**«ОБЛАЧНЫЕ ТЕХНОЛОГИИ»**

**ЛАБОРАТОРНАЯ РАБОТА № 6**

# «Программное управление облачной инфраструктурой с помощью шаблонов сервиса CloudFormation»

***Цели работы:***

1. Ознакомиться с принципами управления облачной инфраструктурой с помощью программных интерфейсов REST API AWS.
2. Познакомиться с подходом автоматического развертывания облачной инфраструктуры “Инфраструктура как код”. Изучить язык описания инфраструктуры JIML (JSON Infrastructure Marrkup Language).
3. Освоить приемы работы с интерфейсом командной строки AWS (AWS CLI).
4. Изучить приемы создания инфраструктуры проекта с помощью SDK AWS для языка высокого уровня, на примере, Node.js.
5. Изучить структуру и шаблона сервиса CloudFormation, используемого для автоматического развертывания инфраструктуры проекта.
6. Познакомиться с описанием шаблонов сервиса CloudFormation с помощью языков JSON и YAML.
7. Создать шаблон сервиса CloudFormation и развернуть инфраструктуру проекта с его помощью.

Выполнение работы

Подготовка к выполнению лабораторной работы

В лабораторной работе используется предварительно созданные шаблоны и программы. Перед выполнением заданий эти шаблоны и программы необходимо скачать с GiHub. Сохраните файл архива приложения на Вашей локальной машине. В дальнейшем используйте файлы из этого архива для выполнения лабораторной работы, следуя соответствующим указаниям.

Подготовка интерфейса командной строки AWS для создания облачной инфраструктуры

Порядок выполнения этого задания зависит от того, какая ОС установлена на машине пользователя и будет использоваться для работы с CLI AWS.

Подготовка к установке CLI AWS для Windows

Со страницы Amazon CLI загрузите установщик ПО AWS CLI (32-bit или 64-bit).

Установка ПО CLI AWS для Windows

Запустить на выполнение установщик и выполнить установку ПО CLI следуя, указаниям мастера.

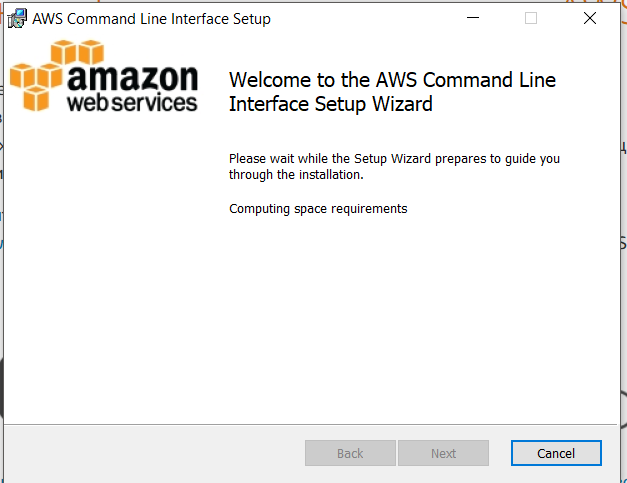


Рисунок 1 – Установка Amazon CLI

Запустить консоль PowerShell от имени Администратора, для чего в меню Start выберите Все программы > Админисирирование > Windows PowerShell Modules и из контекстного меню выбрать пункт "Запуск от имени Администратора".

Перейти в окно консоли PowerShell и выполнить команду:

> Set-ExecutionPolicy Unrestricted

которая, установит политику разрешающую исполнение неподписанных скриптов.

Закрыть консоль PowerShell, и в дальнейшем при выполнении данной лабораторной работы больше никогда не работать с правами Администратора

Вновь открыть консоль PowerShell c помощью меню Start. Проверить версию установленного ПО AWS CLI с помощью команды:



Рисунок 2 – Проверка установленной версии AWS

Если будет выведен номер версии 1.11.36 или выше, то установка прошла успешно, в противном случае установку ПО CLI AWS надо повторить.

Создание нового пользователя CLI AWS

Прежде чем, можно будет начать работу с помощью CLI AWS, требуется создать новый аккаунт пользователя с помощью сервиса IAM (AWS Identity and Access Management).

Для создание нового пользователя IAM необходимо перейти на страницу Management Console и кликнуть ссылку Services в области навигации, затем выбрать сервис IAM. Откроется страница:

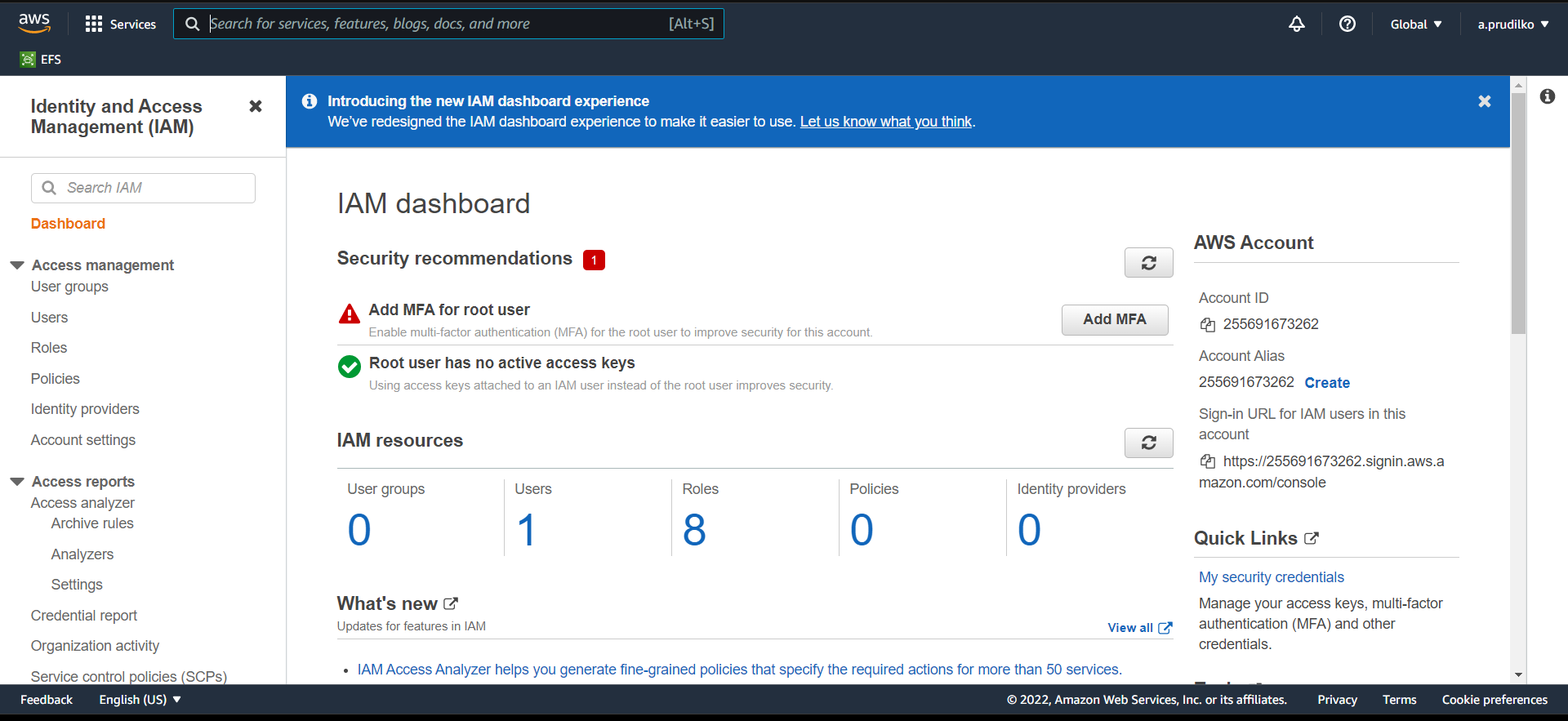


Рисунок 3 – Консоль IAM

в которой, необходимо:

Выбрать ссылку Users в области навигации (левая область страницы);

Кликнуть кнопку Add user, будет выполнен переход на начальную страницу мастера создания учетной записи пользователя.

На этой странице (приведена ниже) необходимо выполнить:

1. Ввести имя пользователя в виде: 02-03-isit-cli

2. В поле Acess Type выбрать Programmatic Access.

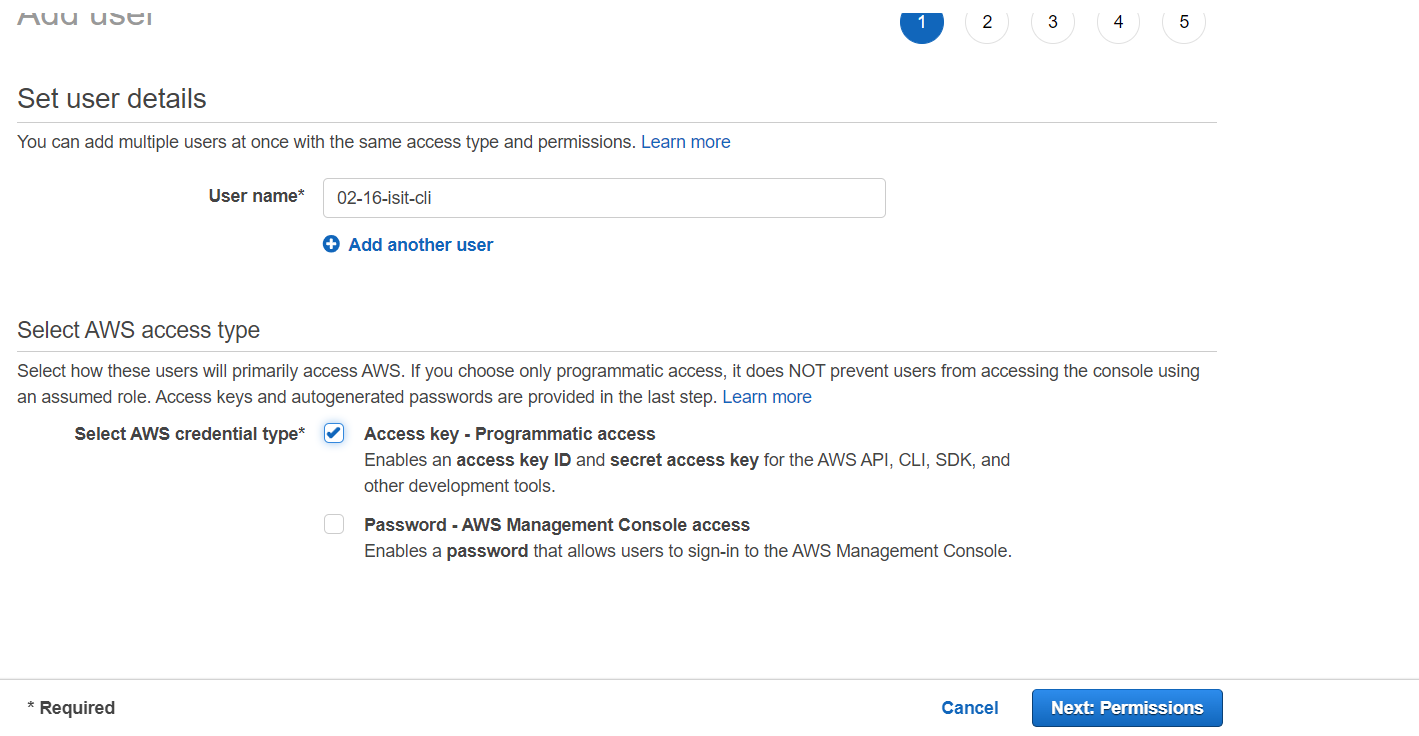


Рисунок 4 – Страница создания пользователя

Кликнуть кнопку Next: Permissions. Будет выполнен переход на следующую страницу мастера (Permissions).

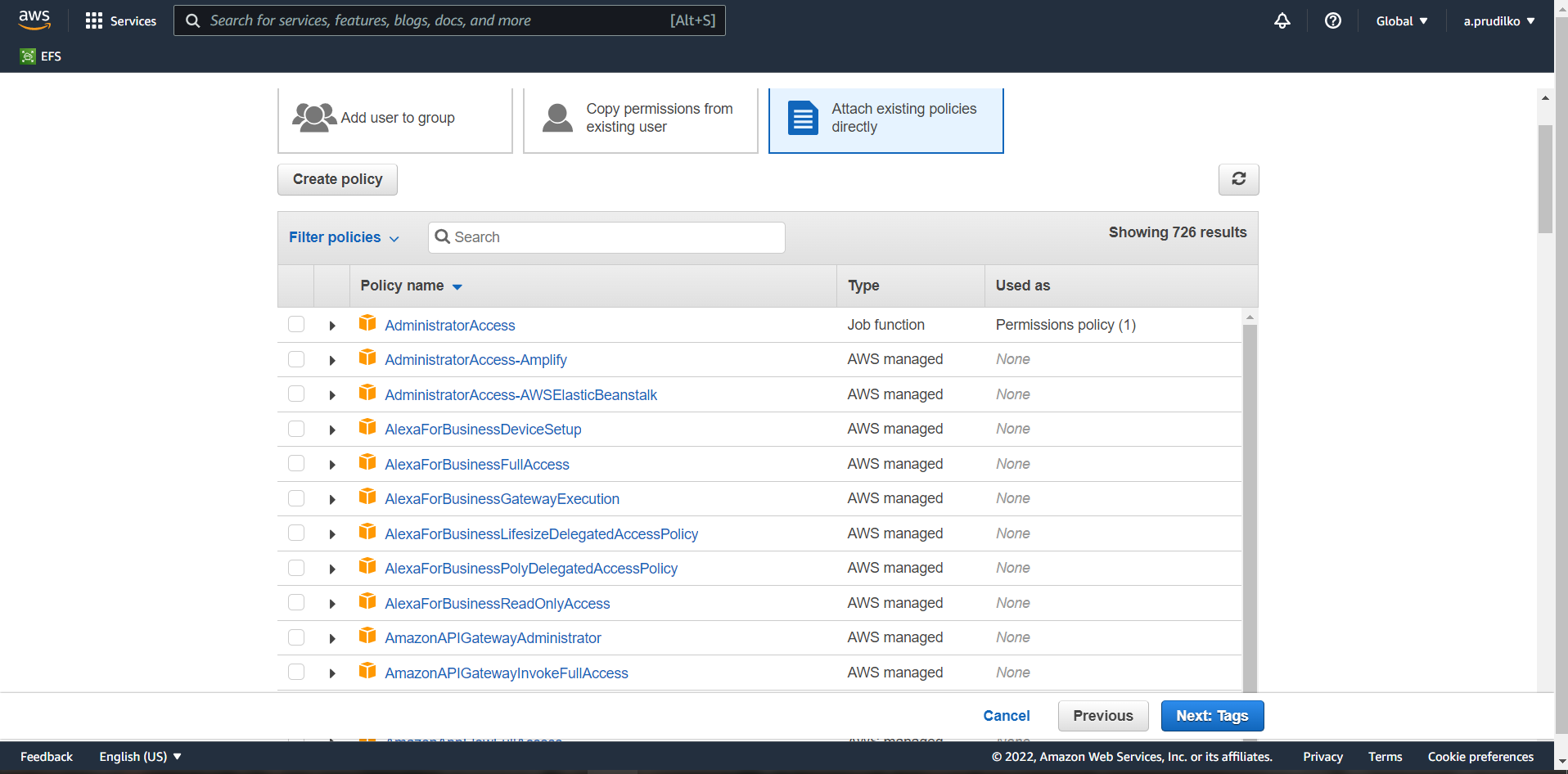


Рисунок 5 – Страница добавления привилегий пользователю

На этой странице надо:

Кликнуть картинку "Attach Existing Policies Directly".

Установить отметку в боксе строки "AdministratorAccess", что соответствует выбору политики, дающей все полномочия доступа.

Кликнуть Next:Review, для перехода на страницу проверки и подтверждения выбранных настоек для учетной записи пользователя.

На странице Review (она не показана) выводятся выбранные настройки аккаунта создаваемого пользователя. Убедитесь в их правильности и нажмите Create User. Откроется страница завершения создания нового пользователя (страница "Complete").

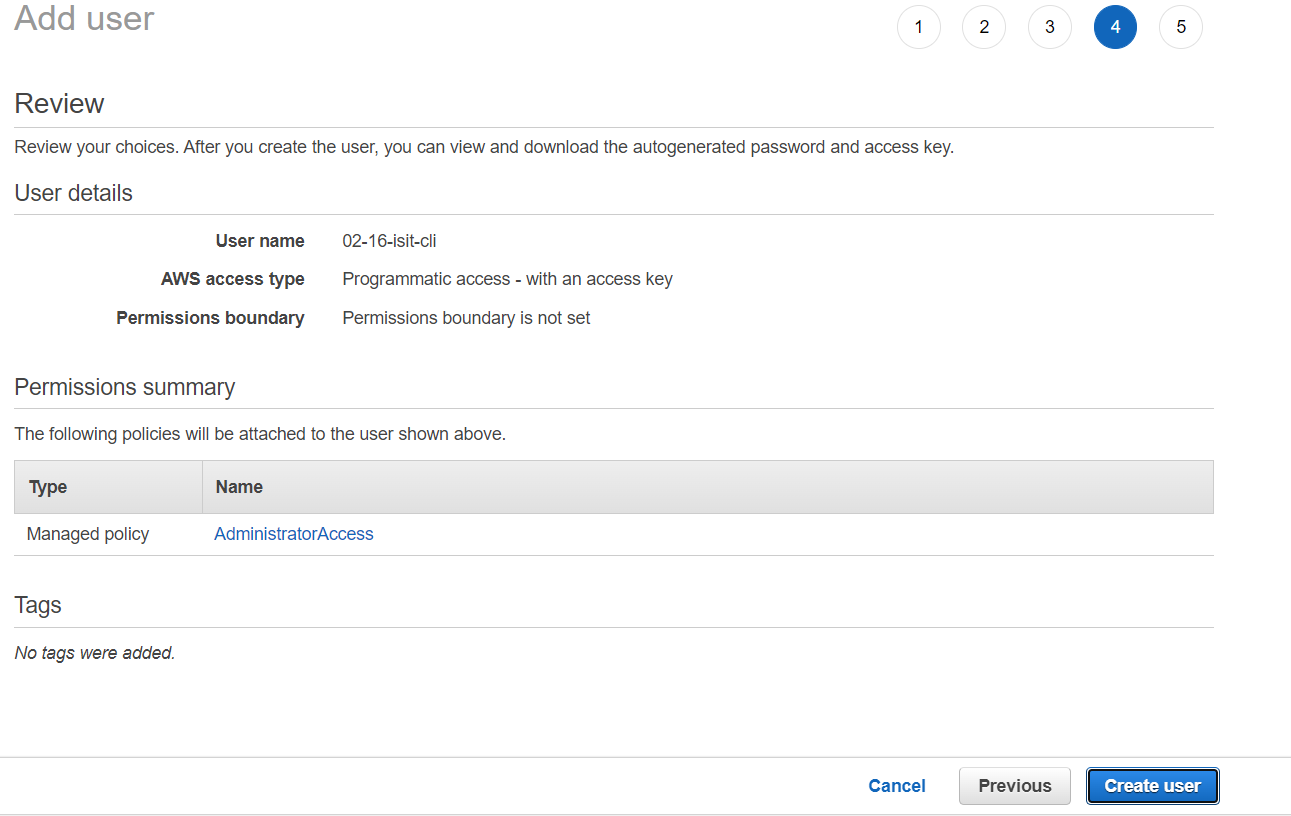


Рисунок 6 – Страница «Review» при создании нового пользователя

Конфигурирование параметров безопасности CLI AWS

На странице "Complete" мастера создания нового пользователя необходимо кликнуть ссылку Show, для того чтобы отобразить значения сгенерированных параметров безопасности (credentials) пользователя, которые будут использованы при работе с CLI AWS.

Эти параметры безопасности необходимо скопировать в конфигурацию CLI AWS. Для этого надо выполнить:

Открыть окно консоли (терминала) локальной машины. Для Linux/Mac OS X это будет консоль bash, а для Windows консоль PowerShell.

В окне локальной консоли (не AWS Management Console) ввести : aws configure

2. Будет запрошено 4 блока информации:

AWS access key ID - необходимо скопировать соответствующий блок из колонки Access key ID страницы "Complete", открытой в браузере и вставить его в соответствующую строку локальной консоли.

AWS secret access key скопировать соответствующий блок из колонки Secret access key ID страницы "Complete", открытой в браузере и вставить его в соответствующую строку локальной консоли.

Default region name — Ввести наименование региона en-central-1.

Default output format – Ввести json/

В результате окно локальной консоли должно выглядеть так:

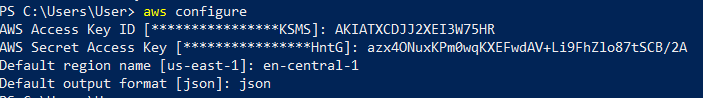


Рисунок 7 – Конфигурирование AWS в консоли

Кликнуть кнопку Close на странице "Complete" мастера создания пользователя.

Проверка правильности развертывания ПО CLI AWS

На предыдущем шаге CLI AWS был сконфигурирован для работы пользователя 02-03-isit-cli.

Сейчас необходимо проверить проверить работоспособность развернутого и настроенного CLI AWS.

Если в ответ будет выдан список регионов, то это значит, что AWS CLI работает !

Использование командной строки AWS для создания облачной инфраструктуры

Знакомство с некоторыми командами CLI AWS

Ввести текст virtualmachine

Для ОС Windows

Текст скрипта предназначенного для исполнения под ОС Linux находиться в файле архива скриптов и ПО, скаченного с GitHub и сохраненного на локальной машине перед выполнением задания 1.

Откройте каталог в котором располагается файл virtualmachine.ps1, выберите его правой кнопкой и в контекстном меню выберите пункт Run with PowerShell, для его запуска на выполнение.

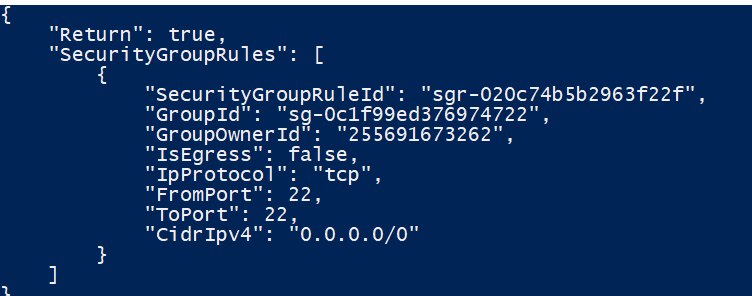


Рисунок 8 – Выполнение скрипта в консоли

Очистка ресурсов AWS

Внимание! После выполнения скрипта virtualmachine независимо от используемой ОС, необходимо убедиться, что виртуальная машина, созданная скриптом действительно была удалена. Это можно сделать либо с помощью CLI AWS, либо с помощью консоли сервиса EC2.

а. Выполните проверку с помощью CLI AWS используя знания полученные при выполнении шага 1, данного задания.

б. Выполните проверку с помощью GUI консоли сервиса EC2, используя знания полученные при выполнении ранее выполненных лабораторных работ.

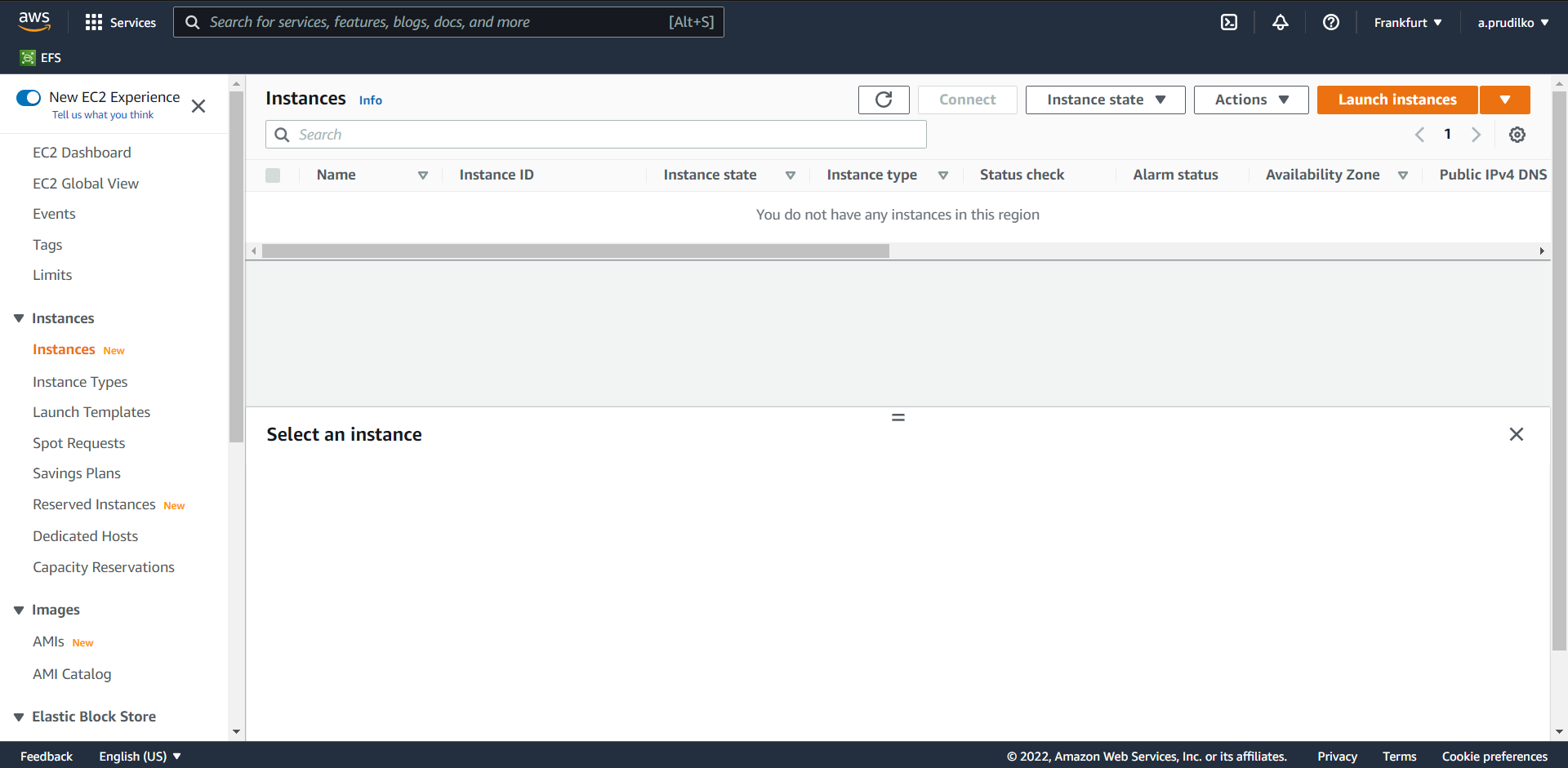


Рисунок 9 – Проверка удаления в консоли EC2

Создание (программирование) облачной инфраструктуры с помощью Node.js SDK AWS

Установка Node.js

Загрузите дистрибутив Node.js для Вашей локальной ОС

Все примеры были проверены для Node.js 8.

Разверните Node.js в своей ОС исполняемой на локальной машине.

Проверьте работоспособность платформы Node.js, выполнив в консоли Вашей ОС команду:

node --version



Рисунок 11 – Проверка версии Node.js

Убедитесь в том, что инструмент пакетного менеджера npm также был установлен:

npm –version

В ответ на эту команду не должно быть сообщений об ошибках.

Установка программы nodecc

Для демонстрации возможностей по управлению облачной инфраструктурой AWS с помощью SDK AWS, будет использоваться программа nodecc (Node Control Center for AWS). Для установки этой программы на локальную машину необходимо:

Перейти в директорию в которой был развернут архив ПО и скриптов скаченный с GitHub перед началом выполнения задания 1, и далее перейти в каталог /chapter04/nodecc/.

Выполнить в консоли ОС команду:

npm install

Для запуска nodecc на выполнение исполнить команду:

node index.js

Откроется окно nodecc. В левой части окна nodecc содержится перечень действий которые можно выполнить, а в правой содержиться список образов виртуальных машин с которыми можно работать. Этот список загружается из файла lib/listAMIs.js

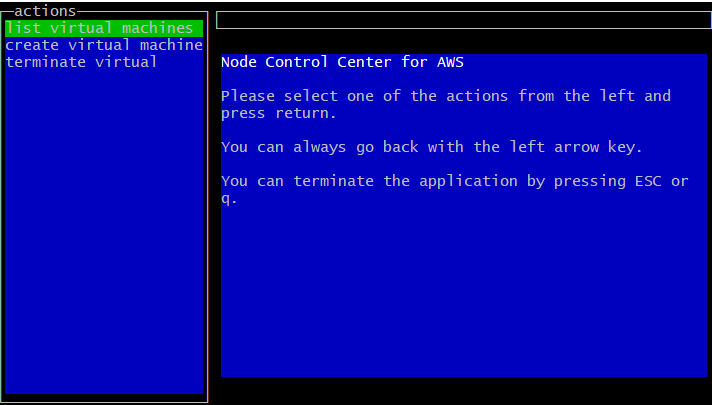


Рисунок 12 – Интерфейс nodecc

В левую часть окна всегда можно перейти с помощью клавиши "Стрелка влево".

Nodecc использует все настройки выполненные при установке AWS CLI для пользователя 02-03-isit-cli

Приложение nodecc можно закрыть с помощью клавиши ESC, или путем ввода символа "q".

Анализ примера кода node.js для работы с SDK AWS

Ознакомиться с кодом модуля lib/listAMIs.js.

Откройте код модуля в любом текстовом редакторе и проанализируйте его код, отметьте операторы обращения к SDK и укажите функции, выполняемые этими операторами. Результат анализа приведите в отчете.

Создание виртуальной машины с помощью nodecc

Для запуска ВМ выберите какой-либо образ ВМ в правой половине окно nodecc, а затем выберите действие "create virtual machine" в левой половине.

В правой части окна nodecc будет отображен список подсетей, в этом списке надо выбрать какую-нибудь сеть, после этого будет запущен экземпляр ВМ.

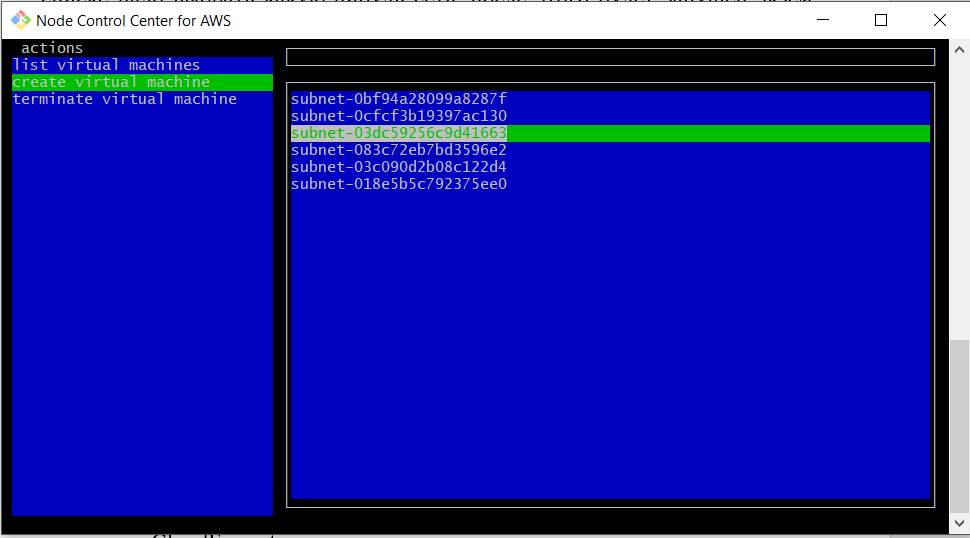


Рисунок 13 – Создание виртуальной машины в nodecc

Запущенные экземпляры ВМ отображаются в правой части окно nodecc в виде списка идентификаторов инстансов.

Использование шаблонов CloudFormation для создания облачной инфраструктуры

Создание шаблона развертывания CloudFormation. Шаблон развертывания CloudFormation можно создать разными способами:

* с нуля, с помощью текстового редактора или средствами визуальной среды разработки (IDE);
* использовать CloudFormation Designer - графический пользовательский интерфейс, предлагаемый AWS;
* начать с шаблона из публичной библиотеки и адаптировать его к своим нуждам;
* использовать инструмент CloudFormer, предназначенный для создания шаблонов, для существующих инфраструктур, представляемый AWS;
* использовать шаблоны, предлагаемые другими поставщиками.

Создание первого шаблона

Откройте AWS Management Console. Кликните Services в области навигации, а затем кликните CloudFormation service.

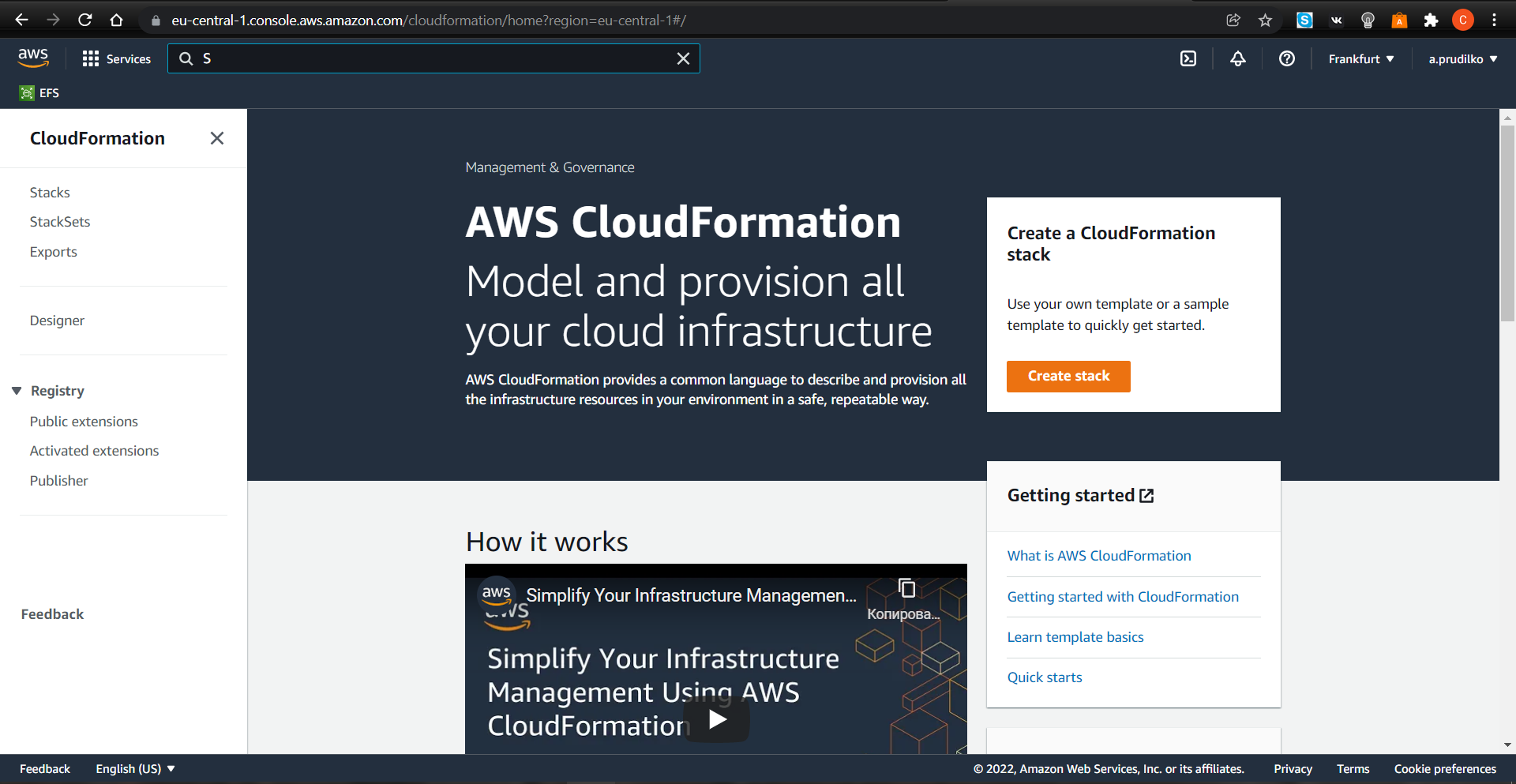


Рисунок 14 – CloudFormation консоль

Кликните кнопку "Create Stack" для запуска мастера. Откроется страница четырех-шагового мастера создания стека. Откроется начальная страница сервиса CloudFormation, в которой отображается состояние всех стеков, запущенных пользователем.

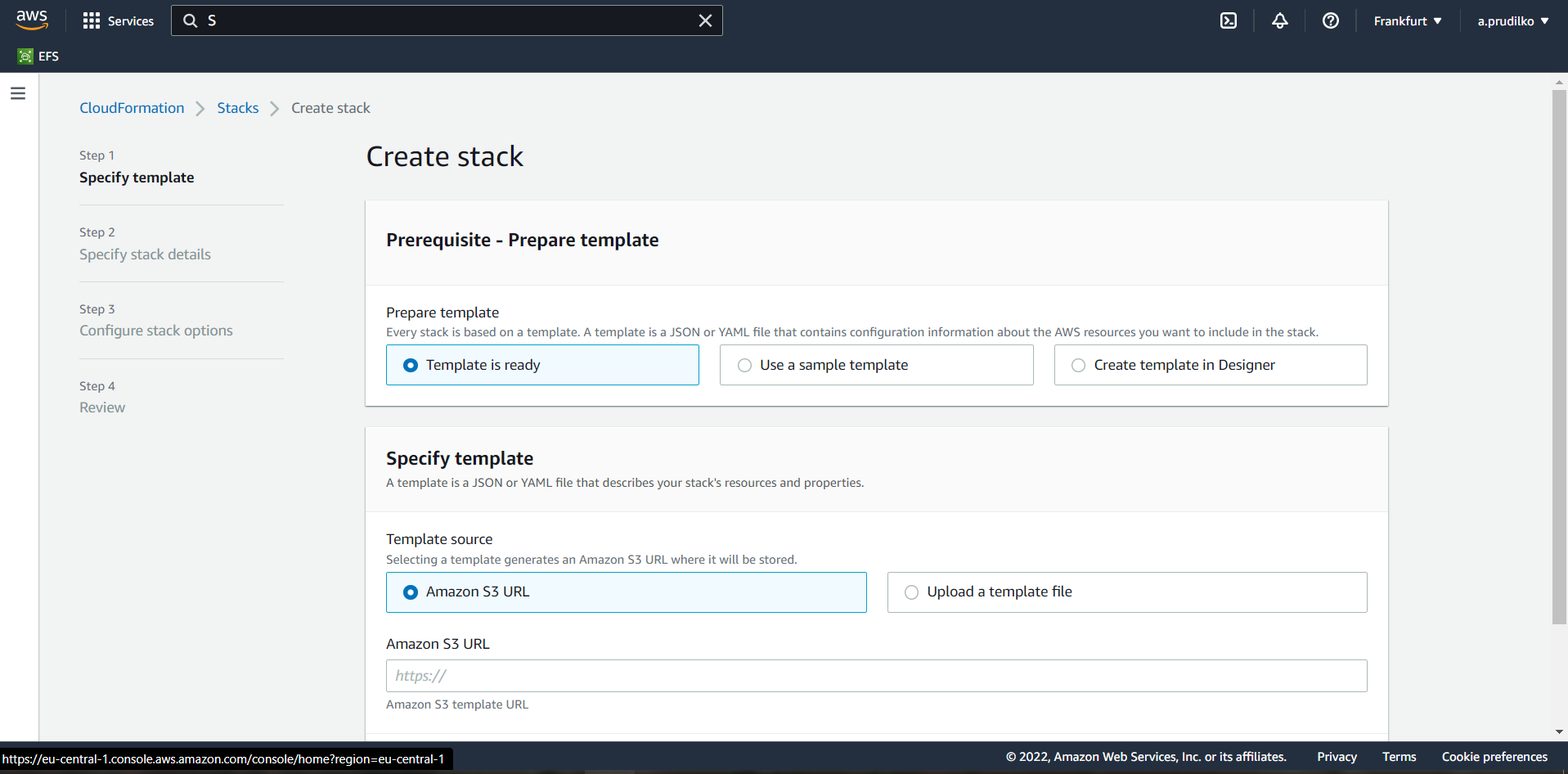


Рисунок 15 – Страница создания стека

На этой странице выберите селектор "Specify in Amazon S3 template URL" и в поле, расположенном ниже селектора, введите место расположения шаблона:

https://S3.amazonaws/com/awsinaction-code2/chapter04/virtualmachine.yaml.

Нажать кнопку Next. Откроется страница "Specify Details".

Далее необходимо определить имя стека и параметры. Рекомендуется следующий формат имени стека:

02-03-isit-Server

И следующие значения параметров:

InstanceType - t2.micro

KeyName - xx-yy-mykey

Subnet - выбрать первое значение из выпадающего списка.

VPC - выбрать первое значение из выпадающего списка.

Кликните Next.

На следующем шаге можно определить опциональные (не обязательные) тэги стека и дополнительные настройки. Этот шаг в данной лабораторной работе пропускается. Нажмите Next в открывшемся окне опционных настроек (третье окно мастера). Настройки, для которых не были указаны значения, будут установлены по умолчанию. Откроется страница "Review".

Проверьте настройки создаваемого стека и кликните "Create" для его создания, иначе измените настройки, и опять "Create" стек.

Начнется процесс создания стека (развертывания инфраструктуры).

Если создание стека пройдет успешно, то откроется страница сервиса CloudFormation, в которой будет отражено текущее состояние стека.

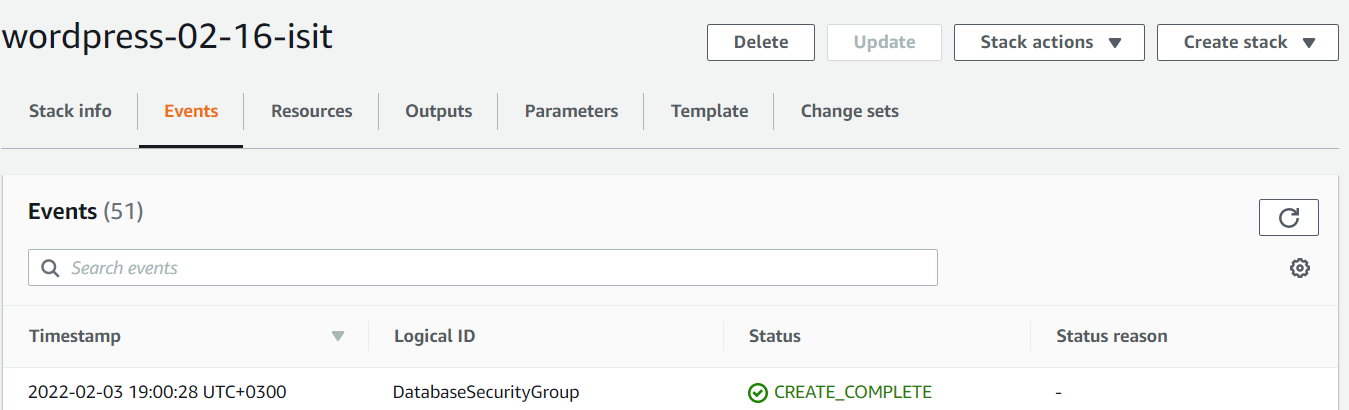


Рисунок 16 – Успешное развертывание стека

CREATE\_IN\_PROGRESS означает, что стек создается, CREATE-COMPLETE - стек успешно создан.

Когда сервис будет создан, выберите стек по его имени и кликните ссылку на закладку Outputs для того, чтобы увидеть публичное имя инстанса ЕС2, которое отображается колонке "Keу".

Сейчас можно протестировать возможность модификации типа инстанса в развернутом стеке.

Изменение стека

На странице CloudFomation Stacks выберите стек и кликните кнопку "Update Stack". Будет запущен мастер, подобный мастеру создания стека.

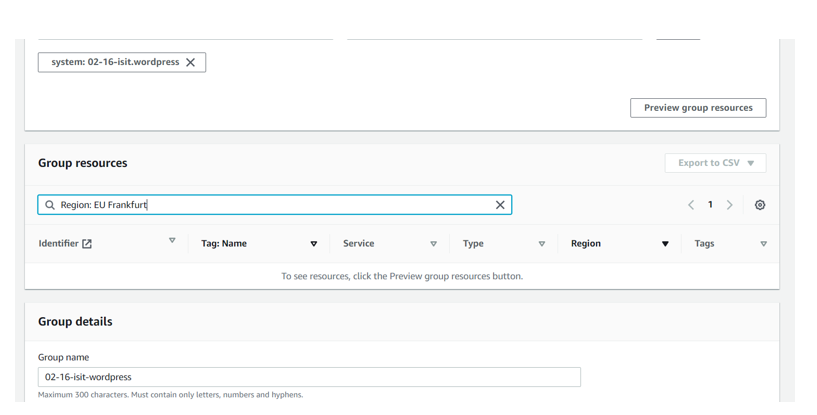


Рисунок 17 – Изменение стека

На первой странице мастера (Select Template) выберите селектор "Use current template" и кликните Next.

На втором шаге мастера можно изменить параметр InstanceType, выбрав значение t2.small для увеличения в 2 раза производительности сервера или t2.medium для четырех кратного увеличения производительности.

В третьем окне мастера модификации стека ничего изменять не надо. Просто кликните Next.

Четвертая страница отображает данные изменения, проверьте их и кликните Update. Дождитесь завершения перезапуска стека. И получите публичное имя стека.

### Шаг 4. Удаление стека

Удалите стек, выбрав его на странице CloudFormation Stack, и кликните кнопку "Delete Stack".

Убедитесь, что стек удалился, иначе повторите процедуру удаления.

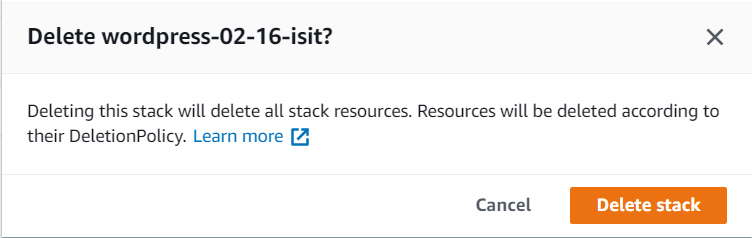


Рисунок 18 – Удаление стека