

# Мессенджер и ВКС

**Развертывание и настройка сервисов групповых политик, версия 26.1**

Назначение документа	4
Дополнительная документация	5
Предварительные условия	6
1. Включите функциональность в веб-интерфейсе установщика VK WorkSpace	7
2. Выполните предварительные действия	8
Шаг 1. Проверьте настройку сервиса Godmod и мини-аппов	9
Шаг 2. Настройте права для административного пользователя	10
Шаг 3. Авторизуйтесь под учетной записью администратора	11
Шаг 4. Настройте сервис Zastava	12
Шаг 5. Создайте топики в Kafka	14
3. Включите политику доступности мини-аппов в приложении VK WorkSpace	16
Шаг 1. Выполните скрипт миграции мини-аппов	16
Шаг 2. Активируйте сервис Расман	18
Шаг 3. Разверните сервис Расман	19
Шаг 4. Анонсируйте политику доступности мини-аппов	23
Шаг 5. Включите политику доступности мини-аппов	24
Шаг 6. Включите политику в панели администратора VK WorkSpace	25
4. Включите политику обязательных чатов и каналов в Мессенджер и ВКС	25
Шаг 1. Активируйте сервис Chat-gp	25
Шаг 2. Разверните сервис Chat-gp	26
Шаг 3. Настройте сервис Mchat-st	28
Шаг 4. Анонсируйте политику обязательных чатов	0
Шаг 5. Включите политику в панели администратора VK WorkSpace	0
5. Включите политику запретов создания чатов/каналов в Мессенджер и ВКС	0
Шаг 1. Активируйте сервис Chat-gp	0
Шаг 2. Разверните сервис Chat-gp	0
Шаг 3. Настройте сервис Mchat-st	0
Шаг 4. Включите политику запретов создания чатов/каналов	0
Шаг 5. Анонсируйте политику запретов создания чатов/каналов	0

Шаг 6. Включите политику в панели администратора VK WorkSpace	0
6. Включите политику привилегий	0
Шаг 1. Активируйте сервис Vipper	0
Шаг 2. Разверните сервис Vipper	0
Шаг 3. Настройте сервис Gbld-mchat	0
Шаг 4. Настройте сервис Nomail	0
Шаг 5. Настройте сервис Boss	0
Шаг 6. Анонсируйте политику привилегий	0
Шаг 7. Включите политику в панели администратора VK WorkSpace	0
7. Настройте переменные окружения на сервере VK WorkSpace	0
8. Выполните настройки в веб-интерфейсе установщика Почты	0
Шаг 1. Выполните шаг upload_pravda_defaults для контейнера pravda	0
Шаг 2. Получите идентификаторы политик в Панели администратора	0
Шаг 3. Выполните настройки в веб-интерфейсе установщика	0
Как выключить групповые политики	0
Как обновить политику доступности мини-аппов в приложении VK WorkSpace	0
Как добавить новый инстанс сервиса групповой политики в карту	0

# Назначение документа

---

В документе описаны шаги для развертывания и включения сервисов следующих групповых политик:

- Политика доступности разделов меню Супераппа VK WorkSpace.
- Политика обязательных чатов и каналов Мессенджер и ВКС.
- Политика запрета создания чатов и каналов.
- Политика привилегий.

В инструкции для упрощения сервер/установщик Почты VK WorkSpace и сервер/установщик Панели администратора VK WorkSpace названы «сервер/установщик VK WorkSpace». Документ предназначен для использования администраторами организации.

Общий план развертывания и включения сервисов групповых политик:

1. Проверьте, что у вас есть [все необходимое](#) для развертывания и включения сервисов групповых политик.
2. [Включите функциональность групповых политик](#) в веб-интерфейсе установщика VK WorkSpace.
3. [Выполните предварительные действия](#) для развертывания и настройки сервисов групповых политик.
4. Разверните и настройте сервисы необходимых групповых политик:
  - [Политика доступности мини-аппов](#).
  - [Политика обязательных чатов и каналов](#).
  - [Политика запрета создания чатов и каналов](#).
  - [Политика привилегий](#).
5. [Настройте переменные окружения](#) на сервере VK WorkSpace.
6. Получите доступ к функциональности групповых политик.  
Функциональность отсутствует в базовой сборке. Чтобы подключить ее для одного или всех доменов инсталляции, [обратитесь в службу поддержки](#).

После развертывания и включения сервисов дальнейшая настройка групповых политик происходит в Панели администратора VK WorkSpace в соответствии с [инструкцией](#).

Действия с политиками после развертывания и настройки:

- Выключить политику привилегий и политику обязательных чатов и каналов можно в панели администратора VK WorkSpace. Чтобы выключить политику доступности мини-аппов, перейдите [в соответствующий раздел данной инструкции](#).
- Если появился новый веб-апп или мини-апп и его нужно добавить в Суперапп, [следуйте инструкции ниже](#).
- Добавление в карту новых инстансов для сервисов Vipper и Распап происходит автоматически. Чтобы добавить в карту новый инстанс сервиса Chat-gr, [следуйте инструкции ниже](#).

## Дополнительная документация

---

[Групповые политики](#) — в документе описана работа с групповыми политиками в Панели администратора VK WorkSpace.

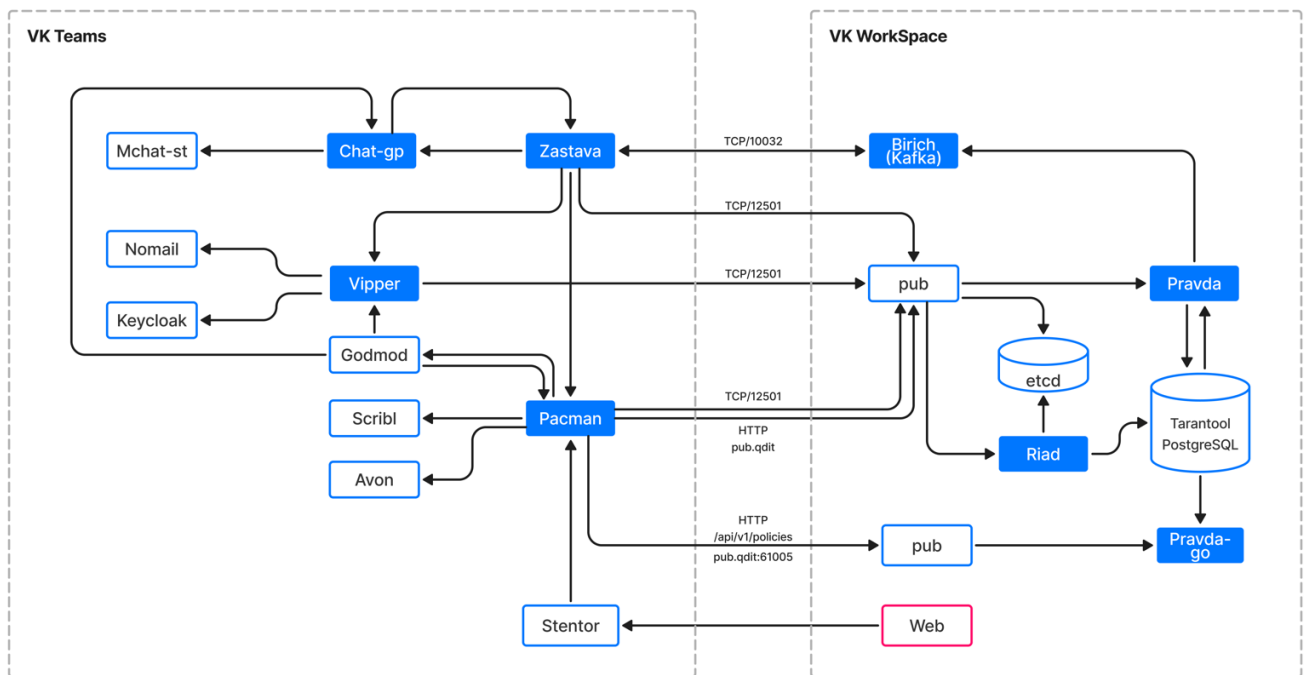
[Настройка интеграции Мессенджер и ВКС и Почты VK WorkSpace](#) — в документе описана настройка интеграции Почты VK WorkSpace и Мессенджер и ВКС.

[Настройка интеграции Мессенджер и ВКС с Панелью администратора VK WorkSpace](#) — в документе описана настройка интеграции Мессенджер и ВКС версии 24.5 и выше и панели администратора VK WorkSpace версии 1.23 и выше.

**Архитектура и описание системы Мессенджер и ВКС** — в документе описаны сервисы Мессенджер и ВКС, обеспечивающие функциональность групповых политик. Не является частью публичной документации, обратитесь к представителю VK, чтобы ознакомиться с документом.

# Предварительные условия

1. Доступ к веб-интерфейсу Kafka.
2. Доступ к серверу Мессенджер и ВКС.
3. Доступ к веб-интерфейсу установщика VK WorkSpace [http://<company\\_domain>:8888](http://<company_domain>:8888).
4. Доступ к серверу VK WorkSpace и к панели администратора VK WorkSpace [https://biz.<company\\_domain>](https://biz.<company_domain>).
5. Сетевой доступ между сервисами Мессенджер и ВКС и сервисами VK WorkSpace:



Откуда	Куда	Порт/протокол
Zastava	Birich (Kafka)	TCP/10032
Birich (Kafka)	Zastava	TCP/10032
Zastava	Pub	TCP/12501
Vipper	Pub	TCP/12501
Pacman	Pub	TCP/12501
Pacman	Pub	HTTP

6. Чат-бот для рассылки уведомлений, если необходимо уведомлять пользователей об изменении политики привилегий и политики обязательных чатов и каналов. [Создайте нового бота](#) или используйте существующего.

Вы можете разрешить боту писать первым — в таком случае уведомления будут получать все пользователи. Это можно сделать командой:


```
echo "row_add bots_can_send_deeplink <ID бота>" | nc stdb.vkteams.svc.cluster.local. 4020
```

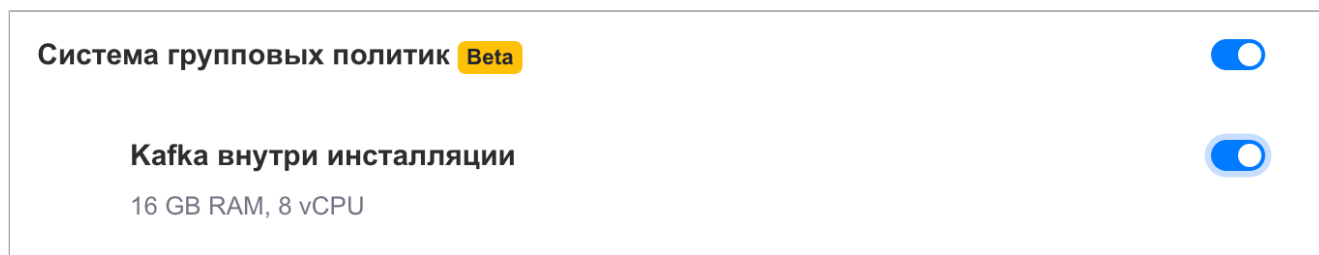
Если бот не может писать первым — пользователи не будут получать уведомления, пока не напишут в боте /start.


Также боту необходимо выдать разрешение на отправку deeplink:

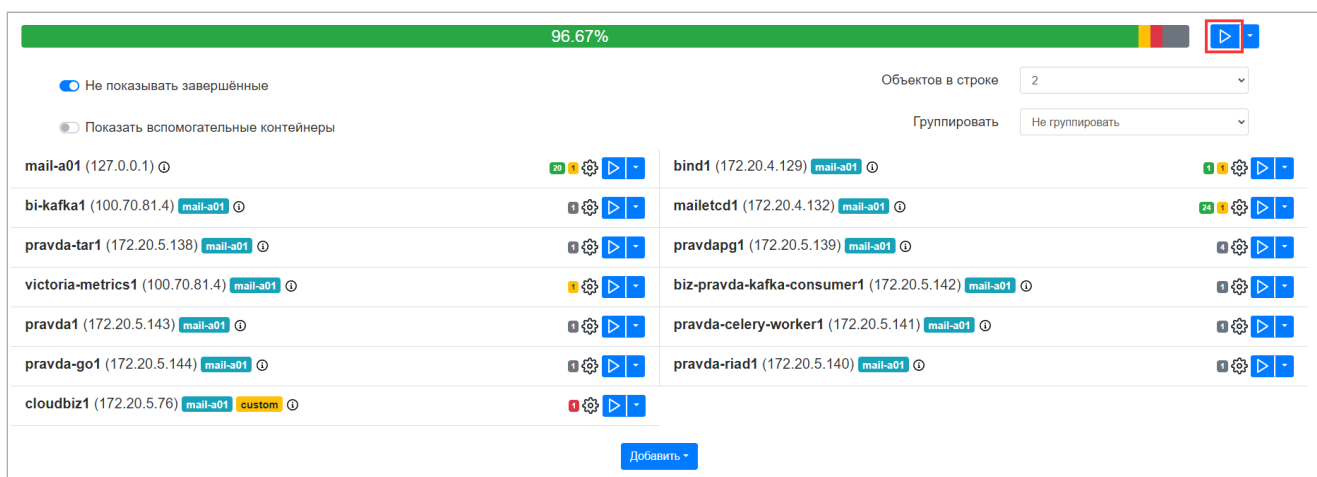
```
echo "row_add bots_can_write_first <ID бота>" | nc stdb.vkteams.svc.cluster.local. 4020
```

## 1. Включите функциональность в веб-интерфейсе установщика VK WorkSpace

1. Перейдите в веб-интерфейс установщика VK WorkSpace по адресу `http://<company_domain>:8888`.
2. Нажмите на кнопку  в правом верхнем углу и выберите пункт **Продукты**.
3. В списке продуктов выберите опцию **Система групповых политик**. Для использования Apache Kafka внутри инсталляции выберите опцию **Kafka внутри инсталляции**:



4. Нажмите на кнопку **Сохранить**.
5. Перейдите на главную страницу веб-интерфейса установщика.
6. Если у вас кластерная инсталляция, распределите контейнеры по гипервизорам. Если инсталляция на одну виртуальную машину, нажмите на кнопку **Сгенерировать автоматически**, чтобы сгенерировать контейнеры для работы групповых политик.
7. Запустите автоматическую установку нажатием на кнопку  в правом верхнем углу:



8. Подтвердите запуск автоматической установки, нажав на кнопку **Запустить** во всплывающем окне.

9. После окончания установки перейдите в список контейнеров и соберите данные, которые вам понадобятся для дальнейшей настройки на стороне Мессенджер и ВКС:

- Адреса Kafka — при использовании Kafka внутри инсталляции необходимо [в конфигурационном файле сервиса Zastava](#) указать адреса машин, на которых установлены сервисы bi-kafka:



- Адрес сервиса PUB1 — адреса машин, на которых установлены сервисы pub, необходимо будет указать далее в конфигурационных файлах сервисов [Pacman](#), [Chat-gp](#) и [Vipper](#).

## 2. Выполните предварительные действия

Перед тем как включать отдельные групповые политики, выполните на сервере Мессенджер и ВКС шаги, описанные ниже:

# Шаг 1. Проверьте настройку сервиса Godmod и мини-аппов

Выполните данный шаг, если настраиваете политику доступности мини-аппов. Если вы настраиваете другую политику, перейдите к [следующему шагу](#).

1. Если у вас новая инсталляция Мессенджер и ВКС, сначала настройте веб-аппы и мини-аппы (см. [дополнительную документацию](#)) и проверьте корректность их работы и только потом переходите к развертке и настройке сервисов групповых политик. Все веб-аппы и мини-аппы инсталляции, такие как почта, календарь, облако, должны быть настроены до запуска миграции мини-аппов и анонсирования групповых политик.

Если Мессенджер и ВКС установлен давно и все веб-аппы и мини-аппы работают корректно, перейдите к пункту 2.

2. Проверьте, что в конфигурационном файле `/usr/local/nginx-im/html/myteam/myteam-config.json` в секциях **desktop** и **mobile** присутствует вкладка **messenger** и стоит на первом месте:

## При наличии Почты VK WorkSpace

```
"disposition":{
  "desktop":{
    "leftbar":[
      "messenger",
      "tasks",
      "calendar",
      "orgstructure",
      "cloud"
    ]
  },
  "mobile":{
    "services":[
      "discover"
    ],
    "tabs":[
      "messenger",
      "tasks"
    ]
  }
}
```

## При отсутствии Почты VK WorkSpace

```
"disposition":{
  "desktop":{
    "leftbar":[
      "messenger",
      "tasks",
      "orgstructure",
    ]
  },
  "mobile":{
    "services":[
```

```
    "discover"
  ],
  "tabs": [
    "messenger",
    "tasks"
  ]
}
```

а также указана версия API  $\geq 120$ :

```
"api-version": 120,
```

Если вы внесли изменения в `/usr/local/nginx-im/html/myteam/myteam-config.json`:

а. 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
```

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

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

## Шаг 2. Настройте права для административного пользователя

Перед анонсом политики убедитесь, что у пользователя, из-под которого будет осуществляться анонс, имеются необходимые права.

1. Перейдите в конфигурационный файл `/usr/local/etc/k8s/helmwave/store/myteam-admin.yml` и в секциях **MiniApps**, **Management** и **Settings** установите для поля **default** значение **allow**:

```
myteamAdminPermission:
  resources:
    Analytics:
      default: allow
      groups: []
    Export:
      default: allow
      groups: []
    Settings:
      default: allow
      groups: []
    Information:
      default: allow
      groups: []
    Management:
      default: allow
      groups: []
    MiniApps:
      default: allow
      groups: []
```

2. Примените изменения.

Для инсталляции на одну виртуальную машину выполните команду:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup --tags godmod
```

Для кластерной инсталляции:

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

## Шаг 3. Авторизуйтесь под учетной записью администратора

1. На сервере Мессенджер и ВКС сгенерируйте OTP с помощью команды:

```
curl 'http://admin.<company_domain>/auth/otp/generate' \
-H 'Content-Type: application/json' \
-d' {"email": "<admin-mail>"}
```

2. Авторизуйтесь:

```
curl 'http://admin.<company_domain>/auth/otp/check' \
-H 'Content-Type: application/json' \
-d'{"email": "<admin-mail>", "password": "<OTP>"}' \
-c cookieFile.txt
```

После этого в `cookieFile.txt` происходит запись `cookie`, который нужно использовать в дальнейших запросах с флагом `-b`. Примеры запросов будут приведены для каждой из политик ниже. В рамках одной сессии достаточно авторизоваться таким образом один раз.

## Шаг 4. Настройте сервис Zastava

Сервис отвечает за взаимодействие с панелью администратора VK WorkSpace и передачу событий групповых политик в сервисы Мессенджер и ВКС. При изменении политики сервис Pravda на стороне панели администратора VK WorkSpace создает событие и отправляет его в Kafka (подробнее см. в документе с описанием архитектуры Мессенджер и ВКС. Не является частью публичной документации, обратитесь к представителю VK Tech, чтобы ознакомиться с документом).

Перейдите в конфигурационный файл сервиса Zastava `/usr/local/etc/zastava-1.yaml` и заполните поля:

### 1. В секции **kafka**:

```
kafka:
  addresses:
    - 100.70.81.231:10032
    - 100.70.81.219:10032
    - 100.70.81.231:10032
  consumer_group: "zastava-consumer"
  topic: "birich"
  retry_topic: "birich-retry"
  dlq_topic: "birich-dlq"
  actions_topic: "action"
  actions_retry_topic: "birich-action-retry"
  poll_limit: 1000
  retry_delay_ms: 3000
  retry_count_max: 3
  consume_preferring_lag: 1
  max_buffered_records: 10000
  fetch_max_partition_bytes: 1000
```

где:

- `address` — адреса машин и порт `bi-kafka` из шага [Настройки в веб-интерфейсе установщика VK WorkSpace](#).
- `consumer_group` — название группы потребителей Kafka. Может быть произвольным, главное, чтобы оно совпадало между всеми инстансами сервиса Zastava.
- `topic` — топик для получения событий изменений, относительно пользователя. Не меняйте название топика.
- `retry_topic` — служебный топик для логики ретраев и отложенной доставки сообщений. Не меняйте название топика.
- `dlq_topic` — служебный топик для недоставленных сообщений. Анализ сообщений из этого топика может быть полезен для определения проблем. Не меняйте название топика.
- `actions_topic` — топик для получения действий, применяемых к пользователям в соответствии с назначаемыми политиками. Не меняйте название топика.

- `actions_retry_topic` — служебный топик для логики ретраев и отложенной доставки сообщений. Не меняйте название топика.
- `poll_limit` — ограничение на количество одновременно забираемых сообщений из брокера.
- `retry_delay_ms` — пауза между попытками повторно доставить сообщение.
- `retry_count_max` — максимальное число попыток доставить сообщение.
- `consume_preferring_lag` — служебный параметр потребителя топиков Kafka, оставьте без изменений.
- `max_buffered_records` — служебный параметр потребителя топиков Kafka, оставьте без изменений.
- `fetch_max_partition_bytes` — служебный параметр потребителя топиков Kafka, оставьте без изменений.

2. В секции **schekolda** укажите для поля **max\_parallel** значение «4»:

```
schekolda:
  max_parallel:
    __default__: 200
```

Поле **max\_parallel** отвечает за количество одновременных запросов в сервис Zastava. Этот параметр можно использовать для ограничения потока исходящих запросов. При выставлении значения «0» количество исходящих запросов не будет ограничено. Рекомендованное значение на данный момент: «4».

3. В секции **etcd** укажите эндпоинты и префикс ключей для доступа в etcd (сервис PUB1):

```
etcd:
  endpoints:
    - 100.70.80.52:12501
  timeout: 1s
  watch_prefix: "/gp/vkt/"
```

где:

- `endpoints` — адрес сервиса PUB1 из шага [Настройки в веб-интерфейсе установщика VK WorkSpace](#).
- `/gp/vkt/` — префикс ключей для доступа в etcd, оставьте без изменений. Он используется [при обновлении групповой политики](#).

#### **Внимание**

Эндпоинты и префикс ключей для доступа в etcd должны совпадать с указанными в конфигурации сервиса Pravda и в конфигурации конкретных сервисов групповых политик.

4. Перезапустите сервис Zastava командой:

```
systemctl restart zastava-1
```

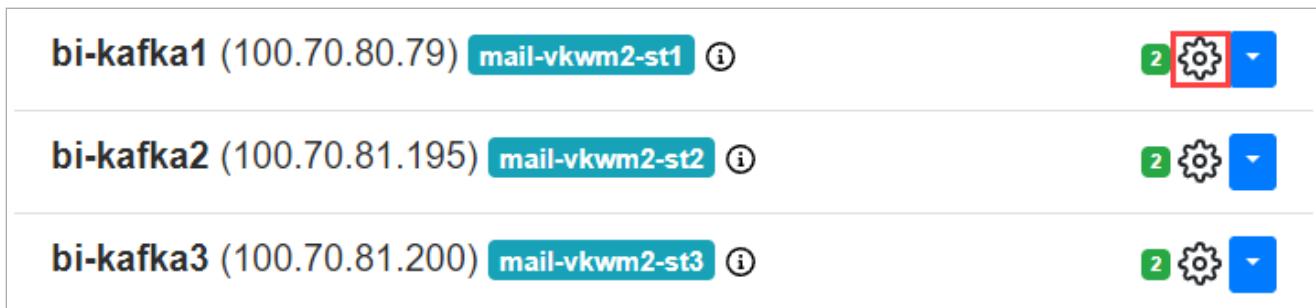
## Шаг 5. Создайте топики в Kafka

### При использовании Kafka внутри инсталляции

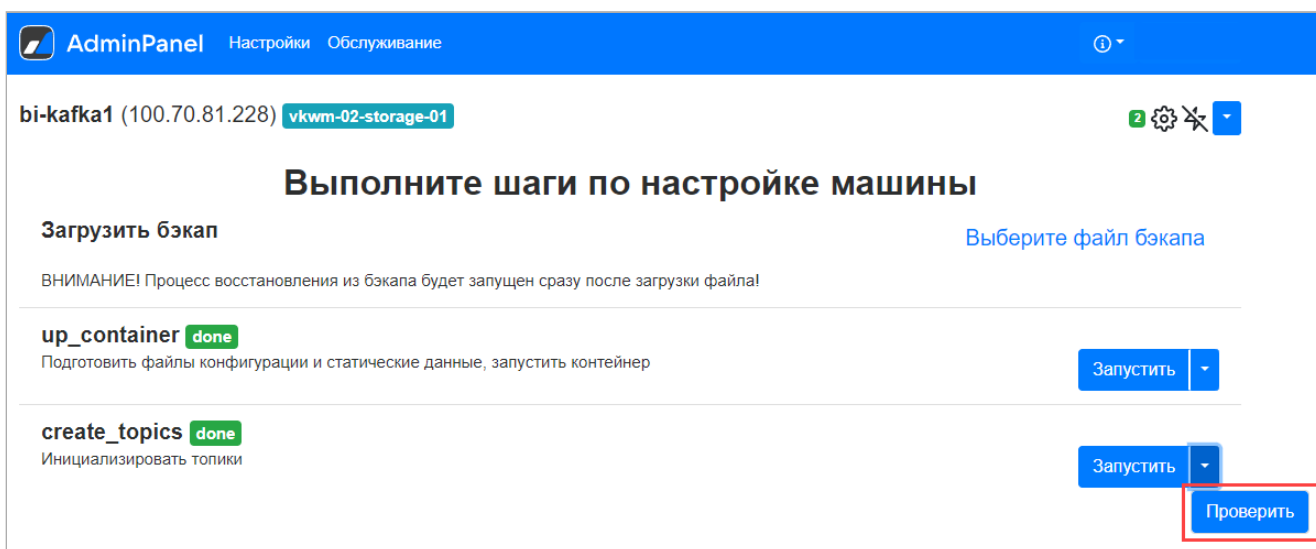
Если вы используете Kafka внутри инсталляции, необходимые топики создадутся автоматически.

Чтобы проверить, что необходимые топики созданы:

1. Перейдите в веб-интерфейс установщика VK WorkSpace по адресу `http://<company_domain>:8888`.
2. Перейдите в список контейнеров, найдите контейнер `bi-kafka` и нажмите на пиктограмму



3. Выберите **Проверить** для графы `create_topics`:



После нажатия на кнопку запустится проверка необходимых топиков.

Если в контейнере не хватает топиков, для графы `create_topics` тег **done** изменится на **needRefresh**. Нажмите на кнопку **Запустить**.

По умолчанию создается 10 партиций для каждого топика. Чтобы увеличить количество партиций:

1. Подключитесь к контейнеру `bi-kafka`:

```
docker exec -it bi-kafkaN bash
```

2. Следуйте [инструкции из документации Apache Kafka](#).

### При использовании внешнего сервиса Kafka

Если вы используете внешний сервис Kafka, перейдите в его веб-интерфейс и создайте все топики из секции **kafka** конфигурационного файла **/usr/local/etc/zastava-1.yaml** (см. [шаг 1 выше](#)).

Количество партиций для каждого топика — не менее **max\_parallel \* <количество инстансов> / 2**.  
Рекомендуется при возможности сразу установить количество партиций равное **max\_parallel \* <количество инстансов>**.

Большее число партиций не является проблемой (и число партиций у существующего топика, как правило, можно увеличить).

Таким образом, для двух инстансов сервиса Zastava с **max\_parallel = 4** рекомендуется для начала установить по восемь партиций в каждом топике.

### Если топиков нет и доступа к веб-интерфейсу Kafka тоже нет

Создайте топики при помощи утилиты `kcat`. Будьте осторожны, так как топики будут иметь некоторое заранее сконфигурированное администратором Kafka количество партиций, которое может не совпадать с желаемым. Прибегать к этой инструкции следует только при отсутствии других вариантов.

Чтобы создать топики:

1. На сервере Мессенджер и ВКС выполните команды ниже:

```
sudo docker run -it --network=host edenhill/kcat:1.7.1 -b b <IP-адрес_Kafka> -P -t birich -Z
# (нажмите ctrl-d, чтобы послать eof)

sudo docker run -it --network=host edenhill/kcat:1.7.1 -b b <IP-адрес_Kafka> -P -t action -Z
# (нажмите ctrl-d, чтобы послать eof)

sudo docker run -it --network=host edenhill/kcat:1.7.1 -b b <IP-адрес_Kafka> -P -t birich-action-retry -Z
# (нажмите ctrl-d, чтобы послать eof)

sudo docker run -it --network=host edenhill/kcat:1.7.1 -b b <IP-адрес_Kafka> -P -t birich-retry -Z
# (нажмите ctrl-d, чтобы послать eof)

sudo docker run -it --network=host edenhill/kcat:1.7.1 -b b <IP-адрес_Kafka> -P -t birich-dlq -Z
#( нажмите ctrl-d, чтобы послать eof)

# Посмотрите список топиков
sudo docker run -it --network=host edenhill/kcat:1.7.1 -b <IP-адрес_Kafka> -L
```

2. Перезапустите сервис Zastava командой:

```
systemctl restart zastava-1
```

### 3. Включите политику доступности мини-аппов в приложении VK WorkSpace

---

На сервере Мессенджер и ВКС выполните шаги, представленные ниже:

#### Шаг 1. Выполните скрипт миграции мини-аппов

1. Для корректной работы скрипта откройте доступ на запросы в сервис Go-files:

a. В конфигурационном файле Nginx `/usr/local/nginx-im/confv2/nginx.conf` в секции `geo $prod_ip` укажите IP-адрес сервера Мессенджер и ВКС:

```
geo $prod_ip {
    <IP-адрес сервера Мессенджер и ВКС>;
    default 0;
}
```

b. Перезапустите Nginx:

```
nginx.sh reload
```

c. Перед запуском скрипта миграции проверьте доступность сервиса Go-files, который хранит файлы. Ниже приведен пример команды для отправки локального svg-файла на удаленный сервер:

```
curl --location --request POST 'http://files-n.<company_domain>/s2s/api/v1/direct_upload?uid=example@mail.ru' --header 'Content-Type: image/svg' -d '@<some_svg_path>'
```

где — путь до локального файла с расширением .svg.

Если сервис недоступен, то

2. Перейдите в директорию `/usr/local/gp_deploy/` и запустите скрипт миграции веб-аппов и мини-аппов командой:

```
python3 miniappCreate.py --baseurl '<companyr_domain>.ru'
```

В результате работы скрипта в сервисах Avon и Gulliver появится информация о веб-аппах и веб-аппах, присутствующих в инсталляции.

### Примечание

Для обновления информации о мини-аппах (в ситуации, когда необходимо исправить конфиг, название мини-аппа или иконку) скрипт можно запустить с ключом `-update`.

Пример:

```
python3 miniappCreate.py --update --baseurl '<company_domain>.ru'
```

Список перенесенных в инсталляцию мини-аппов будет сохранен в файле `created_miniapps.txt`.

3. Чтобы проверить список перенесенных мини-аппов, выполните команду:

```
echo 'box.space.mini_apps_list:select({})' | tarantoolctl eval avon-1
```

В выводе команды отобразятся перенесенные мини-аппы с типом webapp и native:

```
- ['calendar', 5, true, 'webapp']
- ['calls', 3, true, 'native']
- ['cloud', 8, false, 'webapp']
- ['mail', 6, false, 'webapp']
- ['messenger', 17, true, 'native']
- ['miniapp-8fe88e60-4100-4c6d-9a25-5aa40d525c37', 1, true, 'miniapp']
- ['orgstructure', 7, false, 'webapp']
- ['tasks', 2, true, 'webapp']
```

Проверьте, что присутствуют все необходимые мини-аппы для инсталляции.

#### 4. Активируйте все перенесенные мини-аппы

Пример команды для активации мини-аппа календарей:

```
box.space.mini_apps_list:update('calendar', {'=' , 3, true}))
```

## Шаг 2. Активируйте сервис Расман

Сервис отвечает за реализацию групповой политики доступности веб-аппов и мини-аппов и находится в инсталляции в неактивном состоянии. Сервис надо активировать и развернуть.

Чтобы активировать сервис:

1. Перейдите в конфигурационный файл `/usr/local/etc/k8s/helmwave/projects.yml` и удалите из секции **disabled** сервис Расман.
2. Запустите деплой сервиса.

Для инсталляции на одну виртуальную машину выполните команду:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup --tags расман
```

Для кластерной инсталляции:

```
HELMWAVE_ENV_NAME=cluster HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup --tags расман
```

#### 3. Создайте карту сервиса:

```
gic utils mapfiller --map-name расман
```

Пример названия инстанса сервиса в карте: `расман.<имя_ноды>.расман -1`

## Шаг 3. Разверните сервис Pacman

Перейдите в конфигурационный файл сервиса Pacman `/usr/local/etc/k8s/helmwave/projects/pacman/values/pacman.yml` и укажите:

1. В секции **pravda** укажите адрес эндпоинта API групповых политик:

```
pravda:
  addr: "http://biz.<company_domain>.ru/vkwsqp"
  timeout: 5s
```

2. В секции **cache** укажите размер и тип кэша:

```
cache:
  type: lru
  size: 10000
```

На данный момент поддерживаются два типа кэша:

- none;
- lru — рекомендованное значение.

3. Заполните секции **version**, **endpoints** и **connection**:

```
live_config:
  riad:
    version: "/gp/vkt/pacman/1"
    endpoints:
      - "100.70.80.52:12501"
    policies:
      # см. описание в пункте ниже
    connection:
      uri: "pacman"
      protocol: "ipros"
      params: {}
```

где:

- version — ключ, по которому панель администратора VK WorkSpace будет искать конфигурацию. Имеет вид prefix/имя\_сервиса/версия. Важно убедиться, что префикс совпадает с указанным в настройках сервисов Pravda и Zastava.
- endpoints — адрес etcd-сервера PUB1 из шага [Настройки в веб-интерфейсе установщика VK WorkSpace](#).
- connection — секция настроек доставки сообщений для сервиса Zastava. Оставьте как в примере выше.

#### 4. Заполните секцию **policies**:

Если вы выполнили скрипт миграции мини-аппов, возьмите секцию **policies** из примера ниже, скорее всего, вам не потребуется ее менять:

```
live_config:
  riad:
    version: "/gp/vkt/расман/1"
    endpoints:
      - "10.32.0.1:12501"
    policies:
      - name: orderpolicy
        title: Order Policy
        payload:
          permissions:
            - name: desktopOrder
              title: Разделы десктоп и веб-версий
              description: Выбранные сервисы будут отображаться в интерфейсе Супераппа в левом
навигационном баре сверху вниз
              type: SortableMultiSelectList
              list_values:
              meta:
            - name: mobileOrder
              title: Разделы мобильной версии
              description: Выбранные сервисы будут отображаться в интерфейсе Супераппа
              type: SortableMultiSelectList
              list_values:
              meta:
          connection:
            uri: расман
            protocol: ipros
            params: {}
```

Оставьте секции **list\_values** и **meta** пустыми. В таком случае порядок мини-аппов и настройки по умолчанию будут собраны из сервиса Avon.

Чтобы изменить порядок мини-аппов или значения по умолчанию, заполните секции **list\_values** и **meta** для **mobileOrder** и **desktopOrder**. Пример заполненных секций ниже:

```
live_config:
  riad:
    version: "/gp/vkt/расман/1"
    endpoints:
      - "100.70.80.52:12501"
    policies:
      - name: orderpolicy
        title: Order Policy
        payload:
          permissions:
            - name: desktopOrder
              title: Разделы десктоп и веб-версий Супераппа
              description: Выбранные сервисы будут отображаться в интерфейсе Супераппа в левом
навигационном баре сверху вниз
              type: SortableMultiSelectList
              list_values:
                - cloud
                - messenger
                - calendar
                - calls
```

```

- mail
- orgstructure
- tasks
#- miniapp-b885274c-6616-48b0-98f3-f9e5c689b62a
meta:
  messenger:
    default_position: 1
    hidable: false
    movable: false
  tasks:
    default_position: 2
    hidable: false
    movable: true
  calls:
    default_position: 5
    hidable: false
    movable: true
  mail:
    default_position: 4
    hidable: true
    movable: true
  calendar:
    default_position: 3
    hidable: true
    movable: true
  orgstructure:
    default_position: 6
    hidable: true
    movable: true
  cloud:
    default_position: 7
    hidable: true
    movable: true
#miniapp-b885274c-6616-48b0-98f3-f9e5c689b62a:
  #display: Тестовый мини-апп
  #default_position: 8
  #hidable: true
  #movable: true
- name: mobileOrder
  title: Порядок мини-аппов в мобильном клиенте
  description: Пользователям разрешено пользоваться мини-аппами на мобильном
клиенте
  type: SortableMultiSelectList
  list_values:
  - cloud
  - messenger
  - calendar
  - calls
  - mail
  - orgstructure
  - tasks
  - services
#- miniapp-b885274c-6616-48b0-98f3-f9e5c689b62a
meta:
  services:
    default_position: 1
    display: "Сервисы"
    hidable: false
    movable: false
  messenger:
    default_position: 2
    hidable: false
    movable: false

```

```

tasks:
  default_position: 4
  hidable: false
  movable: true
calls:
  default_position: 3
  hidable: false
  movable: true
mail:
  default_position: 6
  hidable: true
  movable: true
calendar:
  default_position: 5
  hidable: true
  movable: true
orgstructure:
  default_position: 7
  hidable: true
  movable: true
cloud:
  default_position: 8
  hidable: true
  movable: true
#miniapp-b885274c-6616-48b0-98f3-f9e5c689b62a:
  #display: Тестовый мини-апп
  #default_position: 9
  #hidable: true
  #movable: true
connection:
  uri: "pacman"
  protocol: "ipros"
  params: {}

```

В примере выше перечислены все поддерживаемые нативные и веб-аппы, у них указаны правильные признаки **hidable** и **movable** и правильный порядок.

Если в инсталляции отсутствует тот или иной веб-апп (например, почта или облако) — достаточно закомментировать соответствующие секции и упоминание в секции **list\_values**.

Мини-аппы можно добавить по аналогии с закомментированным miniapp-b885274c-6616-48b0-98f3-f9e5c689b62a в примере выше. miniapp-b885274c-6616-48b0-98f3-f9e5c689b62a — ID миниаппа, который можно получить в боте Metabot.

### **Внимание**

Все веб-аппы и мини-аппы, не упомянутые в секции **policies**:

- не будут доступны для конфигурирования через групповые политики в панели администратора VK WorkSpace;

- не будут отображаться в Супераппе после включения флага **group-policy-enabled** на [шаге 5](#) ниже.

Перечислите все нативные и веб-аппы, доступные в инсталляции, а также все актуальные мини-аппы в системе, для которых нужна иконка в Супераппе.

Несмотря на то, что все неанонсированные мини-аппы не будут доступны для конфигурирования через групповые политики, они всё ещё будут доступны по ссылке. Групповая политика разделов в текущей реализации управляет отображением разделов, но не доступом.

Информация из раздела **meta** используется только для отображения в интерфейсе панели администратора VK WorkSpace:

- Значения **values** и **icon** в **meta** на данный момент не используются.
- Для **hidable** и **movable** указаны значения по умолчанию — true. При необходимости могут быть скорректированы на false.
- Если **default\_position** прописан хотя бы для одного из аппов, то тогда он должен быть прописан для всех аппов. Кроме того, при выставлении этого параметра через конфиг значения должны быть уникальными. Если нарушено одно из этих условий, при попытке донести в etcd будет код ответа 400. Если **default\_position** не прописывается в конфиге, то он запрашивается из сервиса Avon, в таком случае мини-аппы могут быть не по порядку в клиентском приложении.

#### **Внимание**

**desktopOrder** должен быть описан раньше **mobileOrder**, иначе в панели администратора VK WorkSpace секции настройки поменяются местами. Секция **services** обязательно должна быть описана при помощи поля **display**.

#### 5. Перезапустите сервис Расман.

Для инсталляции на одну виртуальную машину выполните команду:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t расман
```

Для кластерной инсталляции:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t расман
```

## Шаг 4. Анонсируйте политику доступности мини-аппов

1. Убедитесь, что выполнены [предварительные действия](#) для анонсирования политики.
2. В консоли выполните команду:

```
curl 'http://admin.<domain>/api/miniapp/announcePolicy' \  
-H 'Content-Type: application/json' \  
-H 'Accept: application/json' \  
-b cookieFile.txt \  
-d ''
```

Если код ответа **не** 200 — посмотрите логи сервиса Расман:

```
/var/log/vector/k8s/vkteams/расман
```

Если код ответа 200 — перейдите по адресу [https://biz.<company\\_domain>.ru/vkwsgp/api/v1/settings/vkt/расман/](https://biz.<company_domain>.ru/vkwsgp/api/v1/settings/vkt/расман/) и проверьте групповую политику в панели администратора VK WorkSpace.

## Шаг 5. Включите политику доступности мини-аппов

### Для всей инсталляции:

1. В конфигурационном файле `/usr/local/nginx-im/html/myteam/myteam-config.json` укажите для поля `group-policy-enabled` значение `true`:

```
"group-policy-enabled": true
```

Поле `group-policy-enabled` отвечает за то, чтобы приложения начали получать список мини-аппов в соответствии с настроенной групповой политикой.

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
```

### Для определенного домена:

1. В конфигурационном файле `/usr/local/nginx-im/html/myteam/myteam-config.json` укажите для поля `default` значение `false`, а для домена — `true`:

```
"group-policy-enabled": {  
  "$switch-domain": {  
    "default": false,  
    "<company_domain>.ru": 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
```

## Шаг 6. Включите политику в панели администратора VK WorkSpace

Работа с политикой доступности мини-аппов в панели администратора VK WorkSpace описана в документе [Групповые политики](#).

## 4. Включите политику обязательных чатов и каналов в Мессенджер и ВКС

---

На сервере Мессенджер и ВКС выполните шаги, представленные ниже:

### Шаг 1. Активируйте сервис Chat-gp

Сервис отвечает за реализацию групповой политики обязательных чатов и каналов и находится в инсталляции в неактивном состоянии. Сервис надо активировать и развернуть.

Чтобы активировать сервис:

1. В конфигурационном файле `/usr/local/etc/k8s/helmwave/projects.yml` удалите из секции **disabled** сервис Chat-gp.
2. Запустите деплой сервиса.

Для инсталляции на одну виртуальную машину выполните команду:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup --tags chatgp --tags myteam-admin
```

Для кластерной инсталляции:

```
HELMWAVE_ENV_NAME=cluster HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup --tags chatgp --tags myteam-admin
```

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

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

4. Создайте карту сервиса:

```
ic add_map chatgp shard
```

### **Внимание**

Chat-gp не должен иметь "-" в названии карты.

Пример правильного названия инстанса сервиса в карте — chatgp.onpremise.chatgp-1.

Пример **не**правильного названия инстанса сервиса в карте — chatgp.onpremise.chat-gp-1.

5. Создайте сервис для каждого экземпляра инстанса:

```
ic add_srv chatgp.onpremise.chatgp-1 onpremise main
```

где:

- chatgp.onpremise.chatgp-1 — алиас.
- onpremise — имя ipros-ноды.
- main — тип (slave, main, dup, mirror, fallback или none).

Распределите все бакеты по карте равномерно:

```
ic reshard chatgp
```

## Шаг 2. Разверните сервис Chat-gp

Перейдите в конфигурационный файл сервиса Chat-gp `/usr/local/etc/k8s/helmwave/projects/chatgp/values/chatgp.yml` и заполните поля:

1. В секции **zastava**:

```
zastava:  
  timeout: 6s  
  long_tasks_ttl: 1h
```

где:

- timeout — таймаут обработчика сообщений сервиса Zastava. Рекомендуется оставить без изменений.
- long\_task\_ttl — ограничение времени жизни для асинхронных запросов обработки добавления в чат. Рекомендуется оставить без изменений.

### **Примечание**

Значения можно изменить для избежания подвисаний при обработке сообщений из сервиса Zastava и возникновении нежелательных ретраев.

2. В секции **notifier** при необходимости включите уведомления пользователей об изменении политики:

```
notifier:
  bot:
    enabled: true
    token: "${TOKEN}"
    bot_api: "http://[.Release.Store.hosts.botApiExternal ]/bot/v1/"
    timeout: 10s
    retry_count: 3
    retry_pause: 600ms
  text_header: " Настройки Вашего аккаунта были изменены администратором домена:"
  templates:
    add_to_chat: >
      Вы были добавлены в чат [{{escape .ChatName}}]({{.Config.BaseURL}}{{.ChatNick}})
    del_from_chat: >
      Вы были удалены из чата [{{escape .ChatName}}]({{.Config.BaseURL}}{{.ChatNick}})
    base_url: https://%{mapping::host::icq_im}/
    chat_creation_is_allowed_template: " Вам разрешено создание групповых чатов."
    chat_creation_is_not_allowed_template: " Вам установлен запрет на создание
групповых чатов."
    chan_creation_is_allowed_template: " Вам разрешено создание каналов."
    chan_creation_is_not_allowed_template: " Вам установлен запрет на создание
каналов."
```

где:

- enabled: true — включает уведомления пользователей.
- token — укажите токен бота, от которого будут приходить уведомления ([см. предусловия](#)).

3. В секции **live\_config** укажите правильный ключ и адрес etcd-сервера:

```
live_config:
  riad:
    version: "/gp/vkt/chatgp/2"
    endpoints:
      - 100.70.80.52:12501
    policies:
      - name: obligatory_chats
        title: The policy of obligatory chats
        payload:
          permissions:
            - name: obligatory_chats
              title: The policy of obligatory chats
              description: The policy of obligatory chats
              type: MultiSelectList
              limit_scope: false
              value_source:
                type: chat
              actions:
                on_add_item:
                on_remove_item:
        connection:
          uri: chatgp
          protocol: ipros
          params: {}
```

где endpoints — адрес сервиса PUB1 из шага [Настройки в веб-интерфейсе установщика VK WorkSpace](#)

#### **Внимание**

В секции policies может быть указана конфигурация для групповой политики запретов создания чатов/каналов. Не удаляйте ее, если она вам нужна

4. Перезапустите сервис Chat-гр.

Для инсталляции на одну виртуальную машину:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t chatgp
```

Для кластерной инсталляции:

```
HELMWAVE_USE_LOCAL_REPO_CACHE=true hwup -t chatgp
```

## Шаг 3. Настройте сервис Mchat-st

1. Запустите скрипт для выставления creator\_sn у старых чатов (без его запуска старые чаты не будут отображаться в Панели администратора VK WorkSpace):

а. Чтобы понять, где запущен мастер Mchat-st, а где реплика, выполните:

```
ic map mchat-st
```

Пример вывода команды:

```
node mchat-st-1 (2147483648 buckets)
  srv mchat-st.vkt-4vm-1388-standart-01.mchat-st-1 10.32.88.0:2525 alive
main
  srv mchat-st.vkt-4vm-1388-standart-02.mchat-st-1 10.32.64.0:2525 alive
slave
node mchat-st-2 (2147483648 buckets)
  srv mchat-st.vkt-4vm-1388-standart-03.mchat-st-1 10.32.0.1:2526 alive
main
  srv mchat-st.vkt-4vm-1388-standart-04.mchat-st-1 10.32.96.0:2525 alive
slave
```

Мастерами будут инстансы, напротив которых написано «main», репликами — «slave». В примере выше мастера запущены на vkt-4vm-1388-standart-01 и vkt-4vm-1388-standart-03, реплики — на vkt-4vm-1388-standart-02 и vkt-4vm-1388-standart-04.

- b. Подключитесь к виртуальным машинам с репликами,
- c. Остановите сервис Mchat-st, сделайте копию данных и запустите сервис снова. Из примера выше данные команды надо выполнить на vkt-4vm-1388-standart-02 и vkt-4vm-1388-standart-04:

```
systemctl stop mchat-st-1
cp -rp /data/mchat-st-1 cp -rp /data/mchat-st-1.bak
systemctl start mchat-st-1
```

- d. Подключитесь к виртуальным машинам с мастерами (из примера выше это vkt-4vm-1388-standart-01 и vkt-4vm-1388-standart-03) и выставите в real mode:

```
echo set creator_sn_inspector_simulate 0 | nc 0.0.0.0 4081
```

- e. Запустите скрипт:

```
echo creator_sn_inspector_start | nc 0.0.0.0 4081
```

- f. Проверьте, есть ли ошибки в логах **/oap/icq/logs/mchat-st-1.log**. Если есть, то перезапустите скрипт.
- g. Чтобы посмотреть процесс скрипта:

```
tail -f /oap/icq/logs/mchat-st-1.log | grep CreatorSnInspector
```

По завершении скрипта будет вывод:

```
tail -f /oap/icq/logs/mchat-st-1.log | grep "CreatorSnInspector: finished"
```

Данный скрипт запускается один раз при первичной настройке групповой политики.

- 2. Перезапустите сервис Mchat-st командой:

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