

**ИНСТРУКЦИЯ ПО УСТАНОВКЕ И ИСПОЛЬЗОВАНИЮ
МОДУЛЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
PMCONTROLLING: PMCONTROLLING SECURITY**

Оглавление

1. ВВЕДЕНИЕ	3
2. ТРЕБОВАНИЯ К ВИРТУАЛЬНОЙ МАШИНЕ	3
3. НАСТРОЙКА ВИРТУАЛЬНОЙ МАШИНЫ С K3S	4
3.1. Установка k3s	4
3.2. Установка HELM client	4
3.3. Установка NGINX Ingress Controller	5
3.4. Установка Postgres Professional	6
4. УСТАНОВКА И НАСТРОЙКА PMCONTROLLING SECURITY	6
4.1. Создание баз данных	6
4.2. Загрузка докер-образов	7
4.3. Настройка TLS	7
4.4. Заполнение переменных для helm чарт	7
4.5. Применение helm чарта	9
4.6. Применение sql скриптов	10
5. ВХОД В СИСТЕМУ	10

1. ВВЕДЕНИЕ

Настоящее руководство описывает процесс установки и настройки PMControlling Security.

Набор компонентов системы:

pmdatabase-rmq	Брокер сообщений
pmdatabase-redis	База данных Key/Value. Хранилище кеш-данных
oauth2-proxy	Обратный прокси для аутентификации и авторизации.
adm-api	Модуль администрирования, лицензирования
shell-api	Модуль навигации, стилизации.
shell3-web-host	Web-компонент UI
shell3-web-wrapper	Web-компонент внешней интеграции
shell3-adm-host	Web-компонент администрирования

2. ТРЕБОВАНИЯ К ВИРТУАЛЬНОЙ МАШИНЕ

Для установки PMControlling Security потребуется виртуальная машина с Kubernetes или облегченной версией k3s. Необходимо предустановить СУБД Postgres Professional, NGINX Ingress Controller, HELM client.

Рекомендуемые аппаратные требования:

1. Не менее 16 гб оперативной памяти;
2. Не менее 50 gb раздел жесткого диска;
3. Не менее 4 vCPU.

Алгоритм расчета аппаратных требований:

1. 4 ядра;
2. Из расчета 25 пользователей на 1 ядро для расширения;
3. 4 ГБ доступной памяти на 1 ядро системы.

Поддерживаемые ОС:

*nix, Windows (Win2012 R2 и более поздние), в том числе РЕД ОС 7.2, Astra Linux (Орел), ОС РОСА или более поздние.

Поддерживаемые веб-браузеры:

Mozilla Firefox (94 и старше), Microsoft Edge (98 и старше), Apple Safari (15,4 и старше), Google Chrome (98 и старше), Яндекс Браузер (22.9.3.82 и старше).

3. НАСТРОЙКА ВИРТУАЛЬНОЙ МАШИНЫ С K3S

В этом разделе описана настройка виртуальной машины с k3s для запуска программного обеспечения PMControlling Security.

Для корректного выполнения инструкции на виртуальной машине должен быть доступ в интернет. Набор команд выполняется от имени администратора системы.

3.1. Установка k3s

Скачать и запустить скрипт установки k3s с официального сайта:

```
curl -sfL https://get.k3s.io | sh -
```

После окончания установки выполнить проверку командой:

```
k3s kubectl get nodes
```

Результат проверки – успешный запуск k3s со статусом Ready.

Для удобства скопируем конфигурацию подключения k3s в каталог администратора системы.

```
cp /etc/rancher/k3s/k3s.yaml ~/.kube/config.
```

3.2. Установка HELM client

Скачать и запустить скрипт установки с официального сайта:

```
curl -fsSL https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash.
```

После окончания установки выполнить проверку командой:

```
helm version.
```

Результат проверки – успешный вывод версии установленного HELM client.

3.3. Установка NGINX Ingress Controller

Отключаем предустановленный в k3s Traefik:

```
sudo touch /var/lib/rancher/k3s/server/manifests/traefik.yaml.skip  
sudo systemctl restart k3s && kubectl -n kube-system delete helmcharts.helm.cattle.io  
traefik --ignore-not-found.
```

Установим Ingress Controller из официального helm репозитория:

```
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx && helm repo  
update  
helm upgrade --install ingress-nginx ingress-nginx/ingress-nginx \  
--namespace ingress-nginx --create-namespace \  
--set controller.kind=DaemonSet \  
--set controller.hostNetwork=true \  
--set controller.hostPort.enabled=true \  
--set controller.daemonset.useHostPort=true \  
--set controller.service.type=ClusterIP \  
--set controller.metrics.enabled=true \  
--set-string controller.config.use-forwarded-headers=true \  
--set controller.config.allow-snippet-annotations="true" \  
--set controller.config.annotations-risk-level="Critical".
```

После окончания установки выполнить проверку командой:

```
kubectl get po -A | grep nginx.
```

Результат проверки – успешный вывод пода с NGINX Ingress Controller в состоянии Running.

3.4. Установка Postgres Professional

Подключите репозиторий пакетов, предназначенный для вашей операционной системы, предоставленный специалистом технической поддержки Postgres Pro.

Установите пакет postgrespro-std. При этом по зависимостям установятся все требуемые компоненты, будет создана база данных по умолчанию, запущен сервер баз данных и настроен автозапуск сервера при загрузке системы, а все предоставляемые программы станут доступными в пути PATH.

После окончания установки выполнить проверку командой:

psql –version.

Результат проверки – успешный вывод версии установленного Postgres Professional.

4. УСТАНОВКА И НАСТРОЙКА PMCONTROLLING SECURITY

Для установки PMControlling Security необходимо перенести на виртуальную машину с k3s архивы docker-образов PMControlling Security, helm чарты конфигурации и скрипты базы данных.

Необходим установленный и настроенный провайдер идентификации – Keycloak.

Необходимо создать dns-записи для web-приложений PMControlling Security.

4.1. Создание баз данных

Подключиться к Postgres Professional и создать пустые базы данных с указанием владельца.

```
CREATE DATABASE shell WITH OWNER $user;
```

```
CREATE DATABASE adm WITH OWNER $user;
```

```
CREATE DATABASE log WITH OWNER $user;
```

4.2. Загрузка докер-образов

Создать каталог: /opt/images.

Необходимо разархивировать images.tar.gz в каталог /opt/images

```
tar -xvzf images.tar.gz -C /opt/images.
```

Перейти в каталог: /opt/images и выполнить команду импорта образов.

```
find . -type f -exec ctr -n k8s.io image import {} \;
```

Проверка:

```
crictl image ls
```

Результат проверки – успешный вывод списка загруженных в систему образов.

4.3. Настройка TLS

Создадим в k3s namespace demo

```
kubectrl create ns demo.
```

Для работы ingress kubectrl web приложений PMControlling Security потребуется создать secrets с сертификатами, валидными для доменов приложения в сети заказчика.

Выполнить из каталога с файлами сертификата:

```
kubectrl create secret tls csru-secret \  
--cert=file.crt \  
--key=file.key \  
-n demo
```

Проверка:

```
kubectrl get secret -n demo
```

Результат проверки – успешный вывод созданного секрета с именем csru-secret.

4.4. Заполнение переменных для helm чарт

Необходимо разархивировать helm.tar.gz в каталог /opt/.

Файл /opt/helm/values.yaml содержит переменные, необходимые для

корректного запуска PMControlling Security, в том числе параметры подключения к базе данных, Keycloak, oauth2-proxy, redis, rabbitmq-server и dns-имена для web-приложений.

Отредактировать файл /opt/helm/values.yaml.

В блоке keycloak заполнить:

Параметры подключения к keycloak:

authServerUrl:

clientId:

requestBatchSize:

secrets:

realmName:

В блоке db_shell, log_db и administration_db заполнить параметры подключения к ранее созданным базам данных:

db_shell:

name: shell

host:

user:

pass:

port:

administration_db:

name: adm

host:

user:

pass:

port:

log_db:

name: log

host:

user:

pass:

port:

В блоке `oauth_proxu` заполнить:

Параметры подключения к `keycloak`:

keycloak:

baseUrl:

realm:

domains:

allowed:

параметры конфигурации `oauth2-proxu`:

client:

id:

secret:

cookieSecret:

В блоке `shell_web_host_msvc` заполнить dns имя:

host:

В блоке `shell_web_wrapper` заполнить dns имя:

host:

В блоке `adm_web_host` заполнить dns имя:

host:

4.5. Применение helm чарта

После внесения переменных для запуска контейнеров необходимо применить helm чарт.

`helm upgrade --install shell /opt/helm/ -n demo.`

Проверка:

`kubectl get po -n demo.`

Результат проверки – успешный вывод подов PMControlling Security в состоянии Running.

Проверка:

kubectrl get ingress -n demo.

Результат проверки – успешный вывод ingress с полученным IP адресом.

4.6. Применение sql скриптов

Распаковать архив sql.tar.gz в /tmp.

Подключиться к базе данных shell и применить скрипт create-nav-and-dash-shell-adm2.sql.

Подключиться к базе данных adm и применить скрипт create-shell-entity-adm2.sql.

Подключиться к базе данных log и применить скрипт set-secure-all-events.sql.

5. ВХОД В СИСТЕМУ

Для входа в систему необходимо открыть браузер и перейти на dns-имя, указанное в helm чарте для сервиса shell_web_host_msvc.

В открывшемся окне ввести логин и пароль учетной записи, активной в Keycloak (Рисунок 1).



Рисунок 1 Вход в модуль

После авторизации отобразится доступный модуль Администрирования

(Рисунок 2).

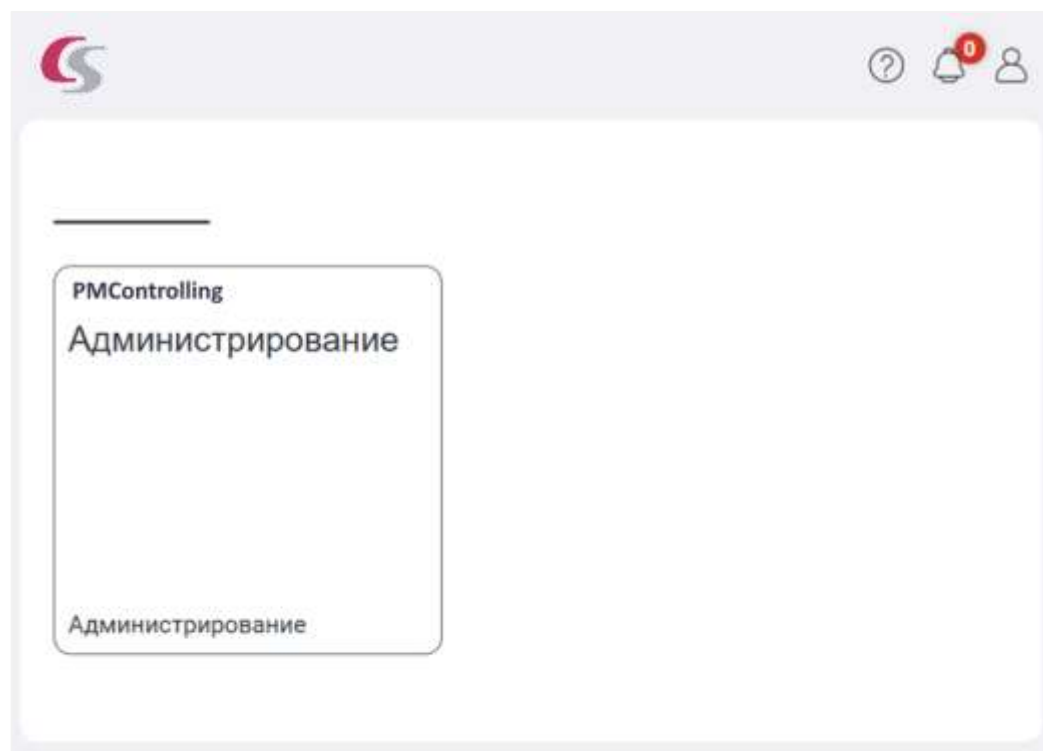


Рисунок 2 Активные модули