

ИССЛЕДОВАНИЕ ПОДХОДОВ К ОБНАРУЖЕНИЮ СОБЫТИЙ В ИНФРАСТРУКТУРЕ OPENSTACK

Петухов Д.А., Докучаев В.А.

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования "Московский технический университет связи и информатики" "МТУСИ", г. Москва, Российская Федерация

Научный руководитель: Докучаев В.А., д-р техн. наук, профессор, заведующий кафедрой СИТuС МТУСИ

Аннотация. Исследование сфокусировано на детализации подходов к выявлению событий, связанных с работоспособностью виртуальной машины в облачной инфраструктуре. Рассматривается среда, построенная на компонентах *OpenStack*. В ходе исследования были определены события, которые могут приводить к перерыву работоспособности виртуальной машины, а также предложены способы проактивного обнаружения данных событий.

Ключевые слова: *OpenStack*, обнаружение событий, мониторинг компонентов, открытый исходный код, свободное программное обеспечение

Введение. Стремление облачных провайдеров в предоставлении услуг на базе собственных разработок часто опирается на продукты с открытым исходным кодом. Такие платформы, как, например, *OpenStack* широко и активно развиваются сообществом разработчиков, однако часто, как и любые другие продукты с открытым исходным кодом имеют пробелы в документации и реализации метрик для оценки их функционирования. Более того, разработка уникального сервиса в дальнейшем требует формулирования особенных подходов и решений для оценки его качества. Поскольку платформа имеет очень широкую область применения, является крайне актуальным вклад в исследование базовых сценариев событий в инфраструктуре, построенной на компонентах *OpenStack*. Результаты исследования позволят определить опорные точки и слабые места инфраструктуры *OpenStack*.

Базовым сценарием применения *OpenStack* является предоставление услуг типа «Инфраструктура как услуга» (*IaaS – Infrastructure as a Service*), т.е. предоставляется набор вычислительных ресурсов, в нашем случае виртуальные машины (VM), как услуга. В обеспечении функционирования этой услуги используется минимальный набор компонентов *OpenStack: Neutron, Nova, Keystone, Glance*. Реализация развёртывания данных компонентов, их составляющих и обеспечение взаимодействия между ними являются ключевыми факторами, которые необходимо уточнить для исследования возможностей обнаружения событий.

Определение целевого состояния виртуальной машины. Уточним, что цель данного исследования направлена на поиск алгоритмов обеспечения показателей качества в облачной инфокоммуникационной инфраструктуре. В рамках этого исследования рассматривается один из аспектов поставленной задачи, необходимый для детализации и описания предметной области услуги вычислений *IaaS*. В ходе ранее проведённых исследований [3] было определено, что в нормативной документации для облачных услуг определяются такие показатели, как доступность, непрерывность, функциональность, обслуживаемость, производительность и другие. В то же время, методы и способы реализации данных сценариев изучены недостаточно. Сложность взаимодействия компонентов инфраструктуры *OpenStack* сужает возможности оценки качества исполнения сценариев с виртуальными машинами. Можно предложить два подхода к оценке качества взаимодействия компонентов:

1. функциональная оценка работоспособности компонентов, задействованных в предоставлении конечной услуги;
2. временная оценка, характеризующая выполнение операций каждым из компонентов за заданное время.

При первом подходе поставленную задачу можно декомпозировать на следующие подзадачи:

- описание информационных потоков и консолидация сведений о состояниях объектов, в том числе, на основе технической документации по устройству механизмов компонентов *OpenStack* [2];

- составление списка сообщений, выявленных в ходе поиска обрабатываемых или необрабатываемых исключений в исходном коде компонентов *OpenStack* [4].

При втором подходе требуется:

- тщательная настройка журналирования всех компонентов с точными временными отметками о событиях;

- корректные настройки получения информации о времени с серверов точного времени;

- использование единого формата записи времени.

Прежде чем приступить к более глубокому исследованию возможностей для реализации данных подходов, необходимо обозначить область определения запущенной виртуальной машины в *OpenStack*. Необходимо обосновать значимость исследуемых взаимодействий между компонентами. Базовая услуга виртуальной машины, как конечный сервис для потребителя, представляет собой следующие ключевые возможности:

- предоставляется виртуальный вычислительный ресурс с заданной конфигурацией (*vCPU, RAM*);

- предоставляются доступы к сетевому управлению виртуальной машиной (*VNC, SPICE, RDP, SSH*);

- обеспечивается доступность виртуальной машины другим виртуальным машинам внутри одной подсети;

- предоставляются доступы к *ACPI* управлению виртуальной машиной (включение, выключение, перезагрузка);

- предоставляется подключение одного или несколько сетевых интерфейсов к виртуальной машине (ввод-вывод осуществляется);

- предоставляется подключение одного или нескольких накопителей к виртуальной машине (ввод-вывод осуществляется).

Вышеизложенные функциональные возможности определим, как целевое работоспособное состояние виртуальной машины. Важно уточнить, что такое определение виртуальной машины позволяет определить границы сервиса, не выходя за его пределы. Например, конкретные типы дисков, а также драйвер взаимодействия с системой хранения данных, выходят за рамки этого определения. Напротив, выделяются только возможности, относящиеся к функционированию виртуальной машины (пребыванию в работоспособном состоянии) в терминах *OpenStack*.

После того, как целевое состояние виртуальной машины определено, необходимо провести исследование с целью определить какие компоненты *OpenStack* обеспечивают его функционирование. Данный процесс представлен в виде схемы компонентов, обеспечивающих функционирование виртуальной машины в инфраструктуре *OpenStack* (Рисунок 1). Пояснения взаимосвязи компонентов между собой на приведённой выше схеме приведены в [5].

Исследование событий недоступности виртуальной машины. Чтобы определить события, сигнализирующие о возможной недоступности виртуальной машины, используем подход исследования исходного кода компонентов [4]. Анализ кода обнаруживает, что *OpenStack* не содержит в кодовой базе какого-либо компонента, контролирующего исполнение последовательности операций какой-либо функции.

Напротив, архитектура компонентов такова, что обмен командами между компонентами осуществляется через очередь сообщений без гарантии исполнения. Это обнаруживает недостаток и сложность при поиске проблем на инфраструктуре. С другой стороны, расширенная отладочная информация позволяет непосредственно сопоставлять отклонения с целевыми состояниями виртуальной машины, хотя это и требует глубокой проработки. Иными

словами, недостаток документации и инструментов управления необходимо компенсировать тщательной настройкой инструментов мониторинга.

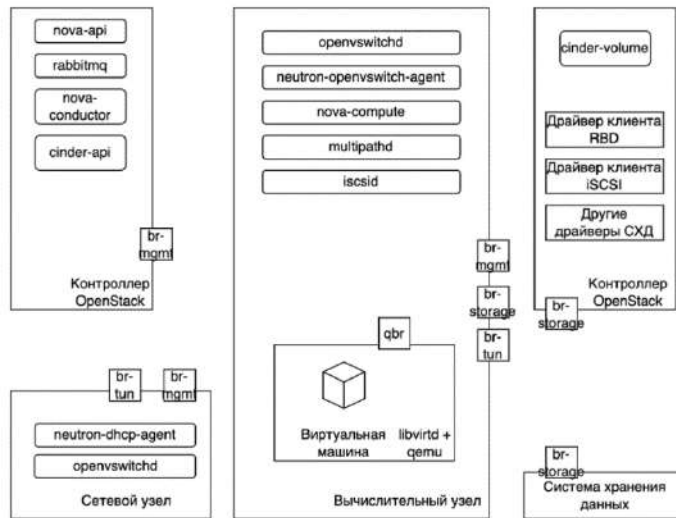


Рисунок 1 – Схема компонентов, обеспечивающих функционирование виртуальной машины в инфраструктуре OpenStack

Например, на рисунке 2 показано, что драйвер *libvirt* содержит 274 журналируемых записи о событиях с виртуальной машиной. Примеры сообщений о невыполнении функций приведены в таблице 1. Анализ и сбор сообщений является централизованной задачей, которую возможно решать такими инструментами мониторинга, как *Zabbix*, *Elasticsearch*, *Grafana* и *Kibana*. Корректная настройка данных инструментов позволит корректно обрабатывать события и устанавливать для них принимаемое решение.

```

files to include
./git-repos/openstack/nova/nova/virt/libvirt/driver.py

files to exclude

274 results in 1 file - Open in editor
LOG.info("Instance soft rebooted successfully.",
LOG.warning("Failed to soft reboot instance. "
LOG.info("Instance shutdown successfully.",
LOG.info("Instance may have been rebooted during soft "
LOG.info("Instance rebooted successfully.",
LOG.info("Instance already shutdown.", instance=instance)
LOG.debug("Shutting down instance from state %s", state,
LOG.info("Instance shutdown successfully after %d seconds.",
LOG.debug("Instance in state %s after %d seconds - "
LOG.debug("Ignoring libvirt exception from shutdown "
LOG.info("Instance failed to shutdown in %d seconds.",
LOG.exception(
    
```

Рисунок 2 – Анализ исходного кода драйвера libvirt компонента OpenStack Nova

Результаты исследования показали, что составление таблицы на основе сообщений в исходном коде и сопоставленных им функций, являющихся показателями работоспособности ВМ, позволит предупредить потенциальные события с виртуальной машиной.

Данный подход не является единственным. Рекомендации сообщества *OpenStack* также включают использование глобального и локального идентификатора запроса для отслеживания цепочки вызовов [7]. Цепочки запросов, или вызовов, приводящих к ошибке, также являются объектами для отслеживания.

Таблица 1 – Функции и сообщения об ошибках в компонентах OpenStack

Функция	Сообщение	Принимаемое решение
Выключение, перезагрузка ВМ	LOG.info("Instance failed to shutdown in %d seconds.", timeout, instance=instance) LOG.warning("Failed to soft reboot instance. Trying hard reboot.", instance=instance)	Определить массовость Проверить доступность СХД и вывести <i>UUID ВМ</i> и информацию о подключенных дисках
Подключение диска	LOG.info("Getting block stats failed, device might have 'been detached. Instance=%(instance_name)s 'Disk=%(disk)s Code=%(errcode)s Error=%(e)s', {'instance_name': instance.name, 'disk': disk_id, 'errcode': errcode, 'e': e}, instance=instance)	Определить подключение диска и вывести <i>UUID ВМ</i> и информацию о подключенных дисках

Дальнейшее развитие исследования обнаруживает перспективу на основе результатов, полученных авторами в [8, 9]. В частности, авторами были выделены риски при эксплуатации облачной инфраструктуры территориально-распределённой компании: сбой контроля выделяемой дисковой ёмкости, задержка операций ввода-вывода при переключении контроллеров системы хранения данных, превышение квот на выделение ресурсов и другие. Данные и многие другие отказы необходимо обнаруживать проактивно для обеспечения качества сервиса.

Заключение. Обеспечение показателей качества функционирования виртуальной машины в облачной инфраструктуре требует подробно проработанного комплекса мер. Архитектура компонентов *OpenStack* не обнаружила компонентов, позволяющих отслеживать функционирование виртуальной машины во всех состояниях и этапах штатным способом. Для оценки работоспособности функций мощным инструментом оказалась обработка определённых сообщений, которые возможно выделить в исходном коде компонентов *OpenStack*. Проверка событий на массовость, а также на основе взаимосвязей компонентов позволит проактивно обнаруживать события на инфраструктуре.

Список литературы

1. *Self-healing SIG//OpenStack: [caim]*. — URL: https://wiki.openstack.org/wiki/Self-healing_SIG
2. *Technical Reference Deep Dives//OpenStack: [caim]*. — URL: <https://docs.openstack.org/nova/latest/reference/index.html>
3. Петухов Д. А. Анализ показателей качества облачных услуг на мировом рынке / Петухов Д. А., Докучаев В. А // *REDS: ТЕЛЕКОММУНИКАЦИОННЫЕ УСТРОЙСТВА И СИСТЕМЫ*. — 2021
4. *Mirrors of opendev.org/openstack*. — GitHub: [caim]. — URL: <https://github.com/openstack>.
5. *Conceptual architecture. Installation Guide documentatio //OpenStack: [caim]*. — URL: <https://docs.openstack.org/install-guide/get-started-conceptual-architecture.html#get-started-conceptual-architecture>
6. *Global event correlation//Zabbix: [caim]*. — URL: https://www.zabbix.com/documentation/5.4/en/manual/config/event_correlation/global
7. *Faults — Compute API Guide 2.1.0 documentation // OpenStack: [caim]*. — URL: <https://docs.openstack.org/api-guide/compute/faults.html>
8. Петухов, Д. А. Исследование подходов к управлению рисками при создании облачной инфраструктуры территориально-распределённой компании / Д. А. Петухов, В. А. Докучаев // *Телекоммуникации и информационные технологии*. — 2019. — Т. 6. — № 2. — С. 140-145.
9. Maklachova, V. V. Risks identification in the exploitation of a geographically distributed cloud infrastructure for storing personal data / V. V. Maklachova, V. A. Dokuchaev, V. Y. Statev // *2020 International Conference on Engineering Management of Communication and Technology, EMCTECH 2020 - Proceedings, Vienna, 20–22 октября 2020 года*. — Vienna, 2020. — P. 9261541. — DOI 10.1109/EMCTECH49634.2020.9261541.

UDC 004.45

A RESEARCH OF APPROACHES FOR EVENTS DETECTION IN OPENSTACK INFRASTRUCTURE

Petukhov D.A., Dokuchaev V.A.

Moscow Technical University of Communications and Informatics “MTUCI”, Moscow, Russian Federation

Scientific supervisor: Dokuchaev V.A. — Dr. Sc, Prof., Head of Department NIT&S, MTUCI

Annotation. This article is focused on detailing the approaches on events detection at the target working state of virtual machine in cloud infrastructure. The environment, which is being researched is based on OpenStack components. As the output of the research the events were revealed, which can result in instance service interruption, and the ways for proactive discovery were proposed.

Keywords: OpenStack, event detection, monitoring, open-source software, free software