**УСТАНОВКА PORTAINER**

Portainer – это очень удобный графический интерфейс для управления Docker или **Docker Swarm**. Устанавливается он практически в одно действие – так как сам точно также является контейнером. Итак:

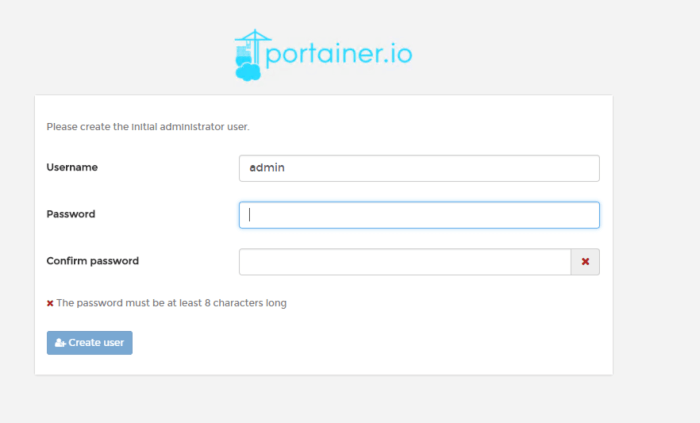
Создаем разметку для Portainer:

docker volume create portainer\_data

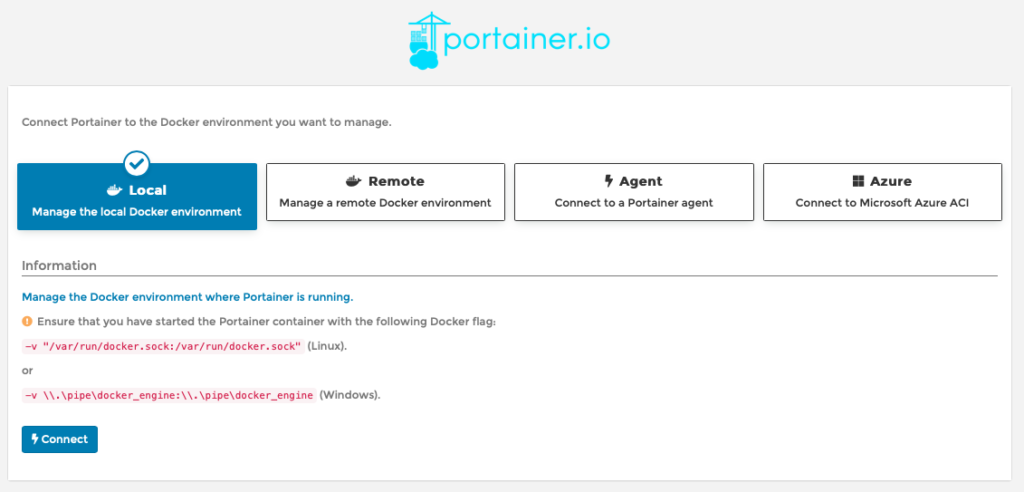
И затем запускаем сам контейнер:

docker run -d -p 9000:9000 -v /var/run/docker.sock:/var/run/docker.sock -v portainer\_data:/data portainer/portainer

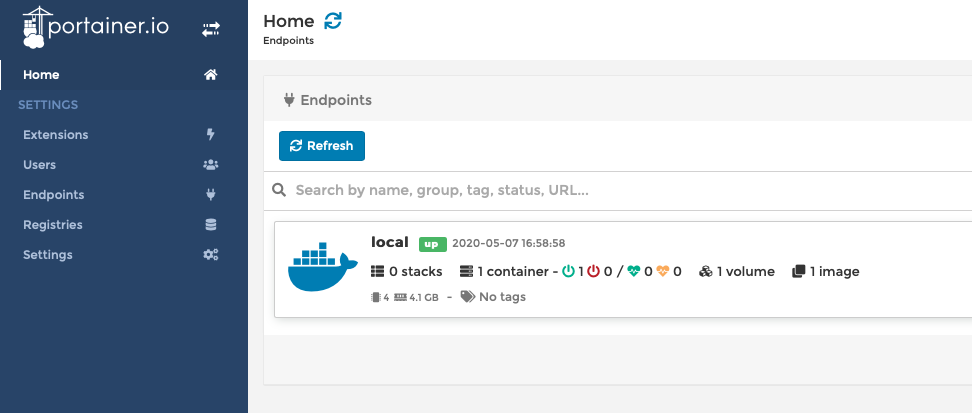
После чего заходите на сетевой адрес вашего сервера на порт **9000**, и вы должны увидеть окно с предложением установить пароль администратора:

[](https://wiki.merionet.ru/images/razbiraemsya-s-docker-ustanovka-i-ispolzovanie/4.png)

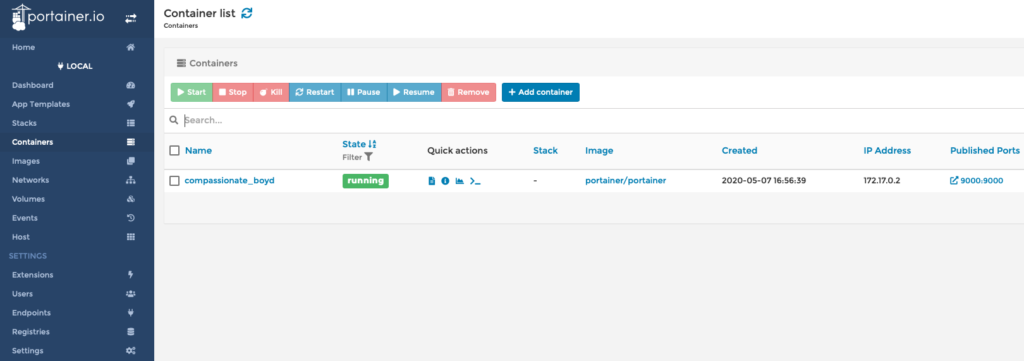
Далее выбираем где находится наш Докер – на этом же сервере, или на другом (в нашем случае – **Local**) и кликаем **Connect**.

[](http://disnetern.ru/wp-content/uploads/2020/05/%D0%A1%D0%BD%D0%B8%D0%BC%D0%BE%D0%BA-%D1%8D%D0%BA%D1%80%D0%B0%D0%BD%D0%B0-2020-05-07-%D0%B2-16.58.48.png)

После чего вас встретит красивый дэшборд:

[](http://disnetern.ru/wp-content/uploads/2020/05/%D0%A1%D0%BD%D0%B8%D0%BC%D0%BE%D0%BA-%D1%8D%D0%BA%D1%80%D0%B0%D0%BD%D0%B0-2020-05-07-%D0%B2-16.59.19.png)

Итак, сначала кликните на **Containers** – вы увидите все имеющиеся контейнеры с информацией о них:

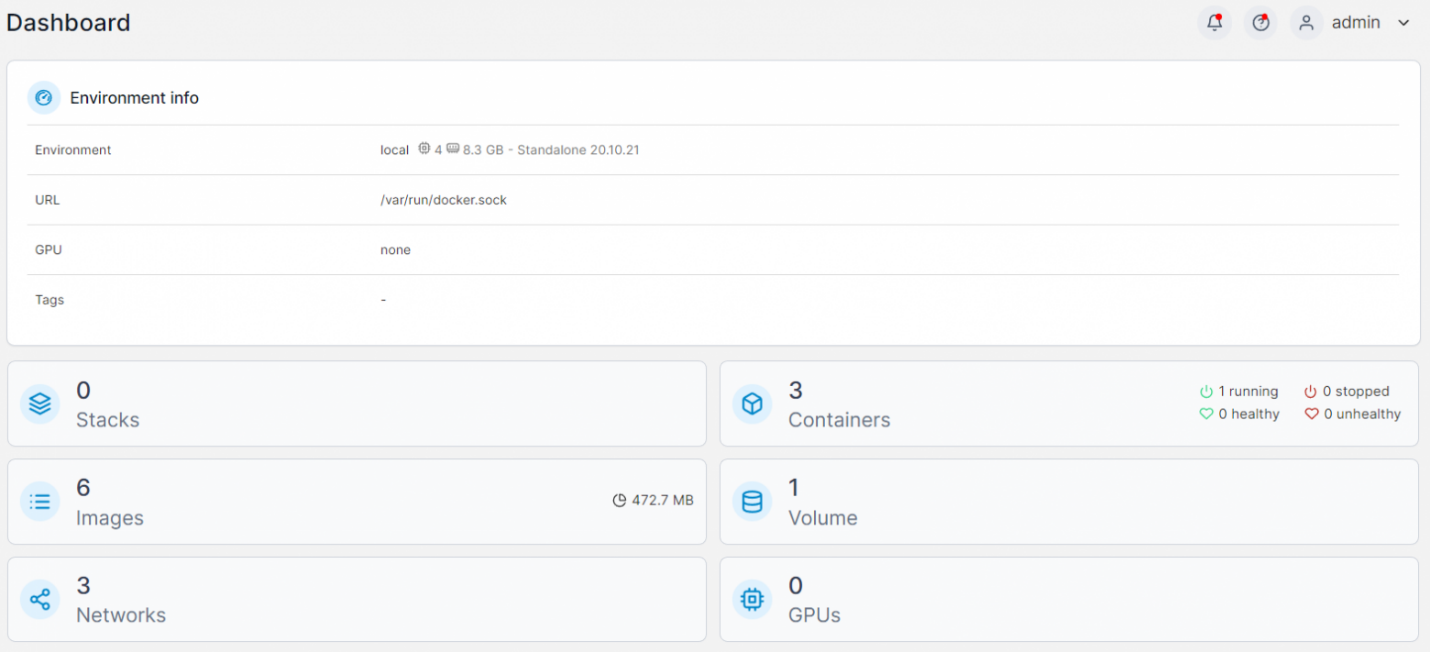
[](http://disnetern.ru/wp-content/uploads/2020/05/%D0%A1%D0%BD%D0%B8%D0%BC%D0%BE%D0%BA-%D1%8D%D0%BA%D1%80%D0%B0%D0%BD%D0%B0-2020-05-07-%D0%B2-17.05.52.png)

Как вы можете видеть, у нас на данный момент запущен только один контейнер – Portainer, и доступ к нему открыт по порту 9000 (столбец Published Ports), и адрес во внутренней сети Docker – 172.17.0.2.

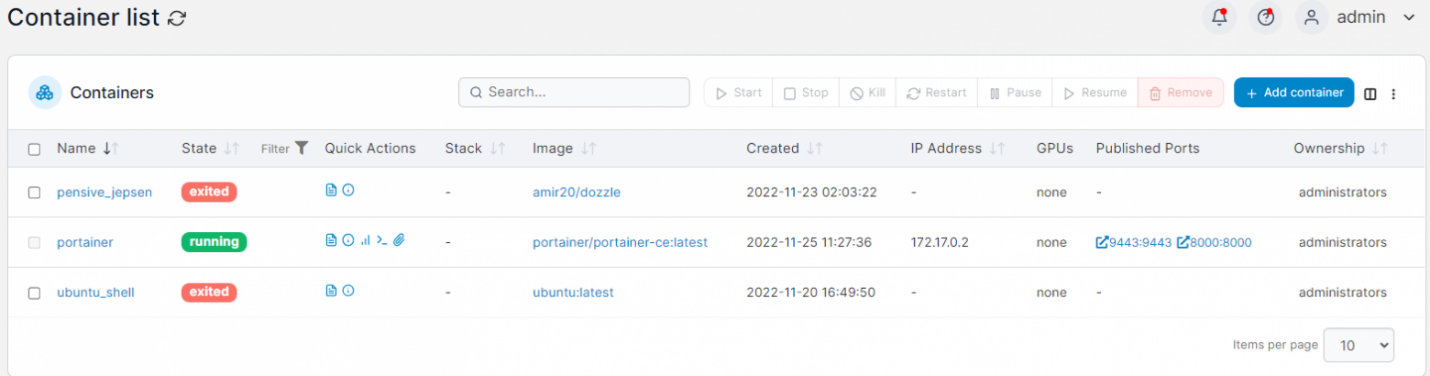
Далее кликните на **App Templates** в меню справа – и вы увидите весь список приложений, который можно запустить одним кликом:

[](http://disnetern.ru/wp-content/uploads/2020/05/%D0%A1%D0%BD%D0%B8%D0%BC%D0%BE%D0%BA-%D1%8D%D0%BA%D1%80%D0%B0%D0%BD%D0%B0-2020-05-07-%D0%B2-17.06.47.png)

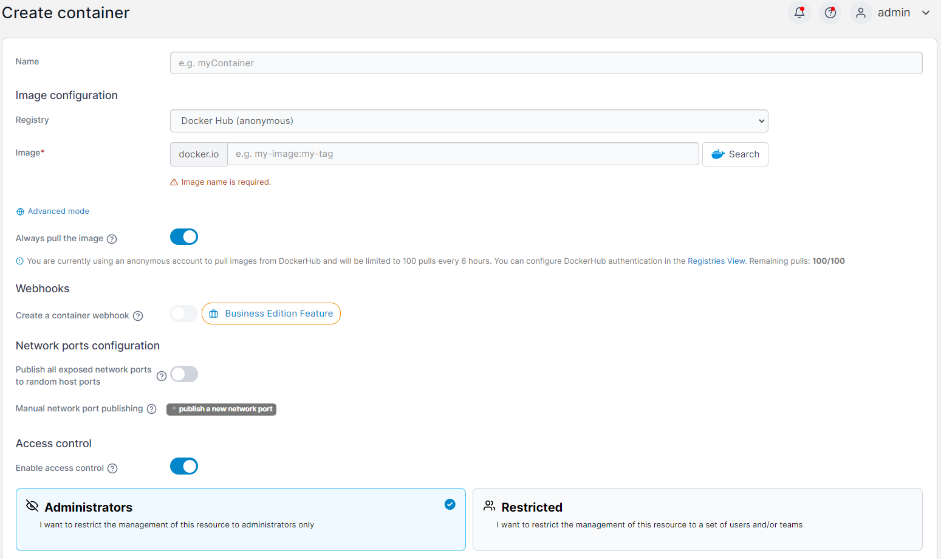
После подключения к необходимому хосту откроется основное меню программы, где будет отображена информация о технических характеристиках сервера, а также — разделы с количеством сервисов, контейнеров, образов, томов и сетей:



В каждый из этих разделов можно перейти и выполнить определенные действия. В разделе **Контейнеры**можно запустить/остановить/завершить/перезапустить/поставить на паузу/удалить контейнер:



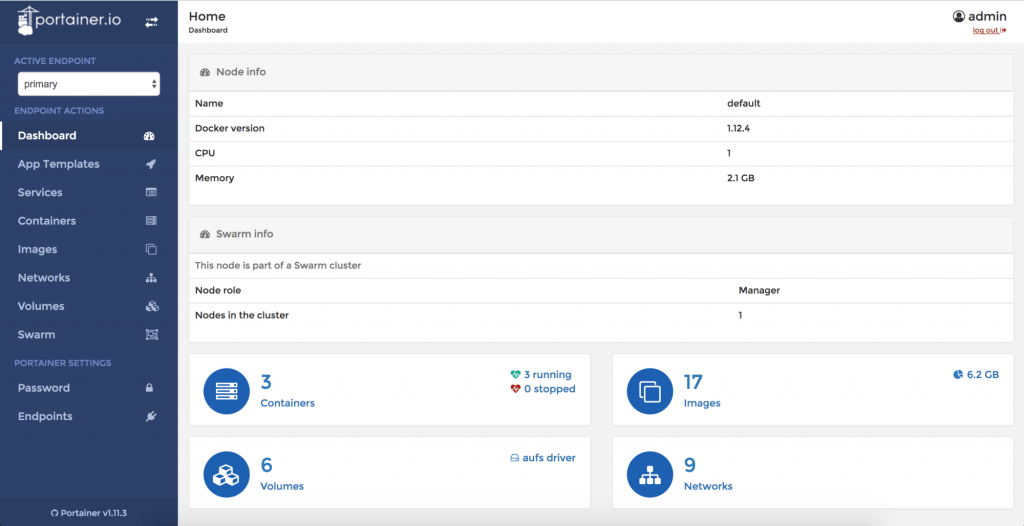
Также в этом разделе можно создать контейнер:



Принцип работы точно такой же, как и у команды **docker run** – задается имя контейнера, образ, при необходимости в контейнере можно открыть порты, а также настроить контроль доступа, только вместо командной строки и параметров здесь используется графический интерфейс.

**Основная панель**

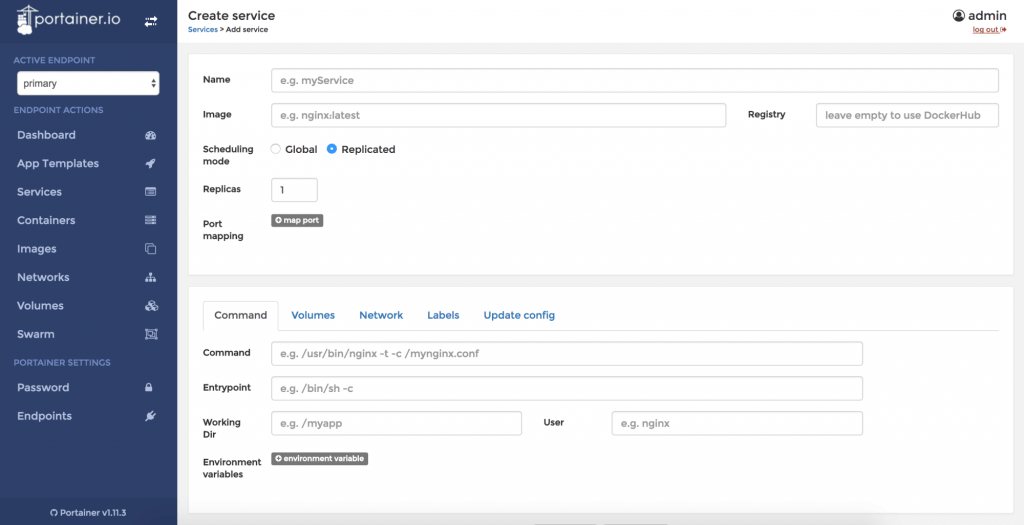
Как видно из экрана, Portainer — это отличное решение, которое позволяет не только очень наглядно предоставлять полную информацию по вашим Docker хостам и кластерам, но и эффективно управлять ими, предоставляя вам множество возможностей, вкратце о которых ниже.



**Сервисы (Services)**

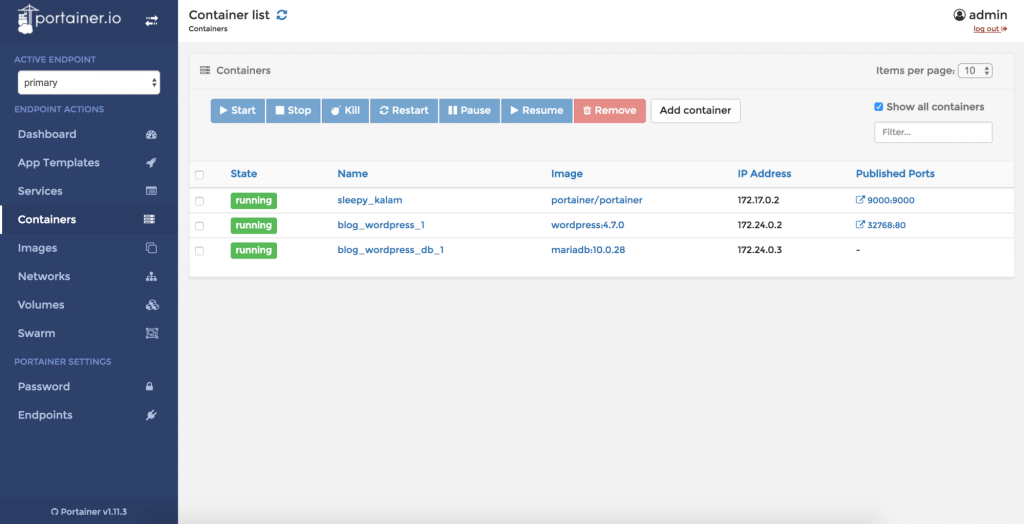
Если ваш Docker Engine запущен в Swarm mode, вам становится доступно меню управления сервисами Swarm кластера. Процесс создания сервиса продемонстрирован на экране ниже. Прямо из Web-UI вы можете задать все значимые параметры сервиса:

* Имя самого сервиса
* Имя образа, из которого будет запущен сервис
* Указать сторонний Docker реестр
* Режим работы планировщика
* Мапинг портов, дисков, сетей
* А также метки



**Контейнеры (Containers)**

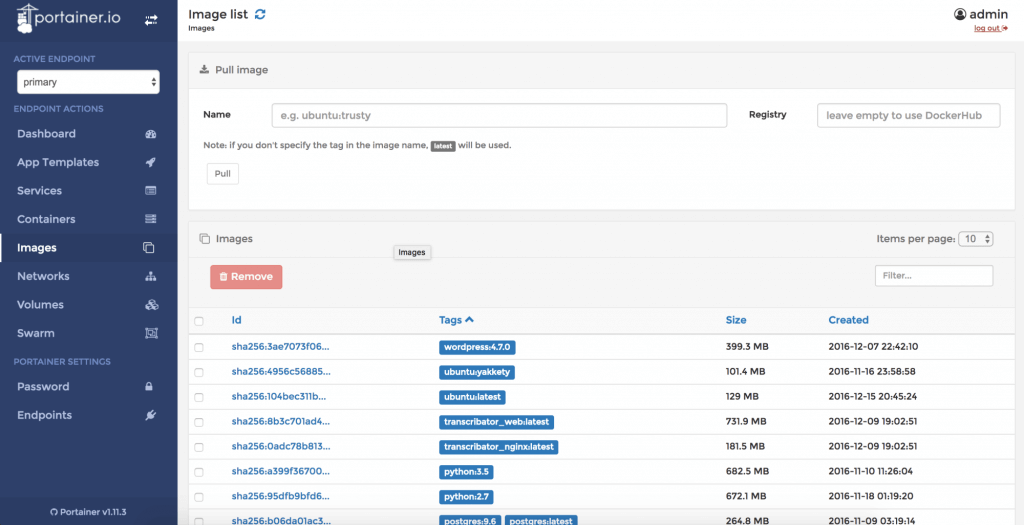
Помимо управления службами Portainer дает возможность управлять жизненным циклом контейнеров, запущенных на вашем хосте или кластере. Интерфейс создания контейнера ничем не отличается от интерфейса создания сервиса.



Если перейти в управление конкретным контейнером, то у вас появится возможность наблюдать за потреблением ресурсов конкретного контейнера, просмотра логов, а также подключению к интерактивной консоли (да, можно зайти внутрь работающего контейнера в терминальную сессию и выполнять нужные вам команды).

**Образы (Images)**

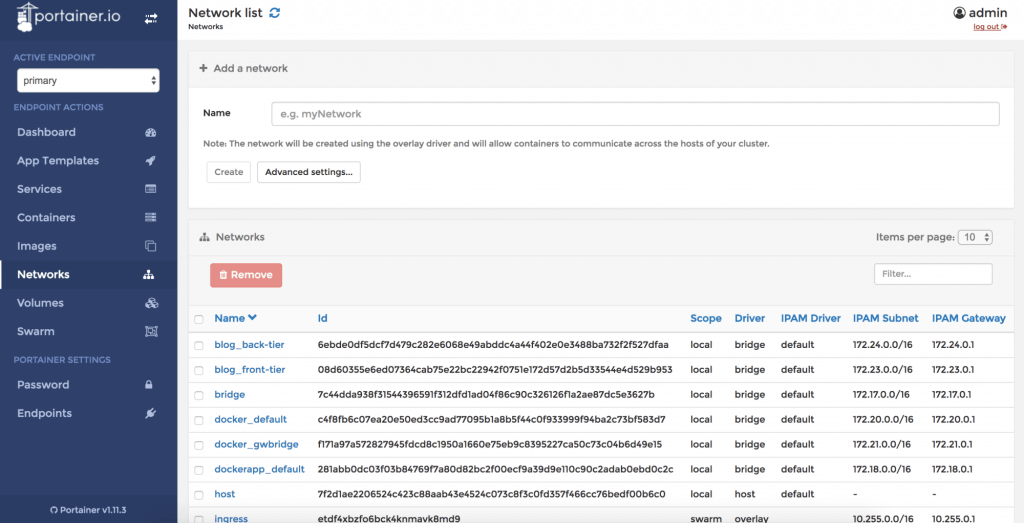
Мне кажется очень удобной возможность выделить мышкой сразу 5-6 не используемых образов, а затем удалить их в один клик.



Более того, при клике на каждый конкретный образ можно изменить его тег, получить информацию о размере, дате создания, а также информацию из Dockerfile, такую как, например: CMD, ENTRYPOINT, EXPOSE, VOLUME и переменные окружения внутри контейнера из ENV.

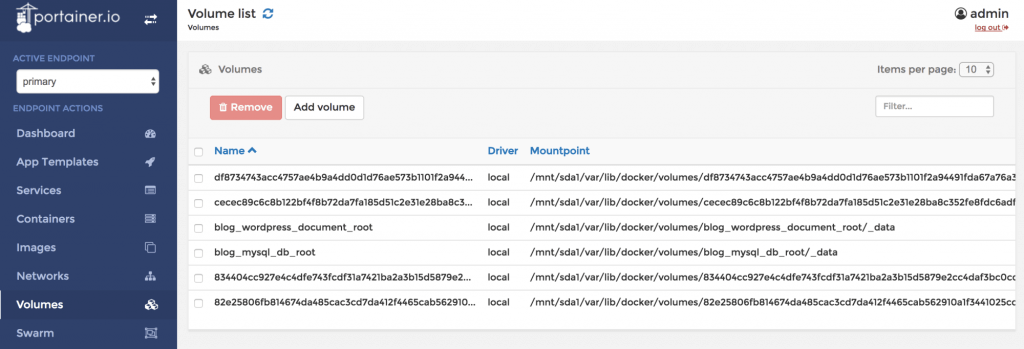
**Сети (Networks)**

Возможностей работы с сетями в Portainer пока не очень много. В списке с сетями отображается лишь скудная информация о типе сети и ее адресации.



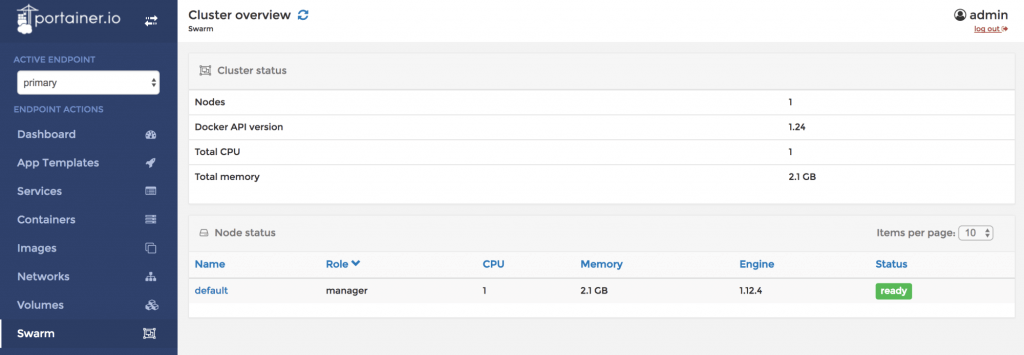
**Диски (Volumes)**

В принципе все, что можно получить из docker volume inspect удобно и наглядно отображается в едином интерфейсе, дополняя остальную функциональность.



**Кластеры (Swarm)**

Состав кластера, а также количество задействованных узлов и доступных в кластере ресурсов можно узнать в меню Swarm. При выборе каждого конкретного узла кластера можно очень удобно очистить его от виртуальных машин и остановить на нем аллокацию.



**Подключения (Endpoints)**

Portainer очень удобно использовать вместе со своей Docker Machine, чтобы иметь возможность управлять сразу всеми вашими Docker хостами и кластерами, определить очередное подключение не составит труда в соответствующем меню. Опция поддержки TLS также имеется.

