



ФИНТЕХ

Потоковое сканирование файлов

применять или сэкономить время



Талипов Руслан

Руководитель центра Киберзащиты

1. Периметр Финтех организации не изолирован

- **Определение современного периметра:**

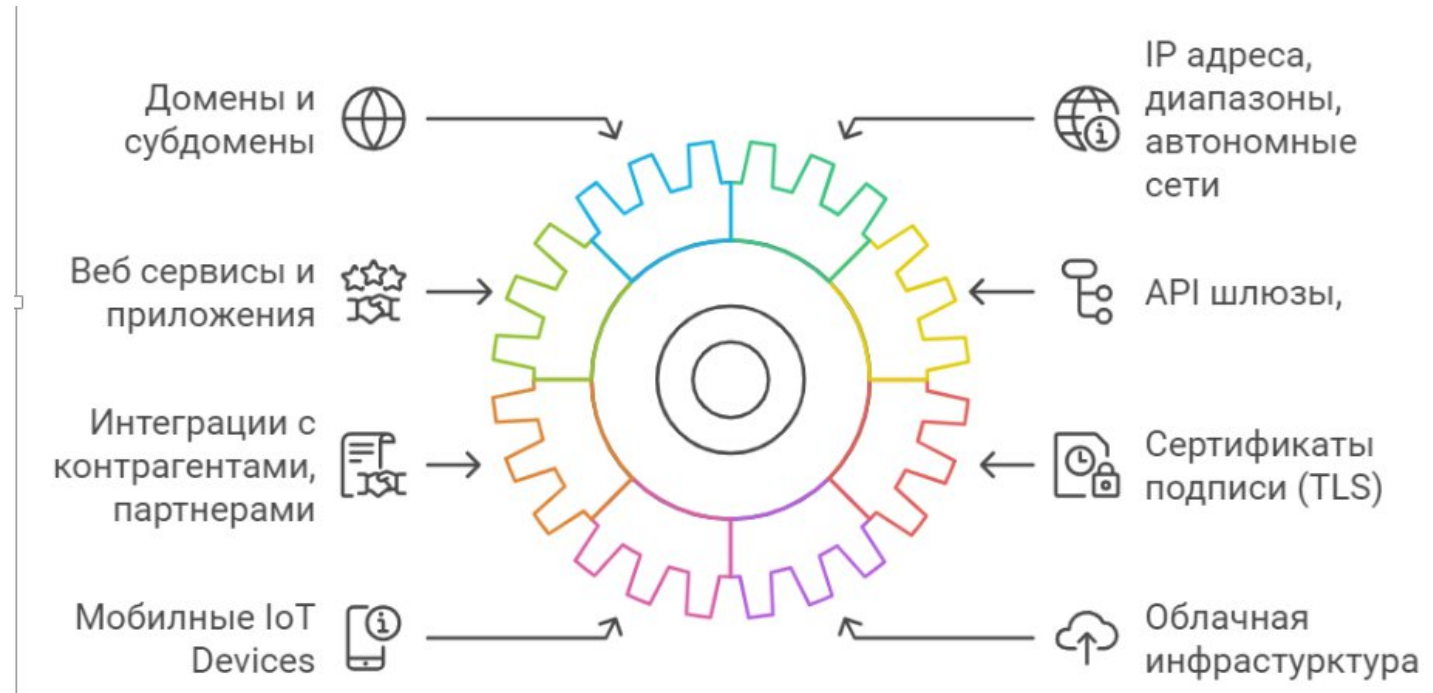
- Расширение границ: облачные сервисы, API, мобильные приложения, устройства IoT и сторонние интеграции.

- **Ключевые проблемы:**

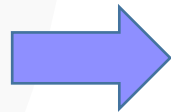
- Постоянно растущая площадь атаки: больше точек входа для потенциальных угроз.
- Сложное управление активами: трудности с отслеживанием и обеспечением безопасности всех активов.

- **Постоянно совершенствующиеся методы атак :**

- RedCanary-Threat-Detection-Report-2024
- IBM-Cost-of-a-Data-Breach-Report
- DNSFilter-Annual-Security-Report



Наиболее известные инциденты



ProxyShell и ProxyLogon в Microsoft Exchange (2021):

Fortinet VPN (2021):

Zero-day уязвимость в Apache Log4j (Log4Shell) (декабрь 2021):

Atlassian Confluence (2021 и 2022):

Citrix ADC и Gateway (2020-2021):

VMware Horizon (2022):

2. Ожидание бизнеса: идеальный мир



1

Синхронность

Ответ сразу в запросе

2

Низкая задержка

Сотни миллисекунд

3

Простота

Без изменения user
experience и
архитектуры

3. Реальность

Время анализа
Не прогнозируемо

Требования к ресурсам
Постоянное
масштабирование

Архитектурные
различия
Не везде единообразие

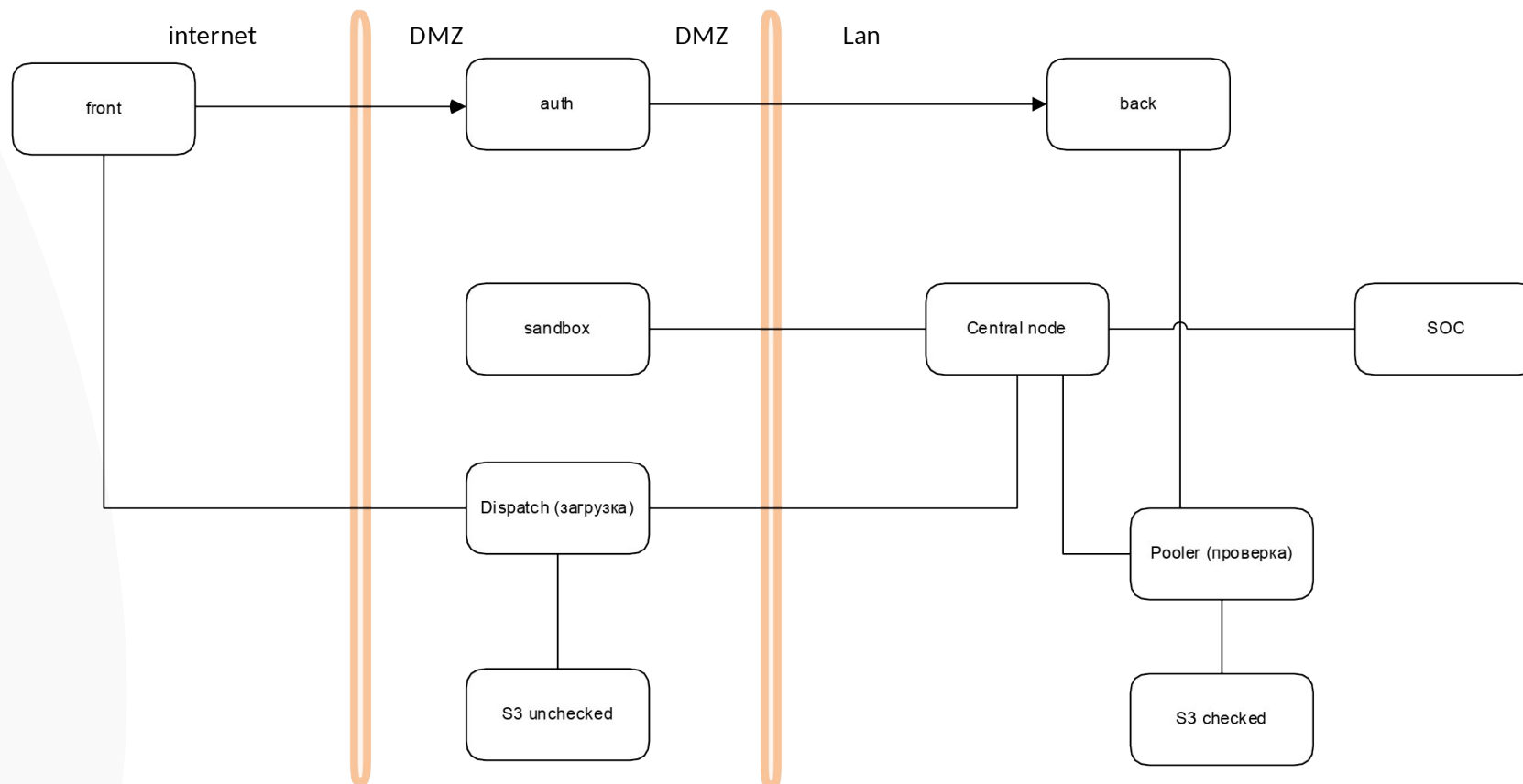
Sandbox evasion
Не серебряная пуля?

Песочница — это инструмент
для **асинхронного, углубленного**
анализа небольшого процента
файлов

4. Асинхронная архитектура

Компонент системы	Функции компонента
Клиент - продуктовые бекенд сервисы	• Передает все входящие файлы • Получает уведомления о завершении проверки. • Прием файлов от сторонних сервисов. • Сохранение копий файлов в хранилище (MinIO) • Передача информации о загруженных файлах в Kafka. • Интеграция с сторонними сервисами посредством API: API должен поддерживать аутентификацию и авторизацию. API должен поддерживать загрузку файлов по частям. API должен поддерживать проверку статуса загрузки. При массовой отправке файлов на сервер через API, проверка должна осуществляться по принципу first in, first out (первым пришел – первым обслужен) • Хранилище для загруженных файлов. • S3-совместимый интерфейс. • Обмен сообщениями для передачи данных между компонентами.
Сервис загрузки файлов	
MinIO	
Kafka	
Сервис перекладчика	• Скачивает файлы из MinIO. • Загружает файлы в песочницу (в KATA по ICAP). • Запрашивает результат проверки у центральной ноды. • Отправляет информацию о зараженных файлах в SOAR. • Отправляет уведомления о завершении проверки. • После проверки файлов на наличие IOC чистые файлы должны быть перемещены в новую папку в MinIO. Эта папка должна быть доступна только для бэкэнд-сервисов.
Центральная нода	• Быстрая проверка по хэшам • Проверка файлов на наличие IOC. • Отправляет результат проверки сервису перекладчика. • Изолированная среда для проверки файлов.
Песочница	

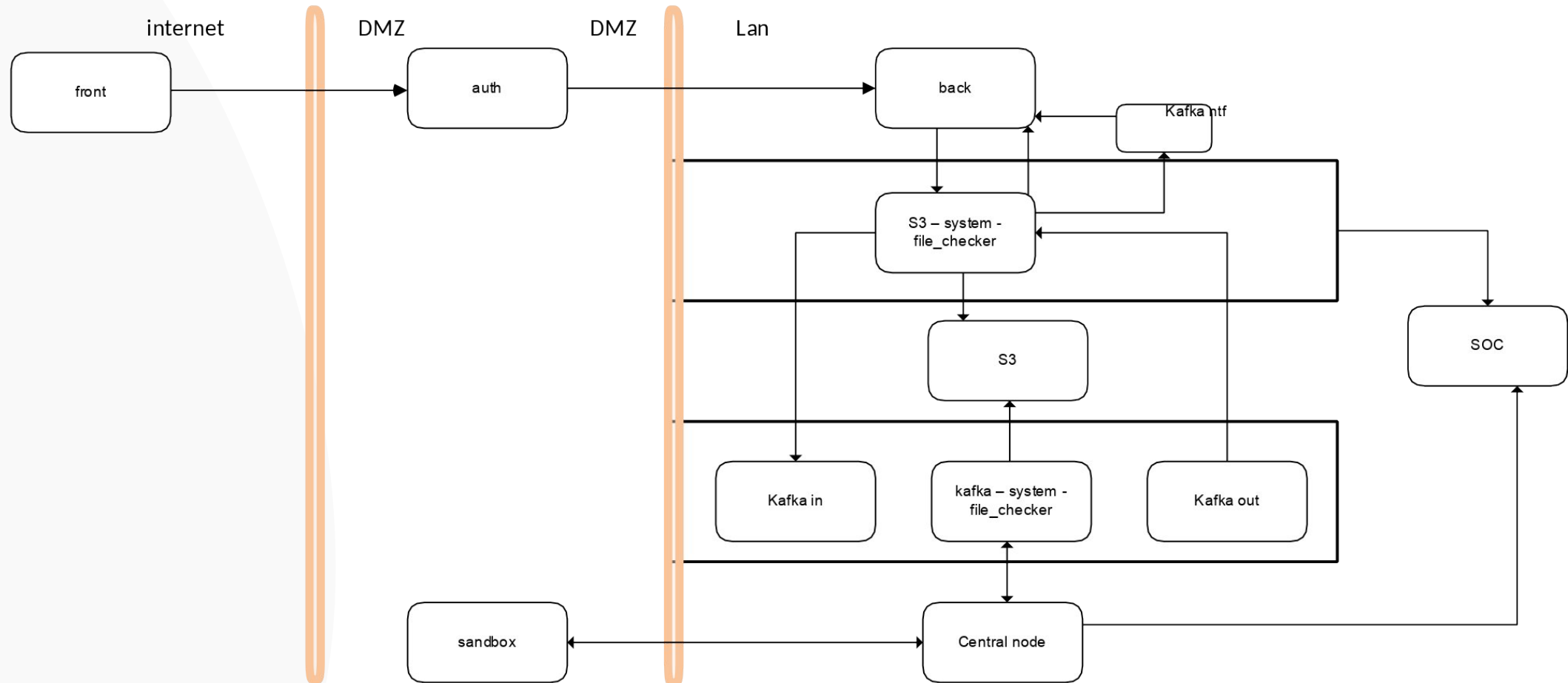
4. Архитектура 1й вариант



До выполнения проверки
файл недоступен

Зараженный файл все равно
попадает в инфраструктуру

4. Архитектура 2й вариант



Асинхронные проверки

Отдельные топики под
каждую систему

6. Выводы

Асинхронная проверка

Не опция а необходимость

Управляемость

Основа – метрики и логирование

Архитектура

API снижает как риски так и затраты

Безопасность

Не только ИБ но и бизнес логика и инфраструктура

Не бывает серебряной пули

Строим процесс, который подойдет именно вашей компании