

# Применение Ceph в современных облачных инсталляциях

В.Г. Ильин, П.С. Цынгалев, А.С. Боронников

**Аннотация**— На сегодняшний день применение систем управления динамических кластеров блочных устройств является одной из неотъемлемых частей прогрессивных облачных инсталляций. Данные технологии позволяют избежать проблем при управлении и конфигурации дисков, например, таких как несогласованность, неструктурированность и низкая доступность хранилища данных. Несмотря на наличие известных программных решений данных задач, на сегодняшний день, с учетом современных реалий, необходимо подбирать аналоги систем, которые свободно распространяются и имеют открытый программный код. Таким программным обеспечением является Ceph. Данный продукт обладает стандартным набором функций подобного класса программ и имеет большое число уникальных полезных методов, что делает его достойным аналогом для использования в облачных инсталляциях.

Целью данной статьи является популяризация использования кластеров Ceph. В работе описаны типичные инсталляции для запуска на них Ceph-кластера и способы его развертывания. Проведено тестирование на задачах, приближенным к реальным, с которыми могут столкнуться пользователи, и выполнен анализ полученных результатов. Полученные результаты применимы как для представления об аналоге систем управления динамическими кластерами блочных устройств в уже существующих облачных инсталляциях, так и для подбора системы в качестве решения для только создаваемых инсталляций.

Настоящая статья демонстрирует краткое представление о возможностях Ceph без физического развертывания необходимой инфраструктуры, что позволит потенциальным пользователям сэкономить ресурсы и время.

**Ключевые слова**— Ceph, облачные вычисления, Private Cloud Solution, Elastic Block Storage, кластер, виртуальная машина, облачная инфраструктура.

## I. ВВЕДЕНИЕ

На сегодняшний день в нашей стране крайне актуальна проблема использования проприетарного прикладного программного обеспечения. Большинство

компаний, которые его поставляют, либо вовсе ушли с российского рынка, либо существенно ограничили свою деятельность. Такие же ограничения коснулись программ, которые позволяют проектировать отказоустойчивые системы хранения данных для управления большим количеством блочных устройств [12-19].

Например, лидер по проектированию и поставке программного обеспечения компания DELL с продуктом ScaleIO также ограничила деятельность на территории Российской Федерации. В связи с чем у компаний, использующих данное решение, возникли проблемы при обращении в поддержку и лицензирования программного обеспечения.

Достойным аналогом является программное обеспечение Ceph. Оно выполняет те же функции, что и большинство программ для управления системами хранения данных в облачных инсталляциях. При этом Ceph является полностью бесплатным решением и имеет полностью открытый исходный код, что позволяет его свободно распространять.

Отметим, что Ceph может работать в двух основных режимах — в режиме объектного хранилища (object storage) и в качестве системы управления блочными устройствами (block device). В данной статье будет рассмотрено использование Ceph в качестве менеджера блочных устройств.

Таким образом, целью данного исследования является популяризация программного обеспечения Ceph для решения конкретного класса прикладных задач [14-17].

## II. КОНФИГУРАЦИЯ ТЕСТОВОГО СТЕНДА

Перед развертыванием системы необходимо описать оборудование, на котором будут производиться тестирование кластера Ceph и последующий анализ результатов.

На сегодняшний день рабочими серверными решениями являются продукты компании Dell и IBM. Рассмотрим несколько серверов, которые подойдут в качестве кандидатов для тестового стенда.

В ходе поставленной задачи были описаны основные общие характеристики абстрактного сервера, который в последующем использовался в качестве тестового стенда. В таблице 1 указаны минимальные требования к узлам Ceph-кластера из официальной документации [3].

Поскольку основой Ceph-кластера являются диски, начнем с описания их параметров. В условиях современных реалий выполнен подбор среди компаний-производителей, которые по-прежнему доступны в

Статья получена 21 октября 2022.

Виктор Георгиевич Ильин, Кафедра вычислительной техники, МИРЭА – Российский технологический университет, Москва, Россия (e-mail: vgilyin@yahoo.com).

Павел Сергеевич Цынгалёв, Кафедра вычислительной техники, МИРЭА – Российский технологический университет, Москва, Россия (e-mail: pstsingalev@mail.ru).

Антон Сергеевич Боронников, Кафедра вычислительной техники, МИРЭА – Российский технологический университет, Москва, Россия (e-mail: boronnikov-anton@mail.ru).

России. Анализ рынка показал, что данными компаниями являются “Samsung” и “Huawei”. Среди продуктов данных компаний осуществлялся выбор блочных устройств.

**Таблица 1. Минимальные требования к узлам Ceph-кластера**

Компонент кластера Ceph	Компонент	Минимальные требования
ceph-osd	Процессор	1 core minimum 1 core per 200-500 MB/s 1 core per 1000-3000 IOPS Results are before replication. Results may vary with different CPU models and Ceph features. (erasure coding, compression, etc) ARM processors specifically may require additional cores. Actual performance depends on many factors including drives, net, and client throughput and latency. Benchmarking is highly recommended.
	Оперативная память	4GB+ per daemon (more is better) 2-4GB often functions (may be slow) Less than 2GB not recommended
	Блочное устройство	1x storage drive per daemon
	DW/WAL	1x SSD partition per daemon (optional)
	Сеть	1x 1GbE+ NICs (10GbE+ recommended)
ceph-mon	Процессор	2 cores minimum
	Оперативная память	2-4GB+ per daemon
	Объем диска	60GB per daemon
	Сеть	1x 1GbE+ NICs
ceph-mds	Процессор	2 cores minimum
	Оперативная память	2GB+ per daemon
	Объем диска	1 MB per daemon
	Сеть	1x 1GbE+ NICs

На основе документации Ceph к требованиям блочных устройств [1] описаны минимальные требования: размер диска – 1 терабайт; интерфейсы подключения – SAS, SATA. Количество операций ввода/вывода зависит от размера самого диска.

Перейдем к описанию спецификаций сервера, на основе которых будет проводиться тестирование Ceph-кластера:

- Процессор: Intel E5-2698v3 (2.3GHz, 16C, 35MB, 9.6GT/s QPI, 120W);
- Оперативная память: 32\*8 Gb PC4 DDR4 ECC RDIMM;
- HDD диски: 600GB 2.5" SAS 6Gbps HS 15k;
- SSD диски: 2TB 2.5" Near Line SAS 12Gbps HS 7.2k;
- Raid контроллер: 1Gb RAID controller (0,1,10,5,50);
- Модуль управления: IMM2 Advanced (KVM);
- Основной адаптер: 4x1 Gb Integrated Ethernet Card;
- Блоки питания: 900W Platinum Hot-swap Power Supply (x3650 M5).

Также стоит упомянуть операционную систему, на базе которой проектировался кластер Ceph. Была выбрана CentOS 7.5 [7, 9] благодаря стабильной версии пакетов и ядра Linux. Разработчики Ceph также рекомендуют использовать CentOS. Тестирование проходило на виртуальных машинах с типом гипервизора QEMU KVM [8, 10]. Использовалась версия Ceph 14 Nautilus, так как на данный момент она является наиболее стабильной и на сегодняшний день имеет поддержку. Выбор поставщиков комплектующих, указанных ранее, зависит от размера бюджета компании для реализации Ceph-кластера и от ситуации на рынке.

### III. ОСНОВЫ ФУНКЦИОНИРОВАНИЯ КЛАСТЕРА CEPH

Перед развертыванием самого кластера необходимо описать принципы работы Ceph-кластера, из каких узлов (частей) он состоит и как узлы между собой взаимодействуют [11]. Ceph – это файловая система [2] для управления хранилищ облачных инстанций. Ceph-кластер включает в себя 4 основных сущности: монитор (monitor), менеджер (manager), демон объектного хранилища (object storage demon – OSD), сервер метаданных (MDS – metadata server). Далее рассмотрим функционал каждой сущности.

Monitor – сущность-наблюдатель, которая работает независимо от других сущностей кластера и необходима для сбора информации. Информацию хранит в так называемых картах (manager map, OSD map, MDS map, CRUSH map). Данные карты описывают правила хранения log-файлов (файлы журнала событий) каждой из сущности, представляя собой критическое состояние кластера, необходимое демонам Ceph для координации друг с другом. Мониторы также отвечают за управление аутентификацией между демонами и клиентами.

Manager – сущность Ceph, которая работает в фоновом режиме и отвечает за отслеживание показателей времени выполнения и текущего состояния кластера Ceph, включая использование хранилища, текущие показатели производительности и загрузку системы. Данные сущности могут размещать модули на основе Python [4] для управления и предоставления информации о кластере Ceph, например Ceph Dashboard [5].

OSD – сущность объектного хранилища, которая, как и Manager, работает в фоновом режиме. Отвечает за хранение данных, обработку репликации данных, восстановление, повторную балансировку и предоставляет некоторую информацию мониторинга для мониторов и менеджеров Ceph, проверяя другие сущности OSD Ceph на предмет доступности. Для каждого диска существует минимум одна OSD-сущность. Обычно физический диск представляет собой одну или несколько OSD сущностей. Затем OSD сущности объединяются в группы размещения (placement group – pg), что позволяет обеспечить высокую доступность, а также упорядочить структуру объектного хранилища для размещения данных.

MDS – сущность, которая хранит метаданные (данные о содержимом узлов кластера) от имени файловой системы Ceph (блочные устройства Ceph и хранилище объектов Ceph не используют MDS). Сущности хранения метаданных Ceph позволяют пользователям с файловой системы, поддерживающей POSIX-стандарты, выполнять операции командной строки, не создавая огромной нагрузки на кластер хранения Ceph.

Ceph хранит данные как объекты в логических пулах хранения. Используя CRUSH (Controlled Replication Under Scalable Hashing – алгоритм, использующий хэширование для расчета хранения и извлечения данных в распределенных системах на основе кластерного типа хранения), Ceph вычисляет, какая группа размещения должна содержать объект, и к какой группе размещения принадлежит конкретная OSD. Алгоритм CRUSH позволяет кластеру хранения Ceph динамически масштабироваться, перебалансироваться и восстанавливаться.

#### IV. ОСОБЕННОСТИ РАЗВЕРТЫВАНИЯ КЛАСТЕРОВ CEPH

При развертывании кластеров Ceph процесс установки необходимо выполнить самостоятельно, в отличие от процесса установки проприетарного программного обеспечения. На сегодняшний день существуют два наиболее эффективных решения для развертывания Ceph-кластера — установка из Ansible playbook (желаемое состояние каких-либо сущностей с использованием файла конфигурации на языке yaml: в данном файле прописываются количества узлов кластера и их характеристики), который можно получить с официального сайта Ceph, и полностью ручная установка.

Поэтому развертывание Ceph полностью вручную не требуется, так как существует Ansible playbook, который при минимальной настройке конфигурационных файлов способен развернуть Ceph-кластер. Однако при большом масштабировании кластера процесс развертывания может быть усложнен. Поэтому в данной ситуации может быть полезна ручная установка. К основным причинам можно отнести:

- безопасность при развертывании большой системы;
- меньше зависимость от других разработчиков Ansible playbook;
- более гибкая настройка, что позволяет получить

более точную желаемую конфигурацию.

#### V. ТЕСТИРОВАНИЕ КЛАСТЕРОВ CEPH

Тестирование запускалось на кластере, развернутом на 3 серверах. Предположим, что развернутый кластер используется для решения некоторого типа прикладных задач. Опишем наиболее частые случаи:

- 1 виртуальная машина, 1 диск — данный пример позволяет отобразить максимальную производительность 1 диска при использовании в нескольких виртуальных машинах.
- 1 виртуальная машина, 4 диска — данный случай является самым распространённым при конфигурации 1 виртуальной машины, так как позволяет реализовывать структуру RAID 10, что обеспечивает отказоустойчивость хранения данных при оптимальной производительности;
- 10 виртуальных машин, 1 диск — распространённый случай, если необходимо разделение дискового пространства для доступа некоторого числа виртуальных машин к одному конкретному разделу диска для совместной записи/чтения.

Таким образом, становится возможным приблизить тестовый Ceph-кластер к тем кластерам, которые в данный момент используются компаниями в своих решениях.

##### *A. Тестирование при конфигурации 1 виртуальная машина, 1 диск (1VM-1D)*

Данный тест позволяет определить максимальную производительность одного блочного устройства (диска) при полностью простаивающем кластере. Более высокой производительности вряд ли удастся достигнуть при иной конфигурации. При тестировании дисков использовалась утилита “fio” [6, 13] со следующими ключевыми параметрами:

- ioengine — тип engine составляющей утилиты “fio”, который содержит основные команды и инструкции для проведения тестирования на чтение/запись. Был использован стандартный engine “libaio”, как рекомендуют сами разработчики данной утилиты;
- buffered 0 — буферизированный ввод отключен, значит что файловая система не будет использовать кэш, что позволит минимизировать влияние системы на производительность диска и провести более точное тестирование;
- bs — размер блока операций чтения, записи;
- rw randrw — данные будут записываться и читаться случайным образом;
- rwmixread 60 — смешанное чтение, запись, 60% процентов операций при тестировании будут приходиться на чтение, 40 на запись;
- iodepth — глубина очереди.

Результаты тестов для размера блока записи 4 килобайт показаны в таблицах 2-3.

**Таблица 2. Результаты тестирования при конфигурации 1VM-1D на 3 серверах, 6 SSD-дисков с размером диска 64 Гбайт, блок записи 4 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	3500	109	10
8	6700	203	23
16	11200	375	25
32	20500	625	30

**Таблица 3. Результаты тестирования при конфигурации 1VM-1D на 3 серверах, 6 HDD-дисков с размером диска 64 Гбайт, блок записи 4 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	3400	90	11
8	6200	92	28
16	6300	85	34
32	6600	80	46

Результаты тестов для размера блока записи 64 Кбайт показаны в таблицах 4-5.

**Таблица 4. Результаты тестирования при конфигурации 1VM-1D на 3 серверах, 6 SSD-дисков с размером диска 64 Гбайт, блок записи 64 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	1200	90	18
8	2000	162	17
16	2600	180	19
32	2800	178	22

**Таблица 5. Результаты тестирования при конфигурации 1VM-1D на 3 серверах, 6 HDD-дисков с размером диска 64 Гбайт, блок записи 64 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	1100	68	17
8	1700	106	11
16	1600	105	11
32	1500	94	13

Анализ результатов тестирования описан в разделе

VI.A.

*В. Тестирование при конфигурации 1 виртуальная машина, 4 диска (1VM-4D)*

Данный случай является самым распространённым, при конфигурации одной виртуальной машины, так как позволяет реализовывать RAID 10, что обеспечивает отказоустойчивость хранения данных при оптимальной производительности. Результаты тестов для размера блока записи 4 Кбайт показаны в таблицах 6-7.

**Таблица 6. Результаты тестирования при конфигурации 1VM-4D на 3 серверах, 6 SSD-дисков с размером диска 64 Гбайт, блок записи 4 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	3900	250	7
8	4200	264	7
16	5700	324	9
32	9900	610	14

**Таблица 7. Результаты тестирования при конфигурации 1VM-4D а на 3 серверах, 6 HDD-дисков с размером диска 64 Гбайт, блок записи 4 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	3700	220	9
8	3900	244	11
16	4600	245	15
32	4500	250	13

Результаты тестов для размера блока записи 64 Кбайт показаны в таблицах 8-9.

**Таблица 8. Результаты тестирования при конфигурации 1VM-4D на 3 серверах, 6 SSD-дисков с размером диска 64 Гбайт, блок записи 64 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	1900	120	17
8	3100	211	24
16	3300	224	35
32	3900	264	41

**Таблица 9. Результаты тестирования при конфигурации 1VM-4D на 3 серверах, 6 HDD-дисков с размером диска 64 Гбайт, блок записи 64 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	1700	90	13
8	1900	92	14
16	2100	95	14
32	2000	95	15

Анализ результатов тестирования описан в разделе VI.B.

*C. Тестирование при конфигурации 10 виртуальных машин, 1 диск (10VM-1D)*

В данном случае необходимо разделение дискового пространства для доступа некоторого числа виртуальных машин к одному конкретному разделу диска для совместной записи/чтения. Результаты тестов для размера блока записи 4 Кбайт показаны в таблицах 10-11. Результаты тестов для размера блока записи 64 Кбайт показаны в таблицах 12-13. Анализ результатов тестирования описан в разделе VI.C.

**Таблица 10. Результаты тестирования при конфигурации 10VM-1D на 3 серверах, 6 SSD-дисков с размером диска 64 Гбайт, блок записи 4 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	6500	203	12
8	7000	219	13
16	7200	225	11
32	7500	234	14

**Таблица 11. Результаты тестирования при конфигурации 10VM-1D на 3 серверах, 6 HDD-дисков с размером диска 64 Гбайт, блок записи 4 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	4700	194	12
8	5000	201	12
16	5100	203	13
32	5000	199	14

**Таблица 12. Результаты тестирования при конфигурации 10VM-1D на 3 серверах, 6 SSD-дисков с размером диска 64 Гбайт, блок записи 64 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	5400	169	8
8	6400	200	8
16	7500	234	9
32	8000	250	8

**Таблица 13. Результаты тестирования при конфигурации 10VM-1D на 3 серверах, 6 HDD-дисков с размером диска 64 Гбайт, блок записи 64 Кбайт**

Глубина очереди	IOPS, кол-во операций ввода/вывода в секунду	Пропускная способность, Мбит/с	Общая задержка, мс
4	3600	95	13
8	4100	109	12
16	4500	110	15
32	4200	105	14

## VI. АНАЛИЗ РЕЗУЛЬТАТОВ ТЕСТИРОВАНИЯ

Для упрощения представления результатов тестов разобьем их по размерам блоков чтения и записи — 4 Кбайт и 64 Кбайт соответственно.

*A. Тестирование при конфигурации 1 виртуальная машина, 1 диск*

Данный тест позволил узнать максимальную производительность диска.

Результаты тестирования при размере блока 4 Кбайт:

- Среднее значение на SSD: 10475 IOPS;
- Среднее значение на HDD: 5625 IOPS.

Результаты тестирования при размере блока 64 Кбайт:

- Среднее значение на SSD: 2150 IOPS;
- Среднее значение на HDD: 1475 IOPS.

*B. Тестирование при конфигурации 1 виртуальная машина, 4 диска*

Также важно проанализировать производительность одного RBD клиента и его масштабируемость. Число дисков, равное 4, было выбрано исходя из большой вариативности конфигурации RAID-массивов.

Результаты тестирования при размере блока 4 Кбайт:

- Среднее значение на SSD: 5925 IOPS;
- Среднее значение на HDD: 4175 IOPS.

Результаты тестирования при размере блока 64 Кбайт:

- Среднее значение на SSD: 3050 IOPS;
- Среднее значение на HDD: 1925 IOPS.

### C. Тестирование при конфигурации 10 виртуальных машин, 1 диск

Распространенный случай организации общего хранилища данных среди множества вычислительных устройств. При тестировании операции чтения/записи выполнялись одновременно с десяти машин.

Результаты тестирования при размере блока 4 Кбайт:

- Среднее значение на SSD: 7050 IOPS;
- Среднее значение на HDD: 4950 IOPS.

Результаты тестирования при размере блока 64 Кбайт:

- Среднее значение на SSD: 6825 IOPS;
- Среднее значение на HDD: 4100 IOPS.

### D. Утилизация CPU RBD серверов

Разработчики Ceph утверждают о достаточности двух ядер процессора для корректного функционирования демона OSD. В результате тестирования выяснилось, что при варианте использования единственного диска множеством клиентов при жестком ограничении ресурсов OSD до двух ядер, демон OSD либо переходил в состоянии ошибки, либо имел место процесс деградации кластера.

### E. Общая оценка результатов тестирования

Ceph показывает хорошую производительность, эффективно расходуя предоставленные ему ресурсы. Из-за фактора мультипликации данных равному трем, результаты при чтении довольно сильно превосходят запись. Производительность системы в достаточной мере оправдывает затраты на оборудование.

## VII. ЗАКЛЮЧЕНИЕ

На сегодняшний день Ceph является лучшим решением для организации сетевого хранилища. Обладая обширным функционалом, он также показывает хорошие показатели при тестировании.

Благодаря гибкой конфигурации кластера Ceph является крайне простым к применению инструментом. В зависимости от требований пользователя способен адаптироваться к любым запросам, например, как выдаче максимального количества IOPS для высоконагруженной базы данных, так и обеспечению надежного хранилища с большим фактором мультипликации для хранения резервных копий.

В зависимости от потребностей эксплуататора Ceph способен предоставить хорошую производительность при превосходном уровне защиты консистентности данных.

Также в пользу Ceph играет фактор свободы от лицензирования. Не произойдет той ситуации, когда инфраструктура не сможет быть обновлена до актуальной версии продукта.

Данное решение будет актуально для IT-компаний среднего уровня на российском рынке, а также образовательных платформ и сетей магазинов.

Все вышеперечисленные факторы и отсутствие конкурента как такового в данном сегменте

программного обеспечения делает Ceph лучшим решением для частных облачных инсталляций.

## БИБЛИОГРАФИЯ

- [1] Ceph Block Device. – URL: <https://docs.ceph.com/en/quincy/rbd/> (дата обращения: 25.09.2022)
- [2] Ceph File System. – URL: <https://docs.ceph.com/en/quincy/cephfs/> (дата обращения: 26.09.2022)
- [3] Installing Ceph. – URL: <https://docs.ceph.com/en/quincy/install/> (дата обращения: 28.09.2022)
- [4] Ceph Dashboard. – URL: <https://docs.ceph.com/en/quincy/mgr/dashboard/> (дата обращения: 29.09.2022)
- [5] API Documentation. – URL: <https://docs.ceph.com/en/quincy/api/> (дата обращения: 30.09.2022)
- [6] FIO's documentation. – URL: <https://fio.readthedocs.io/en/latest/> (дата обращения: 2.10.2022)
- [7] CentOS Documentation. – URL: <https://docs.centos.org/en-US/docs/> (дата обращения: 4.10.2022)
- [8] J. Cannon, "Docker: A Project-Based Approach to Learning Kindle Edition", p.298
- [9] А.М. Кенин, Д.Н. Колисниченко., Самоучитель системного администратора, 6-е изд – СПб.: БХВ-Петербург, 2021. – 608 с.
- [10] V. Dakic, H.D. Chiramal, Prasad Mukhedkar, "Mastering KVM Virtualization", 2020, p. 686
- [11] Ceph network configuration. – URL: [https://access.redhat.com/documentation/en-us/red\\_hat\\_ceph\\_storage/5/html/configuration\\_guide/ceph-network-configuration](https://access.redhat.com/documentation/en-us/red_hat_ceph_storage/5/html/configuration_guide/ceph-network-configuration) (дата обращения: 6.10.2022)
- [12] Analyzing Ceph cluster optimize storage cost. – URL: <https://www.intel.com/content/dam/www/public/us/en/documents/reference-architectures/120315-analyzing-ceph-cluster-optimize-storage-costs.pdf> (дата обращения: 25.09.2022)
- [13] HDD testing prerequisites. – URL: <https://learn.microsoft.com/en-us/windows-hardware/test/hlk/testref/hard-disk-drive-testing-prerequisites>
- [14] X. Zhang, Ya. Wang, Q. Wang, X. Zhao, "A New Approach to Double I/O performance for Ceph Distributed File System in Cloud Computing," in 2019 2nd International Conference on Data Intelligence and Security (ICDIS), pp. 68–75.
- [15] H. Li, Sh. Zhang, Z. Guo; Z. Huang, L. Qian, "Test and Optimization of Large-scale Ceph System," in 2020 IEEE 3rd International Conference of Safe Production and Informatization (IICSPI), pp. 237–241.
- [16] K. Jeong, C. Duffy, J-S. Kim, J. Lee, "Optimizing the Ceph Distributed File System for High Performance Computing," in 2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), pp. 446–451.
- [17] M. Hilmi, E. Mulyana, H. Hendrawan, A. Taniwidjaja, "Analysis of Network Capacity Effect on Ceph Based Cloud Storage Performance," in 2019 IEEE 13th International Conference on Telecommunication Systems, Services, and Applications (TSSA), pp. 22–24.
- [18] M. Oh, S. Park, J. Yoon, S. Kim, K. Lee, S. Weil. H.Y. Yeom, M. Jung, "Design of Global Data Deduplication for a Scale-Out Distributed Storage System," in 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS), pp. 1063–1073.
- [19] M. Oh, J. Eom, J. Yoon, J.Y. Yun, S. Kim, H.Y. Yeom, "Performance Optimization for All Flash Scale-Out Storage," in 2016 IEEE International Conference on Cluster Computing (CLUSTER), pp. 316–325.

# Using Ceph in modern Cloud installation

V.G. Ilyin, P.S. Tsyngalev, A.S. Boronnikov

**Abstract** — To date, the use of control systems for dynamic clusters of block devices is one of the integral parts of progressive cloud installations. These technologies avoid problems in disk management and configuration, such as inconsistency, unstructured and low availability of data storage. Actual software solutions to these problems have long been known to everyone, but today, taking into account modern realities, it is necessary to select analogues of such systems that are freely distributed and have an open source code. Ceph is such software. This product has a standard set of functions of this class of programs and has a large number of unique useful methods, which makes it a worthy analogue for use in cloud installations.

Thus, the purpose of this article is to popularize the use of Ceph clusters. The paper describes typical installations for running a Ceph cluster on them and how to deploy it. Testing was carried out on tasks close to real ones that users may encounter, and an analysis of the results obtained was carried out. The article will be useful, first of all, both for presenting an analogue of control systems for dynamic clusters of block devices in existing cloud installations, and for selecting a system as a solution for newly created installations.

This article provides a glimpse of what Ceph can do without physically deploying the necessary infrastructure, which will save potential users time and resources.

**Keywords** — Ceph, Cloud Computing, Private Cloud Solution, Elastic Block Storage, cluster, virtual machine, cloud infrastructure.

## REFERENCES

- [1] Ceph Block Device [Online]. Available: <https://docs.ceph.com/en/quincy/rbd/>
- [2] Ceph File System [Online]. Available: <https://docs.ceph.com/en/quincy/cephfs/>
- [3] Installing Ceph [Online]. Available: <https://docs.ceph.com/en/quincy/install/>
- [4] Ceph Dashboard [Online]. Available: <https://docs.ceph.com/en/quincy/mgr/dashboard/>
- [5] API Documentation [Online]. Available: <https://docs.ceph.com/en/quincy/api/>
- [6] FIO's documentation [Online]. Available: <https://fio.readthedocs.io/en/latest/>
- [7] CentOS Documentation [Online]. Available: <https://docs.centos.org/en-US/docs/>
- [8] J. Cannon, "Docker: A Project-Based Approach to Learning Kindle Edition", p.298
- [9] A.M. Kenin, D.N. Kolisnichenko., System administrator tutorial, 6 edition. – SP.: BHV-Peterburg, 2021. – 608 c.
- [10] V. Dakic, H.D. Chiramal, Prasad Mukhedkar, "Mastering KVM Virtualization", 2020, p. 686
- [11] Ceph network configuration [Online]. Available: [https://access.redhat.com/documentation/en-us/red\\_hat\\_ceph\\_storage/5/html/configuration\\_guide/ceph-network-configuration](https://access.redhat.com/documentation/en-us/red_hat_ceph_storage/5/html/configuration_guide/ceph-network-configuration)
- [12] Analyzing Ceph cluster optimize storage cost [Online]. Available: <https://www.intel.com/content/dam/www/public/us/en/documents/reference-architectures/120315-analyzing-ceph-cluster-optimize-storage-costs.pdf>
- [13] HDD testing prerequisites [Online]. Available: <https://learn.microsoft.com/en-us/windows-hardware/test/hlk/testref/hard-disk-drive-testing-prerequisites>
- [14] X. Zhang, Ya. Wang, Q. Wang, X. Zhao, "A New Approach to Double I/O performance for Ceph Distributed File System in Cloud Computing," in 2019 2nd International Conference on Data Intelligence and Security (ICDIS), pp. 68–75.
- [15] H. Li, Sh. Zhang, Z. Guo; Z. Huang, L. Qian, "Test and Optimization of Large-scale Ceph System," in 2020 IEEE 3rd International Conference of Safe Production and Informatization (IICSPI), pp. 237–241.
- [16] K. Jeong, C. Duffy, J-S. Kim, J. Lee, "Optimizing the Ceph Distributed File System for High Performance Computing," in 2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), pp. 446–451.
- [17] M. Hilmi, E. Mulyana, H. Hendrawan, A. Taniwidjaja, "Analysis of Network Capacity Effect on Ceph Based Cloud Storage Performance," in 2019 IEEE 13th International Conference on Telecommunication Systems, Services, and Applications (TSSA), pp. 22–24.
- [18] M. Oh, S. Park, J. Yoon, S. Kim, K. Lee, S. Weil. H.Y. Yeom, M. Jung, "Design of Global Data Deduplication for a Scale-Out Distributed Storage System," in 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS), pp. 1063–1073.
- [19] M. Oh, J. Eom, J. Yoon, J.Y. Yun, S. Kim, H.Y. Yeom, "Performance Optimization for All Flash Scale-Out Storage," in 2016 IEEE International Conference on Cluster Computing (CLUSTER), pp. 316–325.