

Последнее, с чем я вас хотел познакомить из Docker SDK, это, соответственно, создание сетей. Напоминаю, что сети в Docker-е..., там не все так просто, существует несколько драйверов для сетей. Существует драйвер Bridge – это, соответственно, сеть, соединяющая несколько Docker-контейнеров внутри себя. Она же дефолтная сеть для Docker compose, если вы создаете Docker compose.

Сеть Host, которая монтирует контейнеры, как часть сетевого оборудования вашего Docker Host-а. То есть, если вы, например поднимаете HTTP сервер на 80 порту, это будет 80 порт именно вашего Host-а, вашей вот этой вот основной главной машины, на которой вы что-то там запускаете. Сеть Overlay, которая позволяет сконектить несколько Docker Daemon-ов вместе.

Ipvlan и Macvlan сети, позволяющие настраивать IP адреса и mac адреса гибко. Я ими не пользуюсь, поэтому не могу ничего конкретного про них сказать.

Соответственно, сеть None, которая ваш..., оставит ваш контейнер вообще без интернета. Соответственно, здесь мы можем создавать сети, опять же, сеть создается с помощью команды NetworkCreate, включающей в себя имя сети, как основной параметр. И всякие параметры NetworkCreate. То есть драйвер сети, соответственно, проверка на дубликаты, которые не позволяют вам создать сети с одинаковым именем, одинаковым типом и там есть большой текст, что у Docker-а нет никаких надежных способов проверить тип сети. Поэтому у вас может так получиться, что сети будут друг на друга накладываться.

Ну и, соответственно, команда NetworkCreate создаст сеть, в которой можно получить ID-шник. И с помощью команды NetworkConnect, вы можете подсоединить ваш контейнер, который здесь стартовал, к этой сети, с какими-то настройками Docker-а. С Link-ами, с сетевым ID-шником, с Gateway-ем. Ну то есть со стандартными настройками Endpoint-а в Docker-е.