

АВТОМАТИЗАЦИЯ РАЗВЕРТЫВАНИЯ СЕРВИСОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ С ИСПОЛЬЗОВАНИЕМ ОРКЕСТРАТОРА NOMAD

Матлаш Т.С., студентка группы 763102

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Саломатин С. Б. – к. т. н., доцент

В современных практиках разработки программного обеспечения используется множество инструментов для компилирования, тестирования и развертывания кода [2]. Прежде чем продукт попадает к конечному пользователю, он должен пройти множество проверок на дефекты в различных средах. Этот факт приводит к сложной конфигурации систем, в которых производится тестирование продукта, увеличению временных и денежных затрат на их поддержку, а также могут возникать ошибки и конфликты из-за различных настроек систем той или иной среды. Для автоматизации этих процессов используются оркестраторы. Оркестратор позволяет автоматизировать создание, мониторинг и развертывание ресурсов в среде разработки.

Ранее повсеместно для развертывания использовались виртуальные машины. Технически было организовано так, что весь код приложений собирался за счет средств автоматической сборки [2]. При помощи конфигурационного менеджера код из системы контроля версий доставлялся на виртуальные сервера. Но в этом способе есть несколько недостатков:

- для каждого сервиса необходимо поддерживать актуальную для него версию виртуальной машины (обновление библиотек и зависимостей);
- для каждого нового сервиса необходимо создавать отдельную новую виртуальную машину, которую также нужно обслуживать.

Nomad - это простой в использовании, гибкий и производительный оркестратор, который позволяет управлять контейнерами, в которых находятся сервисы. Nomad позволяет разработчикам использовать декларативную инфраструктуру как код для развертывания сервисов. Для развертывания сервиса необходим конфигурационный файл. В качестве формата конфигурационного файла используется язык HCL, что расшифровывается как HashiCorp Configuration Language (см. рисунок 1).

```
job "service-job" {
  region = "global"
  type = "service"
  group "service-group" {
    task "service-task" {
      driver = "docker"

      config {
        image = "myimage:1.2.3"
      }

      service {
        name = "my-service"
        tags = [ "1.2.3" ]
      }
    }
  }
}
```

Рисунок 1 – Пример конфигурационного файла

Все конфигурационные файлы хранятся в одном репозитории. Таким образом, конфигурации становятся обозреваемы: их легко поддерживать и можно увидеть, какие сервисы есть в данный момент. В случае необходимости несложно обновить или поменять конфигурации сервиса. Для добавления новой системы необходимо создать конфигурационный файл внутри новой директории (см. рисунок 2).



Рисунок 2 – Структура репозитория конфигурационных файлов

Nomad предоставляет удобный графический интерфейс. На рисунке 3 можно увидеть список развернутых сервисов, а также их состояние, что решает задачу мониторинга сервисов.

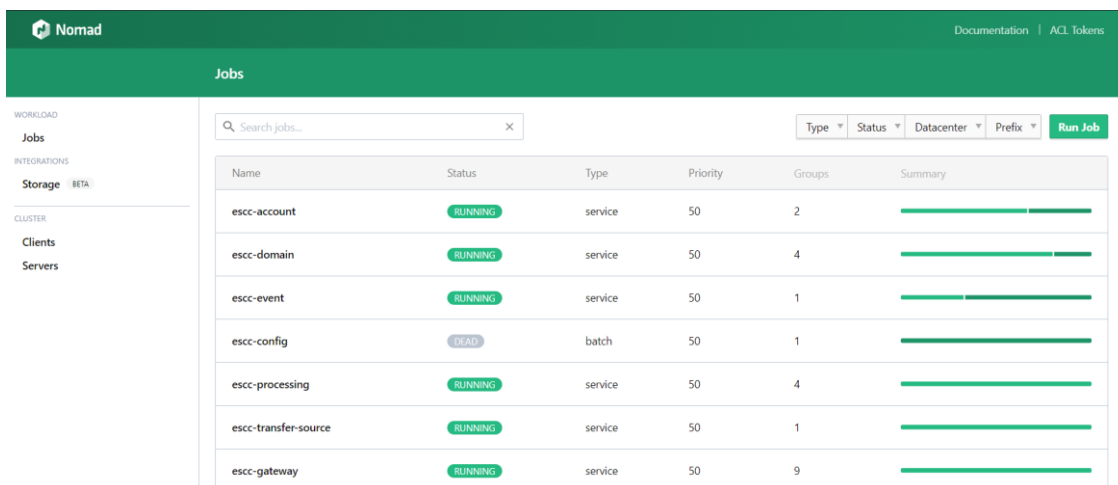


Рисунок 3 – Графический интерфейс оркестратора Nomad

Применение оркестратора Nomad позволяет решить задачу конфигурирования и поддержания систем для развертывания сервисов. Также его использование значительно ускоряет разработку программного обеспечения за счет использования декларативной инфраструктуры.

Список использованных источников:

1. Непрерывная интеграция программного обеспечения [Электронный ресурс]. - https://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D0%BF%D1%80%D0%B5%D1%80%D1%8B%D0%B2%D0%BD%D0%B0%D1%8F_%D0%B8%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B0%D1%86%D0%B8%D1%8F
2. Деллой приложений [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/company/lamoda/blog/451644/>