Урок 8. Протокол Oauth 2.0

Найдите ресурс, на котором реализован протокол OAuth 2.0. Изучите реализацию протокола, ответьте на вопросы:

Вопрос 1:

Какая версия протокола используется (OAuth 1.0, OAuth 2.0 Implicit Grant, OAuth 2.0 Authorization Code Grant, …, или что-то свое)? Кто является клиентом, а кто провайдером?

Ответ:

Для исследования я выбрал avito.ru. Больше похоже на версию протокола **Oauth 2.0 Authorization Code Grant**, хотя можно отнести и к «что-то своё». Клиентом является avito, а провайдером vk.com, т. к. авторизацию я делал через vk.

Вопрос 2:

Опишите каждый шаг протокола. Приложите соответствующие URL и ответы от сервера (чувствительные данные необходимо скрыть).

Ответ:

Первый GET запрос был к **oauth.vk.com** со следующим URI:

/authorize?**client_id**=7777777&display=popup&**redirect_uri**=https%3A%2F%2Fwww.avito.ru%2Fsocial%2Flogin%3Fprovider%3D1&response_type=code&**scope**=email+offline&**state**=667164ef-6292-4702-9c3e-3a1862be36f6

Ответ от сервера: 302 редирект с Location

Location: https://login.vk.com/?act=grant_access&client_id=7777777&settings=4777770&response_type=code&group_ids=&token_type=0&v=&display=popup&ip_h=d0e461a3d1b4066de2&hash=1677777738_077777cdb7fbf1aa77&https=1&state=667164ef-6292-4702-9c3e-3a1862be36f6&redirect_uri=https%3A%2F%2Fwww.avito.ru%2Fsocial%2Flogin%3Fprovider%3D1

Второй GET запрос был по Location из ответа предыдущего запроса

В ответ получили ещё один 302 редирект с Location

Location: https://www.avito.ru/social/login?provider=1&**code=a7777a777aaa7aa7aa**&state=667164ef-6292-4702-9c3e-3a1862be36f6

Третий GET запрос был по Location из ответа предыдущего запроса

В ответ получили… **TOKEN**? Честно говоря хз что я там получил. Так то в ответе html страница, но в ней есть какой-то токен заURL’энкоженый. Вот такой (не менял):

```
%7B%22%40avito-core%2Ftoken%3A2.0.30%22%3A%7B%22name%22%3A%22token
%5B3724078433521%5D%22%2C%22value%22%3A
%221646009238.2400.4baadf8d0fbef149cf311c7779917c333c3650fab3d378d0eb978010d8b120ec
%22%7D%7D
```

Судя по схеме Authorization Code Grant токен не должен быть у нас. Если это токен авторизации, то я бы мог предположить, что у Avito небезопасная реализация протокола. Хотя какой ещё токен там может быть… Мало того, что этот токен у нас, так он ещё особо и не зашифрован.
Вот его расшифровка:

```
{"@avito-core/token:2.0.30":
{"name":"token[3724078433521]","value":"1646009238.2400.4baadf8d0fbef149cf311c7779917c333c36
50fab3d378d0eb978010d8b120ec"}}
```

И последнее. Заинтересовал ещё один запрос. На этот раз POST

```
POST /web/1/social/login HTTP/2
Host: www.avito.ru
```

в данных которого было это (code заменил):

```
{"state":"667164ef-6292-4702-9c3e-3a1862be36f6","providerId":1,"code":"a7777a777aaa7aa7aa"}
```

А в ответ получил JSON со своей аватаркой в нескольких разрешениях, Имя, Фамилию, hashedUserId, socialUserId, registrationTime.

Вопрос 3:

Найдите уязвимости и слабые места в реализации протокола OAuth 2.0. Если уязвимостей и слабых мест нет, объясните, какие механизмы защиты применили разработчики, чтобы избежать каждой из пройденных нами уязвимостей OAuth 2.0.

Ответ:

Мною был проверен метод подмены redirect_uri. Не прошло. Написали типа «некорректный redirect_uri, проверьте его». Ну это вк, они проверяют redirect_uri на стороне сервера.

Если я нашёл именно тот токен, то он не безопасен. Во первых он не зашифрован, во вторых он вообще у нас, хотя он должен быть на

стороне сервиса-клиента. Ну соответственно небезопасная реализация протокола. client_secret вроде нигде не видел, хотя он мог бы и спрятаться в куках, наверное...