Коробочная версия. Логирование событий безопасности

Настройка

Чтобы включить логирование событий безопасности, необходимо добавить в `conf.json` поле `eventLogPath` с указанием пути к файлу, куда будут записываться события безопасности. Пример:

```
...
"eventLogPath": "./events.log",
...
```

События записываются в указанный файл в виде строк в формате json. Пример записи:


```
{"name":"yougile-
events","hostname":"yougile.mysite.com","pid":91925,"level":30,"ip":"192.1
68.1.120","userName":"Иван
Петров","userEmail":"petrov@dev.com","userAgent":"Safari/537.36","clientVe
rsion":"37.5.15","action":"logout","targetType":"user","targetId":"petrov@
dev.com","targetName":"Иван
Петров","result":"accepted","msg":"","time":"2022-05-
25T11:31:37.530Z","v":0}
```


По умолчанию YouGile не будет осуществлять ротацию этого файла, чтобы включить ротацию, необходимо указать в `conf.json` в поле `eventLogDailyRotationLimit` количество файлов лога по дням, которые будут сохраняться (более старые будут удаляться). Пример:

```
...
"eventLogPath": "./events.log",
"eventLogDailyRotationLimit": 30,
...
```

– текущие логи при этом будут писаться в файл `event.log.0`, логи за предыдущий день будут доступны в файле `event.log.1` и т.д. В этом примере файлы логов будут храниться 30 дней.

Описание структуры записи лога

В каждой записи лога есть следующие поля:

Название поля	**Пример**	**Значение**
name	yougile-events	всегда содержит `yougile-events`, это позволяет идентифицировать записи этого лога в том случае если поток логирования этих событий объединён с какими-то другими логами
hostname	myserver.com	имя хоста, на котором запущен процесс YouGile
pid	12345	номер процесса YouGile
level	30	уровень записи, 30 соответствует уровню `info`, необходим для совместимости с форматом логирования bunyan
msg		текстовое описание события, обычно пустое, нужно для совместимости с форматом логирования bunyan
v	0	версия формата лога
time	2022-05-25T11:31:37.530Z	дата и время события в формате ISO 8601
ip	10.17.25.14	ip адрес клиента, инициировавшего событие безопасности, чтобы это поле сохранялось, необходимо записывать ip адрес в заголовок `x-real-ip` в прокси сервере, который находится между клиентом и сервисом YouGile (nginx, IIS, Apache, ...)
userAgent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.64 Safari/537.3	это поле идентифицирует тип и версию клиентского приложения, с которого было инициировано событие (для desktop-приложения, то эта строка будет содержать слово `Electron`, для мобильного – `Dart`)
clientVersion	37.5.15	версия клиентского приложения YouGile, с которого было инициировано событие

Название поля	Пример	Значение
userEmail	petrov@dev.com	email пользователя, который инициировал событие. Поле может быть пустым, если пользователь не авторизован
userName	Иван Петров	имя пользователя, инициировавщего событие
action	`login`	тип события, список событий и их описание см. в разделе Типы событий ниже
targetId	80eed1bd-eda3-4991-ac17-09d28566749d	уникальный id объекта события (инициатор события запрашивает этот доступ к этому объекту на чтение или запись). Это поле может быть пустым, если такого объекта нет для данного типа события, либо если таких объектов несколько (тогда информация об этих объектах записывается в поле `data`, см. ниже)
targetType	`project`	это тип объекта, который указан в `targetId`, возможные значения: `company` (компания), `user` (пользователь), `department` (отдел), `key` (ключ сессии), `project` (проект)
targetName	Разрабока	название объекта, который указан в `targetId`
companyId	a0a93ce4-5b0c-4c53-8200-3070690815e0	уникальный id компании, к которой имеет отношение событие, это поле может отсутствовать для некоторых типов событий, которые не относятся к какой-то отдельной компании
companyName	ООО Производство	название компании, которая указана в поле `companyId`, поле может отсутствовать
result	`accepted`	если сервис принял запрос и произвёл требуемое действие, это поле содержит строку `accepted`, иначе оно содержит строку `rejected`
rejectMessage	Not authorized	если поле `result` содержит `rejected`, то в этом поле содержится пояснение причины отказа выполнения запроса, иначе поле отсутствует
data	`{"details": "all companies"}`	дополнительные данные, которые нужны для некоторых типов событий (см. Типы событий), для других типов событий поле отсутствует

Типы событий

Тип события	Описание события	Особенности события
login	запрос на вход в YouGile и получение ключа сессии	тип события `key`, `targetId` отстутствует (инициатор всегда запрашивает ключ сессии для себя)
logout	запрос на выход из YouGile, ключ сессии при этом инвалидируется	это событие происходит только если в `conf.json` поле `keyExpireOnLogout` установлено в `true`
make-admin	запрос на предоставление статуса администратора в компании для некоторого пользователя	компания указана в `companyId` и `companyName`, пользователь, статус которого меняется указан в `targetId` (email) и `targetName`
make-not-admin	запрос на снятие статуса администратора в компании для некоторого пользователя	компания указана в `companyId` и `companyName`, пользователь, статус которого меняется указан в `targetId` (email) и `targetName`
add-to-company	добавление пользователя в компанию	компания указана в `companyId` и `companyName`, пользователь, статус которого меняется указан в `targetId` (email) и `targetName` (`targetName` может быть пустым, если добавляется новый, не зарегистрированный пользователь)
remove-from-company	удаление пользователя из компании или из всех компаний	компания указана в `companyId` и `companyName`, если пользователь удаляется из всех компаний, то компания не будет указана, вместо этого в поле `data` будет записано значение `{"details": "remove from all companies"}`. Пользователь, статус которого меняется указан в `targetId` (email) и `targetName`
create-company	создание компании	id и название компании указываются в `targetId` и `targetName`
add-user-to-project	добавление пользователя в проект	в `targetId` записывается проект, куда добавляется пользователь, в `data` записываются данные пользователя, которого добавляют (email и имя), также в `data` записывается id роли, с которой добавлен пользователь

Тип события	Описание события	Особенности события
remove-user-from-project	удаление пользователя из проекта	в `targetId` записывается проект, откуда удаляется пользователь, в `data` записываются данные пользователя, которого удаляют (email и имя)
change-role	изменение настроек ролей в проекте, добавление/ удаление ролей	в `targetId` записывается проект, в котором меняется роль, само изменение роли записывается в поле `data` – в этом поле записываются старые и новые значения настроек ролей
change-user-role	изменение роли пользователя в проекте	в `targetId` записывается проект, в котором меняются роли пользователей, в `data` записываются данные по изменению (старая и новая роль для каждого пользователя)
get-all-available-data	загрузка всех доступных пользователю данных по компаниям	если в поле `companyId` указана компания, то запрашиваются данные только по ней, если нет, то запрашиваются данные по всем компаниям, в которых состоит пользователь
get-updates	получение обновлений объектов компании	если в поле `companyId` указана компания, то запрашиваются обновления только по ней, если нет, то запрашиваются обновления по всем компаниям, в которых состоит пользователь. В поле `data` записывается список id всех объектов, по которым пользователь получил обновление в случае успешного выполнения запроса
change-preferences	изменение общих настроек компании	в `data` записываются данные старых и новых настроек
add-departnment	создание нового отдела	в `targetId` и `targetName` записывается id и имя нового отдела, в `data` записываются данные нового отдела
remove-departnment	удаление отдела	в `targetId` и `targetName` записывается id и имя удаляемого отдела
change-departnment	изменение параметров отдела	в `targetId` и `targetName` записывается id и имя отдела, в `data` записываются старые и новые данные отдела
change-password	изменение пароля пользователя	в `targetId` записывается пользователь, чей пароль изменяется

Чтение логов

Логи записываются в формате совместимом с `bunyan`, поэтому для удобного чтения логов можно использовать `bunyan-cli`: node-реализация, rust-реализация. В некоторых дистрибутивах `linux` bunyan включен в основной репозиторий пакетов.