# Установка компонентов KUMA на отдельную машину

# Установка коллектора/коррелятора

Для установки дополнительного коллектора/коррелятора необходимо подготовить машину установив на нее поддерживаемую ОС и создав разделы в системе аналогично разделу «Подготовка» этой инструкции, а также другие компоненты KUMA должны быть доступны по сети от этой машины.

> С машины должен происходить резолв хостнеймов ядра и других компонентов, в случае отсутствия DNS пропишите IP в /etc/hosts. Доступы необходимы согласно **этой** схеме.

1. Скопировать исполняемый файл kuma с любого из установленных компонентов KUMA (либо пакета установки `/kuma-ansible-installer/roles/kuma/files/kuma`), создайте идентичную структуру папок:
   - `/opt/kaspersky/kuma/kuma`

2. Создайте пользователя и группу kuma на новой машине:

   ```
   useradd -Mrs /usr/bin/false kuma
   ```

3. Создайте структуру папок /opt/kaspersky/kuma/ на новой машине, добавьте папки correlator и collector.

4. Поменяйте владельца и группу исполняемого файла и всех подпапок и принимаете соглашение

   ```
   chmod +x /opt/kaspersky/kuma/kuma
   touch /opt/kaspersky/kuma/LICENSE
   chown kuma:kuma -R /opt/kaspersky/kuma/*
   /opt/kaspersky/kuma/kuma license
   ```

Далее процесс установки службы коллектора/коррелятора проходит аналогично стандартному процессу. Пример с DNS коллектором.

> Начиная с версии KUMA 2.1.3 доступен отдельный плейбук и скрипт для централизованной установки компонентов на отдельные сервера. Подробнее по

ссылке: https://support.kaspersky.com/help/KUMA/2.1/ru-RU/222160.htm

# Установка хранилища KUMA

Для установки дополнительного хранилища необходимо подготовить машину установив на нее поддерживаемую ОС и создав разделы в системе аналогично разделу «Подготовка» этой книги, а также другие компоненты KUMA должны быть доступны по сети от этой машины.

Действия необходимо выполнять из машины откуда происходит(ла) централизованное разворачивание системы.

Для актуальных версий (с 2.1.3.49) с помощью expand.inventory:

Перейдите в папку установки `kuma-ansible-installer`

```
cp expand.inventory.yml.template expand.inventory.yml
```

Отредактируйте файл expand.inventory.yml, пример установки хранилища на kuma-additional-storage-1.example.com ниже:

```
kuma:
  vars:
    ansible_connection: ssh
    ansible_user: root
  children:
    kuma_collector:
    kuma_correlator:
    kuma_storage:
      hosts:
        kuma-additional-storage-1.example.com:
```

Запустите начало установки, следующей командой:

```
PYTHONPATH="$(pwd)/ansible/site-packages:${PYTHONPATH}" python3 ./ansible/bin/ansible-playbook -i expand.inventory.yml expand.inventory.playbook.yml
```

Для KUMA от 3.x:

```
PYTHONPATH="$(pwd)/ansible/site-packages:${PYTHONPATH}" python3 ./ansible/bin/ansible-playbook -i expand.inventory.yml expand.playbook.yml
```

Для старых версий:

- Для установки отдельного хранилища (вне кластера ClickHouse) заполните данными о новой машине файл `additional-storage-cluster.inventory.yml.template` из инвентаря (его можно взять отсюда) из папки установки, поменяйте в скрипте установки install.sh плейбук (последняя строчка) с `install.playbook.yml` на `additional-storage-cluster.playbook.yml`, затем запустите установку.
- Для установки хранилища входящий в состав кластера ClickHouse) дополните данными о новой машине файл инвентаря `distributed.inventory.yml` (который ранее использовался для разворачивания системы) из папки установки.

> В случае добавления нового сервера с репликой копирование данных начнется с текущего времени

# Установка агента Linux

## Создание и публикация сервиса агента

1. Зайдите в веб-интерфейс KUMA и перейдите на вкладку **Ресурсы** – **Агенты**.

2. Нажмите на кнопку **Добавить агент**.

3. Задайте необходимые параметры для агента в соответствии с ограничениями Linux-агентов: https://support.kaspersky.ru/help/KUMA/2.1/ru-RU/217776.htm

4. Сохраните созданный ресурс агента.

5. Перейдите на вкладку **Ресурсы** – **Активные сервисы**.

6. Нажмите на кнопку **Добавить сервис**, выберите созданный ресурс агента и нажмите на кнопку **Создать сервис**.

7. Выберите галочкой созданный сервис агента и нажмите в верхней части экрана на кнопку **Копировать идентификатор**.

> Идентификатор будет скопирован в буфер обмена. Сохраните полученный таким образом идентификатор, он потребуется для дальнейшей установки сервиса агента.

---

## Установка агента в качестве службы

> В данном разделе под **<ID>** понимается значение идентификатора агента, который был скопирован в предыдущем разделе в п. 7.  Под **<KUMA-FQDN>** понимается FQDN ядра KUMA.

1. Если установка агента осуществляется на сервер, на котором уже установлены какие-либо компоненты KUMA, то шаги 2, 3, 5, 7 нужно пропустить.

2. Скопируйте из дистрибутива KUMA файл `kuma-ansible-installer/roles/kuma/files/kuma` и переместите его на сервер для установки агента в любую директорию.

3. Создайте пользователя для запуска агента следующей командой

```
useradd -Mrs /usr/bin/false kuma
```

4. Создайте рабочую директорию для агента

```
mkdir -p /opt/kaspersky/kuma/agent/<ID>
```

5. Скопируйте файл kuma в директорию /opt/kaspersky/kuma/

```
cp kuma /opt/kaspersky/kuma/
```

6. Назначьте пользователя kuma владельцем директории

```
chown -R kuma:kuma /opt/kaspersky/kuma/
```

7. Задайте права на выполнение файлу kuma

```
chmod +x /opt/kaspersky/kuma/kuma
```

8. Примите лицензионное соглашение

```
/opt/kaspersky/kuma/kuma license
```

На данном этапе можно выполнить ручной запуск агента для проверки его работоспособности. Для этого необходимо выполните команду

```
/opt/kaspersky/kuma/kuma agent --core https://<KUMA-FQDN>:7210 --id <ID> --wd
/opt/kaspersky/kuma/agent/<ID>
```

Для остановки выполнения воспользуйтесь комбинацией клавиш `Ctrl + C`

9. Создайте файл с описанием сервиса агента

```
touch /usr/lib/systemd/system/kuma-agent-<ID>.service
```

10. Любым удобным способом откройте созданный файл на редактирование и укажите в нем следующую конфигурацию

```
[Unit]
Description=KUMA Agent Syslog
StartLimitIntervalSec=1
After=network.target
```

```
[Service]
Type=notify
Restart=always
RestartPreventExitStatus=99
TimeoutSec=300
RestartSec=5
WatchdogSec=60

User=kuma
Group=kuma

ExecStartPre=+-chown kuma:kuma /opt/kaspersky/kuma/agent
ExecStartPre=+-chown -R kuma:kuma /opt/kaspersky/kuma/agent/<ID>
ExecStart=/opt/kaspersky/kuma/kuma agent --core https://<KUMA-FQDN>:7210 --id <ID> --wd
/opt/kaspersky/kuma/agent/<ID>/

LimitFSIZE=infinity
LimitCPU=infinity
LimitAS=infinity
LimitNOFILE=64000
LimitNPROC=64000
LimitMEMLOCK=infinity
TasksMax=infinity
TasksAccounting=false

[Install]
WantedBy=multi-user.target
```

11. Сохраните полученный файл
12. Выполните обновление конфигурации

```
systemctl daemon-reload
```

13. Запустите сервис агента

```
systemctl start kuma-agent-<ID>.service
```

14. Настройте сервису автозапуск

```
systemctl enable kuma-agent-<ID>.service
```

После установки сервиса агента индикация состояния в веб-интерфейсе KUMA на вкладке **Ресурсы** – **Активные сервисы** изменится на Зеленый. Также будет отображен FQDN и IP-адрес агента.

---

# Диагностика неполадок

В случае возникновения сбоев в работе агента или ошибок установки службы диагностическую информацию можно получить следующими способами:

1. Просмотр состояния службы

```
systemctl status kuma-agent-<ID>.service
```

2. Просмотр журнала службы

```
journalctl -f -u kuma-agent-<ID>.service
```

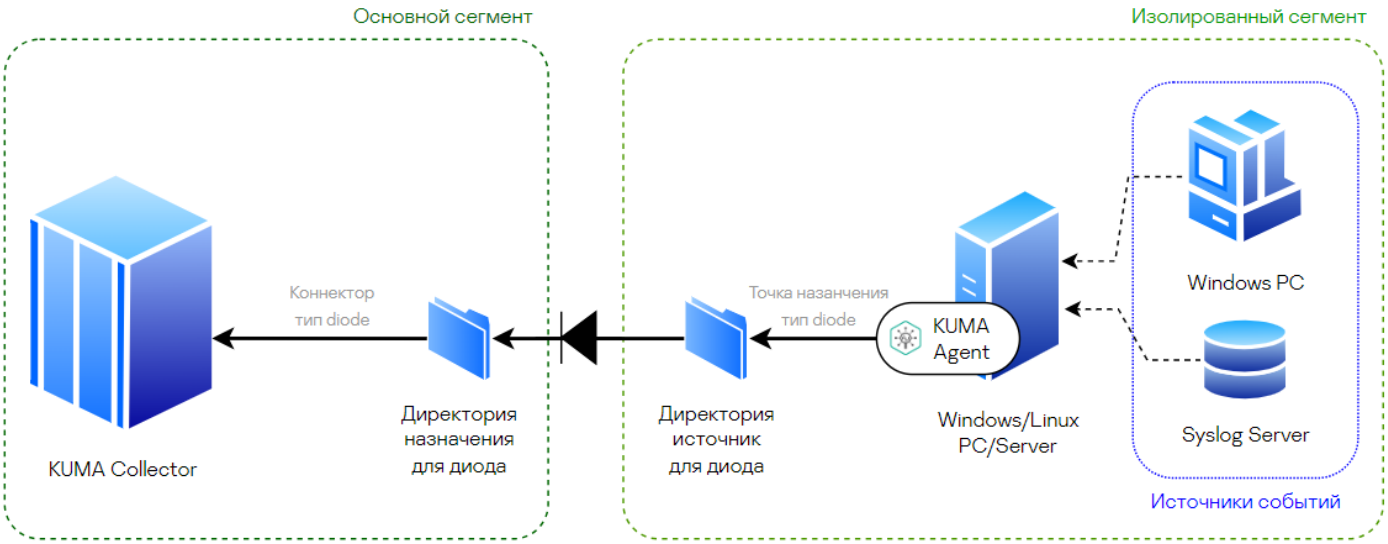
3. Просмотр журнала агента

```
tail -f /opt/kaspersky/kuma/agent/<ID>/log/agent
```

Для получения дополнительной информации в журнале агента необходимо в веб-интерфейсе KUMA в параметрах агента установить галочку рядом с параметром Отладка после чего выполнить обновление параметров сервиса агента.

# Установка агента в режиме диод (Diode)

## Схема работы сбора в режиме diode



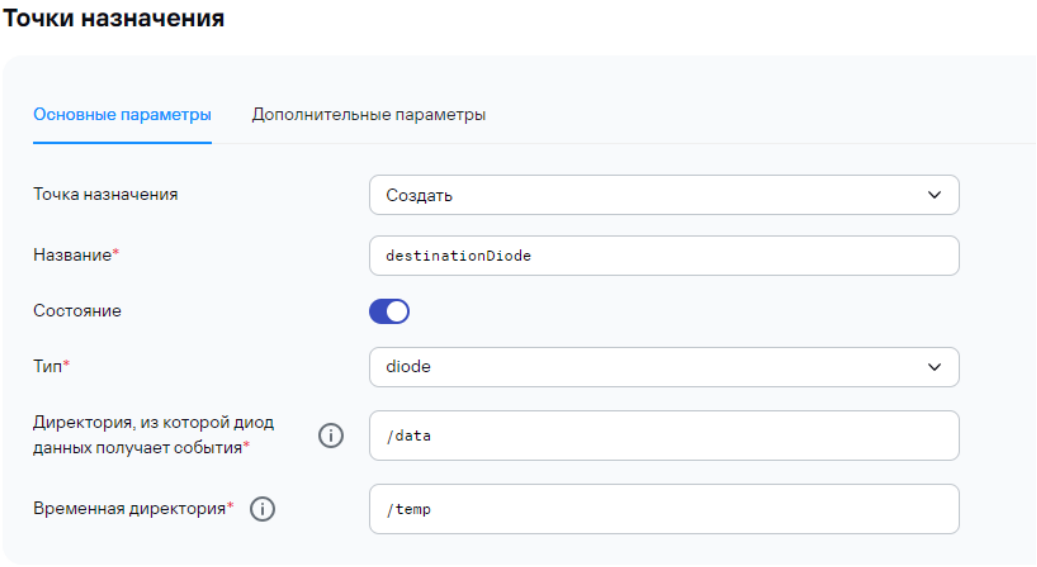

Агент, находящийся в изолированном сегменте сети, собирает события с источников и перемещает их в директорию источник, откуда их забирает диод (дата-диод). Диод переносит файлы в директорию назначения основной сети, удаляя их директории источника. Из директории назначения события собирает коллектор, удаляя их после считывания.

Для осуществления описанного способа передачи событий через диод используется пара destination и connector типа diode, destination на агенте и connector на коллекторе. Агент может иметь любые из возможных типов connector, а коллектор - любые из возможных destination.

Агент при работе накапливает события в буфер. Как только буфер становится размером >=bufferSize (по умолчанию 64 Мб), или с момента предыдущей записи буфера в файл проходит > FlushInterval (по умолчанию 10 сек):

- Агент записывает события в файл во временной директории, указанную пользователем
- Агент переносит файл из временной папки в "Директорию, из которой диод данных получает события (Data diode source directory)" , попутно переименовывая файл. Название файла содержит sha256 хеш содержимого для возможности

осуществления проверки целостности.




По умолчанию сжатие файлов не происходит, но его можно осуществлять, если выбрать в настройках destination данную опцию.  В этом случае для корректной работы тот же алгоритм должен быть указан в соответствующем diode connector.

При считывании (diode connector) sha256 хеш содержимого файла сравнивается с хешем из имени файла, при несоответствии файл удаляется и создается событие аудита.

> Ресурс точки назначения в агенте должен иметь тип diode. В этом ресурсе необходимо указать директорию, из которой диод данных будет перемещать файлы во внешний сегмент сети.

> Для diode-агента невозможно выбрать коннекторы типа sql или netflow.

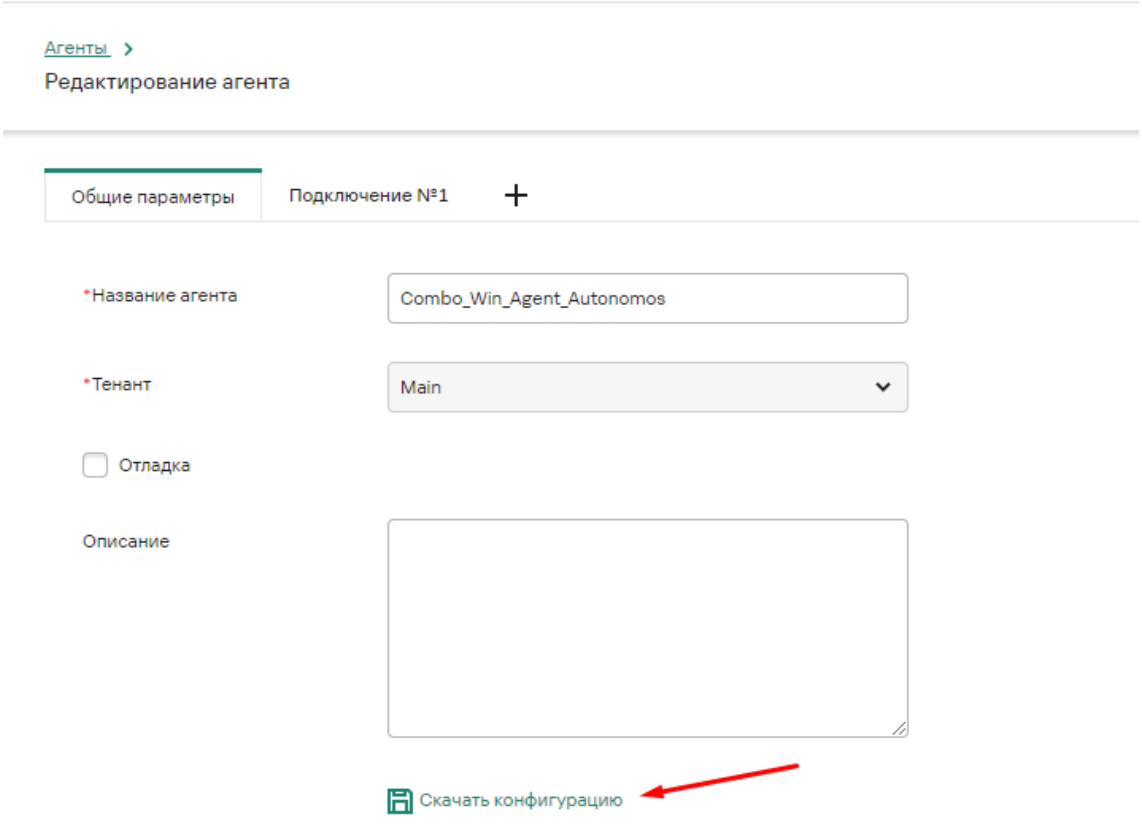
# Конфигурационный файл

Конфигурацинный файл агента сохраняется в человекочитаемом виде для возможности добавления секретов вручную. Для избежания сохранения секретов в открытом виде, содержимое реальных секретов заменяется на шаблоны соответствующего типа секрета, также в конфигурационном файле генерируются шаблоны в местах, где это явно указано (см п. Шаблоны секретов). В ресурсах внутри агента, использующих секреты указан UUID секрета, содержимое секретов находится отдельно в поле secrets. Данные секретов можно заполнить вручную в конфигурационном файле, изменяя поля секретов.

Далее в таблице описаны поля секрета.

| Имя поля | Тип | Описание |
|---|---|---|
| user | строка | Имя пользователя |
| password | строка | Пароль |
| token | строка | Токен |
| urls | массив строк | Список url |
| publicKey | строка | Публичный ключ (используется в PKI) |
| privateKey | строка | Приватный ключ (используется в PKI) |
| pfx | строка, содержащая base64 закодированное содержимое pfx | Содержимое pfx файла, закодированное в base64. На linux получить base64 кодировку файла можно при помощи команды<br>base64 -w0 src > dst |
| pfxPassword | строка | Пароль от pfx |
| securityLevel | строка | Используется в snmp3. Возможные значения: NoAuthNoPriv, AuthNoPriv, AuthPriv |
| community | строка | Используется в snmp1 |
| authProtocol | строка | Используется в snmp3. Возможные значения: MD5, SHA, SHA224, SHA256, SHA384, SHA512 |
| privacyProtocol | строка | Используется в snmp3. Возможные значения: DES, AES |
| privacyPassword | строка | Используется в snmp3 |
| certificate | строка, содержащая base64 закодированное содержимое pem | Содержимое pem файла, закодированное в base64.<br>На linux получить base64 кодировку файла можно при помощи команды<br>base64 -w0 src > dst |

Конфигурационный файл скачивается из веб-интерфейся ядра KUMA в части настроек агента:




# Запуск агента (автономный агент)

> При установке агента его конфигурационный файл не должен находиться в директории, в которую устанавливается агент.

> Необходимо при помощи списка контроля доступа (ACL) настроить права доступа к конфигурационному файлу так, чтобы доступ на чтение файла был только у пользователя, под которым будет работать агент.

> TLS не работает, тк требуется подключение к ядру.

Справочная информация об установщике доступна по команде: `kuma.exe help agent`

# Linux

Примите лицензионное соглашение: `/opt/kaspersky/kuma/kuma license`

Для запуска агента требуется скопировать файл /opt/kaspersky/kuma/kuma с машины, где установлена KUMA на машину с linux, где будет запущен агент и запустить его: `./kuma agent --cfg <path to config file> --wd <path to working directory>`
Через опцию --wd указывается путь для хранения файлов агента, по умолчанию они будут храниться в текущей директории.

Конфигурационный файл может содержать секреты, его следует защищать при помощи ACL, позволяющих чтение только пользователю KUMA (600).

# Windows

Примите лицензионное соглашение: `/opt/kaspersky/kuma/kuma.exe license`

Без установки

```
kuma.exe agent --cfg <path to config file>
```

```
kuma.exe  agent --cfg <путь к конфигурационному файлу агента> --wd <путь к директории, где будут размещаться файлы устанавливаемого агента. Если не указывать этот флаг, файлы будут храниться в директории, где расположен файл kuma>
```

С установкой

```
kuma.exe agent --cfg <path to config file> --user <user to start service as> --install
```

При установке используемый конфигурационный файл перемещается в рабочую директорию (`ProgramData\Kaspersky Lab\KUMA\agent\<serviceID>\`) (ID берется из ресурса агента в конфигурационном файле), `kuma.exe` перемещается в рабочую директорию (`Program Files\Kaspersky Lab\KUMA`).

В дальшейшем используется перемещенный конфигурационный файл.

Конфигурационный файл может содержать секреты, его следует защищать при помощи ACL, позволяющих чтение только пользователю от чьего лица запускается KUMA.

---

# Удаление

```
kuma.exe agent --cfg <path to config file> --uninstall
```

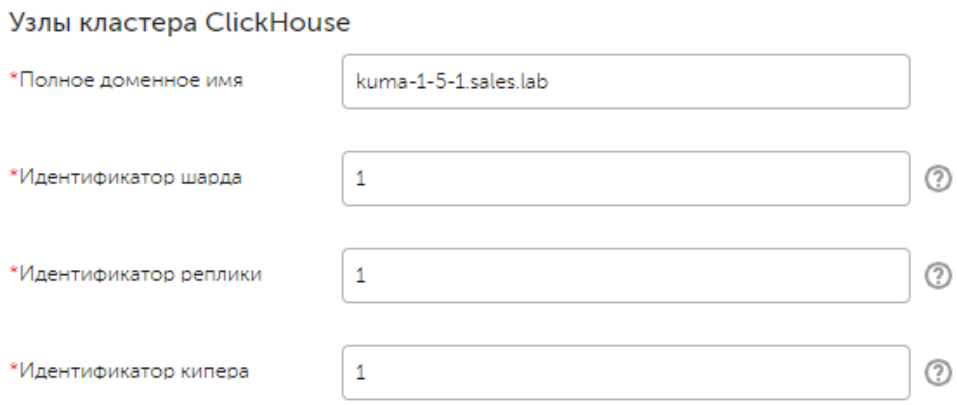
или

```
kuma.exe agent --id <идентификатор сервиса агента, созданного в KUMA> --uninstall
```

# Установка службы хранилища (если этого не произошло при установке)

> Данная инструкция применима только в случае, если KUMA была успешно установлена, но служба хранилища не была развернута из демонстрационных ресурсах. Инструкция приведенная ниже подразумевает, что все действия и команды выполняются на серверах с размещенными файлами clickhouse и созданными учетными записями.
>
> В противном случае следует воспользоваться инструкцией: https://kb.kuma-community.ru/books/ustanovka-i-obnovlenie/page/ustanovka-xranilishha-kuma

Создаем сервис хранилища перейдите в **Ресурсы – Хранилища** затем нажать на кнопку Добавить хранилище придумайте название и затем укажите количество дней хранения событий и событий аудита (от 365 дней срок хранения аудита) и узлы кластера (от версии 2.1) в соответствии с проведенной установкой, ниже пример для All-In-One установки:

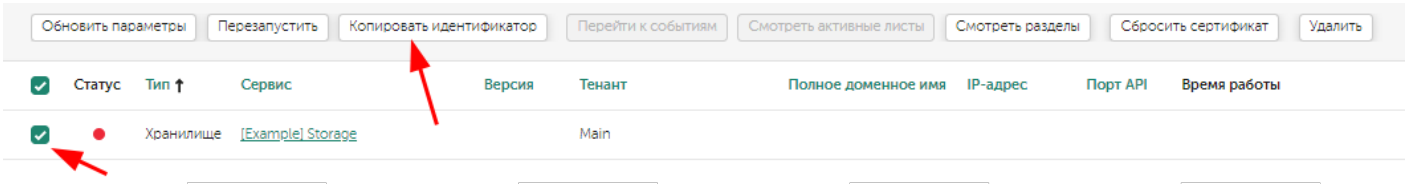



затем нажмите **Сохранить**.

Публикуем созданный сервис **Ресурсы – Активные сервисы** затем выбрать созданный ранее сервис и нажать на кнопку **Создать сервис**.

> В случае кластера хранилищ, опубликуйте этот же сервис по количеству отдельных машин хранилищ и keeper'ов

Скопируйте идентификатор сервиса:




Зайдите по ssh на сервер где будет разворачиваться сервис хранилища и выполните команду (api.port 7230 можно использовать произвольный никем не занятый порт), сначала выполним проверку (без установки, ошибки приналичии будут отображаться в консоли):

> Сначала происходит установка keeper'ов, затем нод Clickhouse

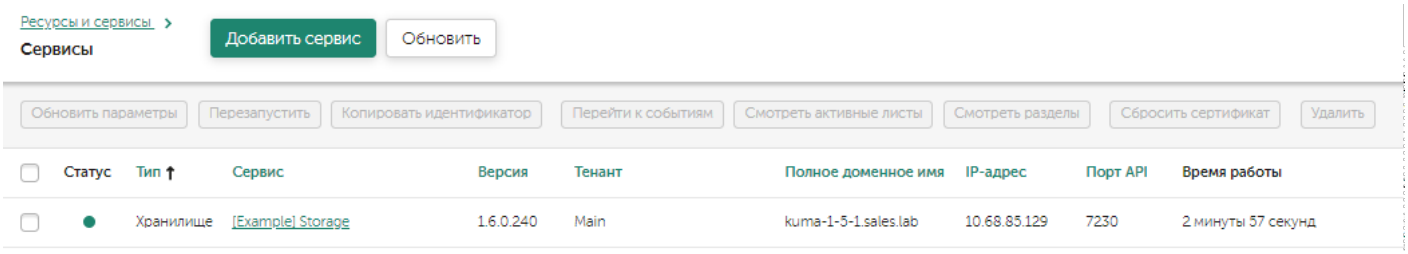
> У каждой из машин хранилищ и keeper'ов  должен быть свой уникальный идентификатор

```
sudo -u kuma /opt/kaspersky/kuma/kuma storage --id <ВАШ_ИДЕНТИФИКАТОР> --core
https://<FQDN/ИМЯ_ХОСТА_СЕРВЕРА_ЯДРА>:7210 --api.port 7230
```

Если ошибок нет, то устанавливаем службу хранилища:

```
/opt/kaspersky/kuma/kuma storage --id <ВАШ_ИДЕНТИФИКАТОР> --core
https://<FQDN/ИМЯ_ХОСТА_СЕРВЕРА_ЯДРА>:7210 --api.port 7230 --install
```

В разделе **Ресурсы – Активные сервисы** убедитесь, что служба работает более 30 секунд с «зеленым» статусом индикатора:




Далее создадим точку назначения, которая используется в маршрутизации событий, перейдите в **Ресурсы – Точки назначения**, затем нажмите на кнопку **Добавить точку назначения**. Придумайте название и затем в поле URL укажите FQDN и порт службы хранилища, например: `kuma-1-5-1.sales.lab:7230`, затем нажмите **Сохранить**.

Аналогичные действия понадобятся для установки остальных компонентов, только в интерфейсе будет доступна команда, которую необходимо будет выполнить для установки службы.

Для удаления службы хранилища (если необходимо), можно использовать следующую команду:

```
/opt/kaspersky/kuma/kuma storage --id <ВАШ_ИДЕНТИФИКАТОР> --uninstall
```