Мессенджер и ВКС

Авторизация в системе

© VK WorkSpace, 2026 г. Все права защищены

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

Оглавление

Назначение документа

В документе описаны варианты доступа к системе и сервисы Мессенджер и ВКС, обеспечивающие авторизацию пользователей. Приведены примеры работы с журналами при проблемах с авторизацией.

Дополнительная документация

Инструкция по интеграции с контроллером домена по протоколу LDAP — в документе описано управление параметрами синхронизации LDAP.

Инструкция по настройке SSO-аутентификации (OIDC, SAML и Kerberos) — в документе описана настройка Single Sign-On аутентификации по протоколам OIDC, SAML и Kerberos.

Инструкция по настройке SSO-аутентификации (Stroma, SWA) — в документе описана настройка SSO-аутентификации по протоколу OpenID Connect (OIDC) в Мессенджер и ВКС и Почте VK WorkSpace при помощи сервиса Stroma в составе Мессенджер и ВКС.

Архитектура и описание системы — в документе описаны механизмы аутентификации в Мессенджер и ВКС. Не является частью публичной документации, обратитесь к представителю VK Tech, чтобы ознакомиться с документом.

Варианты доступа к системе

1. SSO-аутентификация.
2. Аутентификация с помощью одноразовых кодов (OTP).

   При помощи LDAP-коннектора осуществляется подключение к службе каталогов. Пользователи загружаются в сервис Keycloak и далее в БД Tarantool. Пароли для доступа в систему в сервисе не хранятся, при каждой авторизации пользователю на его почтовый ящик отправляется одноразовый пароль. Логином выступает электронная почта пользователя.

   Описание механизма аутентификации представлено в документе «Архитектура и описание системы» (не является частью публичной документации, обратитесь к представителю VK Tech, чтобы ознакомиться с документом).

При эксплуатации системы сложности чаще всего возникают именно с доступами, так как Active Directory является сложной системой, требующей аккуратного отношения.

Компоненты системы авторизации

- Сервис Keycloak (https://www.keycloak.org/). Это приложение осуществляет синхронизацию пользователей через LDAP. Все данные хранятся в БД MySQL.
- Сервис Nomail — основной сервис авторизации внутри системы. Может работать как самостоятельно, так и совместно с Keycloak. Все данные сервиса Nomail хранятся в БД Tarantool.
- MTA Postfix — принимает OTP-сообщения от сервиса Nomail и перенаправляет эти сообщения на SMTP-релей, указанный пользователем.

Сервис Keycloak

Сервис синхронизирует пользователей через LDAP. Расположение журнала: **/var/log/vector/k8s/keycloak/keycloak/* .** При любых ошибках синхронизации имеет смысл в первую очередь смотреть в журнал этого сервиса.

Доступ к веб-интерфейсу

У сервиса есть веб-интерфейс для управления, однако пользоваться этим интерфейсом следует осторожно, так как изменения, сделанные через этот интерфейс, могут быть изменены при обновлении системы. Любые изменения обязательно нужно внести в конфигурационный файл **/usr/local/etc/premsetup/defaults.yaml**.

По умолчанию интерфейс недоступен из внешней сети. Варианты доступа к веб-интерфейсу:

Вариант 1: Доступ через https://mridme.<YOUR_DOMAIN>/auth

Открыть доступ для домена mridme. <DOMAIN> и перейти в браузере https://mridme.<DOMAIN>

> **Примечание**
>
> По умолчанию имя mridme не заведено в DNS, и в настройках nginx выставлено deny all. Не рекомендуется использовать этот способ доступа без крайней необходимости.

Вариант 2: SSH-туннель (предпочтительный)

Выполнить команду:

```
ssh -L 8080:keycloak-http.keycloak.svc.cluster.local:80 centos@<server>
```

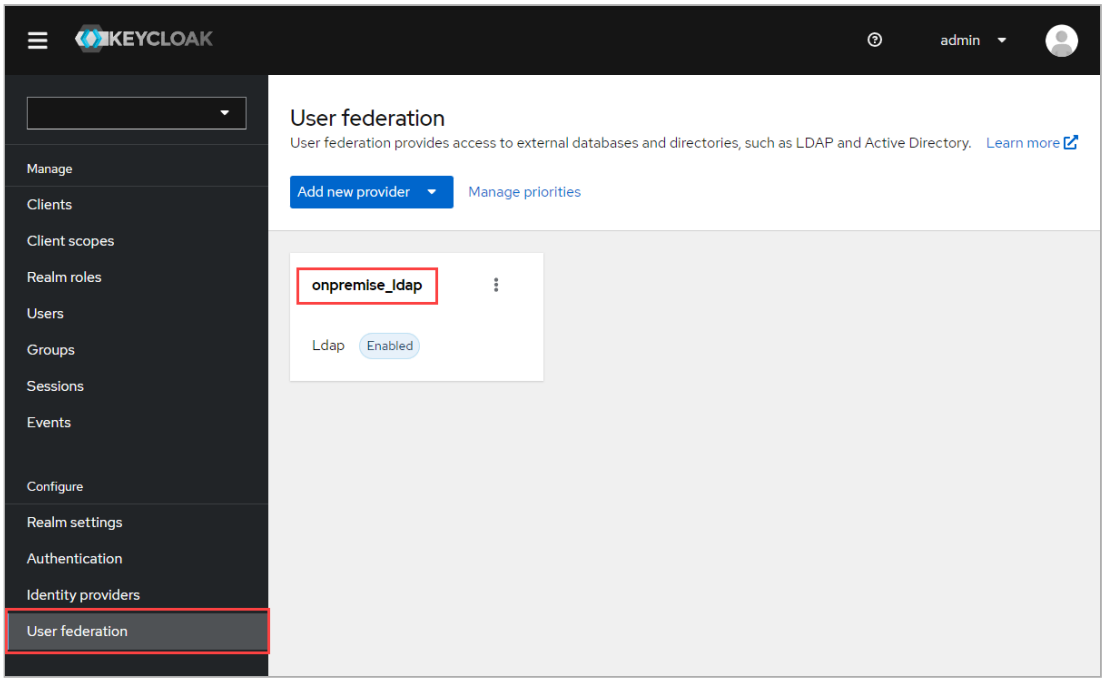
и перейти в браузере http://127.0.0.1:8080/auth

Логин: admin

Пароль: пароль необходимо получить в службе технической поддержки.

Наиболее важные параметры, доступные в веб-интерфейсе:

Настройка LDAP-подключения



Если LDAP-подключение уже создано, но с ним есть проблемы, то основные настройки, с которыми нужно работать:

- baseDN (usersDN) и фильтр — влияют на то, какие объекты будут получены через LDAP.

- Частота синхронизации — влияет на то, как часто будут производиться изменения.

> **Внимание**
>
> При большом количестве пользователей, попадающих под полную фильтрацию, полная синхронизация может быть очень тяжелой для Active Directory. Выполнять такую операцию часто нельзя. Например, Active Directory на 10 тыс. пользователей отрабатывается около трех минут, что является существенной нагрузкой на Active Directory.

Соответствия названий в веб-интерфейсе:

- **Users DN** — base DN в варианте Active Directory.
- **Custom User LDAP Filter** — LDAP-фильтр.
- **Search Scope / one level** — поиск только по дочерним объектам указанного объекта. Поиск по вложенным объектам не производится, также в результаты поиска не попадает сам базовый объект.
- **Search Scope / subtree** — поиск по всем дочерним объектам, включая вложенные. Базовый объект в поиск не попадает.
- **Full Sync Period** — частота выполнения полной синхронизации. Чем меньше пользователей, тем чаще можно выполнять полную синхронизацию.
- **Changed Users Sync Period** — частота выполнения частичного обновления пользователей (оптимальное значение по умолчанию — 60 секунд).

Состояние профилей пользователей (Active Directory + Keycloak конфигурация)

Поддерживается три состояния профиля (на основании состояния профиля (enabled/disabled) и принадлежности к группе). Поддержка состояния suspended включается в конфигурационном файле **nomail.keycloak.suspended.state.enabled**, название группы задается переменной **nomail.keycloak.allowed_group.name**:

- Активный — отсутствие каких либо флагов (в контексте Active Directory аккаунт должен быть enabled=True и принадлежать группе).
- Приостановлен — flag=SUSPENDED (либо enabled и не в группе, либо disabled и в группе).
- Удаленный — flag=DELETED (enabled=false + не в группе).

Если пользователь получает статус SUSPENDED (приостановлен), то осуществляется выход пользователя из системы. На устройства отправляется команда для выхода пользователя из системы и удаления всех локальных данных. Пользователь не может пользоваться сервисом, но в случае необходимости его можно быстро восстановить, и вся история у него сохранится.

При получении статуса DELETED (удален) на устройства пользователя отправляется команда для выхода из системы и удаления всех локальных данных. Вся история переписки на его устройствах удаляется. Также происходит удаление пользователя из всех групповых чатов, где он участвовал, без уведомлений участников групп. В случае необходимости пользователя можно восстановить из статуса DELETED —

история личной переписки восстановится, но пользователю будет необходимо заново вступать в групповые чаты, в которых он ранее участвовал.

Журнал Keycloak

Расположение журнала: **/var/log/vector/k8s/keycloak/keycloak/*.**

В первую очередь следует отслеживать записи уровня ERROR. Пример такой ошибки: дубликат пользователя — когда один и тот же пользователь заведен в нескольких ветках LDAP-каталога. Как правило, после каждой записи ERROR идет трассировка ошибки для более детального анализа.

Пример лога

```
2020-04-14 12:06:12,984 ERROR [org.keycloak.services.error.KeycloakErrorHandler] (default
task-38) Uncaught server error: org.keycloak.models.ModelDuplicateException: Can't import user
'user2' from LDAP because email 'user@vkteams.example.com' already exists in Keycloak.
Existing user with this email is 'user1'
        at
org.keycloak.storage.ldap.mappers.UserAttributeLDAPStorageMapper.checkDuplicateEmail(UserAttri
buteLDAPStorageMapper.java:176)
        at
org.keycloak.storage.ldap.mappers.UserAttributeLDAPStorageMapper.onImportUserFromLDAP(UserAttr
ibuteLDAPStorageMapper.java:107)
        at
org.keycloak.storage.ldap.LDAPStorageProvider.importUserFromLDAP(LDAPStorageProvider.java:517)
        at
org.keycloak.storage.ldap.LDAPStorageProvider.searchForUser(LDAPStorageProvider.java:372)
        at
org.keycloak.storage.UserStorageManager.lambda$searchForUser$2(UserStorageManager.java:556)
        at org.keycloak.storage.UserStorageManager.query(UserStorageManager.java:505)
        at org.keycloak.storage.UserStorageManager.searchForUser(UserStorageManager.java:554)
        at
org.keycloak.models.cache.infinispan.UserCacheSession.searchForUser(UserCacheSession.java:583)
        at
org.keycloak.services.resources.admin.UsersResource.searchForUser(UsersResource.java:247)
        at org.keycloak.services.resources.admin.UsersResource.getUsers(UsersResource.java:
218)
```

Из примера видно, что сервис не может добавить пользователя user2 , так как почта user@vkteams.example.com принадлежит пользователю user1. Как правило, такие проблемы происходят при неправильном формировании фильтров и/или baseDN. Например, под фильтрование могут попадать календари или адресные книги пользователей, либо в Active Directory действительно заведены разные пользователи с одинаковым адресом почты.

Журнал сервиса Nomail

Расположение журнала сервиса: **/oap/icq/logs/nomail-1.log**.

В журнале отображаются записи по синхронизации данных через LDAP — дополнительно к журналу сервиса Keycloak. Ошибки записываются с уровнем логирования ERROR, для просмотра таких записей выполните:

```
grep '] E ' /oap/icq/logs/nomail-1.log
```

Чаще всего ошибки возникают на этапе синхронизации пользователя или на этапе отправки OTP-сообщения, например из-за срабатывания антиспам-систем.

Если появился новый пользователь или произошли изменения в профиле, сервис Nomail оповещает об изменениях все остальные сервисы, которым необходимы эти данные. Например, сервисы поиска или хранения профилей пользователя.

Проблема синхронизации пользователей

В случае проблемы с синхронизацией пользователей ошибка, скорее всего, относится к сервису Keycloak. Поэтому:

1. Сначала проверьте журнал сервиса Keycloak.
2. Если пользователь заведен корректно, тогда проверьте журнал сервиса Nomail.

Проблемы с пользователем

Если есть проблемы с пользователем, изучите вхождения по конкретному пользователю в журнале Nomail.

Пример

```
>grep user@vkteams.example.com /oap/icq/logs/nomail-1.log
[2769513 14.04.2020 14:33:10] W NOMAIL: kkapi::load_user(user@vkteams.example.com)
[2769513 14.04.2020 14:33:11] W KEYCLOAK: user@vkteams.example.com's keycloak profile not
changed
```

Из примера видно, что в 14:33:10 Nomail проверил синхронизацию пользователя и обнаружил, что профиль пользователя не изменился и никаких действий выполнять не нужно.

Проблемы с отправкой OTP

Расположение журнала: **/data/tarantool/logs/nomail-1.log.**

После старта БД в этот журнал записываются ошибки отправки OTP-сообщений. Поэтому любые записи, не связанные со стартом БД, говорят о потенциальных проблемах. OTP-сообщение отправляется в локальный MTA, поэтому дальнейший путь этого сообщения можно найти по логам MTA Postfix:

```
journalctl -t postfix/smtpd -t postfix/smtp
```

Пример отправки сообщения

```
2020-04-14T13:38:33.684799+03:00 onpremise postfix/smtpd[947406]: A72C4B059E:
client=localhost[127.0.0.1]
2020-04-14T13:38:33.724571+03:00 onpremise postfix/cleanup[947410]: A72C4B059E: message-
id=<20200414103833.A72C4B059E@onpremise.localdomain>
2020-04-14T13:38:33.726111+03:00 onpremise postfix/qmgr[554740]: A72C4B059E:
from=<otp@vkteams.example.com>, size=525, nrcpt=1 (queue active)
2020-04-14T13:38:33.730092+03:00 onpremise postfix/smtp[947412]: A72C4B059E:
to=<tester@vkteams.example.com>, relay=10.10.10.10[10.10.10.10]:25, delay=0.05,
delays=0.04/0/0/0, dsn=2.0.0, status=sent (250 2.0.0 Ok: queued as 870A62070BEC)
2020-04-14T13:38:33.730302+03:00 onpremise postfix/qmgr[554740]: A72C4B059E: removed
```

Из примера видно, что локальный MTA отправил почтовое сообщение на релей 10.10.10.10, и релей сообщение принял. Дальнейший путь этого сообщения не зависит от инсталляции Мессенджер и ВКС.

Так как OTP-сообщения отправляются ядром БД, в случае проблем с OTP следует смотреть оба журнала. Но прежде чем изучать проблему внутри этих сервисов, проверьте, что пользователь действительно существует в системе. Резюме процедуры:

1. Проверьте, что пользователь существует в системе.
2. Проверьте журнал БД:

```
/data/tarantool/logs/nomail-1.log
```

3. Проверьте журнал MTA Postfix:

```
journalctl -t postfix/smtpd -t postfix/smtp
```

Проблема с авторизацией

Пример: поступило обращение со стороны клиента, что сотрудник с логином user@vkteams.example.com не может зайти в систему.

Как решить проблему

Проверьте, что пользователь user@example.com есть в системе — через веб-интерфейс сервиса Keycloak или базу данных:

```
mysql –S keycloak-mysql-mysql-cluster-im-db-mysql-master.keycloak.svc.cluster.local -e "SELECT
* FROM USER_ENTITY WHERE EMAIL = 'example@examle.com'\G" keycloak
```

Пользователя в системе нет

Это означает, что есть проблемы с синхронизацией пользователей. Дальнейшие действия:

Шаг 1. Проверьте с помощью `ldapsearch`, что пользователь действительно существует в корпоративном LDAP клиента.

Шаг 2. Если пользователь найден, выполните полную синхронизацию, чтобы убедиться в корректности синхронизации. Учитывайте, что полная синхронизация может занять несколько минут.

Запуск синхронизации:

```
kccli ldap sync --name <ИМЯ LDAP-ПОДКЛЮЧЕНИЯ>
```

В случае если вы хотите запустить принудительную синхронизацию всех LDAP-серверов, выполните:

```
kccli ldap sync
```

Если пользователь появился и был добавлен недавно, то дальнейших действий не требуется. Однако нужно проверить, что у пользователя заработала авторизация. Если пользователь не появился или он заведен давно, то выполните действия далее.

Шаг 3. Проверьте логи сервиса Keycloak — скорее всего, обнаружится ошибка (например, дубликаты записей пользователя). Исправьте ошибку и перезапустите сервис, чтобы закрыть все существующие соединения с LDAP-сервером клиента (так как все существующие соединения продолжают работать со старыми настройками).

Пример использования ldapsearch

```
LDAPTLS_REQCERT=never ldapsearch -H 'CONNECTION_URI' -W -D 'BIND_DN' -b
'BASE_DN''(&(ORIGINAL_FILTER)(mail=example@example.com))'
```

- Флаг `-W` — пароль нужно вводить при выполнении запроса.
- `LDAPTLS_REQCERT=never` — позволяет не регистрировать сертификат LDAP-сервера в ОС при использовании `ldapsearch`.
- `-H (CONNECTION_URI)` — указание полной строки подключения к LDAP согласно данным клиента (из файла конфигурации defaults.yaml или Keycloak).
- `-D (BIND_DN)` — логин для авторизации в LDAP.
- `-b (BASE_DN)` — базовый путь поиска.
- `ORIGINAL_FILTER` — фильтр, который используется в Keycloak (предоставляется клиентом, может быть найден в /usr/local/etc/premsetup/ldap.yaml). В данном примере к оригинальному фильтру добавлено требование найти объект с почтой user@examle.com.

!!! warning "Внимание" Скобки обязательны.

Пользователь в системе есть

1. Проверьте, что запрос на авторизацию со стороны пользователя был выполнен.
2. Проверьте отправку OTP-сообщения.

Пример работы с журналами

Авторизация

Авторизация выполняется с помощью запроса `POST /wim/auth/clientLogin` на `u.<user_domain>` и проходит через лог-файл **nginx-im /oap/icq/domains/local_proxy.icq.com/logs/u-access.log.** Так как POST-запросы неудобно отслеживать через логи Nginx, то при условии, что в них нет явной ошибки (кодов 5xx, 4xx), в журналах сервиса стоит отслеживать `sapi_aim_web_service`, в который перенаправляются запросы этого типа.

Пример

```
grep -h 'POST /wim/auth/clientLogin' /oap/icq/logs/sapi_aim_web_services-*.err.log
14/150409 COMPAT0: sajp_wrapper.c:1539 SAJP_SendResponse >>10.10.10.10 POST /wim/auth/
clientLogin "clientName=Android%20myteam&clientVersion=1.0&devId=on2fah4R-
android&idType=ICQ&pwd=XXXXXXXXXXXXXX&s=tester%40example.com&tokenType=otp_via_email" 200
"200[0] OK" 0.091s [onpremise:2:38826580:1586865849.365] "myteam Android #no_user_id#
on2fah4R-android 1.0(9999999) Android_10_29 SM-G973F"
14/150851 COMPAT0: sajp_wrapper.c:1539 SAJP_SendResponse >>10.10.10.10 POST /wim/auth/
clientLogin "clientName=Android%20myteam&clientVersion=1.0&devId=on2fah4R-
android&idType=ICQ&pwd=XXXXXXXXXXXXXX&s=tester%40example.com&tokenType=otp_via_email" 200
"200[0] OK" 0.094s [onpremise:4:38826580:1586866131.237] "myteam Android #no_user_id#
on2fah4R-android 1.0(9999999) Android_10_29 SM-G973F
```

В примере имеются два запроса `clientLogin` для одного пользователя. В случае использования OTP это правильно, так как первый запрос вызывает отправку OTP-кода, а вторым запросом этот код отправляется на сервер.

Создание сессии

После авторизации пользователь может создавать новые сессии, которых может быть несколько при использовании различных устройств. Создание сессии выполняется запросом `POST /wim/aim/startSession.`

Пример

```
>grep -h 'startSession' /oap/icq/logs/sapi_aim_web_services-*.err.log
14/151808 COMPAT0: sajp_wrapper.c:1539 SAJP_SendResponse >>10.10.10.10 POST /wim/aim/
startSession
"a=%252Fw8BAAAAAADJhgEAAAAAAKz6rxWnlkhWnNX70ISPgqMAAAAXZC5hdmV0aXNvdkBjb3JwLm1haWwucnUAAAAFZW1
hXXXXXXXX&androidExtraPns=certV%3Dmyteam&assertCaps=094613594C7F11D18222444553540000%2C2ACCFA1
AF270424598B39992C6531866%2C1F99494E76CBC880215D6AEAB8E42268%2C0946135A4C7F11D1822244455354000
0%2C0946135C4C7F11D18222444553540000%2C094613574C7F11D18222444553540000%2C094613503c7f11d18222
444553540000%2CB5ED3E51C7AC4137B5926BC686E7A60D&buildNumber=99999&clientName=Android%20myteam&
clientVersion=1.0&deviceId=dit-
cd527a2f2cdeff1a&events=myInfo%2Cpresence%2Cbuddylist%2Ctyping%2CwebrtcMsg%2CMChat%2Creplace%2
CpermitDeny%2Cdiff%2Chist%2ChiddenChat%2CimState%2Cnotification%2Capps&gaid=648bdc13-4ddc-4d70
-b80a-
fb80d2813a56&imf=plain&includePresenceFields=quiet%2Cssl%2CabFriendly%2Ccapabilities%2Cnick%2C
role%2CabPhones%2CaimId%2CautoAddition%2Cfriendly%2ClargeIconId%2Clastseen%2Cmute%2Cofficial%2
Cpending%2Cstate%2CeventType%2CseqNum%2CtimeSave&interestCaps=094613504C7F11D18222444553540000
%2C094613514C7F11D18222444553540000%2C094613503c7f11d18222444553540000%2C8EEC67CE70D041009409A
7C1602A5C84&invisible=false&k=on2fah4R-android&language=ru-
RU&minimizeResponse=0&mobile=1&pollTimeout=30000&rawMsg=0&sessionTimeout=31536000&ts=158686668
8&view=mobile" 200 "200[1] Ok" 0.004s [onpremise:3:39040498:1586866688.837] "myteam Android
tester@example.com on2fah4R-android 1.0(9999999) Android_10_29 SM-G973F"
```

Технический писатель: Белова Ирина

23 сентября 2025 г.