

Инструкция по установке экземпляра программного обеспечения «Скала^р Визион»

Контакты технического специалиста, который может проконсультировать по процессу развёртывания и настройки экземпляра ПО и его функционирования:

Кухтин Андрей

AKukhtin@skala-r.ru

+7 911 129 7339

Telegram ID: <https://t.me/akuhtin>

Развертывание «Скала^р Визион».

1. Развертывание «Скала^р Визион» на выделенный хост из дистрибутива.

В данный момент «Скала^р Визион» работает под управлением ALT 8 SP Server. Дистрибутив «Скала^р Визион» представляет из себя архив, содержащий необходимые пакеты, так что установка может производиться даже на систему, не имеющую доступа к сети Internet.

Состав архива:

- prometheus 2.42.0
- alertmanager 0.25.0
- grafana 9.2.8
- node_exporter 1.5.0
- ha_cluster_exporter 1.3.1
- postgres_exporter 0.11.1
- snmp_exporter 0.21.0
- ipmi_exporter 1.6.1
- blackbox_exporter 0.23.0
- snmp_notifier 1.4.0
- victoria-metrics 1.87.1
- vmutils 1.87.1
- vision-api
- vision-core
- vision-task-daemon
- vision-ui
- utlz_exporter
- postgresql12
- postgresql12-contrib
- python3-module-requests
- python3-module-psycorg2
- python3-module-pip
- python3-module-qunicorn
- python3-module-bottle

- python3-module-cerberus
- python3-module-paramico

2. Установка пакетов

На машине, с которой выполняется развертывание, требуются наличие следующих пакетов:

- ansible
- sshpass
- wget

Установить данные пакеты можно с помощью следующих команд (в процессе установки согласиться на установку зависимостей):

```
sudo apt update
```

```
sudo apt install ansible, wget, sshpass
```

Распаковать файл с дистрибутивом «Скала[^]р Визион» в директорию /root/vision. Дальнейшие команды, если не указано другое, выполняются из этой директории.

2.2 Конфигурирование мониторинга

Для настройки мониторинга необходимо заполнить файл inventory с параметрами развёртывания «Скала[^]р Визион» согласно инсталляционной карте. Для примера ориентируйтесь на файл inventory.yml. Далее по тексту он будет использоваться во всех примерах команд.

В файле укажите:

- в разделе `all` ➤ `vars` (глобальные настройки):

- `ansible_user` и `ansible_ssh_pass` - креды по-умолчанию для доступа по ssh для всех узлов
- `pak_type` - тип ПАКа (значения из списка: PG, ADQM, ADB, ADH, ADS, ADG)
- `pak_name` - имя ПАКа, например "PG-P-1"

- в разделе `vision` (ВМ «Скала[^]р Визион»):

- в разделе `all` ➤ `children` ➤ `vision` ➤ `hosts` ➤ `vm_vision` (настройки ВМ Визиона и его таргетов согласно инсталляционной карте):

- `ansible_host` - ip-адрес целевой ВМ «Скала[^]р Визион»
- `node_targets` - список узлов ПАК с параметрами
- `vm_targets` - список виртуальных машин ПАК (не включая ВМ «Скала[^]р Визион») с параметрами
- `snmp_targets` - список устройств, отслеживаемых системой мониторинга по протоколу SNMP, с параметрами
- `ipmi_targets` - список устройств, отслеживаемых системой мониторинга по протоколу IPMI, с параметрами

- в разделе `all` ➤ `children` ➤ `vision` ➤ `hosts` ➤ `pak_hosts` ➤ `pak_hosts` (настройки узлов согласно инсталляционной карте):

- имя узла
- `ansible_host` - ip-адрес целевого узла
- `mbd_pg_user` и `mbd_pg_pass` - креды для postgres на узлах кластера МБД.П
- `is_ha_cluster` - признак того, что хост является узлом кластера МБД.П

Внимание! При необходимости, для файла `inventory.yml` можно выполнить быстрое развертывание и конфигурирование мониторинга путем запуска скрипта `./deploy_and_config.sh` (этапы 1-6).

3. УСТАНОВКА

Развертывание мониторинга происходит в несколько этапов:

3.1 Подготовительный этап

Подготавливается хост, копируется дистрибутив и устанавливаются требуемые пакеты
`ansible-playbook -i inventory.yml -e "target=vision" 1_initial.yml`

3.2 Установка ядра мониторинга

Производится установка `prometheus`, `node_exporter`, `alertmanager`, `grafana` и `nginx`.

```
ansible-playbook -i inventory.yml -e "target=vision" 2_engine.yml
```

3.3 Установка сервера приложения «Скала^р Визион»

```
ansible-playbook -i inventory.yml -e "target=vision" 3_vision.yml
```

При необходимости выполнить этапы 1-3 можно командой

```
ansible-playbook -i inventory.yml -e "target=vision" 1_initial.yml 2_engine.yml 3_vision.yml
```

3.4 Развертывание агентов

Развертывание выполняется запуском команды

```
ansible-playbook -i inventory.yml -e "target=pak_hosts" 4_agent.yml
```

3.5 Конфигурирование таргетов «Скала^р Визион»

Конфигурирование выполняется запуском команды

```
ansible-playbook -i inventory.yml -e "target=vision" config_vision.yml
```

3.6 Конфигурирование экспортеров «Скала^р Визион»

Конфигурирование выполняется запуском команды

```
ansible-playbook -i inventory.yml -e "target=pak_hosts" config_agent.yml
```

Если установка прошла успешно, то результат будет следующим (пример):

```
PLAY RECAP *****
master      : ok=11  changed=9  unreachable=0  failed=0  skipped=1  rescued=0
  ignored=0
segment1    : ok=11  changed=9  unreachable=0  failed=0  skipped=1  rescued=0
  ignored=0
segment2    : ok=11  changed=9  unreachable=0  failed=0  skipped=1  rescued=0
  ignored=0
segment3    : ok=11  changed=9  unreachable=0  failed=0  skipped=1  rescued=0
  ignored=0
smaster     : ok=11  changed=9  unreachable=0  failed=0  skipped=1  rescued=0
  ignored=0
```

Если в процессе установки возникли ошибки, они будут выделены красным цветом.

Последовательно выполнить команды по настройке агента утилизации

Зайти в веб-интерфейс Prometheus и убедиться в наличии зелёных индикаторов в столбце State (пример):

Prometheus Alerts Graph Status Help

Targets

All Unhealthy Expand All

node_exporter (5/5 up) [View all](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.184.13:9101/metrics	UP	instance="master" job="node_exporter"	1d 18h 59m 2s ago	97.551ms	
http://192.168.184.12:9101/metrics	UP	instance="master" job="node_exporter"	1d 18h 59m 3s ago	93.047ms	
http://192.168.184.6:9101/metrics	UP	instance="supernode1" job="node_exporter"	1d 18h 58m 57s ago	97.924ms	
http://192.168.184.7:9101/metrics	UP	instance="supernode2" job="node_exporter"	1d 18h 59m 5s ago	112.415ms	
http://192.168.184.8:9101/metrics	UP	instance="supernode3" job="node_exporter"	1d 18h 58m 57s ago	108.192ms	

postgres_exporter (1/1 up) [View all](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://127.0.0.1:9187/metrics	UP	instance="127.0.0.1:9187" job="postgres_exporter"	1d 18h 59m 5s ago	1.2s	

prometheus (1/1 up) [View all](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://127.0.0.1:9090/vision/prometheus/metrics	UP	instance="127.0.0.1:9090" job="prometheus"	1d 18h 58m 58s ago	4.720ms	

4. УСТРАНЕНИЕ НЕПОЛАДОК

При заходе на страницу Grafana не отображаются никакие данные

- Проверить и, при необходимости, выполнить синхронизацию времени между хостом с «Скала^р Визион», хостами и АРМ инженера.