

TION IOT CORE. ИНСТРУКЦИЯ ПО ДОСТУПУ К СЕРВЕРУ И РАЗВЕРТЫВАНИЮ СЕРВИСА

Версия документа: 1.0 (17.07.2016)

Содержание

Введение	2
Доступ к работающему на Google Compute Engine сервису	2
Развертывание нового сервиса с нуля	4
Журналирование (логгирование)	5
Настройки базы данных.....	5

Введение

Данный документ содержит следующую информацию:

- информацию, необходимую для доступа к сервису TION IoT Core, развернутому на платформе Google compute Engine специалистами ООО РоботАэро,
- инструкцию по развертыванию сервиса на произвольной платформе с нуля,
- информацию о системе журналирования событий в сервисе,
- информацию о настройке доступа к базе данных.

Для выполнения инструкций некоторых разделов документа от читателя требуются навыки администрирования серверов на Linux.

Доступ к работающему на Google Compute Engine сервису

Готовый сервис TION IoT Core развернут в системе Google Cloud Platform (GCP) / Google Compute Engine (GCE) – IaaS-платформе, предоставляющей сервер с полным доступом к операционной системе.

Документация к системе CGE: <https://cloud.google.com/compute/docs/>

Основная страница управления сервером CGE (см. рисунок 1):

https://console.cloud.google.com/compute/instances?project=tion-iot-core&graph=GCE_CPU&duration=P1D

Для доступа к этой странице, а также для SSH-доступа к серверу, требуются логин и пароль от общей системы аккаунтов Google. Для данного проекта был создан аккаунт со следующими данными:

Логин: **tion.iot.core@gmail.com**

Пароль: **COXEFGU_11**

Платформа GCP позволяет добавлять существующих пользователей Google (например, почты Google) в число администраторов серверов GCE.

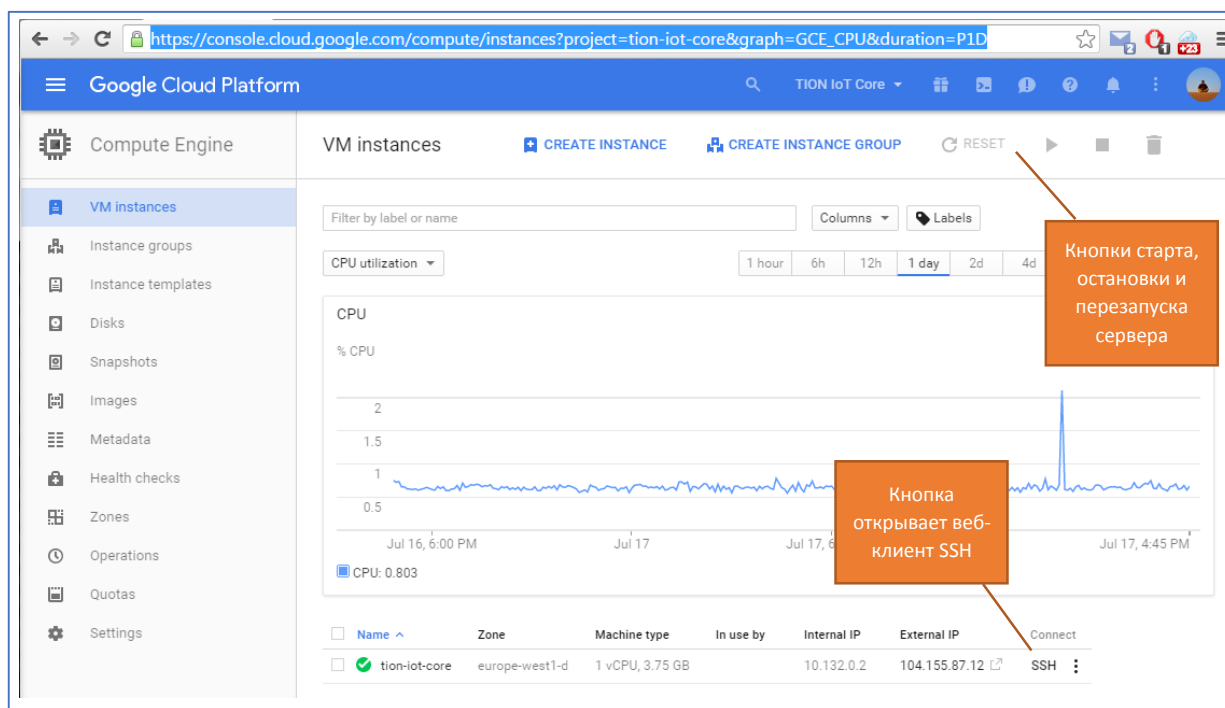


Рис. 1. Страница управления сервером в GCE

Параметры сервера:

- Процессор: Intel Xeon 2.3GHz (Haswell)
- Память: 3.75 Gb
- Диск: 80 Gb SSD
- ОС: Debian GNU/Linux 8
- Внешний IP: 104.155.87.12

Сервис TION IoT Core доступен по адресу <http://104.155.87.12:8080/> (например, рисунок 2).

Управление ПО сервера осуществляется через SSH-канал, который запускается со страницы управления сервером (см. рис. 1).

Исходный код сервиса лежит в папке /opt/tion-iot-core-backend.

Сервис зарегистрирован в ОС с помощью системы systemd, скрипт tion-iot-core.service находится в папке /etc/systemd/system. Старт и остановка сервиса осуществляются с помощью команд systemd:

```
systemctl start tion-iot-core, systemctl stop tion-iot-core.
```

Для отладки возможен ручной запуск (поскольку сервис запускается автоматически при старте, для ручного запуска требуется предварительная остановка сервиса):

```
cd /opt/tion-iot-core-backend && node index.js.
```

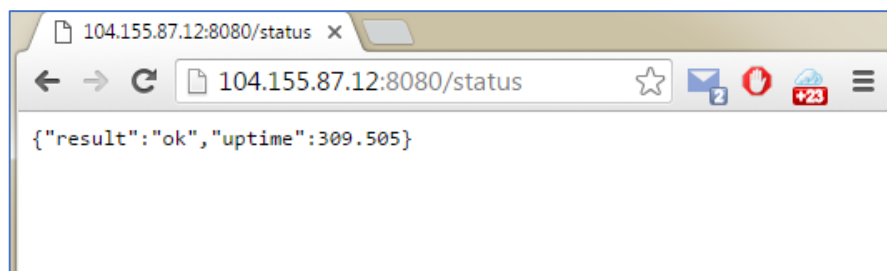


Рис. 2. Результат запроса /status к сервису, работающему в GCE.

Разворачивание нового сервиса с нуля

Для разворачивания сервиса с нуля на произвольной системе необходим предустановленный софт:

- nodejs¹ ($\geq 4.4.7$),
- node package manager, npm² ($\geq 2.15.8$),
- mongodb³ ($\geq 2.4.9$, рекомендуется ≥ 2.6)

На более старых версиях этих программ работоспособность не проверялась.

Программы являются кросс-платформенными, запуск сервиса возможен на всех платформах, для которых существуют версии указанных программ. Ниже приводится инструкция по установке и запуску для Linux-подобных систем.

1. Архив с исходным кодом `tion-iot-core-backend-src.zip` распаковать в папку, например, `/opt/`
2. `cd /opt/tion-iot-core-backend`,
3. Установка библиотек, используемых приложением: `npm install`,
4. Доступ на запись журналов: `chmod a+w logs`,
Лучше дать права на запись только тому пользователю, от которого будет запускаться приложение, с помощью команд `chown` и `chmod`.
5. Запуск:
 - а. для отладочного запуска: `node index.js`,
 - б. для итогового запуска: `NODE_ENV=production node index.js`.
6. Автозапуск.
 - а. Для ОС с `systemd` можно использовать прилагаемый скрипт автозапуска `tion-iot-core.service`. Скрипт проверялся в ОС Debian GNU/Linux 8, для других систем, возможно, придется изменить поля `User` и др. Скрипт необходимо скопировать в `/etc/systemd/system/`, выполнить `systemctl daemon-reload && systemctl enable tion-iot-core`.

¹ <https://nodejs.org/en/>

² <https://www.npmjs.com/>

³ <https://www.mongodb.com/>

- б. Для ОС, использующих Init, launchd, Upstart и другие, также возможен автозапуск.

Журналирование (логгирование)

Для журналирования используются библиотеки `winston`⁴ и `morgan`⁵.

При запуске в production-окружении (с переменной окружения `NODE_ENV=production`), все события уровня выше `debug` добавляются в файл `./logs/app.log` (с ротацией файла каждые 5 МБ).

В другом окружении к журналированию в файл добавляется журналирование в `stdout` с тем же уровнем `debug`.

Изменение настроек журналирования производится в файле `logging.js` в корне проекта.

Настройки базы данных

Изменение настроек доступа к базе данных производится в файле `db.js` в корне проекта.

Сервис использует СУБД `mongodb`, адрес сервера задается при подключении к БД в функции `init()` в файле `db.js`. Там же возможно указание логина и пароля для доступа к СУБД.

Сервис использует базу данных `tiondb`. База создается автоматически, как и все коллекции БД.

⁴ <https://github.com/winstonjs/winston>

⁵ <https://github.com/expressjs/morgan>