

**Описание функциональных и технических характеристик
программного обеспечения
«Интегрированная платформа предупреждения мошенничества
(ИППМ)» (далее – ИППМ)**

для расширения коммерциализации «Платформы цифрового доверия»

**Москва
2024 г.**

Оглавление

1. Общие сведения.....	3
2. Функциональные характеристики ПО	3
2.1. Клиентская библиотека и ее API.....	3
2.2. Ядро и его API.....	5
3. Технические характеристики ПО	5
3.1. Ядро	5
3.2. Клиентская библиотека	5
4. Алгоритм работы системы	6
5. Примеры генерируемых инцидентов	7
6. Пример набора данных, генерируемого клиентской библиотекой ИППМ	7
7. Сторонние компоненты ПО и программные среды.....	8
8. Дополнительные сведения	8

1. Общие сведения

Программное обеспечение «Интегрированная платформа предупреждения мошенничества» (далее – ИППМ) представляет собой программное обеспечение, разработанное ООО «Системы управления идентификацией» (ИНН 7709982222) в 2023 году и предназначено для предупреждения мошеннических действий в интернет-среде. Все права на использование программного обеспечения принадлежат компании ООО «Системы управления идентификацией».

2. Функциональные характеристики ПО

ПО ИППМ состоит из ядра системы и клиентской библиотеки для веб-браузера. Клиентская библиотека встраивается в веб-приложения (например, для дистанционного банковского обслуживания), предназначенные для использования клиентами потребителя системы. Клиентская библиотека собирает данные о характеристиках и настройках устройства клиента, на котором работает приложение (цифрового отпечатка устройства), и поведении клиента при работе с приложением. Ядро системы производит оценку репутации устройства и сессии (промежуток времени, начинающийся и заканчивающийся выполнением целевого действия, например, вход клиента в личный кабинет и выход из него и т.п.). Оценка репутации производится ядром в том числе на базе инцидентов, которые оно формирует в соответствии с набором установленных правил на основе данных, получаемых от клиентской библиотеки, и других доступных сведений, например, информации о сетевом соединении. Ядро анализирует инциденты и прочую информацию, доступную ему по данному устройству или сессии, и по запросу потребителя через API ядра формирует оценку репутации (скоринг) устройства или сессии. Скоринг отражает низкий, средний и высокий уровень риска, который несет для потребителя системы взаимодействие с данным клиентом, и тем самым помогает потребителю сделать вывод о необходимости принятия дополнительных мер предосторожности при работе с клиентом.

2.1. Клиентская библиотека и ее API

Для использования клиентской библиотеки потребителю ИППМ необходимо встроить ее в защищаемое веб-приложение. Пример кода на языке JavaScript для встраивания клиентской библиотеки приведен ниже.

```

<script type="text/javascript">
!function(){var
t=document.createElement("script");t.type="text/javascript",t.async=!0,
t.src=<URL загрузки API ИППМ>,
t.onload=function(){<Объект API ИППМ>.Init("<Код>")},
document.head.appendChild(t)()};
</script>

```

Приведенный выше фрагмент кода загружает клиентскую библиотеку и инициализирует ее API, после чего становятся доступны его функции, приведенные в таблице 2.1.

Таблица 2.1 – Список функций клиентской библиотеки и их описание

Название функции	Описание
GetFingerprintId()	Возвращает идентификатор текущей страницы, который может быть использован для получения скоринга устройства через API Ядра, например, при принятии решения о возможности начала сессии клиента
SetSessionId(<идентификатор сессии>)	Устанавливает идентификатор сессии, которая ранее была начата в веб-приложении. О начале сессии следует также сообщить через API ядра.
ClearSessionId()	Сбрасывает (удаляет) идентификатор сессии при ее окончании. При вызове данного метода также генерируется новый идентификатор страницы. Об окончании сессии следует также сообщить через API ядра.
GetSessionId()	Возвращает идентификатор текущей сессии, который может быть использован для получения скоринга сессии через API Ядра, например, для принятия решения о возможности использования клиентом определенных функций веб-приложения

Пример кода на языке JavaScript для использования клиентской библиотеки приведен ниже.

```

try {
  var sessionId = <Объект API ИППМ>.GetSessionId()
}
catch (e) { }

```

2.2. Ядро и его API

Для получения возможности использования API ядра ИППМ необходимо выполнить инструкцию по его установке, приведенную в документе «Руководство по установке ИППМ», после чего станут доступны его функции, приведенные в таблице 2.2. API использует протокол HTTP, параметры и результаты вызовов имеют формат JSON.

Таблица 2.2 – Функции API ядра

Название функции	Описание
startClientSession	Сообщает ядру, что указанная в параметрах метода сессия началась.
endClientSession	Сообщает ядру, что указанная в параметрах метода сессия закончилась.
getClientSessionIncidents	Возвращает список инцидентов, влияющих на скоринг сессии, указанной в параметрах метода.
getClientSessionAttributes	Возвращает список атрибутов сессии.
reportClientSessionReputation	Сообщает ядру, что указанная в параметрах метода сессия имеет плохую репутацию.
getNormalizedDeviceScore	Возвращает скоринг хоста, ассоциированного со страницей или сессией, указанной в параметрах метода.

3. Технические характеристики ПО

3.1. Ядро

Ядро системы реализовано на языке программирования Java и работает на сервере. Для работы ядра рекомендуется использовать сервер со следующими характеристиками:

- 1) Процессор CPU Intel Core i7 6-th Gen 3.0G Hz или мощнее.
- 2) RAM: 16 Гб оперативной памяти или более.
- 3) HDD: 100 Гб свободного места на диске.
- 4) Тип и версия операционной системы: Ubuntu 18.04 или новее.

3.2. Клиентская библиотека

Клиентская библиотека реализована на языках программирования JavaScript и TypeScript и предназначена для работы во всех современных

десктопных и мобильных браузерах (Google Chrome, Mozilla Firefox, Opera, Safari и др.).

4. Алгоритм работы системы

1. Клиент заходит в веб-приложение, в которое встроена клиентская библиотека ИППМ, при этом осуществляется загрузка и инициализация клиентской библиотеки.

2. При первом запуске клиентская библиотека записывает в память устройства уникальный идентификатор устройства, который затем используется для идентификации данного устройства ядром ИППМ.

3. В своем веб-приложении потребитель системы при необходимости самостоятельно реализует логику начала и окончания сессии (например, авторизация в системе или первый вход на страницу). В момент начала и окончания сессии веб-приложение сообщает об этих событиях ядру и клиентской библиотеке системы через их API.

4. При каждом обновлении страницы (запуске веб-приложения), в момент начала и окончания сессии, или с заданной потребителем системы периодичностью, или по определенным им событиям клиентская библиотека формирует и отправляет в ядро наборы данных, которые характеризуют клиентское устройство, его окружение и настройки (например, язык системы пользователя, локальное время и часовой пояс, тип и версия браузера, список поддерживаемых браузером функций, тип операционной системы, список установленных шрифтов и плагинов, параметры экрана и др.), веб-приложение (адрес страницы, список тэгов *style*, *script*, *iframe* и т.д.) и происходящие события (навигация, ввод данных, нажатия клавиш мыши, сворачивание вкладки и т.д.), и отправляет их в ядро системы для анализа. Каждый набор данных также включает уникальный идентификатор устройства, устойчивый отпечаток устройства, сформированный на основе собранных о нем данных, и уникальный идентификатор сессии (при наличии).

5. При поступлении наборов данных в ядро системы, собираются дополнительные сведения, в том числе о сетевом соединении, а затем вся совокупность данных обрабатывается в соответствии с набором установленных правил: проверяется их полнота и непротиворечивость, допустимость их изменений, значения параметров анализируются с целью оценки риска для потребителя системы. В случае, если какая-либо проверка не пройдена, система генерирует инцидент. Инциденты ранжируются по степени риска для потребителя системы и используются в расчете скоринга устройства и сессии.

6. С помощью механизма сессий осуществляется привязка известного веб-приложению клиента к данным, которые собираются клиентской библиотекой ПО.

7. Информационная система потребителя может получить скоринг клиентского устройства или сессии в режиме реального времени или ретроспективно с помощью серверного API, реализованного в ядре ИППМ. API реализован на базе протокола HTTP, для передачи данных используется формат JSON.

5. Примеры генерируемых инцидентов

В таблице 4.1 представлены некоторые инциденты, которые генерирует ядро ИППМ.

Таблица 4.1 – Некоторые из инцидентов ИППМ

Название	Описание
INCognito_MODE	Клиент включил режим инкогнито
NEW_CLIENT_DEVICE	Клиент впервые использует данное устройство
MULTIPLE_CLIENTS_ON_DEVICE	Устройство было использовано более чем одним клиентом
SIMULTANEOUS_CLIENT_SESSIONS	Клиент имеет более одной сессии одновременно
BAD_DEVICE_REPUTATION	Клиент использует устройство с плохой репутацией

6. Пример набора данных, генерируемого клиентской библиотекой

Далее представлен пример набора данных в формате JSON, генерируемого клиентской библиотекой и отправляемого для обработки в ядро.

```

    "Century Gothic",
    "Franklin Gothic",
    "Haettenschweiler",
    "Leelawadee",
    "Lucida Bright",
    "Lucida Sans",
    "MS Outlook",
    "MS Reference Specialty",
    "MS UI Gothic",
    "MT Extra",
    "Marlett",
    "Microsoft Uighur",
    "Monotype Corsiva",
    "Pristina",
    "Segoe UI Light"
],
"urlHref": "https://demobank.iidx.ru/session/self",
"scrH": 864,
"scrW": 1536,
"navUAg": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36"
}

```

7. Сторонние компоненты ПО и программные среды

Список используемых в составе ПО сторонних программных сред и программных средств с указанием условий их лицензирования и ссылки на них:

- Система контейнеризации docker – лицензия Apache 2.0 (<https://github.com/docker/engine/>);
- СУБД MariaDb – лицензия на свободное программное обеспечение GNU GPL 2 (<https://mariadb.com/kb/en/mariadb-licenses/>);
- Брокер сообщений Apache Kafka – лицензия на свободное программное обеспечение Apache Software Foundation (<https://github.com/apache/kafka/>);
- Программная среда ASP.NET Core – лицензия MIT (<https://github.com/dotnet/aspnetcore/>);
- Spring Framework – лицензия на свободное программное обеспечение Apache 2.0 (<https://github.com/spring-projects/spring-framework/>).

8. Дополнительные сведения

- Полное описание жизненного цикла ПО и его процессы приведены в документе «Описание процессов, обеспечивающих поддержание жизненного цикла программного обеспечения».

- Для установки ИППМ следует выполнить инструкцию из документа «Руководство по установке ИППМ».
- Для работы с ИППМ необходимо следовать инструкции, описанной в документе «Руководство по эксплуатации ИППМ».
- Пользователи с нарушением зрения должны также владеть навыками работы со специальными программами экранного доступа.