NetApp

Confluent Kafka with NetApp ONTAP storage controllers

NetApp Solutions

NetApp
May 15, 2023

Table of Contents

Confluent Kafka with NetApp ONTAP storage controllers

TR-4941: Confluent with NetApp ONTAP storage controllers

Karthikeyan Nagalingam, Joe Scott, NetApp
Rankesh Kumar, Confluent

To make the Confluent Platform more scalable and elastic, it must be able to scale and balance workloads very quickly. Tiered storage makes storing huge volumes of data in Confluent manageable by reducing this operational burden. The fundamental idea is to separate data storage from data processing, which makes it much easier to scale each independently.

Loaded with industry-leading innovations, NetApp ONTAP data management software provides Confluent with many advantages anywhere the data lives.

This document outlines performance benchmarks for the Confluent platform on NetApp ONTAP using a tiered storage benchmarking kit.

Solution

Confluent and NetApp AFF A900 storage controller powered by ONTAP are distributed systems designed for data streams. Both are horizontally scalable, fault tolerant, and provide excellent performance under load. They complement each other in distributed data streaming and stream processing with lower storage costs with data reduction technologies that minimize the data footprint. The AFF A900 storage controller provides great performance, while allowing the decoupling of compute and data storage resources. This simplifies system administration and allows resources to be scaled independently.

Solution architecture details

This section covers the hardware and software used for performance verification in Confluent Platform deployment with NetApp ONTAP for tiered storage. The following table covers the solution architecture and base components.

Platform component	Environment configuration
Confluent Platform version 6.2	• 3 x zookeepers • 8 x broker servers • 5 x tools servers • 1 x Grafana • 1 x control center
Operating system on all nodes	Linux (ubuntu 18.04)
NetApp ONTAP for warm buckets	• 1 x AFF A900 high-availability (HA) pair • 4 x 24 x 800 SSDs • S3 protocol • 100GbE
15 Fujitsu PRIMERGY RX2540 servers	• 2 CPUs; 16 physical cores total • Intel Xeon • 256GB physical memory • 100GbE dual port

Technology overview

This section describes the technology used in this solution.

NetApp ONTAP storage controller

NetApp ONTAP is a high-performance, enterprise- grade storage operating system.

NetApp ONTAP 9.8 introduces support for Amazon Simple Storage Service (S3) APIs. ONTAP supports a subset of Amazon Web Services (AWS) S3 API actions and allows data to be represented as objects in ONTAP-based systems across cloud providers (AWS, Azure, and GCP) and on-premises.

NetApp StorageGRID software is the flagship NetApp solution for object storage. ONTAP complements StorageGRID by providing an ingest and preprocessing point on the edge, expanding the data fabric powered by NetApp for object data, and increasing the value of the NetApp product portfolio.

Access to an S3 bucket is provided through authorized user and client applications. The following diagram shows the application accessing an S3 bucket.

Primary use cases

The primary purpose of supporting S3 APIs is to provide objects access on ONTAP. The ONTAP unified storage architecture now supports files (NFS and SMB), blocks (FC and iSCSI), and objects (S3).

Native S3 applications

An increasing number of applications are able to leverage ONTAP support for object access using S3. Although well-suited for high-capacity archival workloads, the need for high performance in native S3 applications is growing rapidly and includes:

- Analytics

- Artificial intelligence

- Edge-to-core ingest

- Machine learning

Customers can now use familiar manageability tools such as ONTAP System Manager to rapidly provision high-performance object storage for development and operations in ONTAP, taking advantage of the ONTAP storage efficiencies and security as they do so.

FabricPool endpoints

Beginning with ONTAP 9.8, FabricPool supports tiering to buckets in ONTAP, allowing for ONTAP-to-ONTAP tiering. This is an excellent option for customers who wish to repurpose existing FAS infrastructure as an object store endpoint.

FabricPool supports tiering to ONTAP in two ways:

- **Local cluster tiering.** Inactive data is tiered to a bucket located on the local cluster using cluster LIFs.

- **Remote cluster tiering.** Inactive data is tiered to a bucket located on a remote cluster in a manner similar to a traditional FabricPool cloud tier using IC LIFs on the FabricPool client and data LIFs on the ONTAP object store.

ONTAP S3 is appropriate if you want S3 capabilities on existing clusters without additional hardware and management. For deployments larger than 300TB, NetApp StorageGRID software continues to be the flagship NetApp solution for object storage. A FabricPool license is not required when using ONTAP or StorageGRID as the cloud tier.

NetApp ONTAP for Confluent tiered storage

Every data center needs to keep business-critical applications running and important data available and secure. The new NetApp AFF A900 system is powered by ONTAP Enterprise Edition software and a high-resilience design. Our new lightning-fast NVMe storage system eliminates disruptions to mission-critical operations, minimizes performance tuning, and safeguards your data from ransomware attacks.

From initial deployment to scaling your Confluent cluster, your environment demands rapid adaptation to changes that are nondisruptive to your business-critical applications. ONTAP enterprise data management, quality of service (QoS), and performance allow you to plan and adapt to your environment.

Using NetApp ONTAP and Confluent Tiered Storage together simplifies the management of Apache Kafka clusters by leveraging ONTAP as a scale-out storage target and enables independent scaling of compute and storage resources for Confluent.

An ONTAP S3 server is built on the mature scale-out storage capabilities of ONTAP. Scaling your ONTAP cluster can be performed seamlessly by extending your S3 buckets to use newly added nodes to the ONTAP cluster.

Simple management with ONTAP System Manager

ONTAP System Manager is a browser-based graphical interface that allows you to configure, manage, and monitor your ONTAP storage controller across globally distributed locations in a single pane of glass.
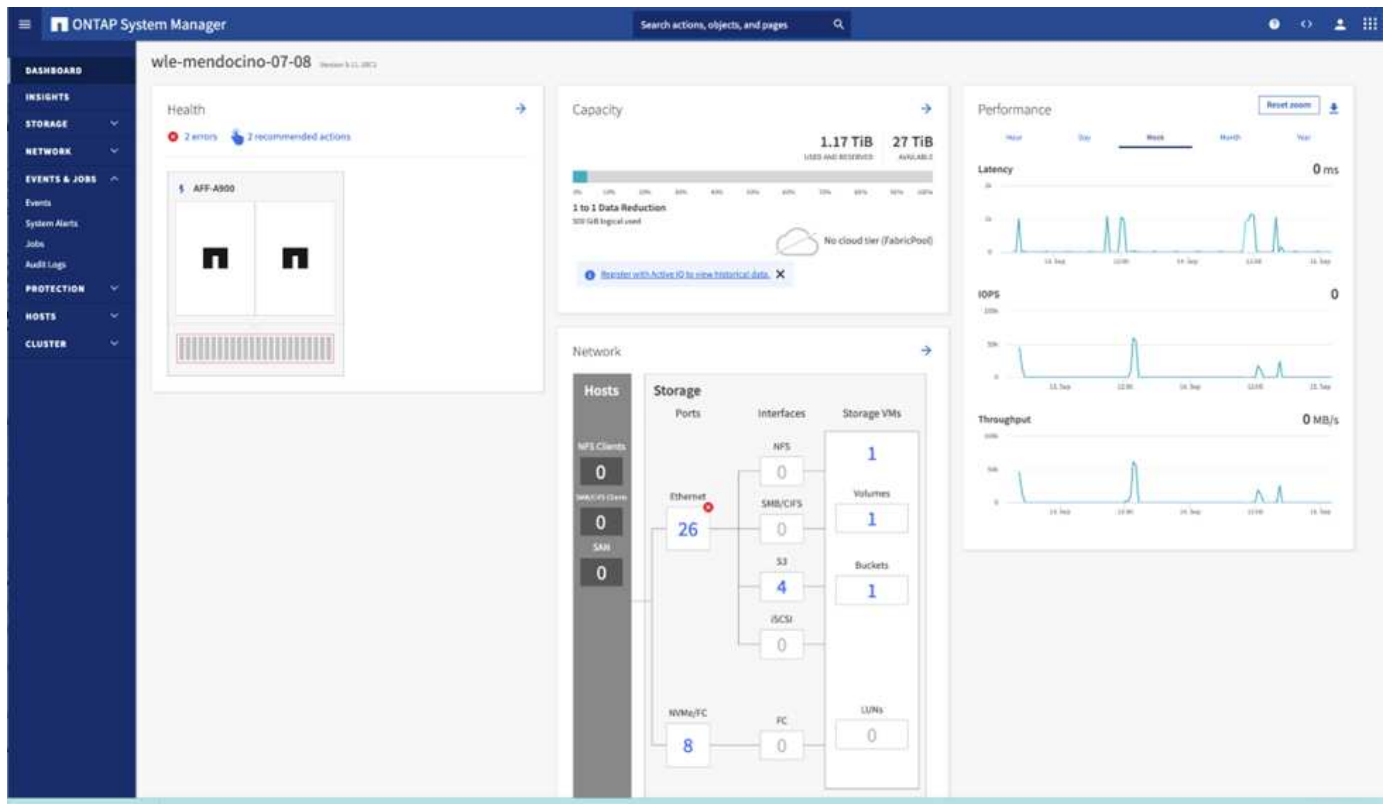
You can configure and manage ONTAP S3 with System Manager and the ONTAP CLI. When you enable S3 and create buckets using System Manager, ONTAP provides best-practice defaults for a simplified configuration. If you configure the S3 server and buckets from the CLI, you can still manage them with System Manager if desired or vice-versa.

When you create an S3 bucket using System Manager, ONTAP configures a default performance service level that is the highest available on your system. For example, on an AFF system, the default setting would be Extreme. Performance service levels are predefined adaptive QoS policy groups. Instead of one of the default service levels, you can specify a custom QoS policy group or no policy group.

Predefined adaptive QoS policy groups include the following:

- **Extreme.** Used for applications that require the lowest latency and highest performance.
- **Performance.** Used for applications with modest performance needs and latency.
- **Value.** Used for applications for which throughput and capacity are more important than latency.
- **Custom.** Specify a custom QoS policy or no QoS policy.

If you select **Use for tiering**, no performance service levels are selected, and the system tries to select low-cost media with optimal performance for the tiered data.

ONTAP tries to provision this bucket on local tiers that have the most appropriate disks, satisfying the chosen service level. However, if you need to specify which disks to include in the bucket, consider configuring S3 object storage from the CLI by specifying the local tiers (aggregate). If you configure the S3 server from the CLI, you can still manage it with System Manager if desired.

If you want the ability to specify which aggregates are used for buckets, you can only do so using the CLI.

Confluent

Confluent Platform is a full-scale data streaming platform that enables you to easily access, store, and manage data as continuous, real-time streams. Built by the original creators of Apache Kafka, Confluent expands the benefits of Kafka with enterprise-grade features while removing the burden of Kafka management or monitoring. Today, over 80% of the Fortune 100 are powered by data streaming technology, and most use Confluent.

Why Confluent?

By integrating historical and real-time data into a single, central source of truth, Confluent makes it easy to build an entirely new category of modern, event-driven applications, gain a universal data pipeline, and unlock powerful new use cases with full scalability, performance, and reliability.

What is Confluent used for?

Confluent Platform lets you focus on how to derive business value from your data rather than worrying about the underlying mechanics, such as how data is being transported or integrated between disparate systems. Specifically, Confluent Platform simplifies connecting data sources to Kafka, building streaming applications, as well as securing, monitoring, and managing your Kafka infrastructure. Today, Confluent Platform is used for a wide array of use cases across numerous industries, from financial services, omnichannel retail, and autonomous cars to fraud detection, microservices, and IoT.

The following figure shows the components of Confluent Platform.

Overview of Confluent event streaming technology

At the core of Confluent Platform is Kafka, the most popular open source distributed streaming platform. The

key capabilities of Kafka include the following:

- Publish and subscribe to streams of records.
- Store streams of records in a fault tolerant way.
- Process streams of records.

Out of the box, Confluent Platform also includes Schema Registry, REST Proxy, a total of 100+ prebuilt Kafka connectors, and ksqlDB.

Overview of Confluent platform enterprise features

- **Confluent Control Center.** A UI-based system for managing and monitoring Kafka. It allows you to easily manage Kafka Connect and to create, edit, and manage connections to other systems.
- **Confluent for Kubernetes.** Confluent for Kubernetes is a Kubernetes operator. Kubernetes operators extend the orchestration capabilities of Kubernetes by providing the unique features and requirements for a specific platform application. For Confluent Platform, this includes greatly simplifying the deployment process of Kafka on Kubernetes and automating typical infrastructure lifecycle tasks.
- **Kafka Connect Connectors.** Connectors use the Kafka Connect API to connect Kafka to other systems such as databases, key-value stores, search indexes, and file systems. Confluent Hub has downloadable connectors for the most popular data sources and sinks, including fully tested and supported versions of these connectors with Confluent Platform. More details can be found here.
- **Self- balancing clusters.** Provides automated load balancing, failure detection and self-healing. It also provides support for adding or decommissioning brokers as needed, with no manual tuning.
- **Confluent cluster linking.** Directly connects clusters together and mirrors topics from one cluster to another over a link bridge. Cluster linking simplifies setup of multi-datacenter, multi-cluster, and hybrid cloud deployments.
- **Confluent auto data balancer.** Monitors your cluster for the number of brokers, the size of partitions, the number of partitions, and the number of leaders within the cluster. It allows you to shift data to create an even workload across your cluster, while throttling rebalance traffic to minimize the effect on production workloads while rebalancing.
- **Confluent replicator.** Makes it easier than ever to maintain multiple Kafka clusters in multiple data centers.
- **Tiered storage.** Provides options for storing large volumes of Kafka data using your favorite cloud provider, thereby reducing operational burden and cost. With tiered storage, you can keep data on cost-effective object storage and scale brokers only when you need more compute resources.
- **Confluent JMS client.** Confluent Platform includes a JMS-compatible client for Kafka. This Kafka client implements the JMS 1.1 standard API, using Kafka brokers as the backend. This is useful if you have legacy applications using JMS and you would like to replace the existing JMS message broker with Kafka.
- **Confluent MQTT proxy.** Provides a way to publish data directly to Kafka from MQTT devices and gateways without the need for a MQTT broker in the middle.
- **Confluent security plugins.** Confluent security plugins are used to add security capabilities to various Confluent Platform tools and products. Currently, there is a plugin available for the Confluent REST proxy that helps to authenticate the incoming requests and propagate the authenticated principal to requests to Kafka. This enables Confluent REST proxy clients to utilize the multitenant security features of the Kafka broker.

Next: Confluent Performance validation.

Confluent Performance validation

We have performed the verification with Confluent Platform for tiered storage on NetApp ONTAP. The NetApp and Confluent teams worked on this verification together and ran the test cases required for it.

Confluent setup

For the setup, we used three zookeepers, five brokers, and five testing servers with 256GB RAM and 16 CPUs. For NetApp storage, we used ONTAP with an AFF A900 HA pair. The storage and brokers were connected through 100GbE connections.

The following figure shows the network topology of configuration used for tiered storage verification.

The tools servers act as application clients that send or receive events to or from Confluent nodes.

Confluent tiered storage configuration

We used the following testing parameters:

```
confluent.tier.fetcher.num.threads=80
confluent.tier.archiver.num.threads=80
confluent.tier.enable=true
confluent.tier.feature=true
confluent.tier.backend=S3
confluent.tier.s3.bucket=kafkabucket1-1
confluent.tier.s3.region=us-east-1
confluent.tier.s3.cred.file.path=/data/kafka/.ssh/credentials
confluent.tier.s3.aws.endpoint.override=http://wle-mendocino-07-08/
confluent.tier.s3.force.path.style.access=true
bootstrap.server=192.168.150.172:9092,192.168.150.120:9092,192.168.150.164
:9092,192.168.150.198:9092,192.168.150.109:9092,192.168.150.165:9092,192.1
68.150.119:9092,192.168.150.133:9092
debug=true
jmx.port=7203
num.partitions=80
num.records=200000000
#object PUT size - 512MB and fetch 100MB - netapp
segment.bytes=536870912
max.partition.fetch.bytes=1048576000
#GET size is max.partition.fetch.bytes/num.partitions
length.key.value=2048
trogdor.agent.nodes=node0,node1,node2,node3,node4
trogdor.coordinator.hostname.port=192.168.150.155:8889
num.producers=20
num.head.consumers=20
num.tail.consumers=1
test.binary.task.max.heap.size=32G
test.binary.task.timeout.sec=3600
producer.timeout.sec=3600
consumer.timeout.sec=3600
```

For verification, we used ONTAP with the HTTP protocol, but HTTPS also worked. The access key and secret key are stored in the file name provided in the `confluent.tier.s3.cred.file.path` parameter.

NetApp storage controller – ONTAP

We configured a single HA pair configuration in ONTAP for verification.

Verification results

We completed the following five test cases for the verification. The first two were functionality tests and the remaining three were performance tests.

Object store correctness test

This test performs basic operations such as get, put, and delete on the object store used for the tiered storage using API calls.

Tiering functionality correctness test

This test checks the end-to-end functionality of the object storage. It creates a topic, produces an event stream to the newly created topic, waits for the brokers to archive the segments to the object storage, consumes the event stream, and validates the consumed stream matches with the produced stream. We have performed this test with and without an object-store fault injection. We simulated node failure by stopping the service manager service in one of the nodes in ONTAP and validating that the end-to-end functionality works with object storage.

Tier fetch benchmark

This test validated the read performance of the tiered object storage and checked the range fetch read requests under heavy load from segments generated by the benchmark. In this benchmark, Confluent developed custom clients to serve the tier fetch requests.

Produce-consume workload generator

This test indirectly generates write workload on the object store through the archival of segments. The read workload (segments read) was generated from object storage when consumer groups fetched the segments. This workload was generated by a TOCC script. This test checked the performance of read and write on the object storage in parallel threads. We tested with and without object store fault injection as we did for the tiering functionality correctness test.

Retention workload generator

This test checked the deletion performance of an object storage under a heavy topic- retention workload. The retention workload was generated using a TOCC script that produces many messages in parallel to a test topic. The test topic was configuring with an aggressive size-based and time-based retention setting that caused the event stream to be continuously purged from the object store. The segments were then archived. This led to many deletions in the object storage by the broker and collection of the performance of the object-store delete operations.

For verification details, see the Confluent website.

Performance tests with produce-consume workload generator

We performed tiered storage testing with either five or eight broker nodes during a produce-consume workload with the one AFF A900 HA pair NetApp storage controller. According to our tests, the time to completion and the performance results scaled with the number of broker nodes until AFF A900 resource utilization reached one hundred percent. The ONTAP storage controller setup required a minimum of one HA pair.

The performance for the S3 retrieve operation increased linearly based on number of Confluent broker nodes. ONTAP storage controller supports up to 12 HA pair in a single deployment.

The following graph shows combined S3 tiering traffic with five or eight broker nodes. We maximized the AFF A900 single HA pair performance.

S3 - Retrieve Performance Trend

The following graph shows the Kafka throughput at approximately 31.74GBps.

We also observed similar throughput in the ONTAP storage controller `perfstat` report.

```
object_store_server:wle-mendocino-07-08:get_data:34080805907b/ s
object_store_server:wle-mendocino-07-08:put_data:484236974b/ s
```

Performance best practice guidelines

- For ONTAP, when possible, use a GET size >=1MB.

- Increasing `num.network.threads` and `num.io.threads` in `server.properties` on broker nodes enables you to push increased tiering activity to S3 tier. These results are with `num.network.threads`

and `num.io.threads` set to 32.

- S3 buckets should target eight constituents per member aggregate.

- Ethernet links driving S3 traffic should use an MTU of 9k when possible on both storage and client.

Conclusion

This verification test reached 31.74GBps of tiering throughput on Confluent with NetApp ONTAP Storage Controller.

Where to find additional information

To learn more about the information that is described in this document, review the following documents and/or websites:

- What is Confluent?

  https://www.confluent.io/apache-kafka-vs-confluent/

- S3-sink parameter details

  https://docs.confluent.io/kafka-connect-s3-sink/current/configuration_options.html#s3-configuration-options

- Apache Kafka

  https://en.wikipedia.org/wiki/Apache_Kafka

- S3 in ONTAP best practices

  https://www.netapp.com/pdf.html?item=/media/17219-tr4814.pdf

- S3 Object storage Management

  https://docs.netapp.com/us-en/ontap/s3-config/s3-support-concept.html

- NetApp Product Documentation

  https://www.netapp.com/support-and-training/documentation/

Version history

Version	Date	Document version history
Version 1.0	September 2022	Initial release.