Инструкция по установке ПО

Системные требования

Минимальные системные требования следующие :

- процессор: Intel i3, i5. i7 не ниже 5-го поколения или его аналог;
- скорость локальной сети: не менее 10 Мбит/сек, скорость доступа в сети Интернет не менее 5 Мбит/сек.

Для обработки потока: до 100 событий в секунду пик, 20 событий в секунду в среднем и примерно 51 млн событий в месяц рекомендуется следующий объем ресурсов:

Сервер	CPU, ядра	ОЗУ, Гб	Диск, Гб
Шлюз	2	2	min
Сессионный мониторинг	8	64	64 SSD
Clickhouse	2	2	128

Поддерживаемые операционные системы для сервера сессионного мониторинга: Debian 11, Ubuntu 20.04 LTS, CentOS 7.9, Astra Linux 1.7, РЕД ОС 7.3 или новее.

Для nginx и Clickhouse - см. их документацию.

Электропитание ПК должно осуществляться от однофазной электрической сети переменного тока с заземлением и номинальным напряжением 220 В и частотой 50 Гц. Рекомендовано использование источника бесперебойного питания.

Процесс установки ПО

Этап 1. Получить учетные данные для доступа к среде по контактам Тел.: +7985 970 44 94 e-mail: Tsavina@technoscore.ru , или развернуть новую.

1. Подготовка инфраструктуры

Типовая инсталляция включает в себя 3 сервера и хранилище данных:

- Шлюз nginx. Находится в отдельной подсети (DMZ). Отвечает за приём запросов, терминацию TLS.
- Сервер сессионного мониторинга. Отвечает за обработку запросов от шлюза. Пишет данные в аналитическую БД и объектное хранилище.Clickhouse (опционально) и в хранилище данных (опционально).
- Сервер аналитики СУБД Clickhouse.
- Объектное хранилище данных (S3-совместимое) для хранения сырых данных..

После развертывания серверов необходимо:

- Добавить учетную запись администратора с входом по ssh key для корректной работы Ansible.
- Настроить DNS запись для шлюза nginx.

2. Установка компонент

Установка осуществляется автоматизированно с помощью Ansible.Перед установкой нужно создать создать конфигурационный файл заказчика (“team”) и инвентарь Ansible.

После создания указанных файлов можно выполнить запуск скрипта Ansible.Он включает в себя следующие роли:

- common – общая настройка сервера (ntp, сетевой экран, обновления и т.д.)
- gate – установка nginx, включая получение TLS сертификата.
- scoring – установка сервиса скоринга
- data_lake_sync – установка служебного компонента (python скрипт запускаемый по cron), обеспечивающего загрузку сырых данных в объектное хранилище.
- filebeat – установка служебного компонента, для отправки логов в ElasticSearch.
- clickhouse – установка Clickhouse.

Запуск: cbt_scoring your_config.yml

Рекомендуется запускать с помощью инструмента, типа supervisord или обернуть в системную службу.

Метрики мониторинга доступны по протоколу Prometheus.

Системные журналы: /var/log/scoring.out.log и /var/log/scoring.err.log

Этап 2. Встроить SDK с свое приложение/сайт (см Встраивание SDK).

Модуль безопасности (далее Модуль, SDK или скрипт) предназначен для сбора информации о состоянии устройства пользователя и о поведении пользователя в мобильном приложении или веб-сайте Заказчика. На основе собранной информации формируется цифровой отпечаток устройства и пользователя, которые используются сервисом фрод-мониторинга в процессе оценки риска операций в канале онлайн банка.

Чтобы интегрировать Ваш сайт/форму/мобильное приложение с Техноскор ИРИС, вам нужно:

1) Добавьте скрипт Техноскор ИРИС на свою страниц:

Для этого желательно встроить наш скрипт в ваше веб-приложение либо же разместить на вашем веб-сервере. В случае, если этого сделать затруднительно, вы можете использовать скрипт с нашего сервера, добавив эту строку в body своей страницы.

2) Инициализируйте модуль с помощью метода:

init({team: string, proxyAddress?: string, userId?: string}): string

Вы должны вызвать его после загрузки страницы (в обработчике window.onload, jQuery.ready и т. д.) или запуска мобильного приложения.

Функция init() принимает следующие аргументы:

- Team (обязательный). Идентификатор заказчика. Для получения, обратитесь в нашу службу поддержки (см Информация для контактов).
- ProxyAddress. Установите этот параметр в случае, для трансляции всего трафика скрипта через прокси-сервер.
- UserId. Вы можете прикрепить к сессии произвольный идентификатор текущего пользователя,чтобы наша система могла группировать сессии по пользователям. Если на момент запуска модуля пользователь еще не известен, он может быть задан после, при помощи метода setUserId().

Настоятельно рекомендуется использование прокси-сервера, расположенного на поддомене заказчика, т.к. это исключает вероятность блокировки данных различными анти-трекерами.

Функция init() возвращает идентификатор сеанса (sessionId). Вы должны передать это значение событию на ваш сервер, потому что это необходимо в API-интерфейсе server-to-server (нам нужно сопоставить вызов API и сбор информации на стороне клиента).

Этап 3. Настроить вызов API внутри своей системы (см Интеграция API сессионного мониторинга).

API на стороне пользователя

Иногда удобно получать сигналы об обнаружении угроз обратно на устройство пользователя. Например, вы хотите учесть эти сигналы в своем WAF или хотите передать их на свой сервер, не выполняя интеграцию API между серверами (например, если вы не можете легко изменить серверное приложение, потому что оно от стороннего поставщика).

В этом случае сервер ИРИС возвращает алерты обратно клиенту в зашифрованном виде. Шифрование важно, не рекомендуется расшифровывать оповещения на стороне клиента, поскольку пользовательское устройство ненадежно и уязвимо для атак. Возвращаются только алерты, а не вся запись сессии целиком, поскольку может быть смысл мгновенно реагировать на алерты.

Зашифрованный текст возвращается в cookie cbtmon. Шифрование: Aes256Gcm. Ключ шифрования: можно получить у нашей службы поддержки (см Информация для контактов).

Формат (в виде JSON):

version: String,

t_ms: u64,

session_id: String,

device_id: String,

alerts: Vec<Alert>,

API на стороне сервера

Наш API можно охарактеризовать как pull (т.е. вы сами инициируете взаимодействие), а также мы используем HTTPS, JSON, и RPC-подобные вызовы.

Аутентификация запросов. В заголовках должен быть передан JWT-токен: ‘Authentication: Bearer {JWT_TOKEN}’ (см https://jwt.io/introduction).

Алгоритм должны быть HS256.

Payload:

iss: {team}

sub: {service_account_name}

exp: {expiration unix time seconds}

Для подписи используйте хэш секретного ключа для вашего service_account. Свяжитесь с нашей службой поддержки во время интеграции для получения team, service_account, секретного ключа и его хэша (см Информация для контактов).

Замечание: не рекомендуется реализовывать генерацию JWT-токенов с нуля, вам лучше использовать готовую, хорошо протестированную библиотеку.

Замечание: вы можете использовать хэш секретного ключа напрямую, но если вы хотите вычислить его динамически из секретного ключа, то см Приложение 1 для описания алгоритма хеширования.

Обработка ошибок:

Коды ответов HTTP соответствуют их обычным значением. Также мы возвращаем описание ошибки в виде строки в text/plain, где это возможно.

Базовый endpoint: POST /api/v3/{method}

Методы API:

1. Метод: score_session

Описание: применяет механизмы обнаружения угроз и возвращает сессию вместе с алертами.

Необходимо передать session_id в теле запроса.

Также вы можете опционально передать значение user_id. Это позволит применить анализ поп пользователю, даже если user_id еще не был задан в SDK (например если пользователь еще не прошел процедуру аутентификации, но известен user_id с которым он пытается войти).

Также вы можете по желанию передать event_id: это может быть ID транзакции или некой другой операции внутри вашей системы. В рамках одной сессии можно передать любое количество различных event_id (но будет хранить максимум 32 последних).

2. Метод: score_ip

Описание: возвращает примерную геолокацию по ip, а также проверяет наличие угроз.

Необходимо передать ip в теле запроса.

Ответ:

Потенциальные угрозы перечисляются в поле tags. Варианты тегов:

- Drop - спам или просто нежелательный адрес;
- Anonymizer - любые средства анонимизации (прокси/впн/тор);
- Webserver - адрес принадлежит диапазонам облачных провайдеров.

Информация для контактов

Для контактов с командой разработчиков просьба обращаться по следующим контактам:
Тел.: +7985 970 44 94
e-mail: Tsavina@technoscore.ru
Фактический адрес разработчиков и тех поддержки:11113 Москва Вн.Тер. Муниципальный округ Перово Ш. Энтузиастов д 52 стр 32