# Мессенджер и ВКС

## Настройки Мессенджера



Портал с документацией: https://biz.mail.ru/docs/on-premises/

# Оглавление

# Назначение документа

В инструкции описаны основные настройки Мессенджера в инсталляции. Документ предназначен для использования администраторами организации.

> **Примечание**
>
> Ранее Мессенджер и ВКС назывался Myteam, что находит отражение в технических моментах (например, команды в консоли).

# Разрешить изменение имени контакта

Чтобы настроить для пользователей возможность изменять имена контактов:

1. Добавьте в конфигурационный файл **/usr/local/nginx-im/html/myteam/myteam-config.json** параметр **allow-contacts-rename** и укажите для него флаг true или false:

   ```
   {
       },
       "allow-contacts-rename": true,
       ....
       },
   }
   ```

   где:

   - false — отключает возможность изменения имени контакта.
   - true — разрешает изменение имени контакта. Измененное имя будет видно только тому пользователю, который внес это изменение.

2. Для инсталляции на одну виртуальную машину выполните команду:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t godmod
   ```

   Для кластерной инсталляции:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true HELMWAVE_ENV_NAME=cluster hwup -t godmod
   ```

3. Перезапустите под в технологическое окно (может приводить к сбою в новых подключениях):

   ```
   kubectl delete pods -n vkteams -l app=myteam-admin
   ```

# Разрешить использование функции шумоподавления

Чтобы настроить для пользователей возможность применять функцию шумоподавления:

1. Добавьте в конфигурационный файл **/usr/local/nginx-im/html/myteam/myteam-config.json** параметр **voip-noise-suppress-enabled** и укажите для него флаг true или false:

   ```
   }
   "voip-noise-suppress-enabled":true,
   ....
   }
   ```

   где:

   - false — отключает возможность настройки шумоподавления.
   - true — включает возможность настройки шумоподавления. В настройках приложения **VK WorkSpace** в разделе **Звонки**, доступна настройка **Шумоподавление**: выключить, слабое, сильное.

2. Для инсталляции на одну виртуальную машину выполните команду:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t godmod
   ```

   Для кластерной инсталляции:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true HELMWAVE_ENV_NAME=cluster hwup -t godmod
   ```

3. Перезапустите под в технологическое окно (может приводить к сбою в новых подключениях):

   ```
   kubectl delete pods -n vkteams -l app=myteam-admin
   ```

# Настроить порядок отображения фамилии, имени и отчества

По умолчанию отображение фамилии, имени и отчества в инсталляциях следующее: имя, отчество, фамилия.

Чтобы изменить порядок отображения фамилии, имени и отчества в клиентском приложении:

**Шаг 1.** Добавьте в конфигурационный файл **/usr/local/nginx-im/html/myteam/myteam-config.json** в секцию myteam-config.json следующие настройки:

```
"leading-last-name": true,
"allow-contacts-rename": false
```

где:

- leading-last-name: false — отображает фамилию контакта в конце;
- leading-last-name: true — отображает фамилию контакта в начале.

**Шаг 2.** Для инсталляции на одну виртуальную машину выполните команду:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t godmod
```

Для кластерной инсталляции:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true HELMWAVE_ENV_NAME=cluster hwup -t godmod
```

**Шаг 3.** Перезапустите под в технологическое окно (может приводить к сбою в новых подключениях):

```
kubectl delete pods -n vkteams -l app=myteam-admin
```

**Шаг 4.** В конфигурационных файлах сервисов Prof-st, Front и Beagle добавьте настройку:

```
swap_person_first_last_name true
```

Расположение конфигурационных файлов:

/usr/local/etc/front-1.conf

/usr/local/etc/front-2.conf

/usr/local/etc/front-3.conf

/usr/local/etc/front-4.conf

/usr/local/etc/beagle-1.conf

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

Для инсталляций, где уже были созданы пользователи и эти пользователи имеют непустые контакт листы, выполните шаги, описанные ниже.

> **Примечание**
>
> Рекомендуется проводить в часы наименьшей активности пользователей, чтобы снизить нагрузку на сервера.

**Шаг 1.** Средствами виртуальной машины выполните бэкап/точку восстановления на случай сбоя.

**Шаг 2.** В каждом инстансе сервиса Cox:

1. Включите уровень логирования в сервисе Cox (main) >= 3 (INFO).

   Уровень логирования при старте сервиса задаётся аргументом сервиса `-l` :

   ```
   systemctl status cox-1
   ```

   Изменить уровень логирования в работающем сервисе можно через ввод в управляющий порт `set log_level 3` .

   Номер командного порта можно узнать из настроек сервиса в значении переменной `compot_bind` . Пример для конфигурационного файла **/usr/local/etc/cox-1.conf**:

   ```
   sudo grep compot_bind /usr/local/etc/cox-1.conf
   ```

   Подключиться к командному порту можно утилитой nc (netcat):

   ```
   nc 0.0.0.0 4221
   set log_level 3
   ```

2. В настройках сервиса Cox (main) присвойте значение переменной **cox.remove_buddy_aliases.dryrun false**.

   В работающем сервисе это значение меняется через введение в управляющий порт команды `set cox.remove_buddy_aliases.dryrun false` :

   ```
   nc 0.0.0.0 4221
   set cox.remove_buddy_aliases.dryrun false
   ```

3. В управляющий порт сервиса Cox (main) введите команду `remove_buddy_aliases` :

   ```
   nc 0.0.0.0 4221
   remove_buddy_aliases
   ```

4. Просмотрете логи сервиса Cox:

   ```
   tail -f /oap/icq/logs/cox-1.log
   ```

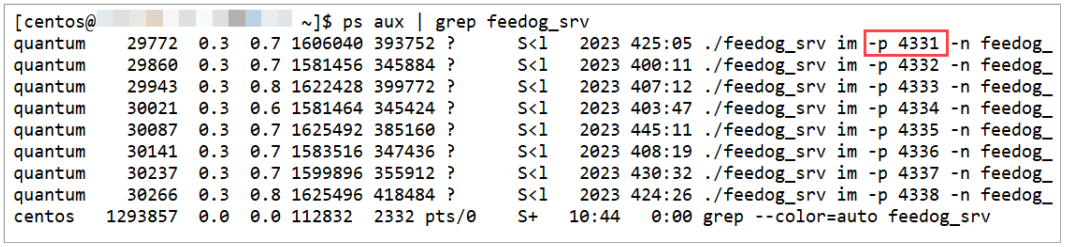
В логах сервиса ожидаются записи вида `Remove buddy aliases for sn`. Окончание операции логируется записью вида `BuddyAliasRemover finished`.

> **Важно**
>
> Не рекомендуется выполнять команды на нескольких инстансах одновременно. Это может привести к избыточной нагрузке.

**Шаг 3.** В каждом инстансе сервиса Feedog:

1. Определите управляющий порт инстанса командой `ps aux | grep feedog_srv`. Порт задаётся аргументом `-p`:

```
ps aux | grep feedog_srv
```



```
[centos@ ~]$ ps aux | grep feedog_srv
quantum    29772  0.3  0.7 1606040 393752 ?      S<l   2023 425:05 ./feedog_srv im -p 4331 -n feedog_
quantum    29860  0.3  0.7 1581456 345884 ?      S<l   2023 400:11 ./feedog_srv im -p 4332 -n feedog_
quantum    29943  0.3  0.8 1622428 399772 ?      S<l   2023 407:12 ./feedog_srv im -p 4333 -n feedog_
quantum    30021  0.3  0.6 1581464 345424 ?      S<l   2023 403:47 ./feedog_srv im -p 4334 -n feedog_
quantum    30087  0.3  0.7 1625492 385160 ?      S<l   2023 445:11 ./feedog_srv im -p 4335 -n feedog_
quantum    30141  0.3  0.7 1583516 347436 ?      S<l   2023 408:19 ./feedog_srv im -p 4336 -n feedog_
quantum    30237  0.3  0.7 1599896 355912 ?      S<l   2023 430:32 ./feedog_srv im -p 4337 -n feedog_
quantum    30266  0.3  0.8 1625496 418484 ?      S<l   2023 424:26 ./feedog_srv im -p 4338 -n feedog_
centos   1293857  0.0  0.0 112832  2332 pts/0    S+   10:44   0:00 grep --color=auto feedog_srv
```

2. Подключиться к управляющему порту можно с помощью утилиты `cpsh`.
3. Выполните `kill_bat_all -delay <delay_milliseconds>`. delay указывать исходя из нагрузки. Значение по умолчанию (если не указать ключ `-delay`) — 256:

```
cpsh 4331
im:feedog_srv-a01.1% kill_bat_all
```

4. Просмотрите логи сервиса Feedog (просмотр логов — **/oap/logs/feedog_srv-a01.<INSTANCE_NUMBER>.err**):

```
tail -f /oap/logs/feedog_srv-a01.1.err
```

Операция сопровождается логированием сообщений вида `BUCKY_DOMAIN: Dropping next bucket`. Прекращение логирования подобных сообщений соответствуют окончанию операции.

> **Важно**
>
> Не рекомендуется выполнять команды на нескольких инстансах одновременно. Это может привести к избыточной нагрузке.

**Шаг 4.** В каждом инстансе сервиса Boss:

1. Определите управляющий порт инстанса командой `ps aux | grep bos_srv`. Порт задаётся аргументом `-p`:

```
ps aux | grep bos_srv
```

```
[centos@ ~]$ ps aux | grep bos_srv
centos   1097888  0.0  0.0 112828  2328 pts/0    S+   09:54   0:00 grep --color=auto bos_srv
root     2472058  0.0  0.0 241456  7088 pts/2    S+    2023   0:00 /usr/bin/sudo -E env LD_LIBRARY_PA
root     2472060  0.0  0.0   4376  1380 pts/2    S+    2023   0:00 scl enable devtoolset-10  'tail' '
root     2472082  0.0  0.0 108112   720 pts/2    S+    2023   0:00 tail -f /oap/icq/logs/bos_srv-a01.
quantum  2907004  0.7  0.8 2476364 425924 ?      S<l   2023 702:23 ./bos_srv im -p 4330 -n bos_srv-a0
quantum  2907006  0.7  0.8 2444616 444080 ?      S<l   2023 721:05 ./bos_srv im -p 4323 -n bos_srv-a0
quantum  2907036  0.7  0.9 2495816 453940 ?      S<l   2023 747:30 ./bos_srv im -p 4329 -n bos_srv-a0
quantum  2907050  0.6  0.8 2521412 425448 ?      S<l   2023 680:12 ./bos_srv im -p 4328 -n bos_srv-a0
quantum  2907075  0.6  0.8 2470212 424704 ?      S<l   2023 694:07 ./bos_srv im -p 4327 -n bos_srv-a0
quantum  2907083  0.7  0.8 2601288 424712 ?      S<l   2023 695:29 ./bos_srv im -p 4325 -n bos_srv-a0
quantum  2907093  0.6  0.8 2520388 424652 ?      S<l   2023 678:00 ./bos_srv im -p 4326 -n bos_srv-a0
quantum  2907103  0.6  0.8 2520388 424588 ?      S<l   2023 690:51 ./bos_srv im -p 4324 -n bos_srv-a0
```

2. Подключиться к управляющему порту можно с помощью утилиты `cpsh`.
3. Выполните `kill_bat_all -delay <delay_milliseconds>`. delay указывать исходя из нагрузки. Значение по умолчанию (если не указать ключ `-delay`) — 256:

```
cpsh 4323
im:bos_srv-a01.1% kill_bat_all -delay 1000
```

4. Просмотрите логи сервиса Boss (просмотр логов — /oap/icq/logs/bos_srv-a01.<INSTANCE_NUMBER>.err.log ):

```
tail -f /oap/icq/logs/bos_srv-a01.1.err.log
```

Операция сопровождается логированием сообщений вида `BUCKY_DOMAIN: Dropping next bucket`. Прекращение логирования подобных сообщений соответствуют окончанию операции.

> **Важно**
>
> Не рекомендуется выполнять команды на нескольких инстансах одновременно. Это может привести к избыточной нагрузке.

**Шаг 5.** Верните в исходное состояние уровень логирования в сервисе Cox.

# Настроить витрину чатов

В витрине отображаются чаты в зависимости от региона пользователя, который вычисляется по геолокации IP-адресов.

Для конфигурации витрины чатов необходимо получить доступ по SSH к машине, на которой запущен сервис Chatexpo:

1. Проверить, что подключились к машине, на которой запущен сервис Chatexpo:

```
pgrep -a chatexpo
6442 /usr/local/bin/chatexpo -c /usr/local/etc/chatexpo-1.conf -l 2 -o /oap/icq/logs/
chatexpo-1.log
```

2. Получить доступ к командному порту сервиса Chatexpo:

```
sudo rg compot /usr/local/etc/chatexpo-1.conf
49:# compot
50:compot_bind 127.0.0.1:4523
```

3. Подключиться к командному порту:

```
rlwrap nc 127.0.0.1 4523
```

Далее можно приступать к конфигурации витрины чатов.

# Добавить чат в витрину

1. Для добавления чата в витрину выполнить:

```
app.tnt.region_chats.add RU 1@chat.agent 1
status=0, reason=ok
```

2. Проверить в клиентском приложении, что чат добавлен.

# Удалить чат из витрины

1. Для удаления чата из витрины выполнить:

```
app.tnt.region_chats.remove RU 1@chat.agent
status=0, reason=ok
```

2. Проверить в клиентском приложении, что чат удален.

# Сменить порядок чатов в витрине

Наиболее простой путь изменение порядка чатов в витрине — удаление чатов из витрины и добавление заново в необходимом порядке.

Предположим, в витрине отображаются 3 чата, например:

```
app.tnt.region_chats.add RU 1@chat.agent 1
status=0, reason=ok
app.tnt.region_chats.add RU 14@chat.agent 2
status=0, reason=ok
app.tnt.region_chats.add RU 26@chat.agent 3
status=0, reason=ok
```

и необходимо поставить третий чат на первое место, первый на второе и второй на третье.

Для этого необходимо:

1. Удалить первый и третий чаты:

```
app.tnt.region_chats.remove RU 1@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 26@chat.agent
status=0, reason=ok
```

2. Добавить удаленные чаты заново в нужном порядке:

```
app.tnt.region_chats.add RU 26@chat.agent 1
status=0, reason=ok
app.tnt.region_chats.add RU 1@chat.agent 2
status=0, reason=ok
```

3. Проверить, что чаты добавились:

```
app.tnt.region_chats.list RU
status=0, reason=ok
{"id":"26@chat.agent","pos":1}
{"id":"1@chat.agent","pos":2}
{"id":"14@chat.agent","pos":3}
```

   Второй чат сам сдвинется на третье место.

4. Проверить порядок чатов в клиентском приложении.

# Удалить всю витрину

1. Проверить, какие чаты добавлены в витрину:

```
app.tnt.region_chats.list RU
status=0, reason=ok
{"id":"26@chat.agent","pos":1}
{"id":"1@chat.agent","pos":2}
{"id":"14@chat.agent","pos":3}
{"id":"2@chat.agent","pos":5}
{"id":"3@chat.agent","pos":6}
{"id":"1586@chat.agent","pos":7}
{"id":"1585@chat.agent","pos":8}
{"id":"1587@chat.agent","pos":9}
{"id":"1787@chat.agent","pos":10}
```

2. Удалить чаты по одному:

```
app.tnt.region_chats.remove RU 26@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 1@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 14@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 2@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 3@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 1586@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 1587@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 1787@chat.agent
status=0, reason=ok
app.tnt.region_chats.remove RU 1585@chat.agent
status=0, reason=ok
app.tnt.region_chats.list RU
status=0, reason=ok
```

3. Проверить в клиентском приложении, что витрина удалена.

# Включить функциональность папок в Мессенджере

Максимальное количество папок по умолчанию — 10. Папки синхронизируются между всеми активными сессиями и платформами.

Чтобы включить отображение папок в клиентском приложении, необходимо:

1. Укажите в конфигурационном файле **/usr/local/nginx-im/html/myteam/myteam-config.json** значение true для секции **folders-enabled**:

   ```
   {
     "api-version": 112,
     "archive-enabled": true,
     "folders-enabled": true,
     //
   }
   ```

2. Для инсталляции на одну виртуальную машину выполните команду:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t godmod
   ```

   Для кластерной инсталляции:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true HELMWAVE_ENV_NAME=cluster hwup -t godmod
   ```

3. Перезапустите под в технологическое окно (может приводить к сбою в новых подключениях):

   ```
   kubectl delete pods -n vkteams -l app=myteam-admin
   ```

# Включить функциональность Retention Policy

Функционал Retention Policy позволяет удалять файлы из хранилища по истечении времени или при условии блокировки системой DLP.

> **Важно**
>
> Функционал Retention Policy доступен с версии 26.1 Мессенджера. Новые условия автоудаления действуют только на файлы, загруженные после изменения условий. К файлам, загруженным ранее, сохраняются условия, которые были установлены в момент их загрузки.

# Автоудаление из хранилища по заданному сроку

Чтобы включить автоудаление по заданному времени из хранилища:

1. Укажите в конфигурационном файле **/usr/local/go.files.icq.com/retention_policy.yaml** следующие параметры в секции **retention_policy** :

```yaml
retention_policy:
  enabled: false
  expiration_check_period: 1m
  expiration_check_files_limit: 100
  expiration_sleep_between_query: 10s
  retention_default_period: 10y
  ext_retention:
    - extension: pdf
      retention_period: 90d
    - extension: txt
      retention_period: 10d
  size_retention:
    - size: "1Mb"
      retention_period: 30d
    - size: "500Mb"
      retention_period: 45d
```

где:

- enabled – параметр включения функциональности (false – выключена, true – включена).
- expiration_check_period – периодичность проверки файлов с истекшим сроком хранения.
- expiration_check_files_limit – количество получаемых файлов за один запрос к базе данных (не рекомендуется ставить менее 100 и более 10000).
- expiration_sleep_between_query – время между запросами в базу данных.
- retention_default_period – срок хранения файлов по умолчанию.
- блок `ext_retention` – определяет периодичность удаления форматов файлов (может быть пустым):
  - extension – расширение файла.
  - retention_period – время, через которое файл будет удален.
- блок `size_retention` – определяет размеры файлов, при достижении которых они будут удалены (может быть пустым):
  - size – размер файла.
  - retention_period – время, через которое файл будет удален.

Правила применения условий:

- к файлу применяется условие хранения, когда файл больше или равен указанному параметру: к файлу размером 100 Мб будет применен параметр 30 дней, а к файлу с размером 600 Мб – 45 дней.
- если файл попадает под условия хранения форматов и размеров, то применяется наименьшее условие: PDF файл размером 3 Мб будет храниться 30 дней.

- если файл не попадает ни под одно из условий, то к нему применяется параметр retention_default_period: файл MP3 размером 200 Кб не попадает ни под одно из указанных условий.

2. Перезапустите сервис **gofiles**:

```
sudo systemctl restart gofiles_httpd
sudo systemctl restart files_mq_consumer
```

## Автоудаление файлов при их блокировке DLP-системой

Чтобы влючить автоудаление файлов по заданному времени из хранилища при их блокировке DLP-системой:

1. Добавьте секцию **dlp_retention_policy** в конфигурационный файл **/usr/local/go.files.icq.com/retention_policy.yaml** со следующими параметрами:

```
dlp_retention_policy:
  enabled: false
  blocked: 1000y
```

где:

- enabled – параметр включения функциональности (false – выключена, true – включена).
- blocked – период, через который файл будет удалён из хранилища.

2. Перезапустите сервис **gofiles**:

```
sudo systemctl restart gofiles_httpd
sudo systemctl restart files_mq_consumer
```

# Включить Security-функции

При использовании security-функций Мессенджера, вы можете:

- Настраивать блокировку скачивания файлов по IP-адресу, операционной системе, устройству и браузеру.
- Ограничить отправку сообщений в Мессенджере по:
  - расширению файла и его размеру;
  - расширению файла без размера;
  - размеру файла для всех расширений.

При включении security-функций проверяются:

- личные чаты;
- групповые чаты;
- сообщения в обсуждениях;
- пересылаемые сообщения;
- сообщения в избранном;
- запланированные сообщения;
- изменение отправленного сообщения;
- ответ на полное сообщение;
- ответ на часть сообщения.

Если сообщение содержит файл, то у получателей нет к нему доступа, пока не придет положительный результат security-проверки. Если результат проверки отрицательный, файл будет заблокирован.

Если сообщение содержит и текст и файл, где файл заблокирован, то пользователи не имеют доступ к файлу, но текст всегда доступен.

## Логика работы вместе с отправкой данных в DLP-систему

Если одновременно настроены security-функции Мессенджера и отправка данных в DLP-систему, то сначала перед отправкой в DLP проверяются условия настроек security-функций.

Пример: включен режим отправки контента сообщений в DLP, и включена настройка в security-функциях на запрет отправки сообщений с файлами с расширением *.exe.

Действие системы: сначала осуществляется проверка настройки в security-функциях. В приведённом примере расширение файла ограничено для передачи и контент (с текстом или без) не отправляется в DLP-систему. Доступ к файлу блокируется. Если бы условия не противоречили security- настройкам, файл отправился бы в DLP-систему и был бы доступен в случае положительного ответа от DLP.

## Ограничить отправку файлов в зависимости от размера и расширения

Ниже описаны настройки в случае, если вам НЕ нужно отправлять файлы и текстовые сообщения в DLP-систему после security-проверок.

## Шаг 1. Активируйте сервис Vahter

Сервис находится в инсталляции в неактивном состоянии. Чтобы активировать сервис:

1. Перейдите в конфигурационный файл **/usr/local/etc/k8s/helmwave/projects.yml** и удалите из секции **disabled** сервис Vahter и строку **strimzi-kafka-operator**;
2. Запустите деплой сервиса Vahter.

   Для инсталляции на одну виртуальную машину выполните команду:

   `HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t vahter`

   Для кластерной инсталляции:

   ```
   HELMWAVE_ENV_NAME=cluster HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t vahter
   ```

3. Проверьте, что сервис Vahter запустился:

   ```
   kubectl -n vkteams get po | grep vahter
   ```

   В выводе консоли pod должен находиться в статусе Running.

## Шаг 2. Настройте конфигурацию адаптера

Перейдите в конфигурационный файл **/usr/local/etc/k8s/helmwave/store/dlp.yml** и заполните секцию **adapters**:

```
vahter:
  adapters:
  - provider: noopadapter
    name: adapter_1
    enabled: true
    always_return: ok
    default_access_level: Internal
    callback_name: multifora-callback
  adapters_manager:
    default_adapter_name: adapter_1
```

где:

- name — наименование адаптера на латинице;
- enabled — если значение true – включены security-проверки, если false – выключены;
- always_return — укажите «ok»;
- default_access_level — укажите «Internal».
- default_adapter_name — адаптер, используемый по умолчанию, если у вас несколько адаптеров.

## Шаг 3. Настройте маршрутизацию по доменам

Данный шаг является опциональным. Вы можете настроить выбор адаптера в зависимости от домена отправителя файла.

В конфигурационном файле **/usr/local/etc/k8s/helmwave/store/dlp.yml** заполните секцию **adapter_chooser**:

```
adapter_chooser:
  enabled: false
  adapters_rules:
  - target: "adapter_1" # адаптер, в который будут направляться сообщения от domain_list
    domain_list:
    - "domain_1"
    - "domain_2"
  - target: "adapter_2"
    domain_list:
    - "domain_3"
    - "domain_4"
```

1. Установите для параметра **enabled** значение true. Если false (конфигурация по умолчанию) — для всех доменов будет использоваться адаптер из параметра **default_adapter_name** конфигурационного файла **/usr/local/etc/k8s/helmwave/store/dlp.yml**.
2. В секции **adapters_rules** укажите правила сопоставления домена и адаптера. Укажите для параметров **target** нужный адаптер и список доменов, для которых этот адаптер будет использоваться.

## Шаг 4. Запретите отправку файлов в зависимости от типа и расширения

Правила безопасности позволяют ограничить отправку файла в зависимости от его разрешения и размера.

**Если вы настроили маршрутизацию по доменам:**

В конфигурационном файле **/usr/local/etc/k8s/helmwave/store/dlp.yml** укажите список исключений в рамках правила выбора адаптера:

```
adapter_chooser:
  enabled: true             # вкл/выкл функциональности.
  adapters_rules:           # правила сопоставления домена и адаптера
  - target: "adapter_1"
    domain_list:
    - "domain_1"
    excluded_users: # добавленная опциональная секция
    - "i.ivanov@domain_1"
    file_blocking_rules:
     - block_types: ['image/png','image/webp']
       block_by_size_limit: 400kb
     - block_types: ['image']
       block_by_size_limit: 10mb
```

```
    - block_types: ['__others__'] # правило для применения ко всем остальным типам файлов
      block_by_size_limit: 4gb
```

где:

- excluded_users — список пользователей, исключенных из проверок. Файлы, отправляемые такими пользователями, не будут проходить проверку и не будут блокироваться.
- block_types — типы файлов и их расширение. Список расширений работает, как черный список, т.е. если расширение не указано, считается, что оно в белом списке.
- block_by_size_limit — размер файла, может быть указан в килобайтах, мегабайтах, гигабайтах.

К файлу применяется ограничение, соответствующее наиболее детализированного типу из заданных. В примере выше к файлу типа png применится ограничение на 400 КБ, к файлу типа jpeg — ограничение на 10 МБ, ко всем остальным типам — ограничение на 4 ГБ.

**Если вы НЕ настраивали маршрутизацию по доменам:**

В конфигурационном файле **/usr/local/etc/k8s/helmwave/store/dlp.yml** укажите список исключений в настройках адаптера по умолчанию:

```
adapters_manager:
  icap_debug: false
  default_adapter_name: adapter_1
  excluded_users:
    - "i.ivanov@domain.example"
    - "p.petrov@domain.example"
  file_blocking_rules:
   - block_types: ['image/png','image/webp']
     block_by_size_limit: 400kb
   - block_types: ['image']
     block_by_size_limit: 10mb
   - block_types: ['__others__'] # правило для применения ко всем остальным типам файлов
     block_by_size_limit: 4gb
```

где:

- excluded_users — список пользователей, исключенных из проверок. Файлы, отправляемые такими пользователями, не будут проходить проверку и не будут блокироваться.
- block_types — типы файлов и их расширение. Список расширений работает, как черный список, т.е. если расширение не указано, считается, что оно в белом списке.
- block_by_size_limit — размер файла, может быть указан в килобайтах, мегабайтах, гигабайтах.

К файлу применяется ограничение, соответствующее наиболее детализированного типу из заданных. В примере выше к файлу типа png применится ограничение на 400 КБ, к файлу типа jpeg — ограничение на 10 МБ, ко всем остальным типам — ограничение на 4 ГБ.

## Шаг 5. Настройте сервис Go-files

Перейдите в конфигурационный файл сервиса Go-files **/usr/local/go.files.icq.com/files.icq.com.config.yaml** и в секции **DLPv2** укажите для параметров **dlp_checks_enable** и **dlp_whitelisting_checks_enable** значения **true**:

```
DLPv2:
    dlp_checks_timeout: 20m
    dlp_access_levels_checks_enable: false
    dlp_whitelisting_checks_enable: true
    dlp_checks_enable: true
    dlp_status_on_timeout: "Outdated"
    dlp_access_level_on_timeout: "External"
```

Перезапустите сервис Go-files командой:

```
sudo systemctl restart gofiles_httpd
```

## Шаг 6. Примените изменения

Для инсталляции на одну виртуальную машину выполните команду:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t vahter
```

Для кластерной инсталляции:

```
HELMWAVE_ENV_NAME=cluster HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t vahter
```

# Заблокировать скачивание файлов

## Шаг 1. Активируйте сервис Watchman

Сервис находится в инсталляции в неактивном состоянии. Чтобы активировать сервис:

1. Перейдите в конфигурационный файл **/usr/local/etc/k8s/helmwave/projects.yml** и установите в поле **watchmanEnabled** значение **true**.
2. Запустите деплой сервиса Watchman.

   Для инсталляции на одну виртуальную машину выполните команду:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t istio-ingress
   ```

   Для кластерной инсталляции:

   ```
   HELMWAVE_ENV_NAME=cluster HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t istio-ingress
   ```

3. Проверьте, что сервис Watchman запустился:

   ```
   kubectl -n istio-ingress get po | grep istio-ingress
   ```

   В выводе консоли pod должен находиться в статусе Running.
4. Проверьте логи сервиса на возможные проблемы:

```
sudo kubectl logs -n istio-ingress $(sudo kubectl get pod -n istio-ingress | grep istio-ingress | awk '{print $1}')
```

Искать необходимо по названию модуля Watchman. Пример проблем в логах:

```
2024-11-21T13:52:28.946719Z error   envoy golang external/envoy/contrib/golang/common/log/cgo.cc:24 failed to parse golang plugin config: ...
2024-11-21T13:52:28.946961Z warning envoy config external/envoy/source/extensions/config_subscription/grpc/delta_subscription_state.cc:269  delta config for type.googleapis.com/envoy.config.listener.v3.Listener rejected: Error adding/updating listener(s) 0.0.0.0_80: golang filter failed to parse plugin config: watchman /var/lib/im/modules/watchman.so
```

Подобная запись в логах говорит о проблемах в конфигурации — необходимо перепроверить конфигурацию или вернуть конфиг к значению по умолчанию.

## Шаг 2. Настройте запрет скачивания файлов

Правила настраиваются в конфигурационном файле **/usr/local/etc/k8s/helmwave/store/dlp.yml** в секции **watchmanAccessLevels.** Пример конфигурации с правилами:

```yaml
watchmanWhitelistingRules:
- subnets:
  - 192.168.0.1/24
  Device: []
  OS:
  - IOS
  Browser: []
- subnets: []
  Device: []
  OS: []
  Browser:
  - Opera
```

Каждое правило состоит из секций селекторов:

- subnets.
- Device.
- OS.
- Browser.

Значения в секциях селекторов не чувствительны к регистру. Возможные значения селекторов:

Device:

- mobile
- desktop
- web

Browser (настраивается для веб-приложений):

- chrome
- firefox
- opera
- ie
- safari
- edge
- yandex

OS (настраивается для веб-приложений):

- android
- windows
- macos
- ios
- linux

Для декстоп-устройств не поддерживается определение операционных систем.

Для мобильных устройств поддерживается определение iOS и Android.

Если секция отсутствует или пуста, проверка на соответствующий признак не будет осуществляться в рамках правила (пример: пустая секция subnets в правиле означает, что под правило попадает любой IP-адрес).

Правила сопоставляются данным пользователя по одному сверху вниз до первого успешного сопоставления. В случае успешного сопоставления — запрос пользователя помечается заголовком белого списка, позволяющим получить доступ к содержимому файла, В случае отсутствия успеха — запрос помечается заголовком, запрещающим доступ к содержимому файла.

## Шаг 3. Примените изменения

Для инсталляции на одну виртуальную машину выполните команды:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t istio-ingress
```

Для кластерной инсталляции:

```
HELMWAVE_ENV_NAME=cluster HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t istio-ingress
```

# Отключить возможность закреплять сообщения в личных чатах

По умолчанию закрепление сообщений в личных чатах включено. Чтобы отключить эту возможность, необходимо:

1. Укажите в конфигурационном файле **/usr/local/nginx-im/html/myteam/myteam-config.json** значение false для параметра **personal-messaging-pins-enabled**:

   ```
   "personal-messaging-pins-enabled": false
   ```

2. Для инсталляции на одну виртуальную машину выполните команду:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t godmod
   ```

   Для кластерной инсталляции:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true HELMWAVE_ENV_NAME=cluster hwup -t godmod
   ```

3. Перезапустите под в технологическое окно (может приводить к сбою в новых подключениях):

   ```
   kubectl delete pods -n vkteams -l app=myteam-admin
   ```

> **Примечание**
>
> Данная настройка не затрагивает закрепленные сообщения в групповых чатах и каналах, поскольку это неконфигурируемый базовый функционал.

# Отключить синхронизацию черновиков между клиентскими приложениями

Начиная с версии Мессенджер и ВКС 25.2 вы можете отключить синхронизацию черновиков сообщений между клиентскими приложениями одного пользователя, чтобы исключить утечку данных. После отключения синхронизации черновики продолжат работать в рамках одного клиентского приложения.

Чтобы отключить синхронизацию черновиков:

1. В конфигурационном файле **/usr/local/nginx-im/html/myteam/myteam-config.json** установите для поля **draft-enabled** значение false.
2. Для инсталляции на одну виртуальную машину выполните команду:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t godmod
   ```

   Для кластерной инсталляции:

   ```
   HELMWAVE_USE_LOCAL_REPO_CACHE=true HELMWAVE_ENV_NAME=cluster hwup -t godmod
   ```

3. Перезапустите под в технологическое окно (может приводить к сбою в новых подключениях):

   ```
   kubectl delete pods -n vkteams -l app=myteam-admin
   ```

Технический писатель: Белова Ирина

23 марта 2026 г.