

VII Респ. науч. конф. студ., маг. и асп. “Актуальные вопросы физики и техники” – Гомель, ГГУ, 25 апреля 2018 г. – Ч.2, С.131-133.

7. Kharin, A.Yu. Robustness of sequential testing of hypotheses on parameters of M-valued random sequences / A.Yu. Kharin // Journal of Mathematical Sciences. – 2013. – Vol. 189(6). – P. 924-931.

8. Kharin, A. Robustness evaluation in sequential testing of composite hypotheses / A. Kharin // Austrian Journal of Statistics. – 2008. – Vol. 37 (1). – P. 51-60.

9. Galinskij, V. On minimax robustness of Bayesian statistical prediction / V. Galinskij, A. Kharin // Probability Theory and Mathematical Statistics. – Vilnius: TEV. – 1999. – P. 259-266.

А.И. Чернышев (ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **А.В. Воруев**, канд. техн. наук, доцент

ОПРЕДЕЛЕНИЕ ГРАНИЦ ДЕЙСТВИЯ ОБЛАЧНОЙ СРЕДЫ

Иерархическая архитектура обслуживания приложений позволяет сочетать аппаратные решения, программные интерфейсы обслуживания клиентов (приложения) и программно-реализованные (виртуализированные) сетевые сервисы для повышения эффективности обслуживания конечного оборудования.

Произведем оценку проекта с четко определенными границами, метриками и включенными устройствами. На рисунке 1 представлена топология сети, включающей в себя все необходимые устройства для анализа разработанной fog computing framework, используемой для проведения экспериментов.

Топология состоит из четырех различных устройств, промежуточного программного обеспечения облачной туманности, узлов управления туманностью, ячеек туманности и датчиков. Среднее промежуточное ПО облачной туманности создает верхний уровень топологии и выполняется на Macbook Pro, подключенном к облачной среде OpenStack, и используется для загрузки и решения задач. Остальные компоненты, включенные в настройку, развернуты на Potatoes Pis.

Узел Fog Control Node 1 (FCN 1) напрямую связан с облачной туманностью (CFM) и используется как контроллер, то есть узел управления облачной туманностью для подсоединенных туманных устройств. Туманность контролируемая и организованная FCN 1,

состоит из FCN 2 и FCN 3. Оба, FCN 2 и FCN 3, контролируют подключенные туманные ячейки, которые обрабатывают данные с подключенных устройств IoT.

В этой установке подключенные устройства IoT представляют собой сенсорные модули, состоящие из датчика температуры и влажности. Эти сенсорные модули подключены к соответствующему картофельному участку с помощью сенсорных модулей.

Топология сети настроена как сеть беспроводной локальной сети точкой доступа Linksys. Эта точка доступа подключена к Интернету и работает как шлюз подключения каждого Potatoes Pi к Интернету.

В разработанном испытательном проекте каждый компонент должен быть подключен к Интернету, поскольку туманные службы требуют возможности загрузки данных Docker Image для создания и развертывания динамических служб.

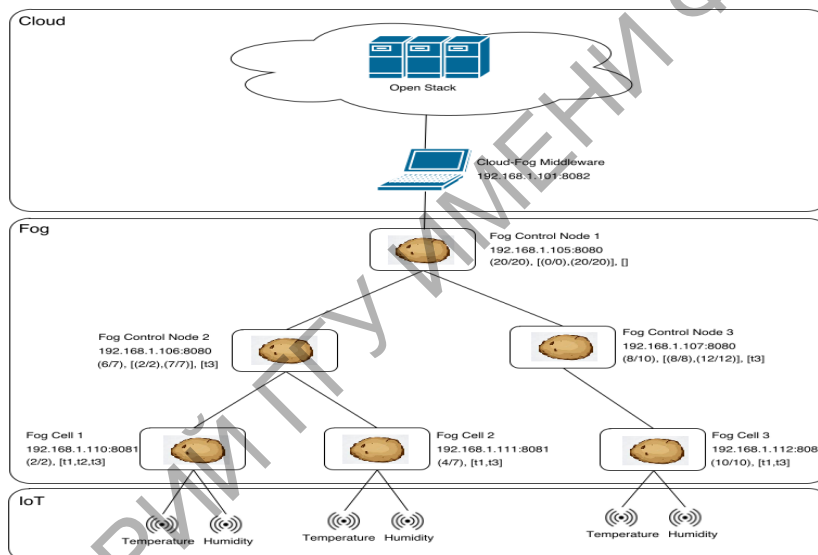


Рисунок 1 – Логические связи в топологии

В топологии сети для оценки, каждое устройство в слое туманности содержит различную информацию (сверху вниз): имя устройства, IP-адрес и порт, местоположение устройства, диапазон местоположения и список типов услуг, которые устройство может обрабатывать.

Расположение устройства и его местоположение необходимы для оценки ответственного родителя для нового соединения устройств туманности. Каждое устройство, запрашивающее родителя с его собственным местоположением устройства, получает возвращаемого родителя, который покрывает область, в которой находится запрашивающее устройство. Таким образом можно наглядно

определить расположение устройств, туманность контролируемая и организованная Fog Control Node 1, состоит из Fog Control Node 2 и Fog Control Node 3. Оба, FCN 2 и FCN 3, контролируют подключенные туманные ячейки Fog Cell 1, Fog Cell 2 и Fog Cell 3 соответственно, которые обрабатывают данные с подключенных устройств IoT. На рисунке 2 представлено визуальное представление размещения устройств.

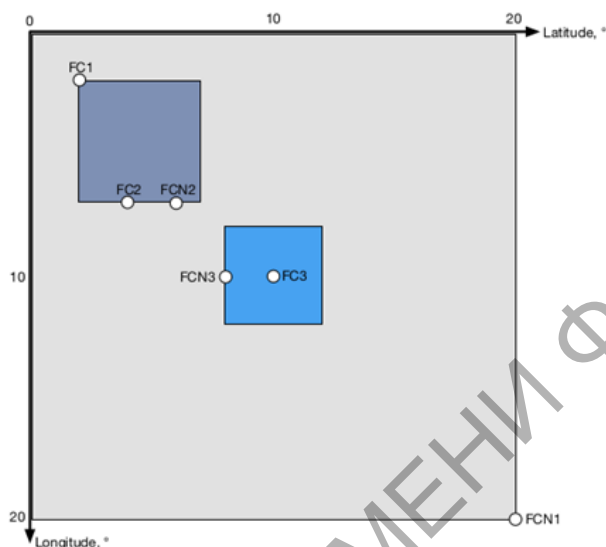


Рисунок 2 – Расположение устройств и их диапазоны

Чтобы была возможность дать оценку, надо определить запросы приложения, состоящие из набора задач и продолжительности приложения, которые будут запрашиваться при управлении FCN и развернуто в системе.

Приложение запрашивает отправлять в систему данные соответствующие следующим условиям:

- запрошенные типы обслуживания туманности должны соответствовать типам услуг устройств туманности;
- запрашиваемые облачные службы должны быть перенаправлены в репозиторий Docker Hub с префиксом «fogframe /» перед запросом;
- все поля приложения должны быть заполнены в соответствии с примерным сценарием.

Даже если выполнение приложения соответствует указанным условиям, приложение может выйти из строя. Возможные причины: перегружена облачная среда OpenStack, исчерпаны плавающие IP-адреса, туманность перегружена. Число развертываемых облачных виртуальных машин теоретически неограниченно, но на практике – ограничено доступными облачными ресурсами и ограничением плавающих IP-адресов.

Литература

1. Соколов, Н.А. Сценарии реализации концепции "Интернет вещей" / Н.А.Соколов// М.: Первая миля. – 2016.– №4. – С. 50-54.

2. Bachmann, K. Design and Implementation of a Fog Computing / K.Bachmann / Technische Universität Wien [Электронный ресурс]. – 2017. – Режим доступа: http://www.infosys.tuwien.ac.at/staff/sschulte/paper/Bachmann_Master.pdf. – Дата доступа: 12.03.2019.

3. Пушкарев М.С. Интернет вещей (IoT): понятие и значение для формирования правовой основы цифровой трансформации экономики / М.С.Пушкарев / Matters of Russian and International Law. - 2018. - Vol. 8. - С. 16-22.

А.Л. Чиркун (ГГУ имени Ф. Скорины, Гомель)
Науч. рук. **М.И. Жадан**, канд. физ.-мат. наук, доцент

СОЗДАНИЕ ANDROID ПРИЛОЖЕНИЯ «AOF»

В наше время большинство людей используют свои мобильные устройства как средство для покупки (продажи) товаров и услуг. Это связано ростом темпа жизни и все делается «на ходу». Также такая популярность мобильных устройств связана с огромным количеством приложений для платформ IOS и Android. В связи с этим пользователь нуждается в простом и удобном приложении, которое не вынуждает его подолгу сидеть и разбираться в принципах его работы.

Для разработки приложения для ОС Android необходимо установить Android Studio версии Portable, требуется среда исполнения Java Runtime Environment, набор разработчика Java Development Kit и эмулятор «Android Virtual Device», который позволяет проводить виртуальное тестирование приложений на эмулированном мобильном устройстве с ОС Android. Разработка приложений началась с тщательного обдумывания удобного и простого в использовании интерфейса, а также способы его реализации.

Приложение получает информацию из баз данных и предоставляется пользователю. Структура пользовательского интерфейса приложения состоит из набора контроллеров, на которых отображаются полученная информация из базы данных и элементы управления. Переход между страницами организован с помощью нижнего меню.