

Исследование и разработка методов развертывания облачной инфраструктуры с использованием Terraform и Terragrunt

Авдеева М.Л. (Университет ИТМО)

Научный руководитель – к.т.н., Грудинин Владимир Алексеевич (Университет ИТМО)

Введение. С постоянным увеличением объема данных и необходимости масштабирования инфраструктуры, организации сталкиваются с необходимостью обеспечения гибкости, надежности и эффективности управления облачной инфраструктурой. Как правило, для этих целей реализуется концепция инфраструктура как код - это подход к управлению и развертыванию инфраструктуры с использованием программного кода и автоматизации [1]. Он помогает упростить процессы управления инфраструктурой, повысить ее гибкость, уменьшить ошибки и обеспечить согласованность данных при развертывании инфраструктуры.

Terraform является одним из инструментов, автоматизирующих развертывание инфраструктуры и реализующих концепцию инфраструктуры как код [2]. На данный момент Terraform обладает некоторыми недостатками, такими как: отсутствие полноценных циклов для работы с ресурсами, невозможность прямой передачи переменных между модулями, а также дублирование кода при использовании модулей, что, в свою очередь, влечет за собой дополнительные издержки в виде увеличения количества времени, затрачиваемого на разработку и, соответственно, увеличение денежных затрат [3]. Terragrunt позволяет нивелировать недостатки Terraform, поэтому использование данного инструмента является более предпочтительным [4].

Основная часть.

В данной работе было проведено подробное исследование подхода к управлению и описанию инфраструктуры как код через конфигурационные файлы, помимо этого будут проанализированы инструменты, предоставляющие функционал для реализации данного подхода, такие как Terraform и Terragrunt.

В ходе выполнения исследования были определены основные недостатки Terraform:

- Проблема передачи переменных между модулями в Terraform заключается в том, что при разработке сложных инфраструктурных конфигураций возникает необходимость передачи данных и параметров между различными модулями и ресурсами. Это может привести к сложностям в обеспечении правильного порядка инициализации данных, управлении зависимостями и к дублированию информации.

- Проблема удобства написания кода часто связана с управлением большими и сложными конфигурациями инфраструктуры. С ростом проекта может возникнуть необходимость в управлении большим объемом кода, повторяющихся структур и настроек, а также управления зависимостями между различными ресурсами и модулями.

Как правило, для решения вышеперечисленных проблем Terraform, используется инструмент Terragrunt. Таким образом, целью работы является проведение эксперимента, состоящего из реализации сценариев развертывания инфраструктуры, для выявления наиболее эффективного метода работы с представленными инструментами.

В ходе работы, используя данные инструменты, были разработаны два программных проекта. Данные проекты применялись для проведения эксперимента с развертыванием нескольких шаблонов инфраструктуры. В ходе выполнения эксперимента были выявлены особенности и недостатки, предложен метод, который позволяет повысить эффективность разработки инфраструктуры как кода, а также проведено сравнение качественных и количественных характеристик данных инструментов, таких как:

- Количество строк кода;
- Время, затраченное на разработку модулей;

УДК 004.04

- Время развертывания инфраструктуры;
- Сложность и структурированность кода;
- Удобство использования.

Выводы. В исследовании были определены особенности инструментов Terraform и Terragrunt, а также был произведен экспериментальный сравнительный анализ данных инструментов. В работе был предложен метод для улучшения эффективности работы с Terraform посредством использования инструмента Terragrunt. Terragrunt показал лучшую эффективность в сравнении с Terraform по следующим параметрам: количество строк кода, время, затраченное на разработку модулей, удобство использования.

Список использованных источников:

1. Garg S., Garg S. Automated cloud infrastructure, continuous integration and continuous delivery using docker with robust container security //2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR). – IEEE, 2019. – С. 467-470.
2. Brikman Y. Terraform: Up and Running. – " O'Reilly Media, Inc.", 2022.
3. Patni J. C., Banerjee S., Tiwari D. Infrastructure as a code (IaC) to software defined infrastructure using Azure Resource Manager (ARM) //2020 International Conference on Computational Performance Evaluation (ComPE). – IEEE, 2020. – С. 575-578.
4. Terragrunt by Gruntwork.io [Электронный ресурс]. – Режим доступа: <https://terragrunt.gruntwork.io/> (дата обращения: 16.01.2024).