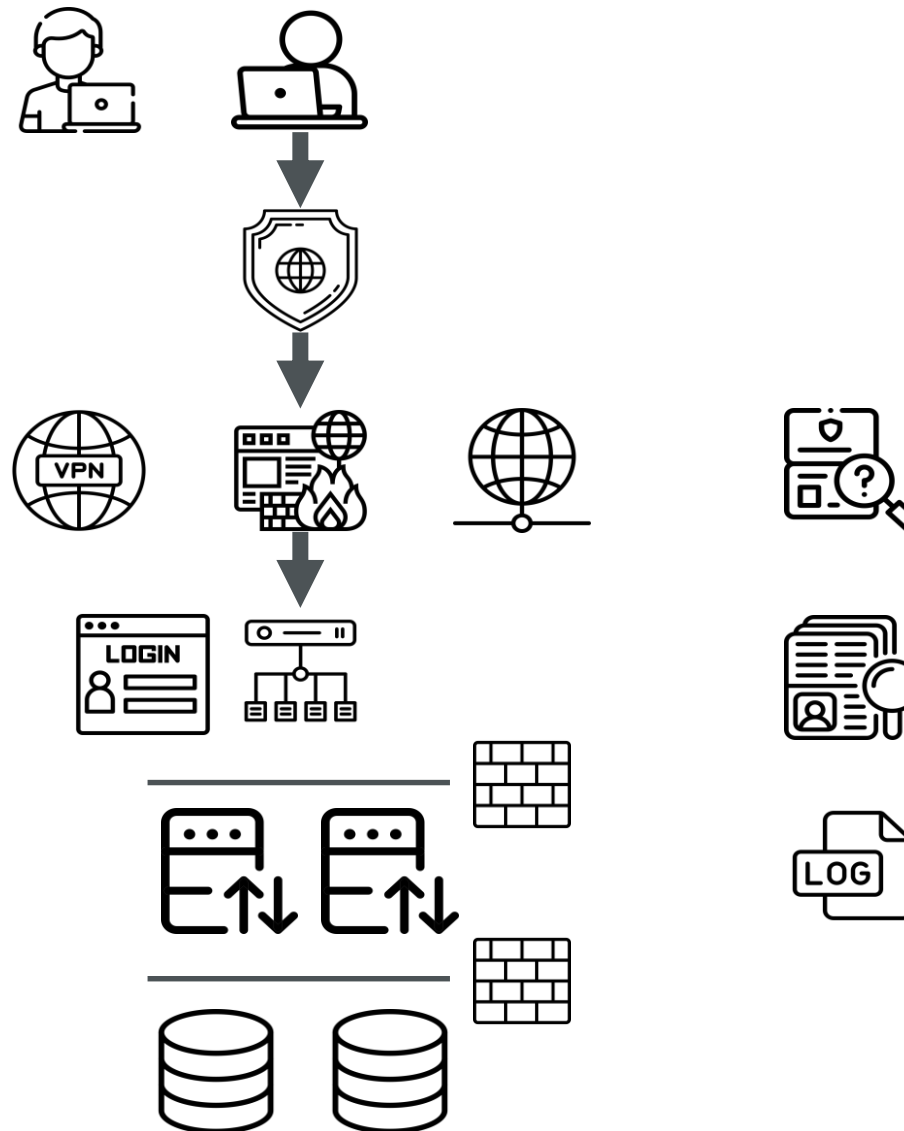
**Цель:** выявлять проблемы безопасности в коде приложения



# Для совместной работы над внедрением приложения / доработками понадобится Task Tracker

**Цель:** оптимизировать совместную работу над задачами в проекте

**Особенность:** большое количество Task Tracker'ов у разных подрядчиков может создать множество проблем



# Для организации процесса поддержки приложения понадобится ITSM Tool

**Цель:** поддержка пользователей в соответствии с SLA

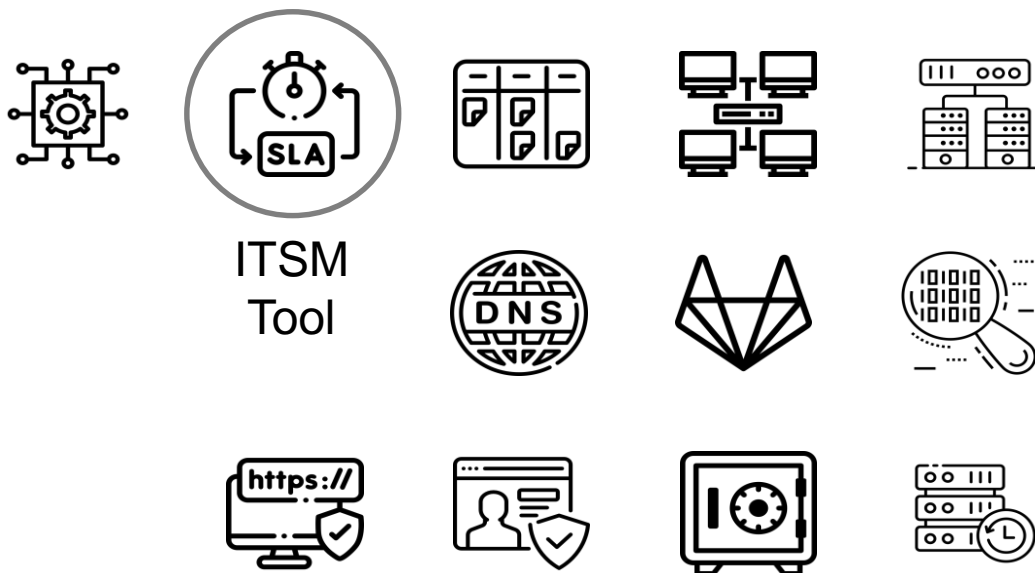
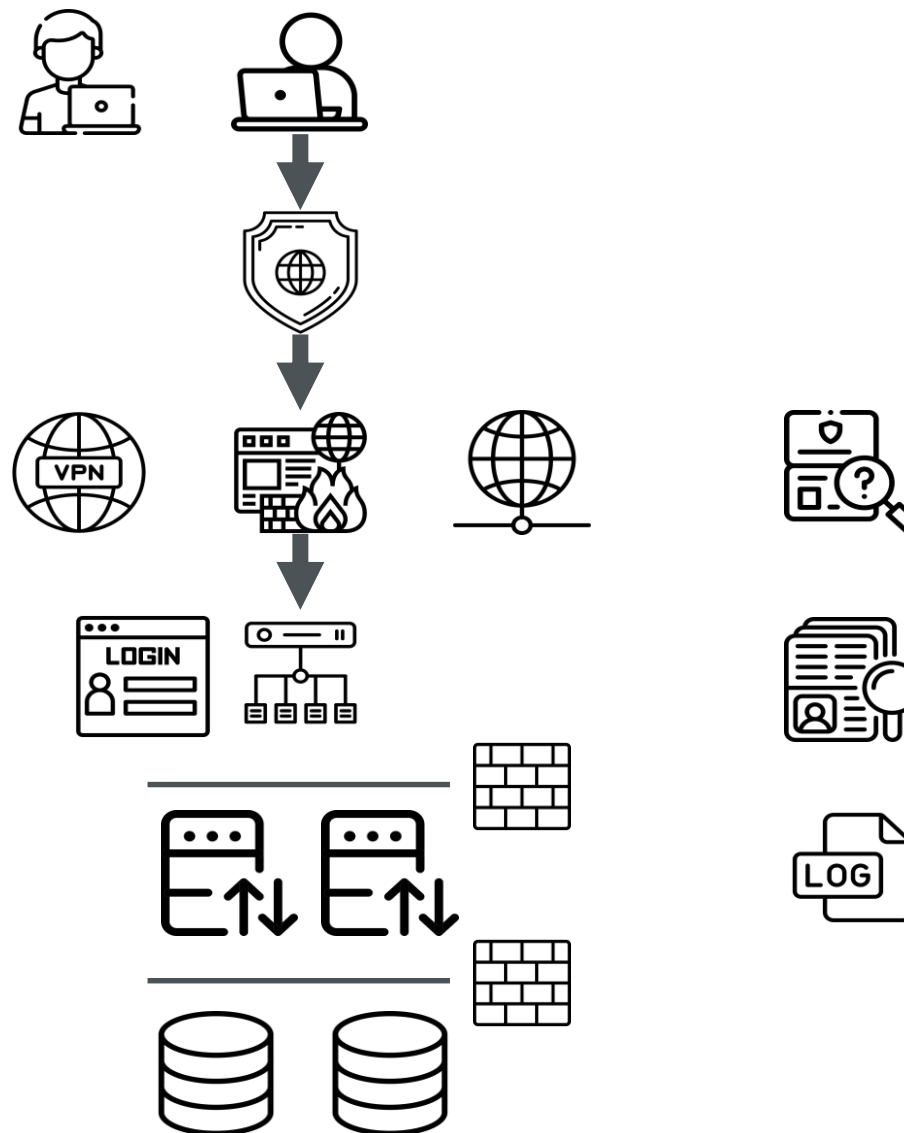
**Особенность:**

1st line – регистрация обращений, диагностика

2nd line – решение стандартных вопросов

3rd line – разработчик приложения

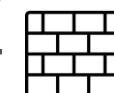
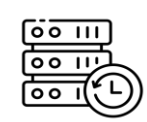
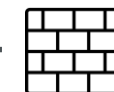
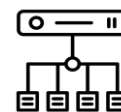
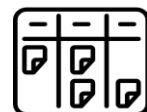
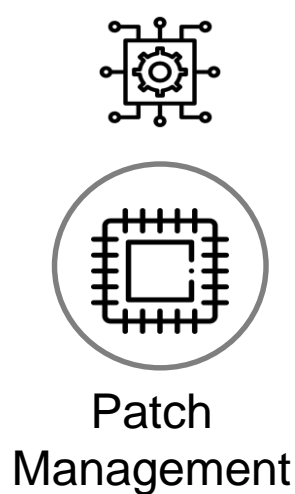
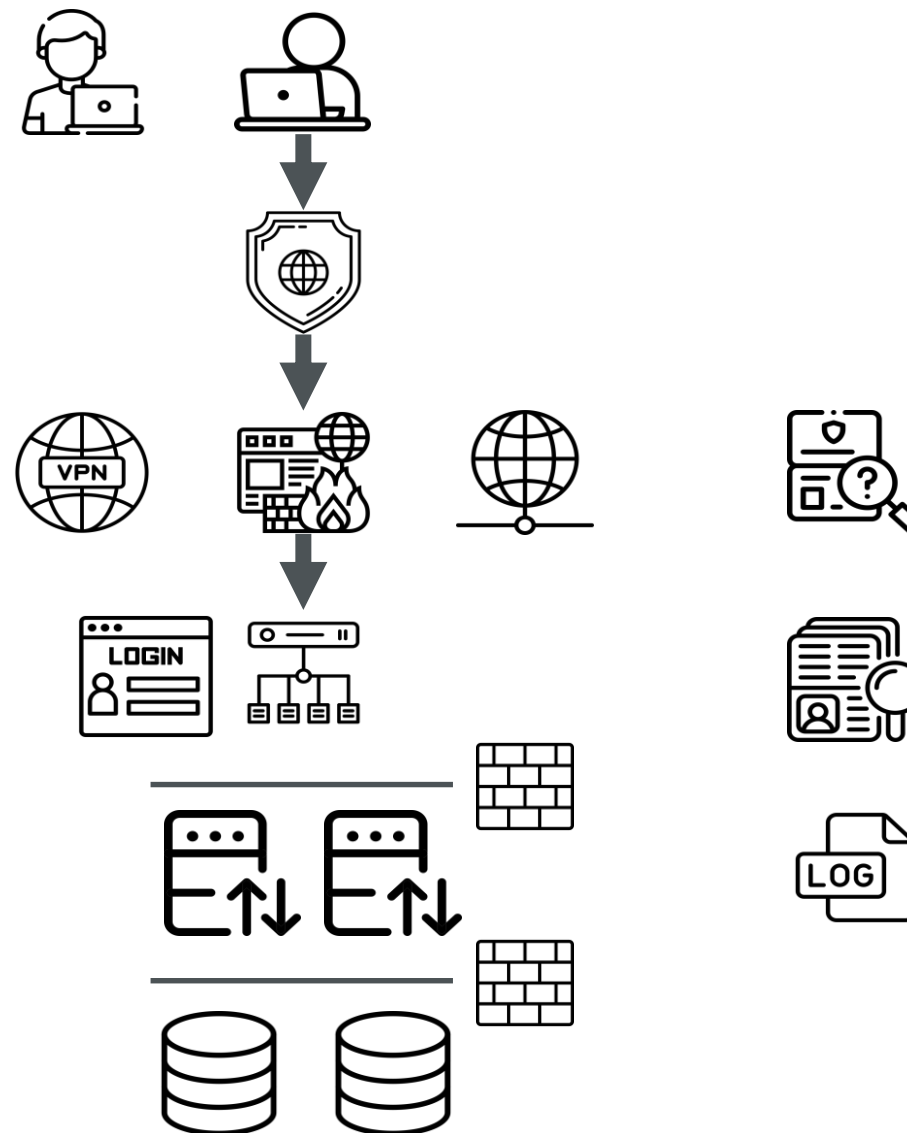
Первые две линии поддержки желательно оказывать самостоятельно



# Для устранения уязвимостей в компонентах приложения понадобится процесс управления исправлениями (Patch Management)

**Цель:** оперативно устанавливать исправления безопасности

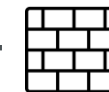
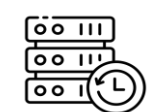
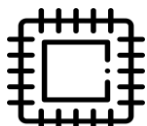
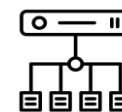
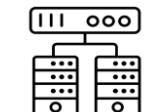
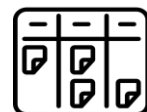
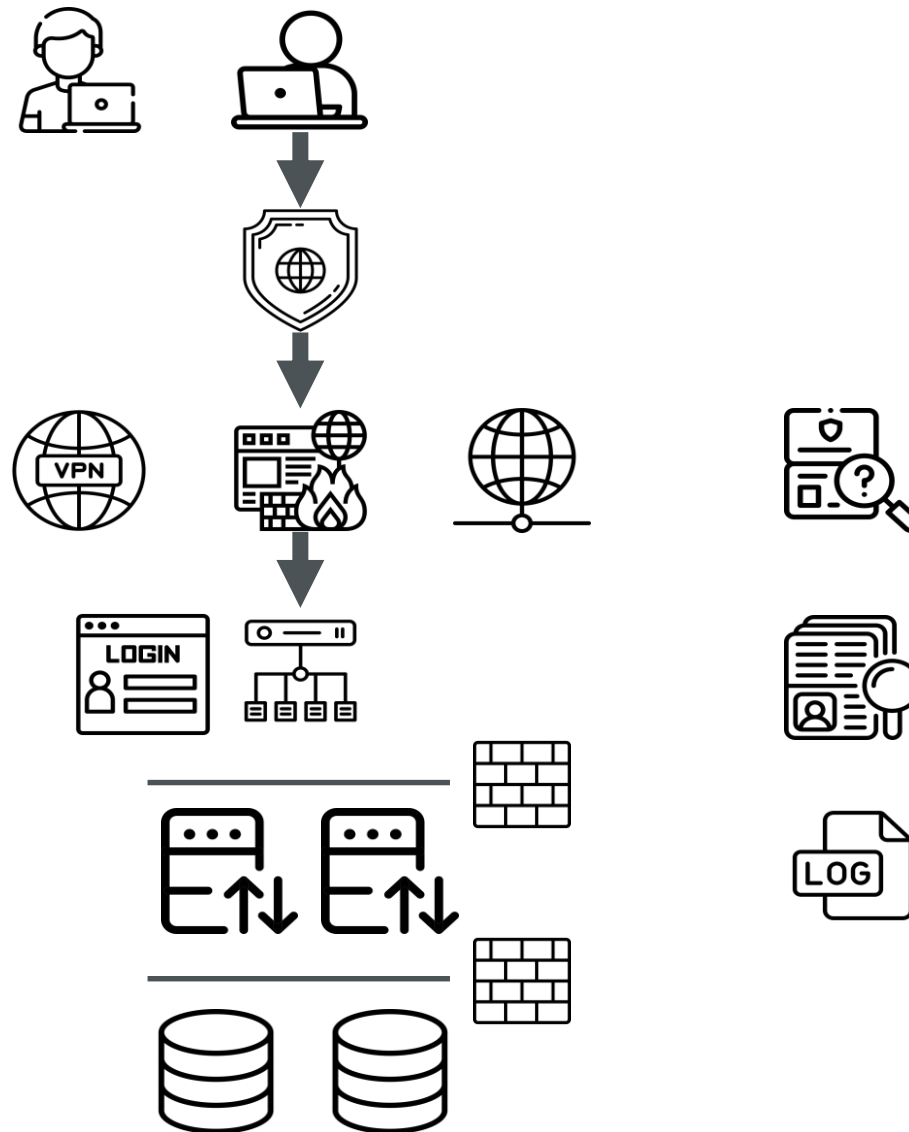
**Особенность:** важно понять за обновление каких компонентов отвечает разработчик. OS, Web Server, Database Server, и т.д..



# Для оперативного выявления уязвимостей в компонентах приложения понадобится процесс управления уязвимостями (Vulnerability Management)

**Цель:** оперативно обнаруживать уязвимости в стандартных компонентах приложения

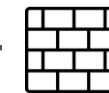
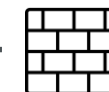
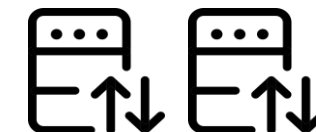
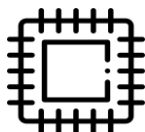
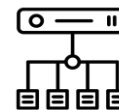
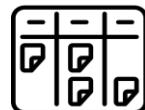
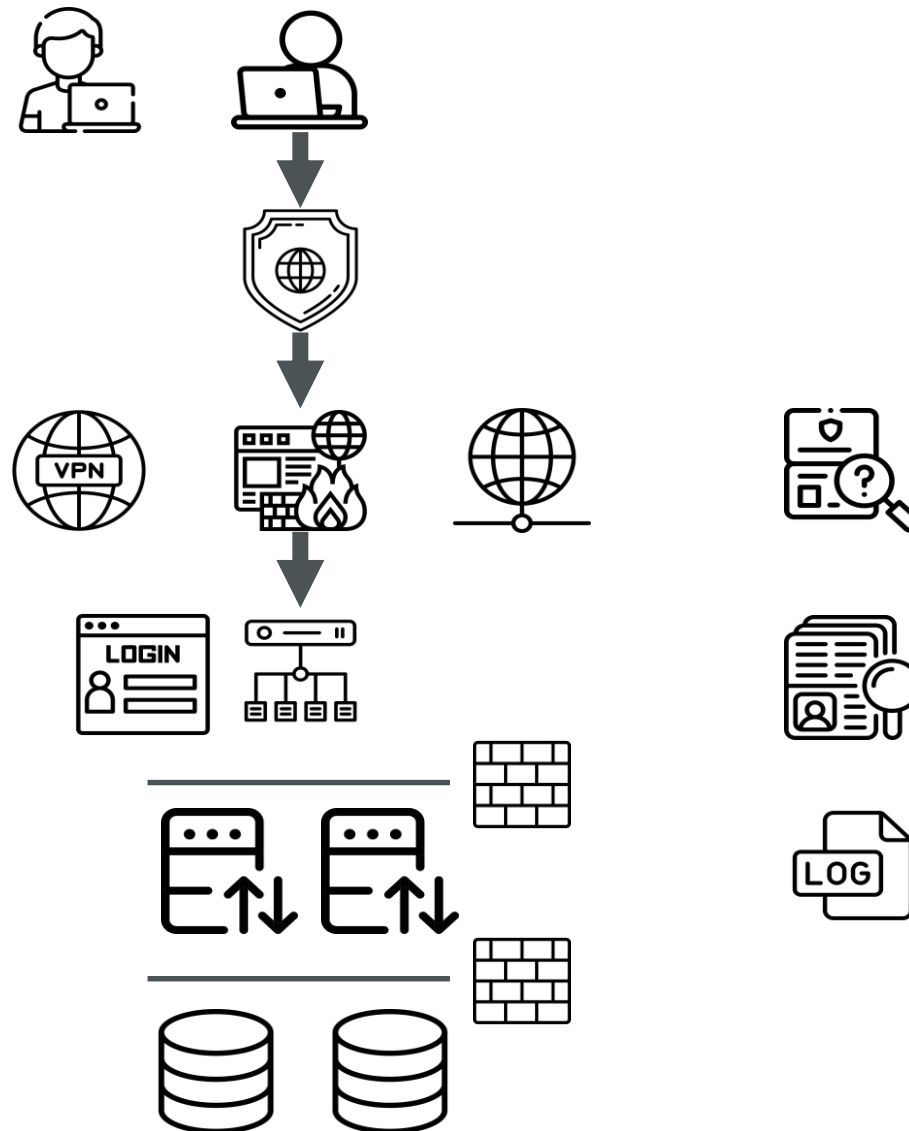
**Особенность:** для полноценного сканирования, сканер должен иметь полный доступ к серверам системы



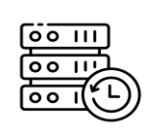
# Для усиления безопасности приложения понадобится применения настроек безопасности к компонентам приложения (Hardening)

**Цель:** повышения безопасности компонент приложения за счёт спец. настроек

**Особенность:** применяется к стандартным компонентам приложения OS, Web Server, Database Server, Component hardening



Hardening

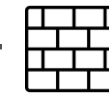
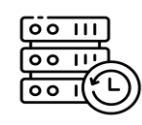
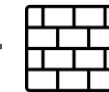
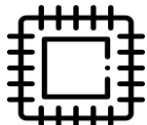
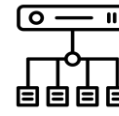
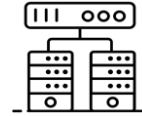
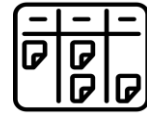
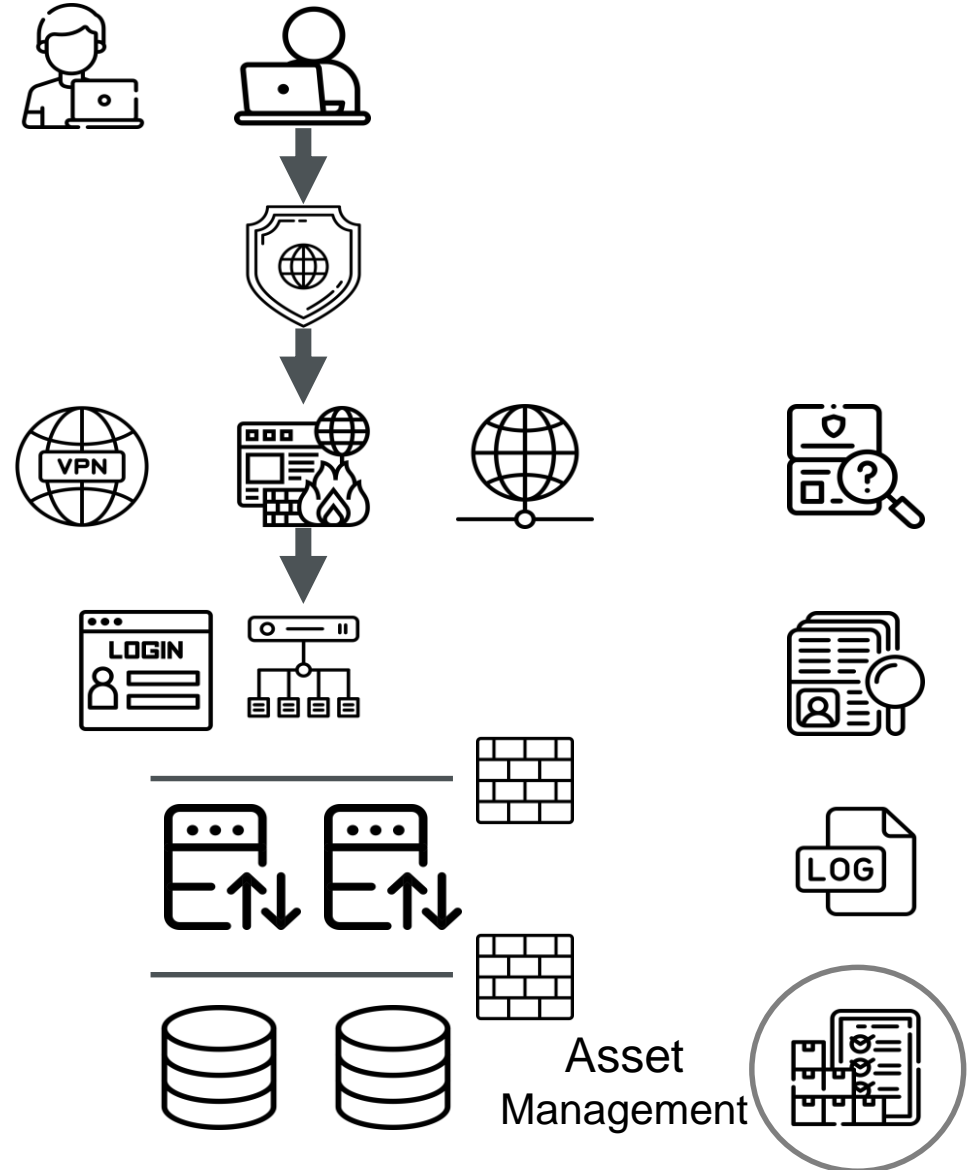


# Для управления жизненным циклом приложения и поддержки других процессов понадобится процесс управления ИТ активами (Asset Management)

**Цель:** управление ИТ активами

**Особенность:** для каждой системы необходимо определить:

- IT Responsible
- Business Owner
- Application Support



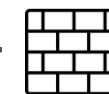
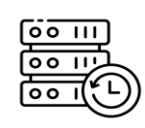
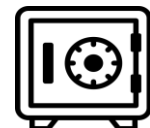
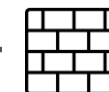
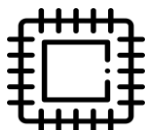
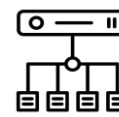
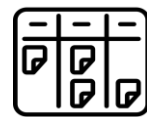
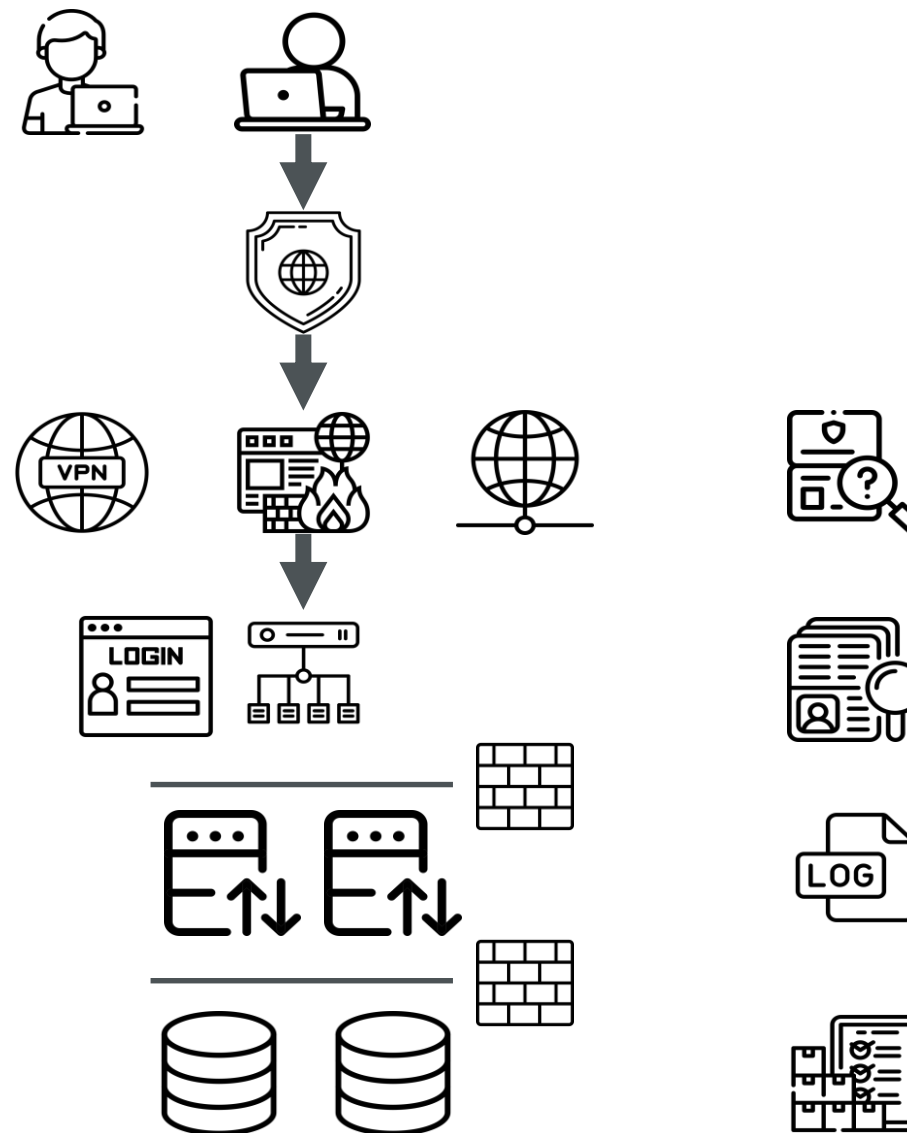
Asset  
Management



# За требования к компонентам приложения отвечает заказчик

Что делать с указанными компонентами?

- внедряет и обслуживает разработчик
- внедряет и обслуживает заказчик
- используется IaaS/PaaS/SaaS
- не используется



## Выводы

- ✓ Комплексно прорабатывайте все аспекты безопасности приложения до внедрения
- ✓ Используйте готовые компоненты внутри компании или SaaS
- ✓ В идеале разработчик должен отвечать только за код приложения, его безопасность и 3ю линию поддержки
- ✓ Код должен принадлежать заказчику
- ✓ Будьте готовы сменить разработчика

Андрей Минаев

+79255032728

Andrey.Minaev@vwgroup.ru

www.vwgroup.ru

