F6

ПО «F6 Session Fraud Protection»

Руководство администратора

Оглавление

Термины и сокращения

АС	Автоматизированная система
Заказчик	Лицо, использующее программное обеспечение на основании заключённого договора и эксплуатирующее его в своей инфраструктуре
Исполнитель	Работы Исполнителя на протяжении всего жизненного цикла могут исполняться: • АО «БУДУЩЕЕ»; • Компанией-интегратором, по выбору Заказчика
Операция	Отдельное действие или событие в системе, связанное с выполнением бизнес-процесса или обработкой данных и подлежащее анализу и учёту
ПО	Программное обеспечение «F6 Session Fraud Protection»
Пользователь	Лицо, взаимодействующее с цифровыми каналами обслуживания Заказчика, в отношении которого применяется антифрод-защита
Разработчик	АО «БУДУЩЕЕ»
СУБД	Система управления базами данных
API	(«Application Programming Interface») Программный интерфейс, который позволяет разным системам обмениваться данными и взаимодействовать друг с другом по заранее определенным правилам
IP-адрес	(«Internet Protocol Address») Уникальный числовой идентификатор устройства в компьютерной сети, используемый для обмена данными в интернете или локальной сети
Mobile SDK	Модуль программного обеспечения «F6 Session Fraud Protection» для встраивания в мобильные приложения
MQ	(«Message Queue») Технология обмена данными через брокера сообщений, при котором системы взаимодействуют асинхронно, отправляя и получая сообщения через очереди
On-premises	Модель развёртывания программного обеспечения, при которой система устанавливается и эксплуатируется в собственной инфраструктуре Заказчика, без использования внешних облачных сервисов
SaaS	(«Software as a Service») Модель предоставления программного обеспечения, при которой система размещается и эксплуатируется в облачной инфраструктуре поставщика, а

	доступ к ней осуществляется через сеть без установки в инфраструктуре Заказчика
Web Snippet	Модуль программного обеспечения «F6 Session Fraud Protection» для встраивания в веб-приложения

#1 Общие сведения

1.1. Введение

Настоящий документ содержит описание реализации программного обеспечения «F6 Session Fraud Protection» (далее — ПО, Session Fraud Protection).

1.2. Назначение ПО

«F6 Session 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;
- Яндекс Браузер версии 23.1.1 и выше.

В браузере устройства Заказчика должно быть разрешено исполнение скриптов JavaScript.

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

В случае использования облачной интеграции требования к оборудованию отсутствуют.

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

- Серверы приложений (3 шт.):
  - CPU: 4 core, 2Mhz и выше;
  - RAM: 32 GB;
  - HDD: 200 GB;
  - ОС: Ubuntu, РЕД ОС.
- Серверы баз данных (3 шт.):
  - CPU: 4 core, 2Mhz и выше;
  - RAM: 32 GB;
  - HDD: 1 TB;

- ОС: Ubuntu, РЕД ОС.

Требования к развёртыванию предоставленной тестовой среды:

- Среда виртуализации:
  - Система контейнеризации Docker;
  - Система управления контейнерами Kubernetes.

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 рисунке 1 изображены общие принципы функционирования ПО.

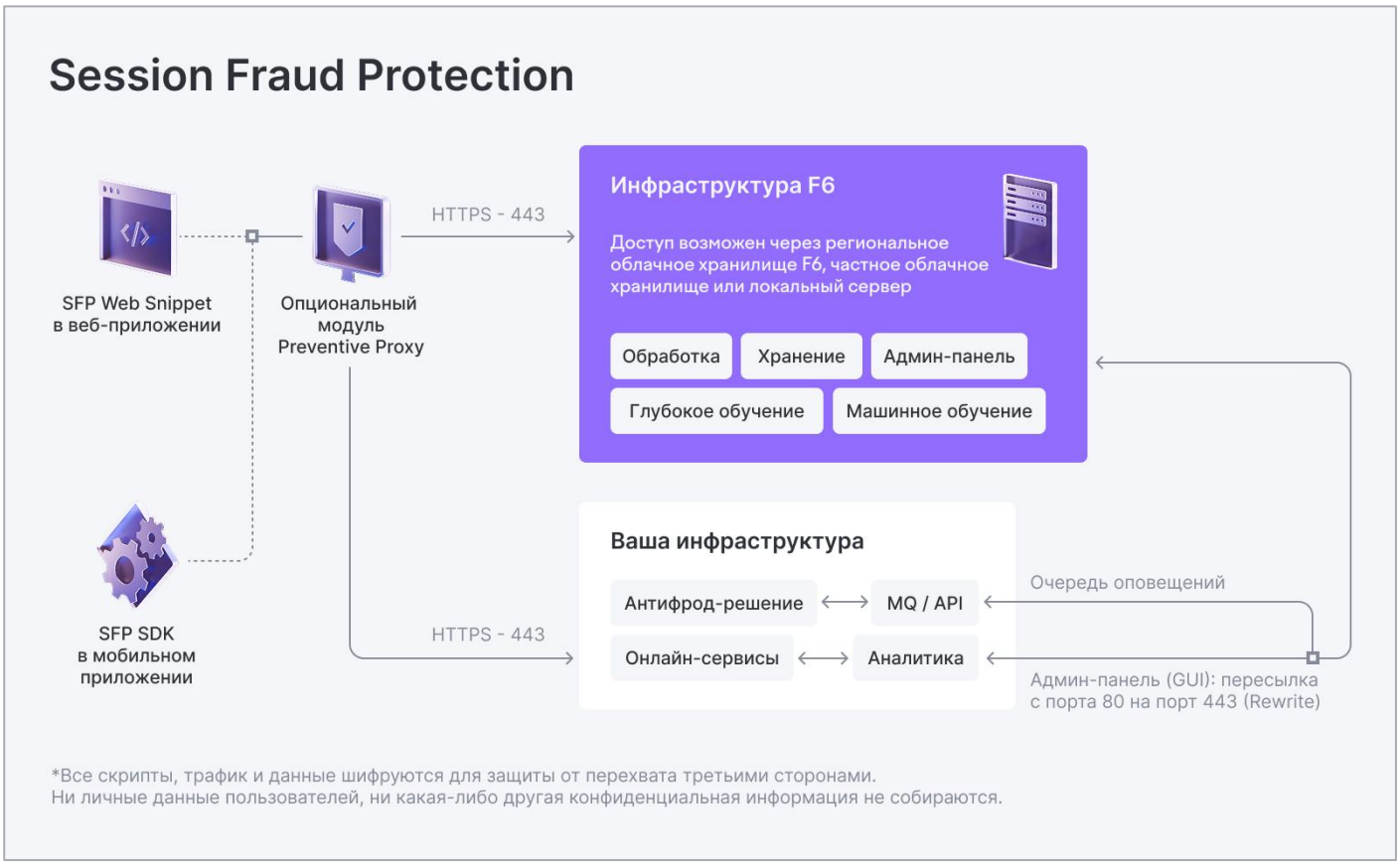

Рисунок 1. Общие принципы функционирования ПО "F6 Session Fraud Protection"

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

Web Snippet (далее — скрипт) — пользовательский модуль Session Fraud Protection для защиты веб-ресурсов, реализованный на языке JavaScript. Модуль загружается совместно со страницами защищаемого веб-ресурса.

Mobile SDK (далее — SDK) — пользовательский модуль Session Fraud Protection для защиты мобильных приложений, реализованный на языке Java (для устройств на операционной системе Android) и Swift (для устройств на операционной системе iOS). Модуль запускается совместно с мобильным приложением.

ПО осуществляет сбор контрольных данных со страниц защищаемых веб-ресурсов или мобильных приложений, а также с устройства Пользователя, и направляет их для дальнейшего анализа в автоматизированную систему (далее — АС) АО «БУДУЩЕЕ» (далее — Разработчик).

При выявлении признаков работы вредоносного ПО на устройстве Пользователя либо иных мошеннических атак АС Разработчика незамедлительно уведомляет Заказчика.

ПО может быть представлена Заказчику двумя способами:

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

Модули Web Snippet и SDK требуют встраивания в защищаемое приложение. ПО не требует установки на устройстве пользователя и функционирует без его участия.

Все данные передаются посредством протокола HTTPS с использованием порта 443.

#3 Обязанности и функции администратора Заказчика

В обязанности администратора входит следующее:

1. Произвести встраивание ПО в защищаемый веб-ресурс;
2. Произвести встраивание ПО в защищаемое мобильное приложение;
3. 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. Сгенерировать приватный и публичный RSA-ключи;
3. Создать две тестовые учётные 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 Заказчика пользовательский модуль Web Snippet;
7. Получить в ответ ссылку на настроенный под мобильное приложение Заказчика пользовательский модуль SDK;
8. Сконфигурировать веб-серверы Заказчика на дублирование заголовков HTTP-запросов от пользователя на адрес https://fp-back.fp.f6.security;
9. Вставить в каждую необходимую страницу защищаемого веб-ресурса ссылку на пользовательский модуль;
10. Встроить модуль 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.

4.1.1. Схема 1. Загрузка клиентского модуля и передача контрольных данных происходит на домены F6

На рисунке 2 представлена схема загрузки клиентского модуля и передача контрольных данных на домены *.f6.security

Рисунок 2. Схема загрузки клиентского модуля и передача контрольных данных на домены *.f6.security

1. Отправка GET-запроса с устройства конечного Пользователя на сервер Заказчика через балансировщик нагрузки для получения данных;
2. Отправка GET-запроса в backend-приложение ПО, расположенное в облачной инфраструктуре 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.

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

В случае выбора схемы 2, если SSL-сертификат защищаемого веб-ресурса не распространяется на домены следующего уровня, требуется выпуск отдельного SSL-сертификата для созданного домена.

На рисунке 3 показана схема передачи данных на поддомен Заказчика.

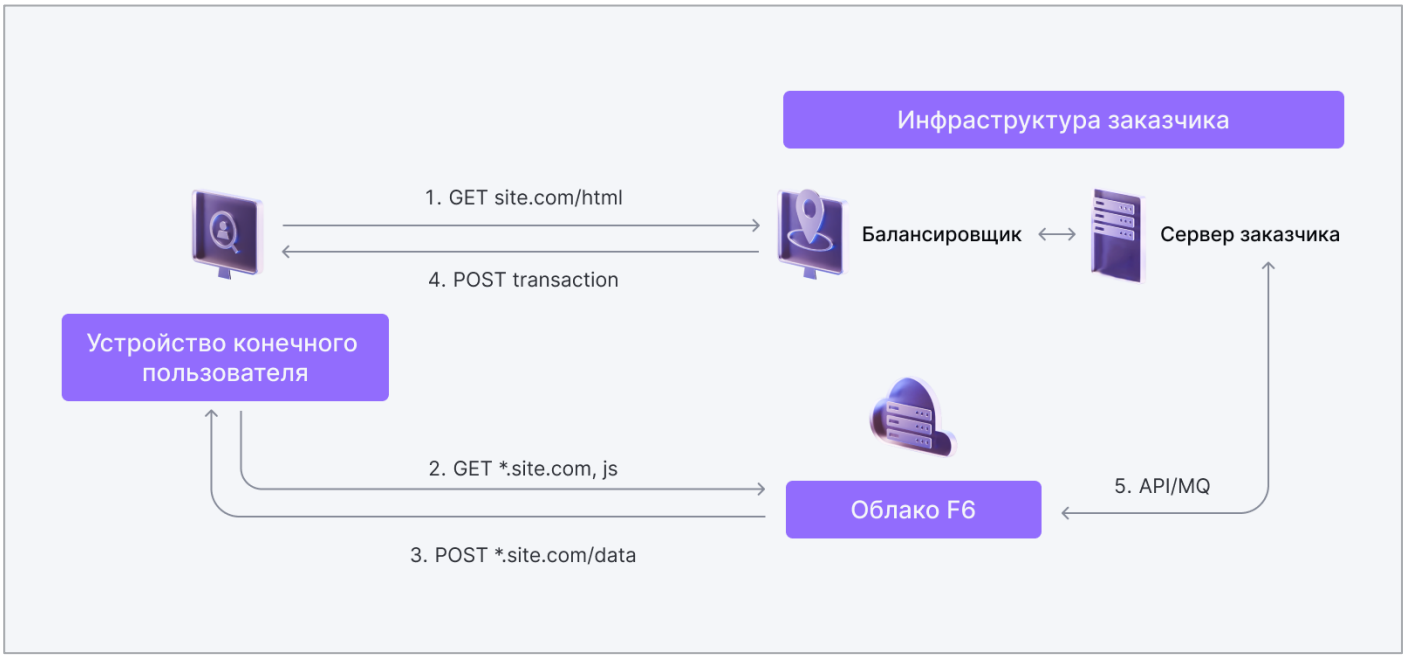

Рисунок 3. Схема передачи данных на поддомен Заказчика

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

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

В случае выбора схемы 3 необходимые настройки будут предоставлены отдельно по запросу Заказчика.

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

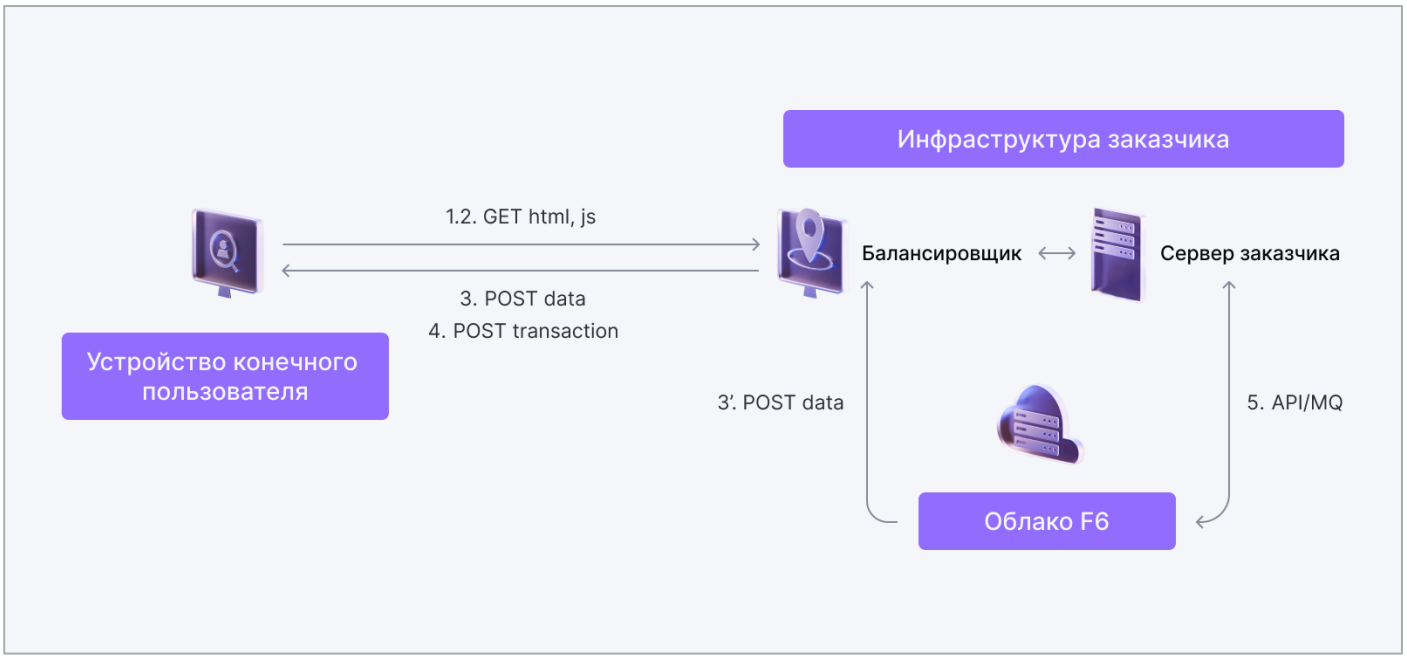

Рисунок 4. Схема передачи контрольных данных через IT-инфраструктуру Заказчика

1. Отправка GET-запроса с устройства конечного пользователя на сервер Заказчика через балансировщик нагрузки для получения данных;
2. 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;
4. Обмен данными между сервером Заказчика и облачной инфраструктурой Разработчика происходит посредством методов API, либо с использованием брокеров сообщений MQ.

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

Далее описаны 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 учётной записи Пользователя. Шифрование производится на устройстве Пользователя. Зашифрованное имя учётной записи Пользователя передается в АС Разработчика с другими контрольными деталями страницы защищаемого веб-ресурса.

Приватный RSA-ключ Заказчика используется для расшифрования имени учётной записи Пользователя, если получены извещения из АС Разработчика о признаках подозрительного события. Расшифрование производится на стороне Заказчика. Таким образом, обеспечивается конфиденциальность пользовательских учётных данных.

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

Далее приведены команды для генерации ключей на примере свободного программного обеспечения OpenSSL (версии 3.0 и выше) (www.openssl.org):

- Для создания приватного RSA-ключа необходимо выполнить команду:

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

- Для получения публичного RSA-ключа необходимо выполнить команду:

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

4.3. Создание тестовых учётных 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 учётные записи.

Условия предоставления тестовых учётных записей определяются Заказчиком.

4.4. Определение IP-подсетей используемых при взаимодействии с тестовой версией ПО

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 рисунке 5 представлена принципиальная схема взаимодействия между АС Заказчика и Разработчиком:

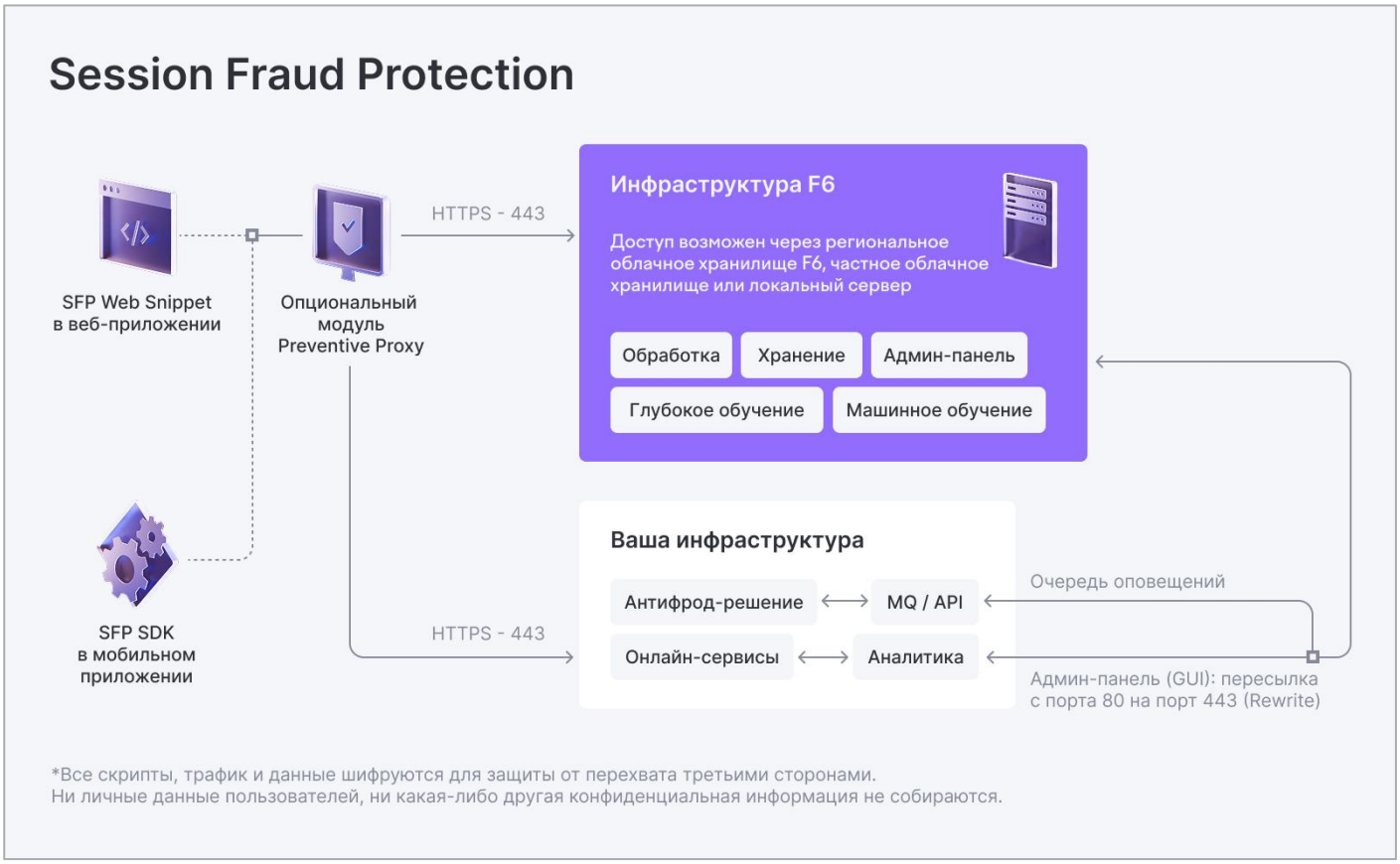

Рисунок 5. Принципиальная схема взаимодействия между АС Заказчика и F6 Session Fraud Protection

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

- Веб-серверами АС Заказчика и серверной инфраструктурой АС Разработчика;
- Модулем 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 компонентов АС Заказчика определяется Заказчиком самостоятельно. При этом необходимо учитывать следующее:

- Всё взаимодействие с АС Разработчика инициируется со стороны компонентов 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 «F6 Session Fraud Protection»

Через портал защищенной электронной почты F6 (https://smail.f6.ru) необходимо отправить письмо с заголовком «Session Fraud Protection registration» со следующими сведениями:

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

Если портал защищенной электронный почты F6 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 защищённой 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, выполненных в рамках установки ПО. Иных регламентных мероприятий со стороны Заказчика ПО не требует.