

Корпоративный мессенджер VK Teams

Инструкция по настройке интеграции с DLP-
системой

Оглавление

Назначение документа	3
Дополнительная документация	3
Общее описание	4
SearchInform	6
InfoWatch	8
Настройка отправки данных в DLP-систему	11

Назначение документа

В данном документе представлено описание механизма отправки запросов в DLP-систему, а также процесс активации отправки данных в DLP-систему.

Документ предназначен для использования системными администраторами.

Дополнительная документация

Архитектура и описание системы — в документе представлено описание сервисов, обеспечивающих отставку данных в DLP-систему, и расположение log-файлов данных сервисов. Не является частью публичной документации, обратитесь к представителю VK Tech, чтобы ознакомиться с документом.

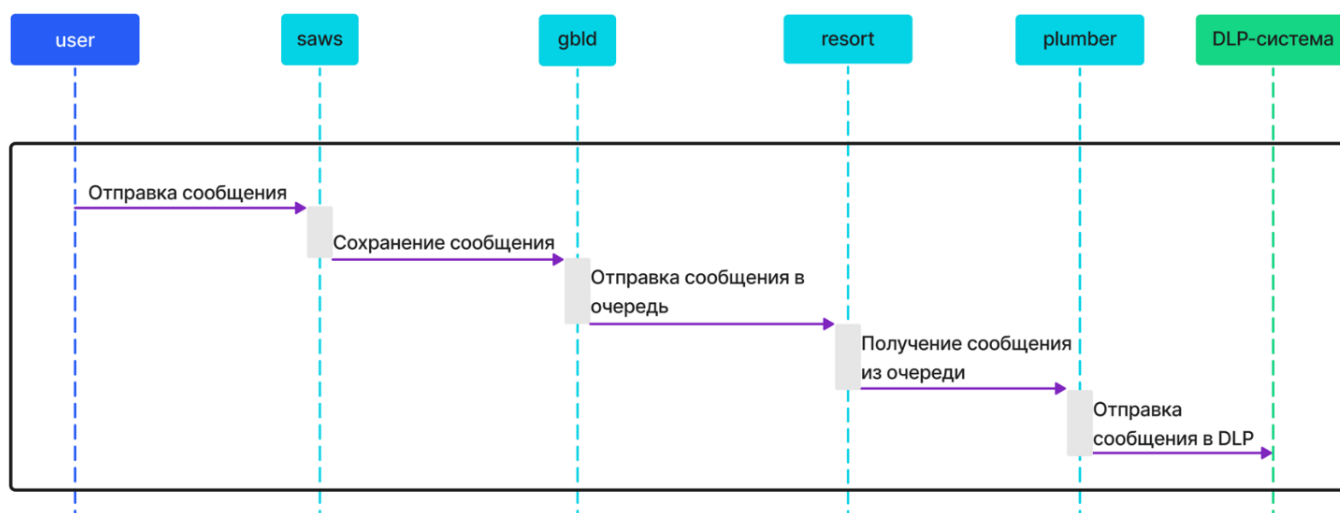
Общее описание

DLP-система — специализированное программное обеспечение, предназначенное для защиты компании от утечек информации. Со стороны VK Teams в DLP-систему отправляются запросы при отправке пользователями сообщений или загрузке файлов.

Есть два типа запросов, отправляемых в DLP-систему:

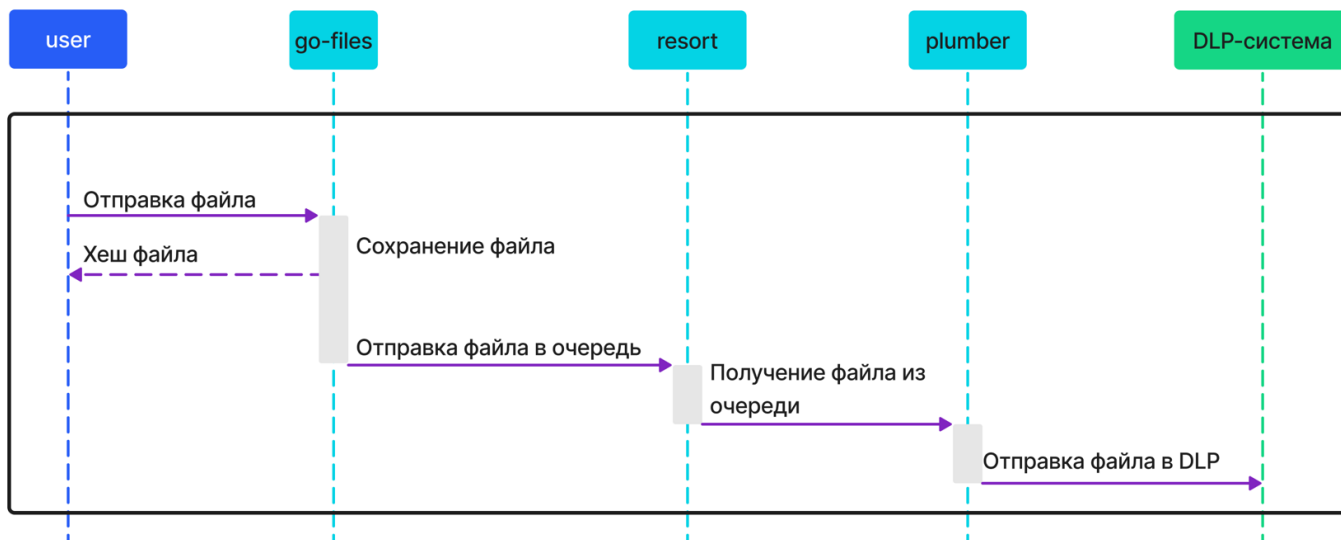
- message — обычное текстовое сообщение;
- file — файл, загруженный пользователем.

Ниже представлены схемы взаимодействия сервисов VK Teams при отправке сообщений/файлов во внешнюю DLP-систему заказчика.

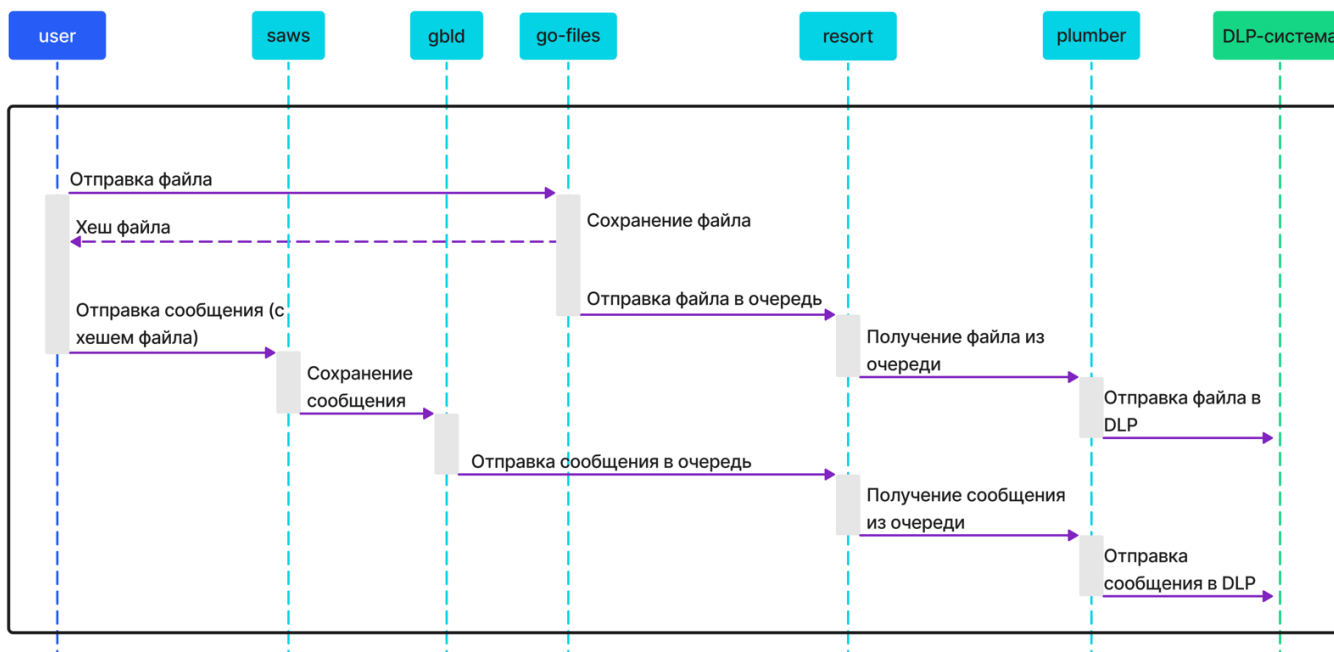


При отправке данных типа **message** (схема выше) сообщение пользователя сохраняется в хранилище сервиса Saws и в сервисе Gbld-mchat (для групповых чатов)/Gbld-st (для 1-to-1). Далее сообщение складывается в очередь для отправки в DLP-систему. Очередь реализована в сервисе Resort и обрабатывается сервисом Plumber.

Plumber — сервис отправки данных в DLP-систему, запускается в Kubernetes.



При отправке данных типа **file** (схема выше) файл пользователя сохраняется в сервисе Go-files. Далее сообщение складывается в очередь сервиса Resort и обрабатывается сервисом Plumber.



При отправке пользователем сообщения с прикрепленным файлом (**message + file**, схема выше) файл сохраняется в сервисе Go-files. Далее сервис Go-files возвращает хеш файла, после чего сообщение пользователя сохраняется в сервисах Saws и Gbld-mchat /Gbld-st с хешем файла и обрабатывается сервисами Resort и Plumber.

После отправки данных в DLP-систему сервис Plumber присваивает запросу статус в зависимости от результата отправки.

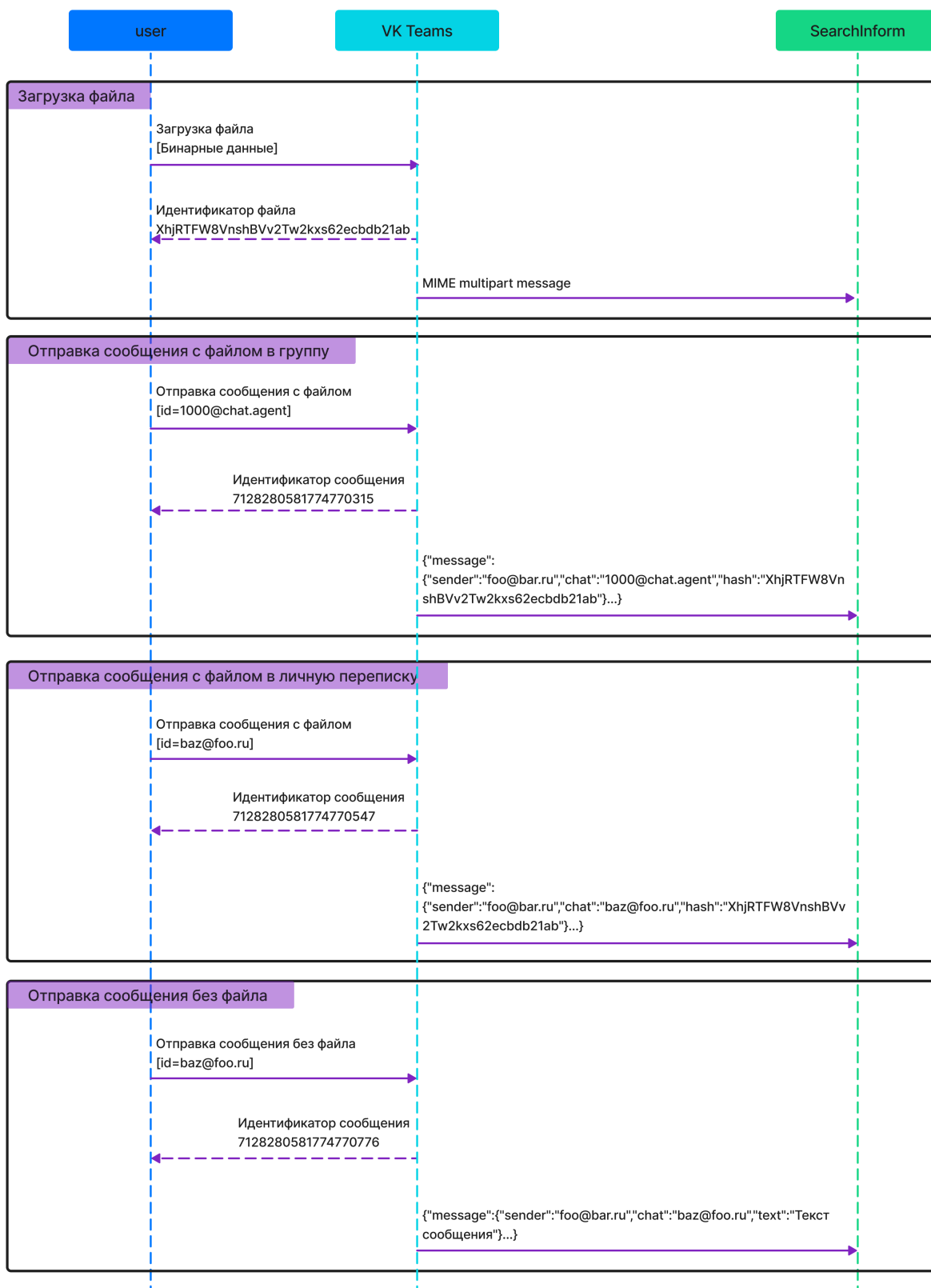
VK Teams поддерживает интеграцию со следующими поставщиками DLP-систем:

1. SearchInform
2. InfoWatch

Описание взаимодействия с DLP-системами и настройка интеграции представлены ниже.

SearchInform

Схема взаимодействия с DLP-системой компании SearchInform представлена ниже.



Передача данных осуществляется по протоколу ICAP.

При отправке данных типа **message** сервис Plumber формирует HTTP-запрос со следующими параметрами:

- "url" — http://vkteams/;
- тело запроса в json-формате. Пример:

```
{
  "sender" : "dev1@example.com", // отправитель сообщения
  "chat" : "123@chat.agent", // получатель сообщения (может быть как чатом, так и
идентификатором пользователя)
  "hash" : "23r2312fr3", // идентификатор файла, если был отправлен
  "text" : "Hello!", // текст сообщения
  "parts": "[{\"mediaType\": \"text\", \"text\": \"https://files-n-asemyonov.v3.im-
sandbox.devmail.ru/get/gPgp0YSoNTm1Rbn206aE4V6332d3281bb wef\", \"captionedContent\":
{\"caption\": \"wef\", \"url\": \"https://files-n-asemyonov.v3.im-sandbox.devmail.ru/
get/gPgp0YSoNTm1Rbn206aE4V6332d3281bb\"}}]", // системная информация о сообщении
  "ip": "192.168.1.1", // ip отправителя
  "mid": 12341, // идентификатор сообщения
}
```

- заголовок Content-Type — "application/json".

Далее полученный HTTP-запрос упаковывается внутрь ICAP ReqMod-запроса. Добавляются следующие заголовки:

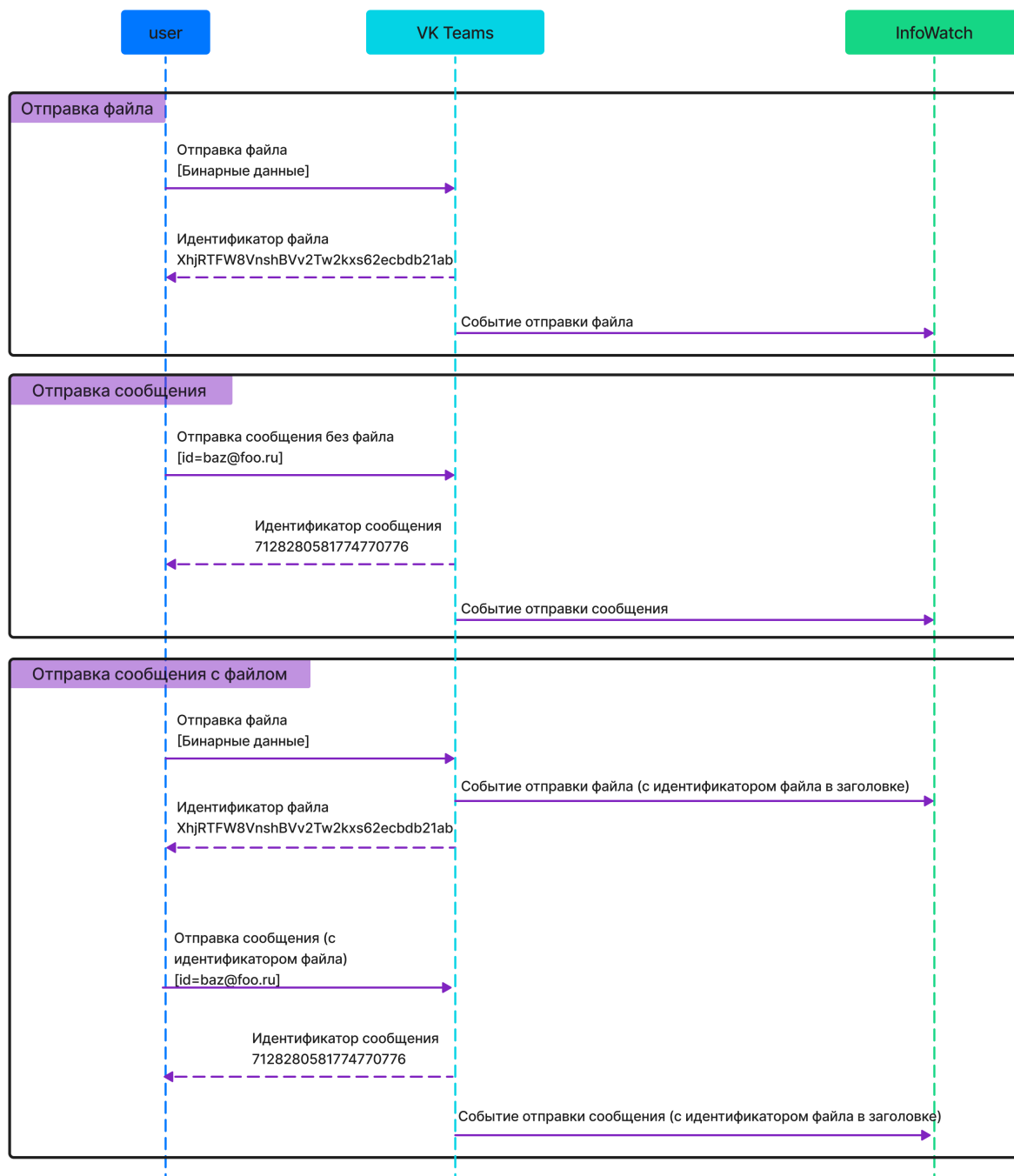
- X-Authenticated-User // отправитель сообщения
- X-Client-IP // ip адрес отправителя письма

Файлы отправляются в DLP-систему, когда файл загружен на сервер VK Teams. Запрос в DLP-систему отправляется в формате multipart. При отправке данных типа **file** сервис Plumber формирует HTTP-запрос со следующими параметрами:

- "file" — информация о содержимом файла:
 - заголовок Content-Disposition — "form-data; name="file"; filename="{имяФайла}";
 - заголовок Content-Type — "application/octet-stream";
 - содержимое файла.
- "hash" — информация об идентификаторе файла:
 - заголовок Content-Disposition — "form-data; name="hash";
 - hash файла.

InfoWatch

Схема взаимодействия с DLP-системой компании InfoWatch представлена ниже.



Передача данных в DLP-систему осуществляется через InfoWatch Traffic Monitor SDK методом pushAPI SDK.

Данные, отправляемые сервисом Plumber в DLP-систему InfoWatch, представлены в документации к продукту <https://kb.infowatch.com/pages/viewpage.action?pagelId=165545261>.

Связывание событий

DLP-система от InfoWatch использует плагины для подключения дополнительных перехватчиков. При сценарии «Отправка сообщения с файлом» необходимо связывать события общим признаком. Связь выставляется добавлением заголовка с хешем файла в каждое событие при данном сценарии.

Подробное описание представлено в документации к плагинам продукта по ссылке <https://kb.infowatch.com/pages/viewpage.action?pagelId=165546582>.

Для создания плагина необходимо создать файл **manifest.json**, содержащий информацию о плагине. Пример файла **manifest.json** представлен ниже:

```
{
  "PLUGIN_ID": "346227C2657C4701B86892CAE732805D",
  "DISPLAY_NAME": "Плагин для события мессенджера",
  "DESCRIPTION": {
    "eng": "IM events reception",
    "rus": "Прием событий менеджера"
  },
  "VERSION": "0.0.1",
  "VENDOR": "VKteams",
  "LICENSE": [
    {
      "PATH": "license/tm_license.license"
    }
  ],
  "PATTERN_SEARCH_LICENSE": {
    "operator": "and",
    "conditions": [
      {
        "common_name": "VKteams"
      },
      {
        "object_type": "im_VKteams"
      },
      {
        "protocol": "NONE"
      }
    ]
  },
  "ADDS_SERVICES": {
    "SERVICE_TYPE": [
      {
        "SERVICE_MNEMO": "im_VKteams",
        "DATA_CLASS": [
          "kChat", "kFileExchange"
        ],
        "ICON": "icon/acme_messenger.png",
        "LOCALE": {
          "rus": "Мессенджер VKteams",
          "eng": "VKteams messenger"
        },
        "CONTACT_TYPE": [
          {
            "MNEMO": "im_VKteams",
            "SCOPE": [
              "person"
            ],
          }
        ],
      }
    ],
  }
}
```

```
"ICON": "icon/acme_messenger.png",
"LOCALE": {
  "rus": "Аккаунт VKteams",
  "eng": "VKteams account"
}
}
]
}
],
"OBJECT_HEADER": [
{
  "NAME": "VKteams_file_hash_header",
  "NOTE": {
    "rus": "Хеш файла",
    "eng": "File hash"
  },
  "DATA_CLASS": ["kChat", "kFileExchange"],
  "USE_IN_POLICY": "1",
  "USE_IN_QUERY": "1",
  "USE_IN_NOTIFICATION": "1",
  "USE_IN_LIST": "1",
  "USE_IN_SHOW": "1",
  "USE_IN_DETAIL": "1",
  "TYPE": "string",
  "FORMAT": "string",
  "IS_MULTIPLE_VALUE": "1"
}
]
}
```

Настройка отправки данных в DLP-систему

Шаг 1. Активировать отправку сообщений в DLP-систему:

1. В конфигурационных файлах сервисов Gbld-st (для чата 1-to-1) и Gbld-mchat (для группового чата)

/usr/local/etc/gbld-st-1.conf

/usr/local/etc/gbld-mchat-1.conf

включить флаги (изменить или добавить строки):

```
gbld.check_dlp true
gbld.check_dlp_sync true
```

, где:

- `gbld.check_dlp` — отвечает за отправку сообщений в DLP-систему;
- `gbld.check_dlp_sync` — отвечает за синхронную отправку сообщений в сервис Resort. Если установлено значение `true` и сервис Resort не доступен, сообщения пользователей в чатах/группах отправляться не будут. Отправка сообщений в DLP-систему также не будет производиться.

2. Выполнить перезапуск сервисов Gbld-st, Gbld-mchat:

```
systemctl restart gbld-st-1 gbld-mchat-1
```

Шаг 2. Активировать отправку файлов в DLP-систему:

1. В конфигурационном файле сервиса Go-files **/usr/local/go.files.icq.com/files.icq.com.config.yaml** включить флаги (изменить или добавить строки) :

```
DLP:
  isActive: true
  syncSend: true
resortClient:
  timeout: "3s"
```

, где:

- `isActive` — отвечает за отправку сообщений в DLP-систему;
- `syncSend` — отвечает за синхронную отправку сообщений в сервис Resort.

2. Выполнить перезапуск сервиса Go-files_httpd:

```
systemctl restart gofiles_httpd
```

Шаг 3. Для работы функционала необходимо:

1. Указать в конфигурационном файле сервиса Plumber `/usr/local/etc/k8s/helmwave/projects/plumber/values/plumber.yml` значение секции **prefixes**:

```
etcd:  
  endpoints: {{.Release.Store.etcdEndpoints}}  
  prefixes: ["EXAMPLE"]  
  timeout: 5s
```

2. Установить значения по ключам в ETCD :

Пример команды:

```
etcdctl --endpoints=[ХОСТ_ETCD]:[ПОРТ_ETCD] put /vars/services/plumber/[PREFIXES]/public/  
service/DLP/{KEY} {VALUE}
```

, где:

- PREFIXES — значение соответствующей секции в конфигурационном файле сервиса Plumber `/usr/local/etc/k8s/helmwave/projects/plumber/values/plumber.yml`;
- KEY/VALUE — см. таблицы ниже:

key	value
DLP/system	Имя DLP-системы, например: icar или infowatch.
DLP/address	Endpoint DLP-сервера.

Данные ключи необходимо указать как при интеграции с DLP-системой SearchInform, так и с DLP-системой InfoWatch.

В случае настройки интеграции с DLP-системой InfoWatch также дополнительно необходимо указать следующие ключи:


key	value
DLP/token	Токен доступа. Используется в случае настройки интеграции с InfoWatch. Необходимо получить в админ-панели сервиса InfoWatch.
DLP/ company	Имя компании из лицензии. Используется в случае настройки интеграции с InfoWatch.
DLP/ imservice	Имя сервиса в плагине, указанное в manifest.json. Используется в случае настройки интеграции с InfoWatch.

Пример команды для DLP-системы SearchInform:

```
etcdctl --endpoints=etcd.im-etcd.svc.cluster.local:2379 put /vars/services/plumber/development/public/service/DLP/address icap:\\\\90.239.107.151:1344/request
```

Пример команды для DLP-системы InfoWatch:

```
etcdctl --endpoints=etcd.im-etcd.svc.cluster.local:2379 put /vars/services/plumber/development/public/service/DLP/address infowatch.beicq.net:9101
```

 **Внимание**

Переключение DLP- системы должно производиться при выключенном сервисе Plumber.

Шаг 4. Настроить конфигурацию сервиса Plumber:

При необходимости изменить настройки по умолчанию — в конфигурационном файле сервиса Plumber / **usr/local/etc/k8s/helmwave/projects/plumber/values/plumber.yml** изменить или добавить строки:

```
dlpClient:
  timeout: 5s // таймаут отправки данных в dlp
  debug: false //детализация логов
files:
  url: "files-c.<DOMAIN>" // адрес сервиса Go-files, откуда можно скачать отправляемый файл
  timeout: 5s // таймаут на загрузку файла
```

Шаг 5. Если при установке VK Teams не была произведена настройка отправки данных в DLP-систему, выполнить команду:

```
premssetup.py --no-init --skip-check --install -m helmwave
```

Дождаться завершения работы скрипта. По окончании процесса в консоли отобразится перечень установленных модулей. В случае возникновения ошибок процесс установки прервется, в консоли отобразится информация об ошибках.

Если вы производите процедуру включения впервые, то необходимо увеличить количество реплик:

```
kubectl -n vkteams scale deployment plumber --replicas=1
```

Дата обновления документа: 04.07.2024 г.