

Логикор. Автоматическое развертывание с помощью Ansible

В данной статье описано автоматическое развертывание системы хранения и обработки данных Логикор с помощью плейбуков Ansible.

Подготовка серверов

На каждом сервере должна быть выполнена [интеграция с корпоративным NTP-сервером](#).

Настройка SSH-подключения

OpenSSH должен быть настроен на стандартный 22 порт.

Для настройки SSH-подключения необходимо выполнить следующие действия:

Шаг 1. Если для развертывания Логикор имеется потребность в использовании непривилегированного пользователя, отличного от root, то необходимо создать целевого SSH-пользователя на всех **control-** и **managed-серверах**, если этого не было сделано ранее. Для этого на всех **control-** и **managed-серверах** выполните следующие команды:

```
sudo useradd -m -s /usr/bin/bash -g users -G sudo <целевой_ssh_пользователь>
sudo groupadd <целевой_ssh_пользователь>
```

Шаг 2. Если целевой пользователь на **control-** и **managed-серверах** является непривилегированным, отличным от root, то на каждом из них необходимо добавить пользователя в sudo с отключением ввода пароля:

```
echo '<целевой_ssh_пользователь> ALL=(ALL) NOPASSWD: ALL' | sudo tee -a /etc/sudoers
```

Шаг 3. На **control-сервере** выполните генерацию SSH-ключей под целевым пользователем, под которым будет запускаться развертывание Логикор:

```
su - <целевой_ssh_пользователь>
ssh-keygen -t rsa
```

Шаг 4. Скопируйте публичный SSH-ключ с **control-сервера** на каждый **managed-сервер** под целевым SSH-пользователем:

```
ssh-copy-id <целевой_ssh_пользователь>@<IP_адрес_или_FQDN_managed_сервера_1>
ssh-copy-id <целевой_ssh_пользователь>@<IP_адрес_или_FQDN_managed_сервера_2>
ssh-copy-id <целевой_ssh_пользователь>@<IP_адрес_или_FQDN_managed_сервера_N>
```

Загрузка сборки Логикор для офлайн-развертывания

Для офлайн-развертывания Системы необходимо загрузить на **control-сервер** офлайн-версию Логикор и затем распаковать ее. Для этого выполните следующие действия:

Шаг 1. Подключите USB-накопитель к устройству, с которого имеется доступ по SSH на control-сервер и загрузите офлайн-сборку на него:

```
scp <путь_к_офлайн-сбоке_на_USB-накопителе> <целевой_ssh_пользователь>@<адрес_control_сервера>:/opt/logiq_<версия_Логикор>-offline_amd64.deb
```

Шаг 2. Вернитесь в терминал управления control-сервера и установите deb-пакет:

```
sudo dpkg -i /opt/logiq_<версия_Логикор>-offline_amd64.deb
```

Подготовка конфигурационного файла Логикор

Общая информация

Конфигурация Логикор описывается в конфигурационном файле, представляющем собой Ansible Inventory-файл.

В каталоге `./ansible/inventories/presets` хранятся пресеты для различных конфигураций Логикор, которые можно использовать как есть (за исключением пресетов, в которых необходимо задать параметры хостов) для развертывания Логикор или для клонирования одного из пресетов с последующей перенастройкой Логикор под специфические условия и окружение:

- AIO-docker-offline.yml** — All-in-One-конфигурация для офлайн-развертывания Логикор через Docker Compose локально на одном сервере. В качестве control- и managed- выступает один и тот же сервер;
- AIO-helm-offline.yml** — All-in-One конфигурация для офлайн-развертывания Логикор через Helm Chart в кластере K8s;
- AIO-localhost-offline.yml** — All-in-One-конфигурация для офлайн-развертывания Логикор локально на одном сервере. В качестве **control** и **managed** выступает один и тот же сервер/

Путь к конфигурационному файлу Логикор можно указать через флаг `-f </conf/path>`. Если параметр не указан, то по умолчанию используется путь `/etc/logiq/config.yml`.

Рекомендуется использовать путь по умолчанию `/etc/logiq/config.yml` для хранения конфигурации Логикор на **control-сервере**, чтобы исключить риск перезаписи конфигурационного файла при обновлении каталога со сборкой Логикор.

Процесс подготовки конфигурационного файла

Для того чтобы подготовить конфигурационный файл, выполните на control-сервере следующую команду находясь в каталоге сборки Логикор:

```
sudo cp /usr/share/logiq/ansible/inventories/presets/<имя_выбранного_пресета>.yml /etc/logiq/config.yml
```

Пример конфигурационного файла

Ниже приведен пример структуры конфигурационного файла Логикор и значение параметров:

```
# Определение целевых хостов для Логикор. По умолчанию определяется только локальный хост `logiq-control`.
hosts: {}
# Примеры:
#
# - name: "<имя_целевого_хоста>"
#   address: "<IP-адрес|FQDN_целевого_хоста_или_переменная_ansible>"
#   connection: "<local|ssh|docker>" # Опциональный параметр с указанием типа подключения ansible на целевые хосты. Если не указать, то будет использоваться значение 'ssh'.
#   user: "<ssh_username>" # Обязательный параметр с указанием имени SSH-пользователя, если `connection: "ssh"`. Альтернатива параметру: задать переменную `ansible_user` в секции `variables_overr`
#
# - name: "logiq-control"
#   address: "127.0.0.1"
#   connection: "local"
# - name: "logiq-storage-01"
#   address: "10.10.10.10"
#   connection: "ssh"
#   user: "root"

# Переопределение целевых хостов для конкретных компонентов Логикор. По умолчанию для всех компонентов используется локальный хост `logiq-control`.
# Список доступных компонентов: clickhouse, postgres, s3, apigw, cadvisor, aptly, registry, vault, portainer, mpsiem_rabbitmq, venus, earth, saturn, mercury, prometheus, grafana, openldap, mailhog
targets_overrides: {}
#
# Пример:
#
# <наименование_компонента>:
#   hosts:
#     - <значение_параметра `name` целевого хоста из секции `hosts` >
#
# clickhouse:
#   hosts:
#     - example-logiq-storage-01
#     - example-logiq-storage-02

# Переопределение переменных с параметрами развертывания Логикор и его компонентов.
# Перечень доступных переменных можно посмотреть в /usr/share/logiq/ansible/inventories/defaults.yml.
variables_overrides:
  deployment_type: "docker-compose" # Обязательный параметр с указанием типа развертывания. Возможные значения: docker-compose, helm
  offline_mode: "true"
  init_aptly: "true"
  use_repo_aptly: "true"
  logiq_taxonomy: "mpsiem" # В данном параметре можно выбрать формат регистрации событий. Возможные значения: mpsiem, json.

Для того чтобы выбрать формат записи событий, необходимо установить его значение для параметра logiq_taxonomy в конфигурационном файле. Для отображения в формате MPSIEM установите значение "mpsiem" (значение по умолчанию), для формата JSON — "json".
```

Вы можете изменить формат записи событий после развертывания системы, следуя [ИНСТРУКЦИИ](#), но в таком случае все записанные события и пользовательские настройки системы будут удалены.

Запуск процесса подготовки хостов к развертыванию Логикор

Данный подготовительный шаг необходимо выполнять только для Docker_compose-развертывания.

Для того чтобы запустить процесс подготовки хостов к развертыванию Логикор, выполните на **control-сервере** следующую команду находясь в каталоге сборки Логикор:

```
# опциональный параметр: -f <путь_к_конфигурационному_файлу_Логикор>:
logiq prepare-host
```

Процесс развертывания Логикор

Развертывание Логикор в любом из режимов запускается из каталога сборки. Для этого на **control-сервере** выполните следующую команду:

```
# опциональный параметр: -f <путь_к_конфигурационному_файлу_Логикор>:
logiq deploy --components <список_сервисов_и/или_модулей>
```

Примеры запуска команд

Установка всех компонентов Логикор

```
logiq deploy --components all
```

Установка службы vault и модуля services

```
logiq deploy --components vault,services
```

Установка нескольких модулей и служб

```
logiq deploy --components vault,s3,services,openldap,monitoring
```

Справка об использовании скрипта развертывания Логикор

```
logiq --help
Справка об использовании скрипта развертывания Логикор
```

Команды:

```
prepare-host                Подготовить хосты и окружение к развертыванию или обновлению Логикор
Использование: logiq prepare-host [--ansible-tags <tag1,tag2,...>]
Оptionальные параметры:
--ansible-tags <tag1,tag2,...>          Указать теги для запуска определенных задач Ansible. Если параметр не указан, то по умолчанию указываются все доступные Ansible теги
Доступные Ansible теги:
security_disable,set_sysctl,init_aptly,os_packages_install,docker_install
```

```
deploy|uninstall           Развернуть/обновить или удалить компоненты Логикор
Использование: logiq <deploy|uninstall> --components <module1,module2|service1,service2...>
Обязательные параметры:
--components <module1,module2|service1,service2...>  Список модулей и/или сервисов для развертывания
```

```
ansible                    Прямой вызов ansible, путь к инвентарю необходимо передать через -f </conf/path> или создать его по пути /etc/logiq/config.yml
Использование: logiq ansible <ansible commands and flags>
```

```
ansible-playbook           Прямой вызов ansible-playbook, путь к инвентарю необходимо передать через -f </conf/path> или создать его по пути /etc/logiq/config.yml
Использование: logiq ansible-playbook <ansible-playbook commands and flags>
```

```
get-config                 Вывести текущую полную конфигурацию Логикор
Использование: logiq get-config
```

```
get-example-conf          Вывести пример конфигурационного файла Логикор
Использование: logiq get-example-conf
```

Дополнительные флаги:

```
-f </conf/path>          Путь к конфигурационному файлу Логикор. Если параметр не указан, то по умолчанию используется /etc/logiq/config.yml
--ansible-opts '<--param1 --param2 -P3>'  Дополнительные параметры Ansible, например, '-diff --check --list-tasks --limit'
-h|--help|help          Вывести эту справку
-v|--version|version    Вывести версию Логикор
```

Список доступных модулей и сервисов Логикор:

```
registry: registry
tools: portainer,vault
storage: clickhouse,s3,postgres,mpsiem_rabbitmq
external: openldap,mailhog
services: earth,apigw,mercury,saturn,venus,saturn-agent
monitoring: prometheus,grafana,cadvisor,saturn-metrics-exporter
```

Для выбора всех вышеперечисленных модулей укажите: all

Пример развертывания Логикор с внешним слоем хранения

```
# Выполняем развертывание Docker Registry и tools (portainer, vault)
logiq deploy --components registry,tools
```

```
# Создаем файл export_hashi_vault_secrets.json с секретами Логикор с тестовыми значениями
cat <<EOF | sudo tee export_hashi_vault_secrets.json
{"clickhouse": {"CLICKHOUSE_DB": "logger", "CLICKHOUSE_PASSWORD": "testpass", "CLICKHOUSE_USER": "logger"}, "ldap": {"LDAP_ADMIN_USERNAME": "admin", "LDAP_PASSWORDS": "testpass1,testpass2", "LDAP_L
EOF
```

```
# Запускаем процесс импорта секретов в Hashicorp Vault из файла export_hashi_vault_secrets.json
logiq ansible-playbook ./ansible/playbooks/mgmt/import_hashi_vault_secrets.yml
```

```
# Заходим в веб-интерфейс Hashicorp Vault и актуализируем значения секретов (s3, postgres, clickhouse) на целевые значения
# http://<IP_адрес_или_FQDN_менеджера_vault>:8200
# Для того чтобы получить Vault Root Token используйте команду
logiq ansible-playbook ./ansible/playbooks/mgmt/get_hashi_vault_token.yml
```

```
# После актуализации секретов на целевые значения выполняем экспорт всех секретов Логикор из Hashicorp Vault и копируем их в надежное место для возможности быстрого восстановления в случае п
logiq ansible-playbook ./ansible/playbooks/mgmt/export_hashi_vault_secrets.yml
cat export_hashi_vault_secrets.json
```

```
# Запускаем развертывание служб и мониторинга
logiq deploy --components services,monitoring
```

Если вы используете внешний ClickHouse и хотите использовать JSON-формат записи событий, необходимо в конфигурационном файле **config_clickhouse_profile.xml** (или его аналоге) активировать флаг работы с JSON: **<enable_json_type>1</enable_json_type>**.

FAQ

Что такое control-сервер?

Control-сервер — это хост, с которого вы запускаете `logiq` и инструменты Ansible CLI (ansible-playbook, ansible, ansible-vault и другие).

Вы можете использовать любой компьютер, который соответствует требованиям к программному обеспечению, в качестве control сервера: например, ноутбук с Ubuntu/Debian (или Windows с WSL2), виртуальную машину с Ubuntu/Debian. Вы также можете запускать Ansible в контейнерах, известных как [Execution Environments](#)

Что такое managed-серверы?

Managed-серверы — это целевые хосты, которыми вы собираетесь управлять с помощью Ansible и на которые планируется развертывание компонентов Логикор.

Что означает параметр `--components` командах `logiq deploy` или `logiq uninstall`?

В обязательном параметре `--components` необходимо перечислить список модулей и/или сервисов LogIQ, которые планируется развернуть, например, `logiq deploy --components <список_сервисов_и/или_модулей>` или `logiq uninstall --components <список_сервисов_и/или_модулей>`. Подробнее описано в справке `logiq --help`.

Какие модули необходимо указать в параметре `--components` при запуске команд `logiq deploy --components <список_сервисов_и/или_модулей>` или `logiq uninstall --components <список_сервисов_и/или_модулей>`?

Прочитайте справку `logiq --help`.

Как указать нужные версии сервисов?

Посмотрите файл stand.env, в нем можно указать какие версии сервисов использовать при развертывании Логикор.

Как изменить конфигурацию Логикор при развертывании?

Конфигурация Логикор описывается в конфигурационном файле, представляющем собой Ansible Inventory-файл. В каталоге presets хранятся пресеты для различных конфигураций Логикор, которые можно использовать как есть для развертывания Логикор или для клонирования одного из пресеты с последующей перенастройкой Логикор под специфичные условия и окружение.

Как остановить работу служб?

Для остановки работы служб используйте следующую команду:

```
`logiq uninstall --components <список_служб_и/или_модулей> -f <путь_к_конфигурационному_файлу_Логикор>`
```

ID статьи: 151

Последнее обновление: 20 мар., 2026

Обновлено от: Егоров В.

Ревизия: 16

База знаний Логикор -> Документация -> Логикор. Автоматическое развертывание с помощью Ansible

<https://docs.axel.pro/entry/151/>