УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет математики, информационных и авиационных технологий

НАУЧНО-ИССЛЕДОВАТЕЛЬСКАЯ РАБОТА

НА ТЕМУ: «РАЗРАБОТКА МОДУЛЯ АУТЕНТИФИКАЦИИ ВЕБ-РЕСУРСОВ ПРЕДПРИЯТИЯ»»

Выполнил студент 6 курса Е.И. Шабров

Научный руководитель А.М Иванцов

Ульяновск -2022

Цели и задачи работы

- Цель научно-исследовательской работы – разработка модуля аутентификации веб-ресурсов предприятия.

 Задачи, реализующие цель работы:

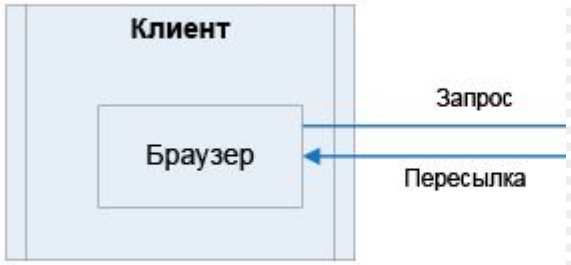
1. Провести анализ угроз веб-ресурсам.
2. Выявить требования к защите веб-ресурсов.
3. Проанализировать способы и криптографические протоколы аутентификации.
4. Выявить основные угрозы безопасности при аутентификации.
5. Разработать и обосновать структуру криптографического модуля веб-приложения предприятия – ее функционал и алгоритм работы.
6. Осуществить программную реализацию криптографического модуля веб-приложения предприятия.

Понятие о веб-ресурсе

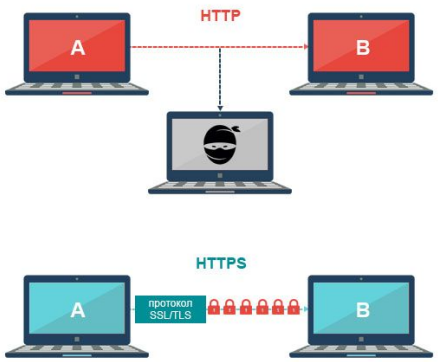
Веб-ресурс это:

Одна или несколько логически связанных между собой веб-страниц; также место расположения контента сервера

IP-адрес



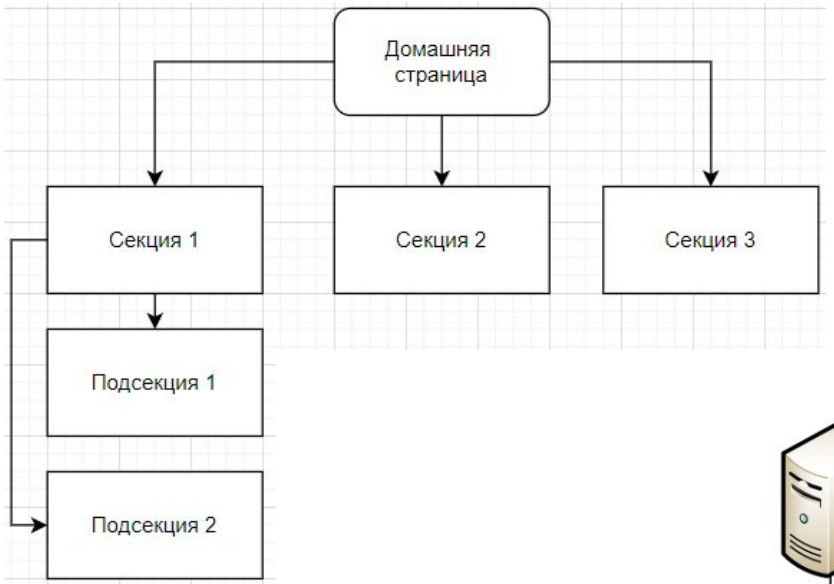
Клиент



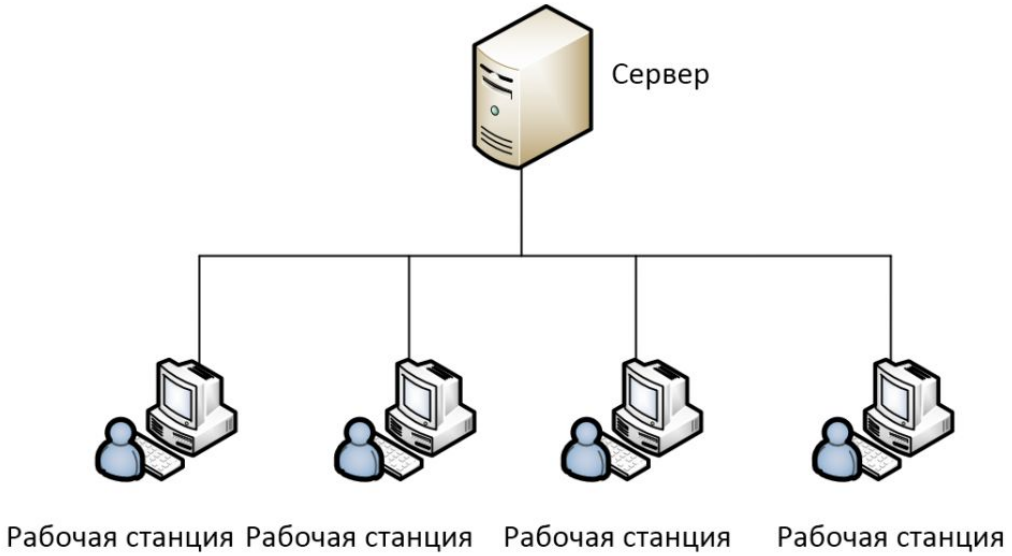
Протокол

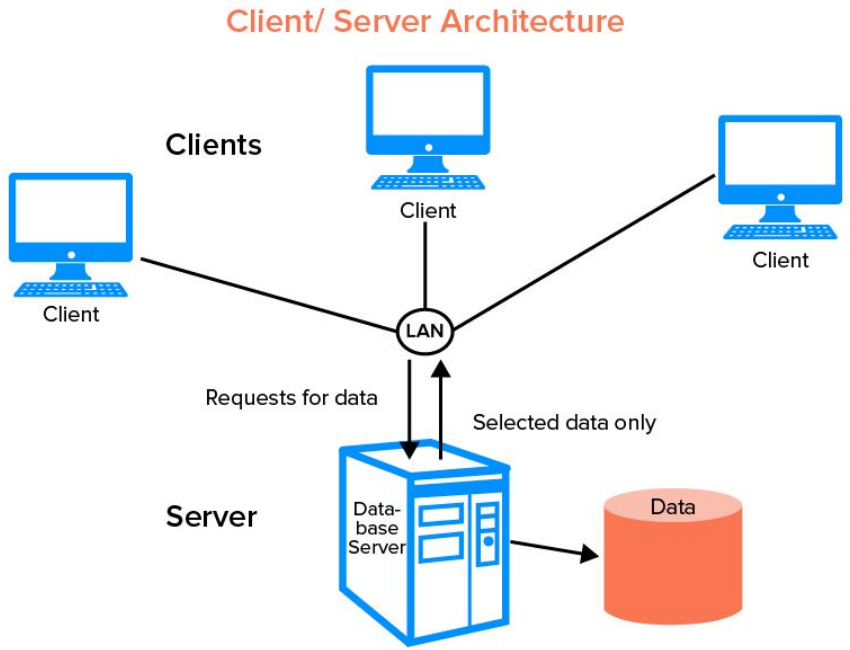
Авторизация и аутентификация

Структура клиентской части веб-ресурса



Процедура взаимодействия клиент-сервер





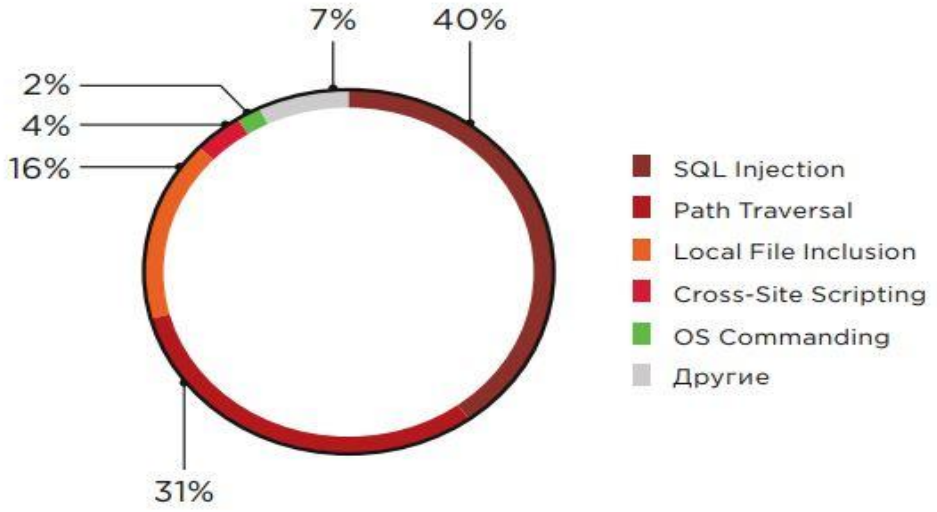
Основные угрозы для веб-ресурсов предприятия

Основные типы угроз информационной безопасности веб-приложения :

- Угрозы конфиденциальности – несанкционированный доступ к данным.
- Угрозы целостности – несанкционированное искажение или уничтожение данных.
- Угрозы доступности – ограничение или блокирование доступа к данным.

Тройка наиболее распространенных атак на веб-ресурсы не меняется из года в год:

1. «Внедрение SQL-кода»
2. «Выход за пределы каталога»
3. «Межсайтовое выполнение сценариев»



Основные требования к защите веб-ресурсов

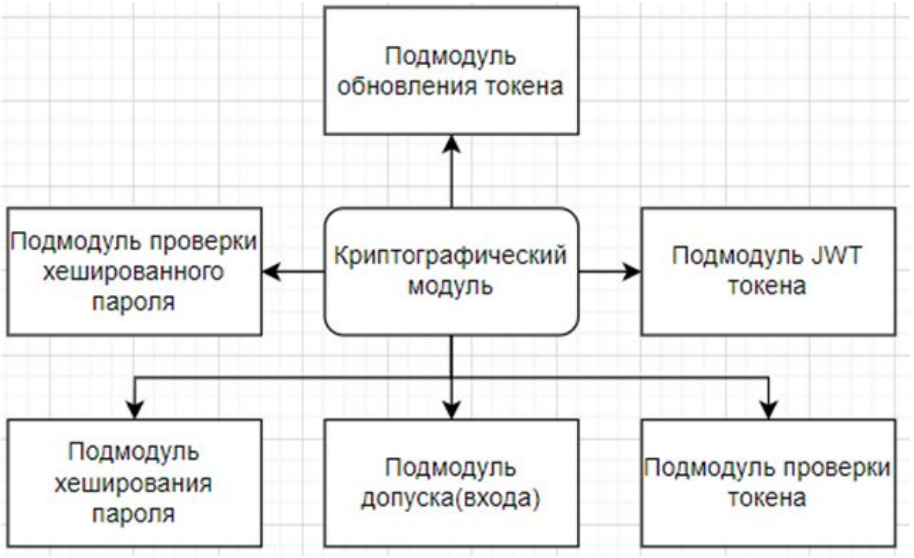
Требования	Документ
• Разработка на основе модели угроз системы защиты информации, обеспечивающей нейтрализацию предполагаемых угроз с использованием методов и способов защиты информации.	Приказ от 31 августа 2010 года N 489 [13]
• Модификация сайта должна быть возможна за кратчайшие сроки. • Необходимо использовать протоколы безопасности при передаче данных между локальным компьютером и сервером. • Должна проводиться регулярная проверка файлов журнала доступа и ошибок сервера (log-файлы) на предмет несанкционированного доступа. • Должна проводиться проверка подлинность идентификации осуществлять аутентификацию • Управлять обновлениями программного обеспечения	Технические требования по обеспечению безопасности веб-ресурсов [14]
• Безопасность персональных данных при их обработке в информационной системе обеспечивается с помощью системы защиты персональных данных, нейтрализующей актуальные угрозы.	Постановление Правительства РФ от 01.11.2012 N 1119 "Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных" [9]
• Все криптографические протоколы должны реализовать в среде специальных аппаратных средств СКЗИ • Для каждого конкретного средства регистрации должны проводится работы по оценке влияния данной модели на выполнение СКЗИ предъявляемых требований • Специальные аппаратные средства СКЗИ должны реализовать в себе совокупность аппаратно-программных систем защиты • В СКЗИ должен быть реализован форматно-логический контроль информации	Выписка «Требования к средствам криптографической защиты информации, предназначенным для обеспечения некорректируемой регистрации информации, не содержащей сведений, составляющих государственную тайну» ФСБ России от 23.09.2019
• Идентификация и аутентификация включают в себя распознавание пользователя средств доверенной третьей стороны, члена группы администраторов средств или процесса и проверку их подлинности. Механизм аутентификации должен блокировать доступ этих субъектов к функциям средств ДТС при отрицательном результате аутентификации.	Приказ ФСБ России от 04.12.2020 N 556 (ред. От 13.04.2021) [25]

Основные способы аутентификации веб-ресурсов

Способ	Основное применение	Протоколы
По паролю	Аутентификация пользователей	HTTP, Forms
По одноразовым паролям	Дополнительная аутентификация пользователей (для достижения two-factor authentication)	Forms
По ключам доступа	Аутентификация сервисов и приложений	-
По токенам	Делегированная аутентификация пользователей; делегированная авторизация приложений	SAML, WS-Federation, OAuth, OpenID Connect
По сертификатам	Аутентификация пользователей в безопасных приложениях; аутентификация сервисов	SSL/TLS
По электронной подписи	Аутентификация пользователей	Протокол CCITT X.509, Транспортный протокол Шамира
По биометрическим данным	Аутентификация пользователей в безопасных приложениях	Протоколы PAKE, BAKE
По GPS	Аутентификация пользователей	Протокол NTRIP

Криптографического модуль защиты веб-ресурсов предприятия

Структуре криптографического модуля



В основной состав криптографического модуля веб-приложения предприятия входят такие элементы как:

- Подмодуль хеширования пароля
- Подмодуль валидации хешированного пароля
- Подмодуль допуска
- Подмодуль JWT токена
- Подмодуль проверки токена
- Подмодуль обновления токена

Устройство и алгоритм работы JWT токена

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.ey
JzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6Ikpva
G4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.wFt3a
GAovoJT6kApMQNitoHYRwZM2Q0nlwrCV8Cd3rE
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

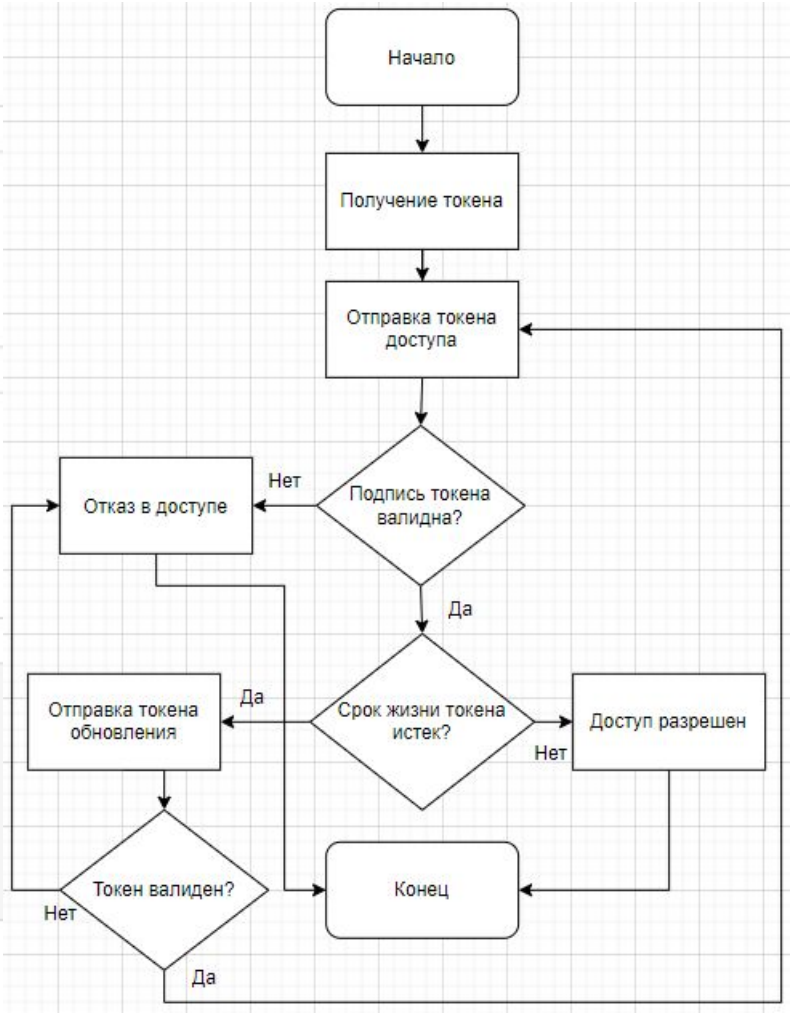
```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

PAYLOAD: DATA

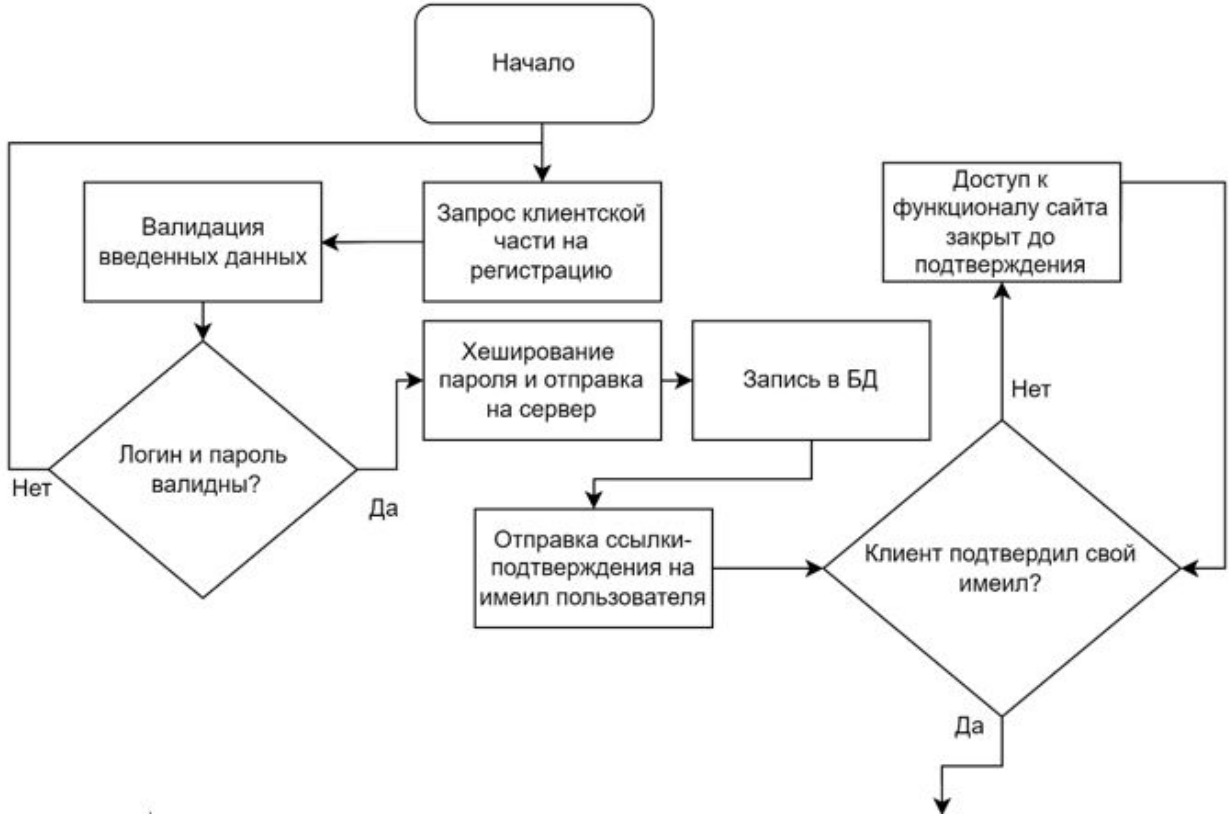
```
{
  "sub": "1234567890",
  "name": "John Doe",
  "iat": 1516239022
}
```

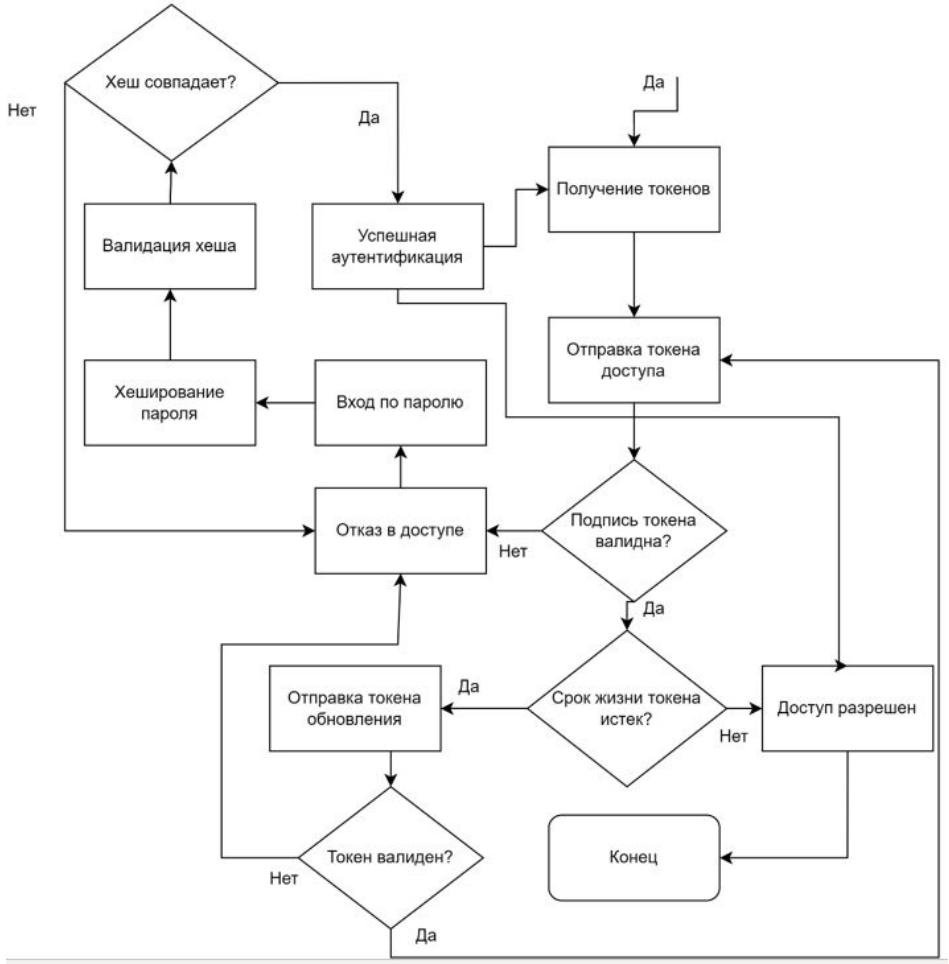
VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  my_secret_string
) ☐ secret base64 encoded
```



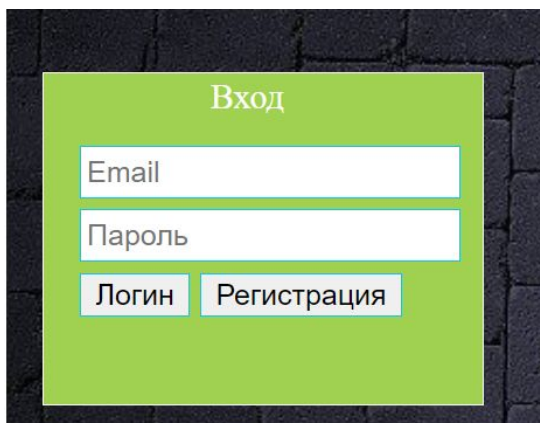
Алгоритм работы криптографического модуля веб-ресурса предприятия

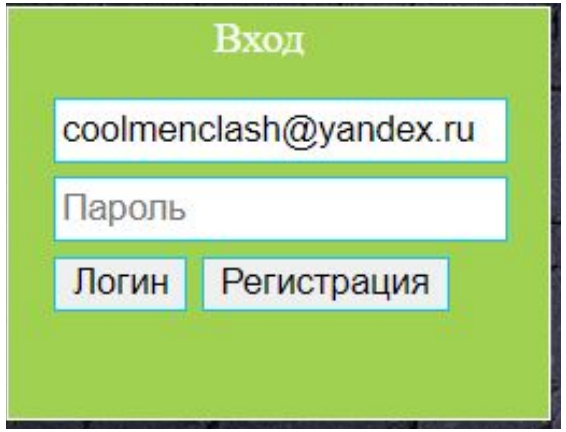




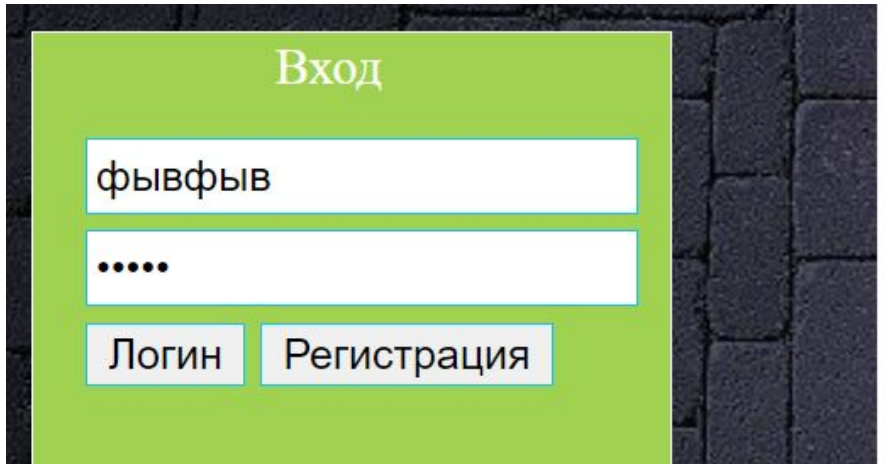
Пример работы программы

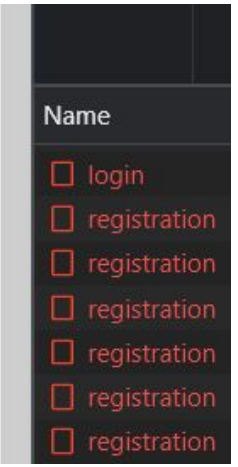
С начала происходит вход пользователем на стартовую страницу





В случае если данные не проходят валидацию на сервере, выводится ошибка:

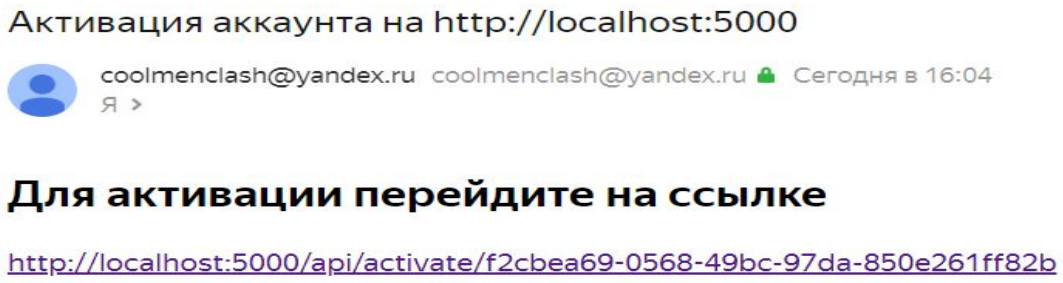


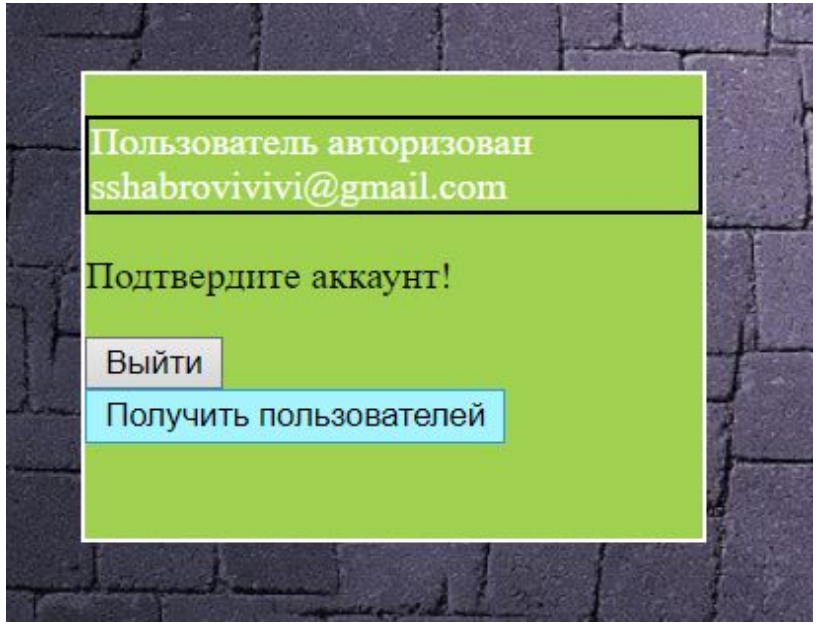


После регистрации и успешного пройденного теста на валидацию пароля и логина, пароль хешируется функцией стрибог и заносится в БД, вот как это выглядит:

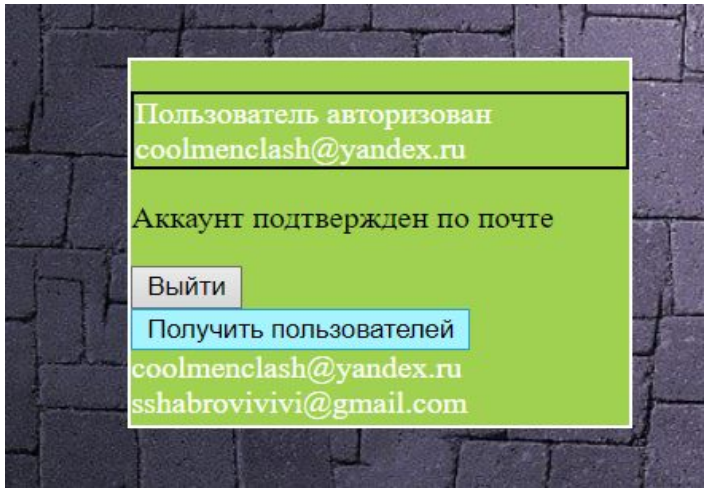
```
_id: ObjectId("631c94c2c2148c45aef64bf4")
email: "sshabrovivivi@gmail.com"
password: "d##$n#d6Fc/cPEf.#eW0D$94mD1d#$$y##2lwN7/./5/$I.AF53V0U$03$.m2o.$"
isActivated: false
```

После входа пользователя пускает на сервер, но функциями веб-ресурса он пользоваться не может. Здесь в качестве демонстрации таких функций присутствует кнопка «получить пользователей». Поскольку пользователь не подтвердил свой аккаунт, не перейдя по ссылке активации, отправленной ему на имеил





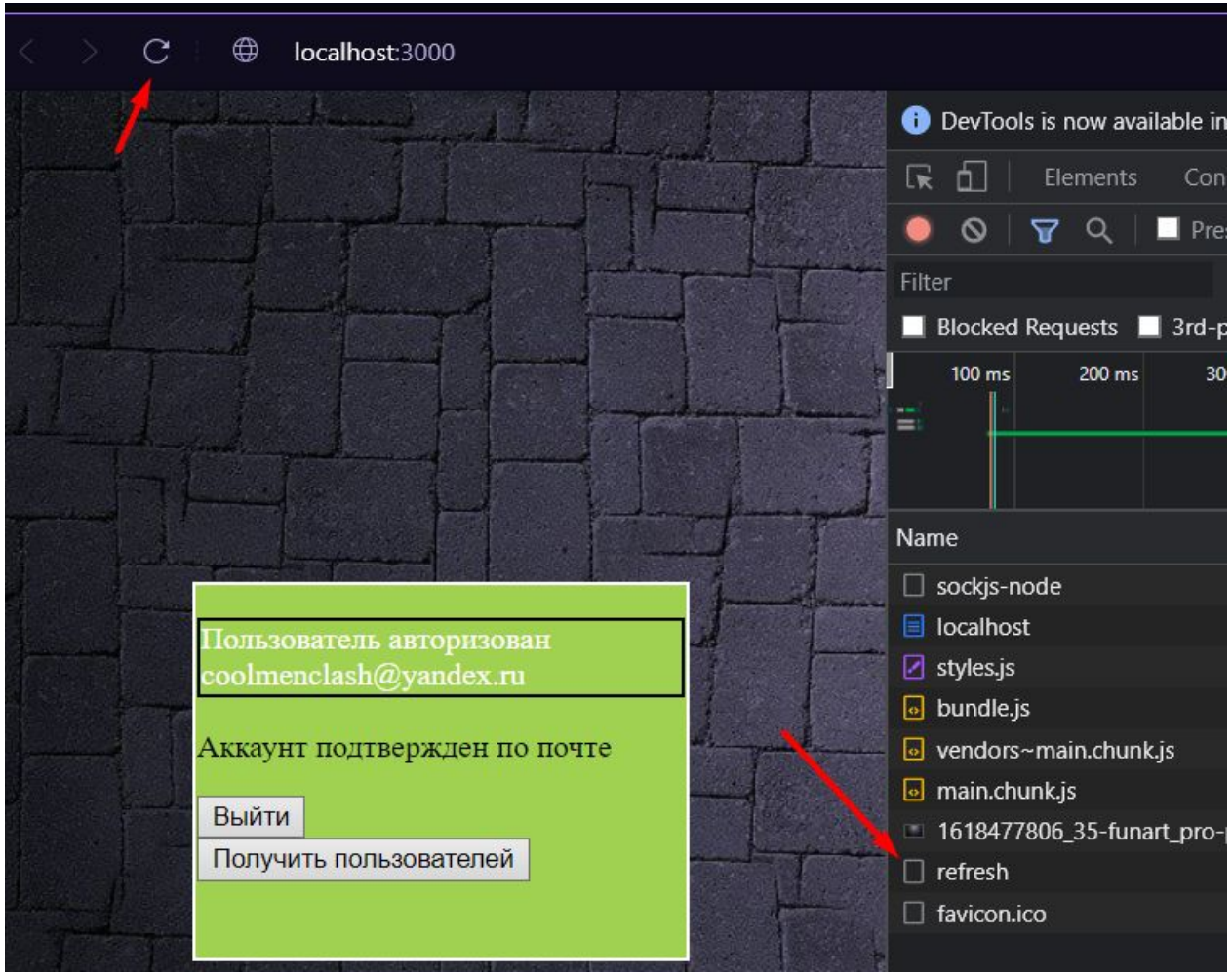
После подтверждения аккаунта можно пользоваться функциями ресурса, причем функции могут находится на стороннем сервере, пользователь все равно сможет ими пользоваться из-за наличия в куки токена доступа:



Активация аккаунта также прописывается в БД:

Также возможно возникновение ошибок, связанных с невнимательностью самого пользователя во время входа в аккаунт, таких как:

При обновлении страницы, в случае если токен доступа истек, клиент отправляет токен обновления, его отправление можно увидеть в панели разработчика:



Еще один пример обращения к функции ресурса при истекшем токене доступа (пункт user) и немедленная отправка токена обновления (refresh):

users	200	xhr	xhr.js:...	26...	11...
users	401	xhr	xhr.js:...	40...	5 ...
refresh	200	xhr	xhr.js:...	1....	41...
users	200	xhr	xhr.js:...	26...	14...

Представление токенов в куки клиента:


```
"accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI
6IkpXVCJ9.eyJuYW1lIjoiSmltIiwiaWF0IjoxNTY4NzU
4MTg0LCJleHAiOjE1Njg3NTgxOTl9.uWF4goP2qKLv4bb
JB2b2Ebmzu8NAkII4FB7d0qjElzY",
"refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cC
I6IkpXVCJ9.eyJuYW1lIjoiSmltIiwiaWF0IjoxNTY4Nz
U4MTg0fQ.Jjv1aQsSgmVRc3E7i1rXCl5usKq8DMcSzxFE
Mj__0p4"
}
```

Спасибо за внимание!