F.A.C.C.T.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

«F.A.C.C.T. Fraud Protection»

Руководство администратора

Содержание

ТЕРМИНЫ И СОКРАЩЕНИЯ 3

1 ОБЩИЕ СВЕДЕНИЯ 4

1.1 Введение 4

1.2 Назначение ПО 4

1.3 Программно-аппаратные среды функционирования ПО 4

1.4 Технические требования к составу оборудования при размещении в инфраструктуре Заказчика 5

1.5 Требования к базам данных 5

2 ОБЩИЕ ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ ПО 6

3 ОБЯЗАННОСТИ И ФУНКЦИИ АДМИНИСТРАТОРА ЗАКАЗЧИКА 8

4 ПОРЯДОК ВСТРАИВАНИЯ 9

4.1 Выбор схемы встраивания в инфраструктуру 9

4.2 Выработка RSA-ключей 14

4.3 Создание тестовых учетных записей 14

4.4 Определение IP-подсетей Заказчика, используемых при взаимодействии с АС Разработчика 14

4.5 Передача регистрационных данных Заказчика в F.A.C.C.T. Fraud Protection 15

4.6 Получение настроенных пользовательских модулей 16

4.7 Вставка ссылки на пользовательский модуль в страницы защищаемого веб-ресурса . 16

5 ПОДДЕРЖАНИЕ ФУНКЦИОНИРОВАНИЯ ПО 17

ТЕРМИНЫ И СОКРАЩЕНИЯ

Термин	**Описание**
АС	Автоматизированная система АО «БУДУЩЕЕ»
ПО	Программное обеспечение «F.A.C.C.T. Fraud Protection»
Заказчик	Лицо, которое использует на законных основаниях ПО на основании заключенного договора
Исполнитель	Работы Исполнителя на протяжении всего жизненного цикла могут исполняться: • АО «БУДУЩЕЕ»; • Компанией-интегратором, по выбору Заказчика
Разработчик	АО «БУДУЩЕЕ»
Mobile SDK (далее – SDK)	Модуль программного обеспечения «F.A.C.C.T. Fraud Protection» для встраивания в мобильные приложения
RSA	Криптографический алгоритм с открытым ключом, основывающийся на вычислительной сложности задачи факторизации больших полупростых чисел
Web Snippet (далее – скрипт)	Модуль программного обеспечения «F.A.C.C.T. Fraud Protection» для встраивания в WEB приложения

1 ОБЩИЕ СВЕДЕНИЯ

1.1 Введение

Настоящий документ содержит описание реализации программного обеспечения «F.A.C.C.T. Fraud Protection» (далее — ПО, F.A.C.CT. Fraud Protection).

1.2 Назначение ПО

F.A.C.C.T. Fraud Protection — система для противодействия мошенничеству и защиты цифровой личности пользователя в цифровых каналах обслуживания, а также защиты цифровых ресурсов от ботов и предотвращения мошенничества. ПО позволяет выявлять и предотвращать мошенническую активность, а также улучшать пользовательский опыт в автоматизированных системах Заказчика.

1.3 Программно-аппаратные среды функционирования ПО

Для корректного функционирования ПО необходим веб-браузер.

ПО поддерживает работу на следующих версиях браузеров:

- Internet Explorer версии 8.0 и выше;
- Google Chrome версии 4.0 и выше;
- Mozilla Firefox версии 3.5 и выше;
- Apple Safari версии 4.0 и выше;
- Opera версии 10.5 и выше;
- iOS Safari версии 3.2 и выше;
- Opera Mobile версии 11.0 и выше;
- Google Chrome for Android версии 11.0 и выше;
- Mozilla Firefox for Android версии 26.0 и выше;
- Windows Internet Explorer Mobile версии 10.0 и выше.

В браузере устройства пользователя должно быть разрешено исполнение скриптов JavaScript.

1.4 Технические требования к составу оборудования при размещении в инфраструктуре Заказчика

При **размещении в инфраструктуре Заказчика** требуется выделить следующие минимальные мощности для установки системы в промышленную эксплуатацию:

Три сервера приложений:

- CPU: 4 core 2Mhz+;
- RAM: 32 Gb;
- HDD: 200Gb;
- ОС: Ubuntu, Ред ОС.

Три сервера для размещения Баз данных:

- CPU: 4 core 2Mhz+;
- RAM: 32 Gb;
- HDD: 1Tb;
- ОС: Ubuntu, Ред ОС.

1.5 Требования к базам данных

ПО функционирует с использованием следующих СУБД:

- Apache Cassandra 4.0 и выше;
- Elasticsearch 7.0 и выше;
- ClickHouse 20.1 и выше.

2 ОБЩИЕ ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ ПО

На Рисунок 1 изображены общие принципы функционирования ПО.

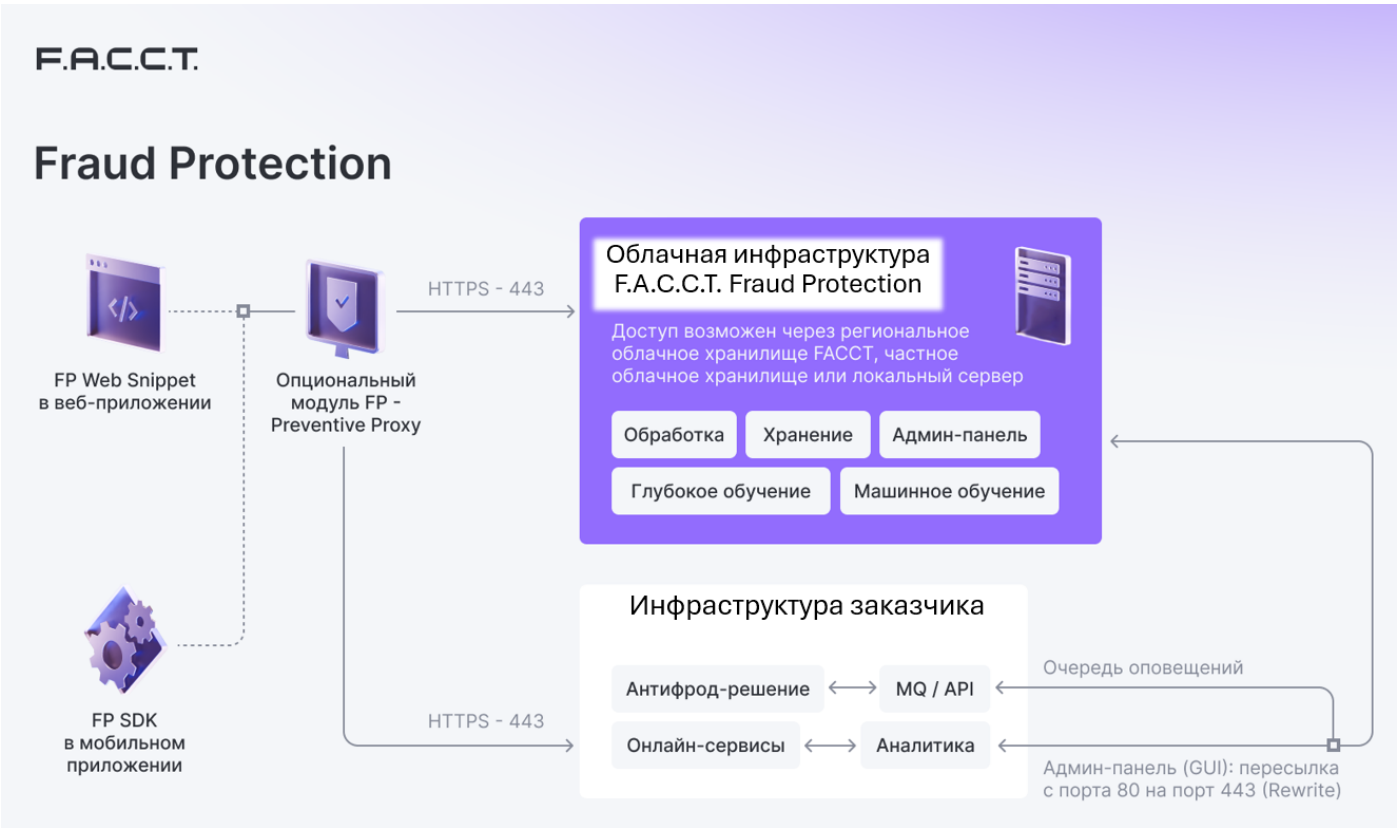

Рисунок 1. Общие принципы функционирования ПО.

ПО состоит из пользовательских модулей, реализованных на языках программирования Java и JavaScript.

WebSnippet (далее – скрипт) — клиентский модуль Fraud Protection для защиты веб-ресурсов, реализован на языке JavaScript. Модуль загружается совместно со страницами защищаемого веб-ресурса.

Mobile SDK (далее – SDK) — клиентский модуль Fraud Protection для защиты мобильных приложений, реализован на языке Java. Модуль запускается совместно с мобильным приложением.

ПО производит сбор контрольных данных со страницы защищаемого веб- или мобильного приложения и устройства клиента, и отсылает их для дальнейшего анализа в автоматизированную систему (далее – АС) АО «БУДУЩЕЕ» (далее – Разработчик). В случае выявления работы вредоносного ПО на устройстве пользователя или проведения иных мошеннических атак, АС Разработчика незамедлительно извещает об этом Заказчика.

ПО может быть представлена Заказчику двумя способами:

1. ПО как услуга (SaaS) – облачный интернет-сервис;
2. Размещение ПО в инфраструктуре Заказчика (On-premise).

Модули WebSnippet и SDK требуют встраивания в защищаемое приложение. ПО не требует инсталляции на устройстве пользователя и работает совершенно прозрачно для него.

Все данные передаются посредством протокола HTTPS с использованием порта 443.

3 ОБЯЗАННОСТИ И ФУНКЦИИ АДМИНИСТРАТОРА ЗАКАЗЧИКА

В обязанности администратора входит следующее:

- Произвести встраивание ПО в защищаемый веб-ресурс;
- Произвести встраивание ПО в защищаемое мобильно приложение;
- Поддерживать функционирование ПО.

4 ПОРЯДОК ВСТРАИВАНИЯ

Для встраивания ПО в защищаемый веб-ресурс или мобильное приложение необходимо выполнить следующие шаги:

1. Выбрать схему встраивания в инфраструктуру;
2. Выработать приватный и публичный RSA-ключи;
3. Создать две тестовые учетные записи на защищаемом веб-ресурсе;
4. Определить перечень IP-подсетей Заказчика, которые будут использоваться при взаимодействии с АС Разработчика;
5. Передать полученные ранее регистрационные данные Заказчика Разработчику;
6. Получить в ответ ссылку на настроенный под веб-ресурс пользовательский модуль WebSnippet;
7. Получить в ответ ссылку на настроенный под мобильное приложение пользовательский модуль SDK;
8. Сконфигурировать веб-серверы Заказчика на дублирование заголовков HTTP-запросов от пользователя на адрес https://fp-back.facct.ru;
9. Вставить в каждую необходимую страницу защищаемого веб-ресурса ссылку на пользовательский модуль;
10. Встроить SDK в мобильное приложение.

4.1 Выбор схемы встраивания в инфраструктуру

Существует три схемы встраивания ПО в инфраструктуру Заказчика. У каждой из схем есть свои достоинства и недостатки, оптимальное сочетание которых определяется Заказчиком исходя из условий использования защищаемого веб-ресурса.

Схема 1. Загрузка клиентского модуля и передача контрольных данных происходит на домены *.facct.ru.

На Рисунок 2 представлена схема загрузки клиентского модуля и передача контрольных данных на домены *.facct.ru.

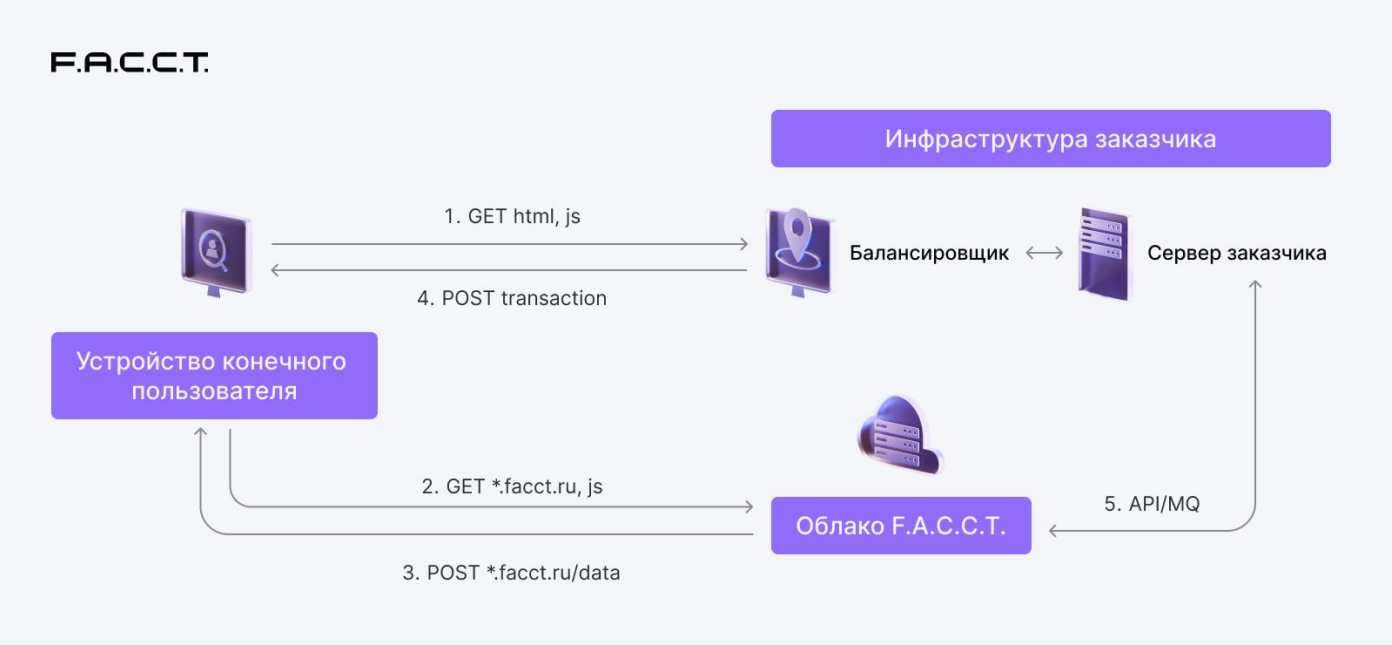

Рисунок 2. Схема загрузки клиентского модуля и передача контрольных данных на домены *.facct.ru.

1. Отправка GET-запроса с устройства конечного пользователя на сервер Заказчика через балансировщик нагрузки для получения данных;
2. Затем происходит отправка GET-запроса в бекенд приложение ПО, расположенного в облачной инфраструктуре Разработчика;
3. Далее из облачной инфраструктуры Разработчика на устройство конечного пользователя поступает POST-запрос с данными;
4. Сервер Заказчика в свою очередь возвращает POST-запрос, содержащий информацию о транзакции, на устройство конечного пользователя;
5. Обмен данными между сервером Заказчика и облачной инфраструктурой Разработчика происходит посредством методов API, либо с использованием брокеров сообщений MQ.

Достоинства:

– Минимальные настройки на стороне Заказчика;
– Отсутствие дополнительной нагрузки на ИТ-инфраструктуру Заказчика;
– Быстрый вариант для пилотного использования ПО.

Недостатки:

– На стороне браузера видны обращения на сторонние по отношению к Заказчику ресурсы;
– Неработоспособность ПО при использовании дополнительных настроек политики или плагинов браузера, которые ограничивают обмен со сторонними веб-ресурсами по отношению к основному;

– Неработоспособность в IE6 и IE7, или в некоторых режимах обратной совместимости с более ранними версиями в IE8 и старше.

Схема 2. IP-адреса серверов Разработчика регистрируются как домен следующего уровня в основной домен Заказчика.

В случае выбора **схемы 2**, если действие SSL-сертификат защищаемого веб-ресурса не распространяется на домены следующего уровня, то необходимо выпустить отдельный SSL-сертификат на созданный домен.

На Рисунок 3 показана схема передачи данных на поддомен Заказчика.

Рисунок 3. Схема передачи данных на поддомен Заказчика.

1. Отправка GET-запроса с устройства конечного пользователя на сервер Заказчика через балансировщик нагрузки для получения данных.
2. Затем происходит отправка GET-запроса в облачную инфраструктуру Разработчика
3. Из облачной инфраструктуры Разработчика на устройство конечного пользователя поступает POST-запрос с данными.
4. Сервер Заказчика в свою очередь возвращает POST-запрос, содержащий информацию о транзакции, на устройство конечного пользователя.
5. Обмен данными между сервером Заказчика и облачной инфраструктурой Разработчика происходит посредством методов API, либо с использованием брокеров сообщений MQ.

Достоинства:

- Весь обмен между браузером пользователя и веб-ресурсом происходит с использованием доменов Заказчика;
- Отсутствие блокировок работы клиентского модуля сторонним ПО;
- Средний уровень скрытности использования ПО для мошенника;
- Отсутствие дополнительной нагрузки на ИТ-инфраструктуру Заказчика.

Недостатки:

- В некоторых случаях требуется дополнительно выпускать SSL-сертификат на новые домены (описано ниже);
- Неработоспособность в IE6 и IE7, или в некоторых режимах обратной совместимости с более ранними версиями в IE8 и старше.

Схема 3. Загрузка клиентского модуля и передача контрольных данных производится через веб-серверы Заказчика.

Необходимые настройки в случае выбора схемы III будут предоставлены отдельно по запросу Заказчика.

На Рисунок 4 представлена схема передачи контрольных данных через IT-инфраструктуру Заказчика.

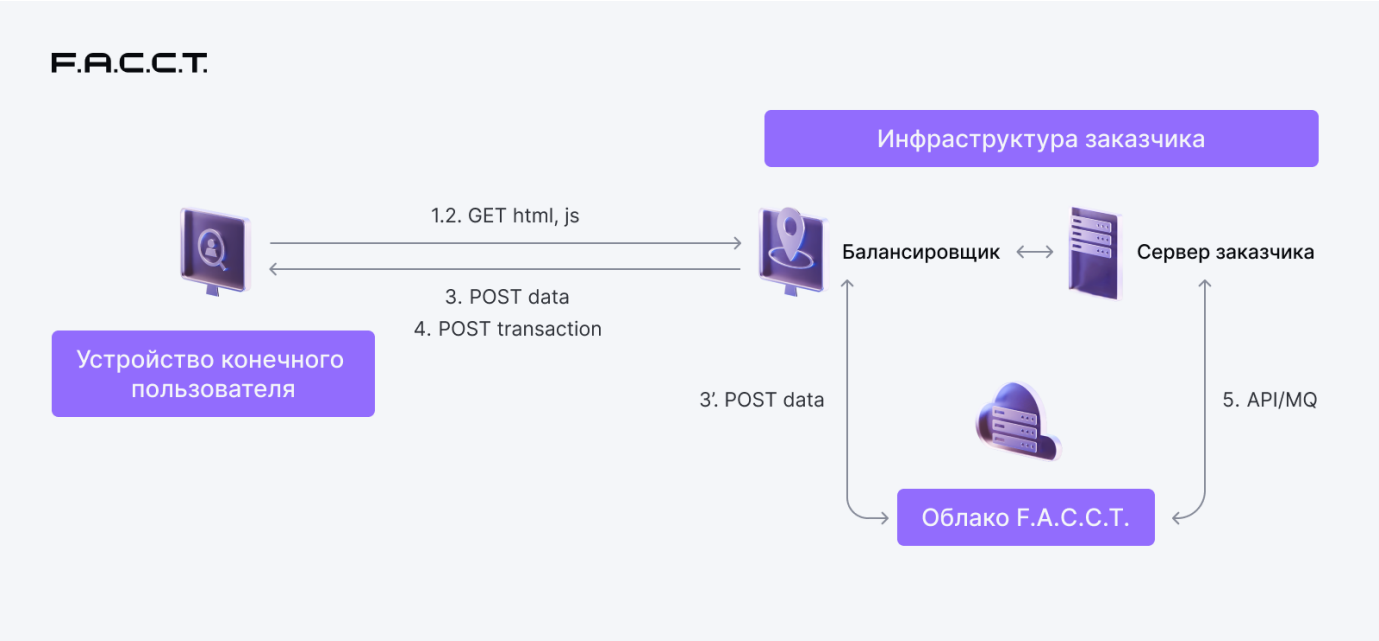

Рисунок 4. Схема передачи контрольных данных через IT-инфраструктуру Заказчика.

1. - 2. Отправка GET-запроса с устройства конечного пользователя на сервер Заказчика через балансировщик нагрузки для получения данных.

3. - 4. Сервер Заказчика в свою очередь возвращает POST-запрос, содержащий запрошенные данные, и информацию о транзакции на устройство конечного пользователя.

3'. Балансировщик перераспределяет POST-запросы с облачной инфраструктуры Разработчика

5. Обмен данными между сервером Заказчика и облачной инфраструктурой Разработчика происходит посредством методов API, либо с использованием брокеров сообщений MQ.

Достоинства:

- Весь обмен между браузером пользователя и веб-ресурсом происходит с использованием его домена;
- Отсутствие блокировок работы клиентского модуля сторонним ПО;
- Высокий уровень скрытности использования ПО для мошенника;
- Работоспособность в IE6 и IE7, а также в режимах обратной совместимости с более ранними версиями в IE8 и старше.

Недостатки:

- Требуются дополнительные настройки по трансляции запросов, относящихся к ПО, на web-серверах Заказчика;
- Дополнительная нагрузка на инфраструктуру Заказчика.

Примечание:

По выбору Заказчика незначительная часть серверной функциональности ПО, генерации полиморфного клиентского модуля и его раздачи может быть передана Заказчику. Это дает Заказчику полный контроль над изменениями клиентского модуля и перечнем передаваемых данных с устройства пользователя. Инструкции по настройке вышеуказанного функционала будут предоставлены Заказчику отдельно по запросу.

Далее указаны общие шаги по внедрению ПО вне зависимости от выбранной схемы встраивания.

4.2 Выработка RSA-ключей

Публичный RSA-ключ Заказчика используется пользовательским модулем для шифрования имени учетной записи пользователя. Шифрование производится на устройстве пользователя. Зашифрованное имя учетной записи пользователя передается в АС Разработчика с другими контрольными деталями страницы защищаемого веб-ресурса.

Приватный RSA-ключ Заказчика используется для расшифрования имени учетной записи пользователя, если получено извещения из АС Разработчика о признаках подозрительного события. Расшифрование производится на стороне Заказчика. Таким образом, обеспечивается конфиденциальность пользовательских учетных данных.

Размерность ключей, срок действия и выбор программного обеспечения для выработки пары RSA-ключей определяется Заказчиком.

Далее приведены команды для выработки ключей на примере свободного программного обеспечения OpenSSL (www.openssl.org):

a) Для создания приватного RSA-ключа необходимо выполнить команду:

```
openssl genrsa -out privkey.pem 1024
```

b) Для получения публичного RSA-ключа необходимо выполнить команду:

```
openssl rsa -pubout -in privkey.pem -out pubkey.pem
```

4.3 Создание тестовых учетных записей

Для настройки пользовательских модулей необходим доступ в защищаемый веб-ресурс. Для достоверной проверки, что пользовательский модуль не будет собирать контрольные данные, которые зависят от пользователя, необходимо использовать две различные учетные записи.

Условия предоставления тестовых учетных записей определяются Заказчиком.

4.4 Определение IP-подсетей Заказчика, используемых при взаимодействии с АС Разработчика

В целях обеспечения информационной безопасности, помимо использования протокола HTTPS при взаимодействии между компонентами АС Заказчика и Разработчика, используется ограничение на публичные IP-адреса/подсети Заказчика, с которых это взаимодействие возможно.

На Рисунок 5 представлена принципиальная схема взаимодействия между АС Заказчика и Разработчиком.

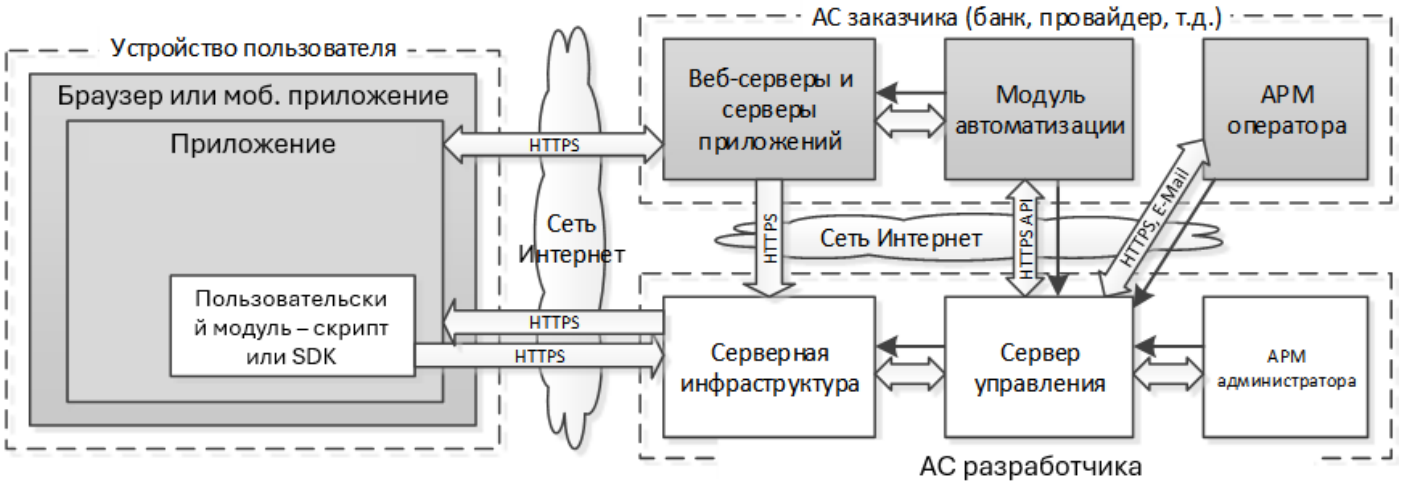

Рисунок 5. Принципиальная схема взаимодействия между АС Заказчика и F.A.C.C.T. Fraud Protection

Необходимо определить все IP-адреса/подсети Заказчика, которые будут участвовать в обмене между следующими компонентами АС:

- Веб-серверы АС Заказчика и Серверной инфраструктурой АС Разработчика;
- Модуль автоматизации АС Заказчика и Сервером управления АС Разработчика;
- АРМ оператора АС Заказчика и Сервером управления АС Разработчика.

При определении IP-адресов/подсетей необходимо учесть существующие сценарии обеспечения непрерывности функционирования АС Заказчика.

Политика ограничений по доступу к АС Разработчика со стороны компонент АС Заказчика определяется Заказчиком самостоятельно. При этом необходимо учитываться следующее:

- все взаимодействие с АС Разработчика инициируется со стороны компонент АС Заказчика по протоколу HTTPS;
- доменным именам АС Разработчика соответствует несколько IP-адресов в целях обеспечения бесперебойности работы АС и распределения нагрузки на нее.

4.5 Передача регистрационных данных Заказчика в F.A.C.C.T. Fraud Protection

Через портал защищенной электронной почты F.A.C.C.T. (https://smail.facct.ru) необходимо отправить письмо с заголовком «Fraud Protection registration» со следующими сведениями:

– Публичный RSA-ключ Заказчика. Передача приватного RSA-ключа строго запрещена и потребует выработки новой пары RSA-ключей;
– Две тестовых учетных записи с паролями к защищаемому веб-ресурсу;
– Публичные IP-адреса/подсети Заказчика, которые участвуют в обмене с АС Разработчика.

Если портал защищенной электронный почты F.A.C.C.T. используется в первый раз, то необходимо пройти процесс регистрации на портале.

4.6 Получение настроенных пользовательских модулей

Для настройки пользовательского модуля под защищаемое приложение или веб-ресурс потребуется некоторый период времени, который зависит от сложности веб-ресурса и мобильного приложения. Данный период согласовывается с Заказчиком отдельно.

В ответ на исходное письмо Заказчика с регистрационными данными, по окончании настройки пользовательского модуля, Разработчик вышлет ссылку на скачивание настроенных модулей через портал защищенной почты.

4.7 Вставка ссылки на пользовательский модуль в страницы защищаемого веб-ресурса

Пользовательский модуль написан на языке JavaScript. Для его использования необходимо вставить в раздел HEAD необходимых HTML-страниц защищаемого веб-ресурса следующую директиву:

```
<script type=”text/javascript” src=[ссылка на пользовательский модуль]></script>
```

ВНИМАНИЕ: указанная директива должна находиться сразу за тегом <HEAD>.

5 ПОДДЕРЖАНИЕ ФУНКЦИОНИРОВАНИЯ ПО

Поддержание функционирования ПО заключается в контроле настроек, проведенных в рамках установки ПО. Иных регламентных мероприятий со стороны Заказчика ПО не требует.