VK WorkSpace

Корпоративный мессенджер VK Teams

Инструкция по настройке интеграции с антивирусом

Оглавление

Назначение документа

В данном документе представлена архитектура и способы антивирусной проверки, а также ее конфигурирование.

Документ предназначен для использования системными администраторами.

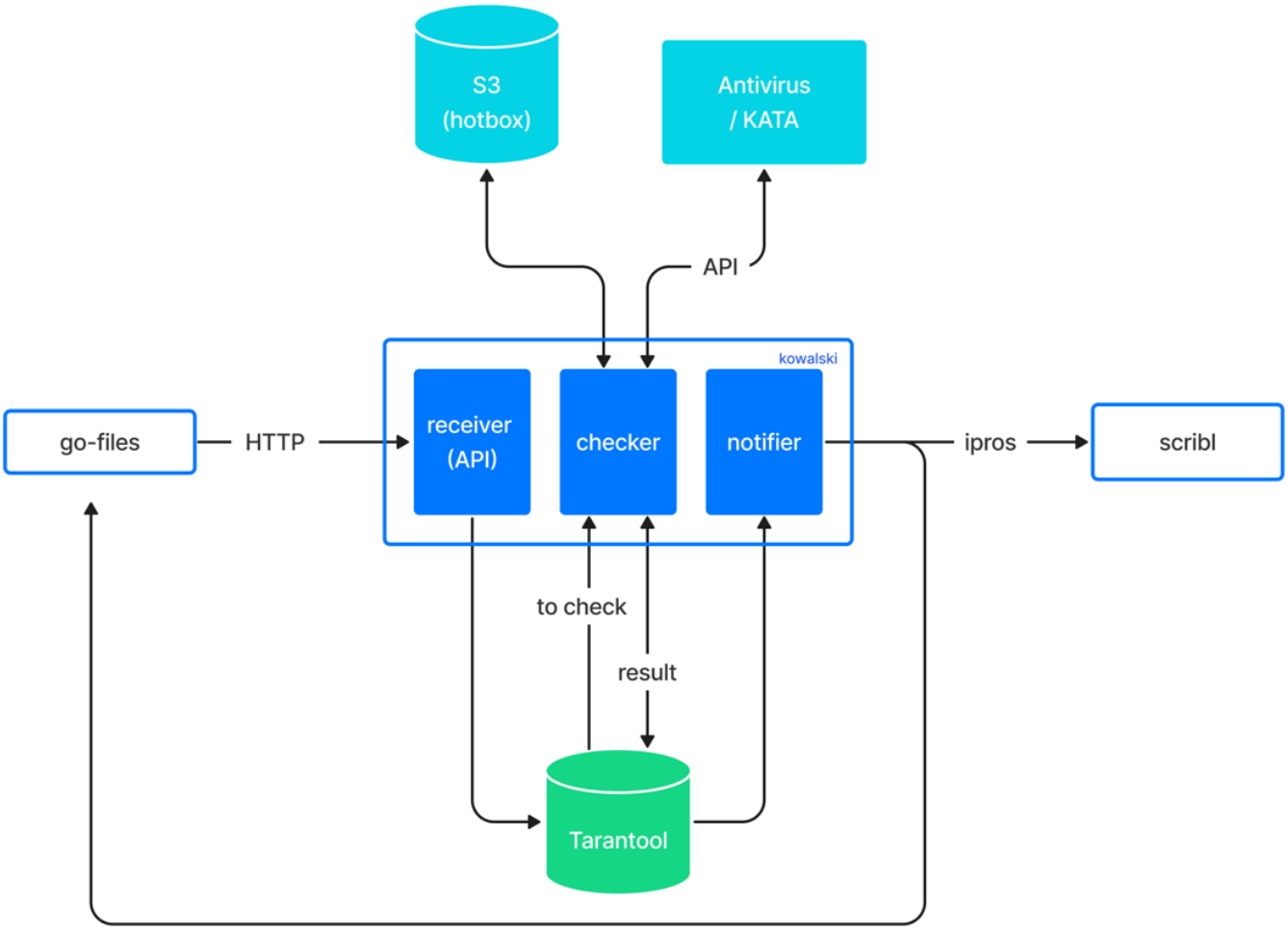
Дополнительная документация

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

Архитектура и способы антивирусной проверки

За проверку файлов антивирусом отвечает сервис Kowalski . Сервис работает на собственных локальных очередях и общается с файловым бэкендом ub.<DOMAIN> и универсальным сервисом подписок Scribl.

Архитектура:



После загрузки файла в сервис Go-files идёт HTTP-запрос в сервис Kowalski с информацией о файле (hotbox object ID, bucket name, fileID). Сервис Kowalski помещает этот файл к себе в очередь на проверку.

Способы антивирусной проверки

В данный момент представлено два вида проверки файлов:

1. Через протокол ICAP.
2. Через интеграцию с KATA (Kaspersky Anti Targeted Attack) по API.

Описание конфигурирования для каждого способа представлено ниже.

Конфигурирование с использованием ICAP

Для конфигурации сервиса в режиме проверки файлов по протоколу ICAP необходимо:

1. В конфигурационном файле сервиса Kowalski **/usr/local/etc/k8s/helmwave/projects/kowalski/values/kowalski.yml** в переменной `antivirusType` задать значение «ICAP»:

   ```
   antivirusType: "ICAP"
   ```

2. Задать таймаут для соединения:

   ```
   kowalski:
     antivirusType: "ICAP"
     icap:
       timeout: 5s
   ```

3. Задать параметр с адресом и портом ICAP антивируса в ETCD:

   ```
   etcdctl --endpoints=[ХОСТ_ETCD]:[ПОРТ_ETCD] put /vars/services/demo/development/public/
   service/antivirus/ICAP/address icap:\/\/ [DLP_SERVER_ENDPOINT]
   ```

   Пример:

   ```
   etcdctl --endpoints=etcd.im-etcd.svc.cluster.local:2379 put /vars/services/demo/
   development/public/service/antivirus/ICAP/address icap:\/\/90.230.107.150:1344/reqmod
   ```

4. Зафиксировать количество реплик сервиса Kowalski:

   ```
   kubectl -n kowalski scale deployment kowalski --replicas=1
   ```

Конфигурирование с использованием KATA

Файлы на проверку отправляются асинхронно, без ожидания результата проверки. Затем отправляется дополнительный запрос с информацией о проверке.

Информация о состоянии файла проверяется отдельной горутиной. Горутина делает запрос о состояниях с частотой, указанной в `kata: checkDelay`. После получения информации помещает файл в очередь на оповещение о состоянии проверки.

Чтобы начать взаимодействовать с KATA, необходимо в конфигурационном файле сервиса Kowalski **/usr/local/etc/k8s/helmwave/projects/kowalski/values/kowalski.yml**:

1. Установить для переменной `antivirusType` значение «KATA»:

```
kowalski:
  antivirusType: "KATA" // на латинской раскладке
```

2. Задать интервал проверки состояния файла в переменной `kata: checkDelay`, а также общий таймаут на запросы `kata: timeout` рекомендуемое значение 5 сек:

```
kata:
  checkDelay: 5s
  timeout: 5s
```

3. Путь до сертификата прописать в переменную `kata: certFilePath`. Наличие файла необязательно, его создаст сам сервис и поместит туда сгенерированный сертификат:

```
kata:
  certFilePath: "/data/kowalski"
```

4. Путь до системы с KATA прописать в переменную `kata: endpoint`:

```
kata:
  endpoint:"<https://EXAMPLE.ru>"
```

5. Чтобы сервис доверял серверу KATA, необходимо:

   - добавить его сертификат в доверенные сервису Kowalski. Для поля `enabled` установите значение true:

```
customCAcert:
  enabled: true
    secretName: custom-secret
    certName: my-cert.pem
```

   - положить сертификат сервера KATA или цепочку сертификатов в Kubernetes (файл **my-cert.pem** на машине). Пример, как получить только сертификат сервера KATA и положить его в файл:

```
openssl s_client -connect <ДОМЕННОЕ ИМЯ СЕРВЕРА KATA>:443 -showcerts < /dev/null 2>/
dev/null |openssl x509 -outform PEM > my-cert.pem
```

- отправить сертификаты в Kubernetes командой:

```
kubectl -n kowalski create secret generic custom-secret --from-file=my-cert.pem=my-
cert.pem
```

6. Если необходимо добавить домен для поиска, то используйте параметр `customDomainSearches`. Пример:

```
customDomainSearches:
    - EXAMPLE.com
```

7. Примените изменения конфигурации:

```
premsetup.py --install -m helmwave
```

8. Зафиксируйте количество реплик сервиса Kowalski:

```
kubectl -n kowalski scale deployment kowalski --replicas=1
```

Сервис отправляет следующие запросы к KATA:

- https://support.kaspersky.com/KATA/3.7.2/ru-RU/176838.htm
- https://support.kaspersky.com/KATA/3.7.2/ru-RU/176830.htm
- https://support.kaspersky.com/KATA/3.7.2/ru-RU/176836.htm

В случае неавторизованной системы KATA возвращает 401 (подробнее см. https://support.kaspersky.com/KATA/3.7.2/ru-RU/176825.htm). Если авторизация системы в сервисе KATA не происходит, то сервис об этом сообщает в логах и постоянно шлет запросы до успешной авторизации.

> **Внимание**
>
> Если поменять в системе sensorId (удалить или изменить файл), а сертификаты оставить старые, то любой запрос будет возвращать 400. Необходимо удалить сертификаты, чтобы избавиться от проблемы.

Подробная документация о взаимодействии с KATA по API доступна по ссылке: https://support.kaspersky.com/KATA/3.7.2/ru-RU/181505.htm

Место хранения отправляемых файлов

Файлы, отправляемые пользователями, хранятся в S3. При пересылке используется тот же файл, экземпляр файла не копируется и не пересоздается.

Режим проверки отправляемых файлов

Режим проверки отправляемых файлов антивирусом определяется конфигурационным файлом сервиса Go-files **/usr/local/go.files.icq.com/files.icq.com.config.yaml**

```
antivirus:
  work_mode: sync
```

В секции `antivirus:` необходимо поставить один из флагов:

- sync (препроверка) — файл нельзя скачать, если он не проверен или заражен.
- async (постпроверка) — файл нельзя скачать, если он заражен.

Дата обновления документа: 04.07.2024 г.