Cải Thiện Hệ Thống Stream Processing

# Khái niệm

Resource = CPU, memory, disk access.

Kafka

* Replication-factor: (default=1) số bản lưu trên kafka cluster. Repl-fact = số copy -1. Ví dụ rep-fact=2 tức là tạo thêm 1 copy trên 1 broker khác. Số broker phải >= rep\_factor để thỏa mản copy dữ liệu qua backup qua broker khác.

Rsyslog: v8.37, cần đọc hiểu các khái niệm action, module, queue

# Giới thiệu

**Vấn đề với hệ thống cũ**

Dữ liệu chênh lệch lớn giữa hệ thống stream và batch processing. Ví dụ, CCU cả ngày thống kê ngày 2018-09-04 là 9.9M, còn batch là 12.3M. -> Hệ thống streaming processing cũ gồm logcourrier, logstash, kafka có khuyết điểm deliver thiếu log tới spark streaming dẫn tới ảnh hưởng số liệu thống kê batch và streaming.

**Giả thuyết nguyên nhân**

Hệ thống cũ

1. không có document đầy đủ dẫn đến khó khăn trong việc debug nguyên nhân mất log nằm ở công đoạn nào.
2. Không có tools để monitor trợ giúp phần debug.

Logstash trong hệ thống cũ có khả năng là điểm thắt cổ chai do lúc cao điểm phải làm 4 chuyện: 1-nhận log từ logcourrier, 2-xử lý và parse log, 3-ghi log ra file trên disk, 4-theo dõi file để ghi log qua kafka.

**Hướng giải quyết**

Bỏ logstash. Dùng rsyslog để deliver log từ servers vì rsyslog có sẵn trên linux servers và là tool nhẹ ít tốn resource sẽ tốt cho server lúc phải chịu high load.

Setup + test hệ thống mới gồm rsyslog, kafka để deliver log tới spark streaming.

Quy trình log đi: rsyslog trên cùng server với log file theo dõi file log, nếu có dữ liệu mới sẽ bắn qua kafka topic định trước trên một server khác, spark streaming consumes dữ liệu từ kafka topic để xử lý cho ra data gần realtime.

# Logs

**Tóm tắt quá trình setup và tinh chỉnh hệ thống streaming mới gồm rsyslog và kafka.**

* Tóm tắt vấn đề và cách giải quyết ngắn gọn trong bảng dưới
* Chi tiết từng ngày được ghi bên dưới bảng

## Tóm tắt

|  |  |  |
| --- | --- | --- |
| Ngày | | Log |
| T6 08-31 | | [3h-7h] Rsyslog không gửi msg sau khi logrotate chạy lúc khoảng 3h50. Rsyslog process không bị tắt. tới khoảng 7h thì rsyslog gửi log lại. |
| CN 09-02 | | [17h] Kafka full disk nen shutdown. Thanh deleted msg roi restart lai kafka.  Đổi qua sử dụng rsyslog stable mới nhât v8.37 từ 8.\*\*  Bỏ các lệnh gửi signal –HUP (HANGUP) tới rsyslog trong postrorate script của logrotate. |
| T2 09-03 | Vấn đề rsyslog k gửi log sau lograte được giải quyết. | |
| T3 09-04 | | [20-21h] message rsyslog gửi qua kafka failed lúc load nginx lên cao.  Debug chi tiết ở dưới. |
| T4 09-05 | | Sửa: cho action queue vào mem, size 1M. cũ là trong disk hay direct mode k biết, size 100k. kafka thì giàm replication-factor của HTV xuống 1, k copy, 9 partitions vẫn phân đều ra 3 brokers, mỗi broker 3 partitions.  Mong đợi: topic HTV tối nay k gặp vấn đề failed khi submit msg.  Kết quả:   * 3 queues đầy: main q size 100k, action q size 1M, librdkafka q size 100k. * Toan bo data.failed cua action log-to-kafka giam xuong con 0 * Omkafka data.failures van xuat hien |
| T5 09-06 | | Kết luận:   * Rsyslog k có đủ CPU cycle để thi hành lệnh, nên có tăng q size lên thì q vẫn đầy. có 2 options là  1. tăng NICE + main q worker threads lên cho rsyslog để xử lý data đi nhanh hơn. 2. Tăng main queue size và giảm polling interval xuống để trữ dữ liệu trong main q của rsyslog.   Lưu lượng của cache50 lúc cao điểm: 2.8k-5k msg/s ~ 3MB-5MB/s với 1msg ~ 1kB trong khoảng 7h-9h.  Thực hiện:   * Tăng main queue lên 200k size ~ 2GB RAM để chứa msg để chạy polling mode mỗi 1 phút (5MB/s \* 60s \* 5 = 1.5GB). |
| T2  09-10 | | Rsyslog bị lỗi do logrotate trên cache55, 57 khi Thạnh add thêm config nén logfile vô trong conf của logrotate   * Bỏ 3 dòng xóa imfile, stop và start rsyslog trong logrotate config. Lý do: rsyslog thiết kế hỗ trợ logrotate, nếu xóa imfile và restart manually sẽ làm rsyslog khong track được logrotate dẫn đến việc dùng lại offset cũ -> ngưng bắn log tới lúc file log đầy như cũ. Quan trọng là phải restart rsyslog sau khi đổi config để rsyslog chạy config mới, sau đó mới logrotate để rsyslog biết mà tự restart monitoring trên file mới, sau đó log sẽ dc bắn lại. * Không nén dc file do process nào đó (có khả năng là nginx) vẫn ghi vào file log cũ, do đó logrotate sẽ k truy cập vào file đó được để nén file. Nếu nén theo kiểu delaycompress thì sẽ dc do sau 1 ngày file log đó chắc chắn k còn được ghi nữa -> chạy script riêng sau logrotate khoảng 30p để nén file thì sẽ giải quyết được vấn đề. * Thanh sua lai config cua logrotate cho tat ca cac con log. |
| T3  09-11 | | 1. Loi khong ban log o 4 con app20, 21 (drm) va cache 55, 56.   Phát hiện rsyslog không ổn định lúc handle logrotate (lý do tại sao thì chưa biết, có thế liên quan đến inotify của system và watch descriptor của linux).   * Sửa config của logrotate lại, phải stop và start rsyslog trong phần postrotate để đảm bảo sau khi rotate rsyslog thấy file mới và them watch cho file mới này. áp config này của logrotate cho tất cả servers.  1. Anh Thuận kêu Thạnh thử test polling mode trên con HTV sau khi nghe giải thích là inotify mode dùng thêm notify của linux. |
| T5 9-13 | | Kafka full đĩa. Tính dung lượng cần thiết cho ổ đĩa của kafka cho các topic khác nhau cần chú ý đến khoảng thời gian có nhiều log nhất (12PM-12AM) chứ không phải lấy dung lượng cả ngày chia 2. Nếu lấy dung lượng cả ngày chia ra thì sẽ ra dung lượng trung bình thấp hơn dung lượng log lúc cao điểm nhất (buổi tối).  1 lý do full nữa là topic ts\_playtv\_other không dc chia ra làm 3 mà chỉ tập trung vào 2 con 01 và 02 nên làm cho 2 con đó đầy -> dẫn đến việc sai lệch so với kết quả tính toán dung lượng.   * Giảm dung lượng lưu trữ xuống 6 tiếng và restart lại kafka để load được chia ra đều cho 3 con. |
| T2 9-17 | | Enable them config reopenonTruncate() |
| T6 9-21 | | Lag ts-other lên tới 12M. => bỏ df.coalesce() trong spark streaming consumer app để không gom data về master trước khi ghi ra file parquet giúp giảm lag max xuống còn 1M |
| T3 9-25 | | -App21 bình thường, app20 rsyslog ngưng bắn sau 4h20. -> do app20 full disk.  -Tăng số partition trong topic st lên 18 -> quan sát thấy buổi tối tốc độ xử lý lên 1.6-1.8 lần => chứng tỏ tăng số partitions để tăng số thread => tăng khả năng xử lý song song của Spark Streaming. Lý do là Spark Streaming chia partitions theo partitions của kafka mặc định. Kafka có 9 partitions thì Spark Streaming JVM sẽ mở 9 thread để xử lý song song 9 luồng này. |

## T2 Sep-03

Tested: rsyslog imfile module hỗ trợ logrotate.

Đổi qua sử dụng rsyslog stable mới nhât v8.37 từ 8.\*\* (k biết chắc, tùy máy, có thể máy cũ có bản v7) trong các con cache servers. Các module cũ có thể có lỗi với đường dẫn file log có \* (wildcard). Đổi config của imfile module sang path không có \* cho chắc ăn.

Bỏ các lệnh gửi signal –HUP (HANGUP) tới rsyslog trong postrorate script của logrotate. Lệnh này có khả năng làm rsyslog không nhận ra việc logrotate tạo ra file mới dẫn tới việc track theo số dòng đã gửi cũ trong file log mới -> file log tới 7h số lượng log mới tăng lên như số rsyslog đang track nên rsyslog gửi tiếp.

## T3 Sep-04

[20h20] Thạnh chỉnh rsyslog omkafka parameters **closeTimeout** từ 10s lên 1 phút, **resubmitOnFailure** của rsyslog trên cache50 HTV từ off qua on.

## T4 Sep-05

Xác đinh nguyên nhân lỗi gửi msg hôm trước:

1. Xem impstats của om-kafka trên kibana -> action queue của om-kafka bị đầy ở 100,000. Sau đó msg submitted = 0 một khoảng thời gian. -> 2 khả năng: kafka xử lý không kịp, hàng đợi không đủ nhiều.
2. Xem kibana, Impstats gửi rất nhiều stats ngoài 2 thứ quan trọng cần thiết [log-to-kafka, om-kafka]: global, omelasticsearch, parse\_rsyslog\_stats, push\_rsyslog\_stats, resource\_usage, main Q.

Có 2 cách để giải quyết giả thuyết nguyên nhân là kafka không replicate kịp lượng lớn msg nên gửi response failed cho rsyslog.

1. Giảm replication cho topic HTV xuống 1 => test giải pháp này.
2. Chuyển replication sang async thay vì sync = chờ replication xong rồi mới acknowledge cho client

Thay đổi:

1. Chỉnh topic HTC **replication-factor** của kafka xuống 0 để giảm thời gian kafka gửi request lại được trước khi timeout của rsyslog omkafka producer. Giữ **replication-factor=2** cho các topic còn lại.
2. Thay đổi action queue thành In-Memory Queue, type Fixed Array, size=1,000,000 cho rsyslog trên cache50 HTV từ k chỉnh mode, size=100,000. -> mem dành cho rsyslog action queue khoảng 1GB=1K/msg\*1,000,000msg

**Triệu chứng**

Lúc cao điểm rsyslog không bắn được qua kafka. Message nhận bị failed. Msg in per sec (trung binh 1 phut) va msg comsumed by spark bị răng cưa.

Sep 04 20:50:26 cache50.prod.hcm.fplay rsyslogd[60483]: omkafka: kafka error message: -185,'Local: Timed out','118.69.190.39:9092/3: 12 request(s) timed out: disconnect' [v8.37.0 try http://www.rsyslog.com/e/2422 ]

Sep 04 20:50:33 cache50.prod.hcm.fplay rsyslogd[60483]: action 'log-to-kafka' suspended (module 'omkafka'), retry 0. There should be messages before this one giving the reason for suspension. [v8.37.0 try http://www.rsyslog.com/e/2007 ]

**Rsyslog config**

1. Omkafka
   * requestTimeout=10s. sau khi msg gửi cho topic HTV bị failed thì mới sửa lại 60s.
   * reSubmitOnFailure=off, sau khi failed thì sửa lại on

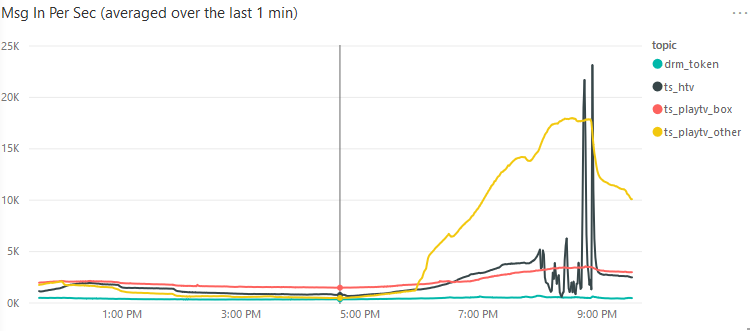
**Kafka config**

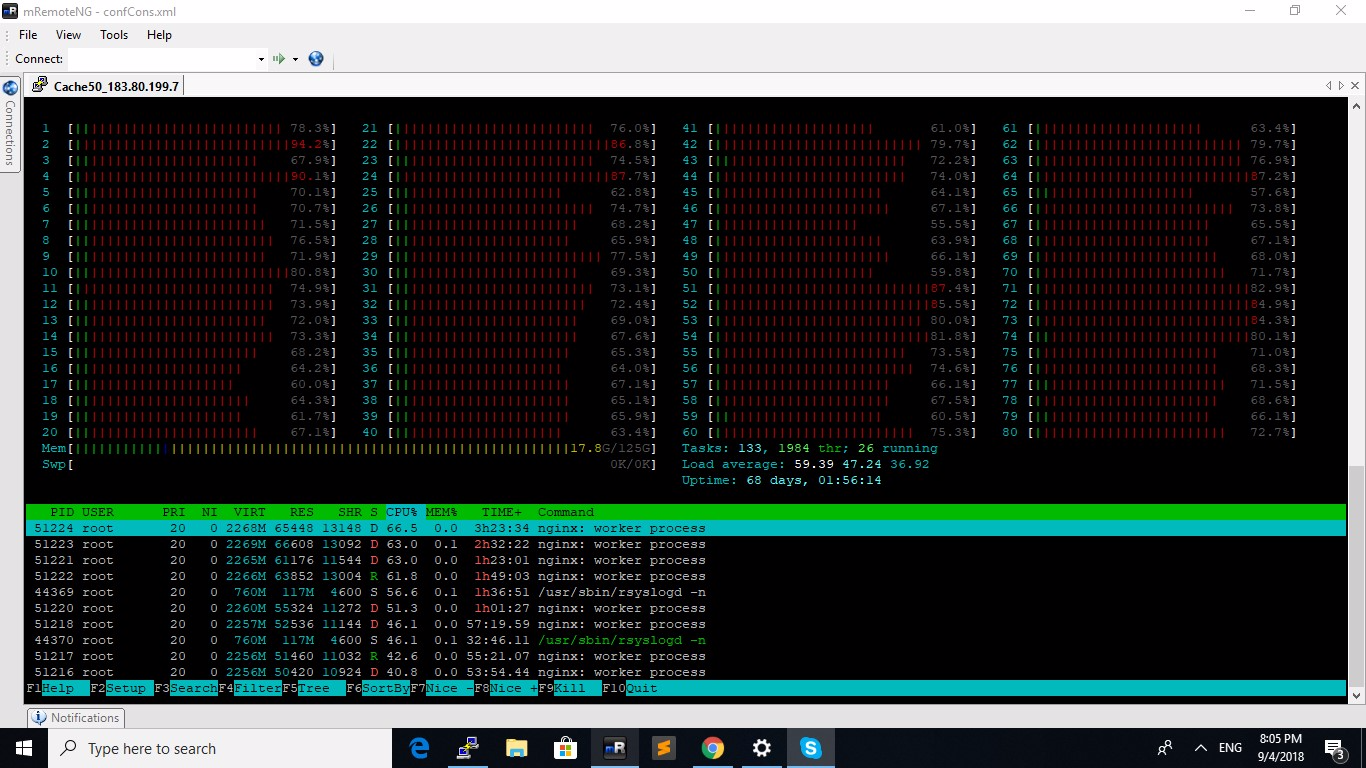
#request.timeout.ms=60000 trong **producer.config (config/producer.properties)**

* config nay khong lien quan cho rsyslog do do la config cho producer neu start consumer bang bash script.

num.replica.fetchers=2

replica.fetch.max.bytes=2048576 # server.properties





Lúc cao điểm nginx chiếm rất nhiều resource.

**Kibana**



**Chẩn đoán nguyên nhân**

có khả năng chính do 1 trong 2 hoặc cả 2 nguyên nhân

1. Rsyslog không có resource để bắn log qua kafka. Lúc có resource thì record quá nhiều do đó bắn một lần quá nhiều qua kafka. Kafka không handle nổi dẫn đến timeout nên failed.
2. Kafka topic HTV đang nhận quá nhiều nên lúc đó không có thời gian để response lại cho rsyslog. Timeout cho response của kafka là 60s, có khả năng kafka nhận quá nhiều msg từ HTV nên không replicated kịp. Do timeout khi đang replicate quá nhiều msg nên không thành công, do đó kafka gửi lại response failed cho rsyslog.

**Thay đổi chi tiết**

[root@cache53 kafka\_2.11-1.1.1]# /opt/kafka\_2.11-1.1.1//bin/kafka-reassign-partitions.sh --zookeeper 183.80.199.4:2181,183.80.199.5:2181,118.69.190.39:2181 --reassignment-json-file increase-replication-factor.json --execute

Current partition replica assignment

{"version":1,"partitions":[{"topic":"ts\_htv","partition":5,"replicas":[2,1],"log\_dirs":["any","any"]},{"topic":"ts\_htv","partition":1,"replicas":[1,2],"log\_dirs":["any","any"]},{"topic":"ts\_htv","partition":7,"replicas":[1,2],"log\_dirs":["any","any"]},{"topic":"ts\_htv","partition":2,"replicas":[2,3],"log\_dirs":["any","any"]},{"topic":"ts\_htv","partition":4,"replicas":[1,3],"log\_dirs":["any","any"]},{"topic":"ts\_htv","partition":8,"replicas":[2,3],"log\_dirs":["any","any"]},{"topic":"ts\_htv","partition":0,"replicas":[3,1],"log\_dirs":["any","any"]},{"topic":"ts\_htv","partition":3,"replicas":[3,2],"log\_dirs":["any","any"]},{"topic":"ts\_htv","partition":6,"replicas":[3,1],"log\_dirs":["any","any"]}]}

Đổi replication-factor của topic HTV về 1

Topic:ts\_htv PartitionCount:9 ReplicationFactor:1 Configs:

Topic: ts\_htv Partition: 0 Leader: 1 Replicas: 1 Isr: 1

Topic: ts\_htv Partition: 1 Leader: 2 Replicas: 2 Isr: 2

Topic: ts\_htv Partition: 2 Leader: 3 Replicas: 3 Isr: 3

Topic: ts\_htv Partition: 3 Leader: 1 Replicas: 1 Isr: 1

Topic: ts\_htv Partition: 4 Leader: 2 Replicas: 2 Isr: 2

Topic: ts\_htv Partition: 5 Leader: 3 Replicas: 3 Isr: 3

Topic: ts\_htv Partition: 6 Leader: 1 Replicas: 1 Isr: 1

Topic: ts\_htv Partition: 7 Leader: 2 Replicas: 2 Isr: 2

Topic: ts\_htv Partition: 8 Leader: 3 Replicas: 3 Isr: 3

* + Broker 1 chứa: partition 0, 3, 6
  + Broker 2 chứa: 1, 4, 7
  + Broker 3 chứa: 2, 5, 8

**Rsyslog**

module(load="imfile" PollingInterval="10")

module(load="omelasticsearch")

module(load="omkafka")

input(type="imfile"

#File="/var/log/nginx/cdn/fplay.live.ts.json.log"

File="/var/log/nginx/cdn/htv.live.ts.json.log"

#Tag="ts\_playtv\_other"

Tag="ts\_htv"

freshStartTail="on"

)

template(name="forwardFormat" type="string"

string="%msg%"

)

action(

name="log-to-kafka"

broker=["183.80.199.4:9092","183.80.199.5:9092","118.69.190.39:9092"]

type="omkafka"

#topic="ts\_playtv\_other"

topic="ts\_htv"

Partitions.Auto="on"

resubmitOnFailure="on"

KeepFailedMessages="on"

failedMsgFile="/etc/rsyslog.d/file-failed/file.log"

template="forwardFormat"

closeTimeout="60000"

queue.size="1000000"

queue.type="FixedArray"

action.resumeRetryCount="-1"

action.resumeInterval="60"

action.reportSuspension="on"

action.reportSuspensionContinuation="on"

errorFile="/var/log/rsyslog-kafka-error.json"

)

## T2 Sep-10

**Config logrotate của tất cả server luc bi loi (config gay ra loi to mau do)**

/var/log/cron

/var/log/maillog

/var/log/messages

/var/log/secure

/var/log/spooler

/var/log/nginx/\*.log

/var/log/nginx/cdn/\*.log

/var/log/nginx/json\_access/\*.log

/var/log/nginx/playstats/\*.log {

daily

missingok

rotate 15

compress

compresscmd /bin/lz4

compressext .lz4

notifempty

create 640 nginx adm

sharedscripts

postrotate

[ -f /var/run/nginx.pid ] && kill -USR1 `cat /var/run/nginx.pid`

rm -rf /var/lib/rsyslog/imfile\*

systemctl stop rsyslog.service

systemctl start rsyslog.service # 3 dong nay gay ra van de

rm -rf /var/log/nginx/\*.log. rm -rf /var/log/nginx/\*.log- && rm -rf /var/log/nginx/cdn/\*.log- && rm -rf /var/log/nginx/json\_access/\*.log-

endscript

**Config rsyslog sua lai de k bi loi nua (xoa may dong ben duoi di), bỏ compress ra khỏi config của logrotate do lúc đó có process nào khác đang giữ file này nên k nén được.**

/var/log/cron

/var/log/maillog

/var/log/messages

/var/log/secure

/var/log/spooler

/var/log/nginx/\*.log

/var/log/nginx/cdn/\*.log

/var/log/nginx/json\_access/\*.log

/var/log/nginx/playstats/\*.log {

daily

missingok

rotate 15

~~compress~~

~~compresscmd /bin/lz4~~

~~compressext .lz4~~

notifempty

create 640 nginx adm

sharedscripts

postrotate

[ -f /var/run/nginx.pid ] && kill -USR1 `cat /var/run/nginx.pid`

~~rm -rf /var/lib/rsyslog/imfile\*~~

~~systemctl stop rsyslog.service~~

~~systemctl start rsyslog.service~~

rm -rf /var/log/nginx/\*.log. rm -rf /var/log/nginx/\*.log- && rm -rf /var/log/nginx/cdn/\*.log- && rm -rf /var/log/nginx/json\_access/\*.log-

endscript

## T3 Sep-11

**Config đã sửa của logrotate (thêm 2 dòng màu đỏ).** Mục đích của việc sửa này là đảm bảo rsyslog check lại file log cần bắn sau khi rotation (lúc này đã có file mới với inode mới) mà monitor file này. Rsyslog không ổn định ở bước này.

/var/log/nginx/\*.log

/var/log/nginx/cdn/\*.log

/var/log/nginx/json\_access/\*.log

/var/log/nginx/playstats/\*.log

/var/log/cron

/var/log/maillog

/var/log/messages

/var/log/secure

/var/log/spooler

{

daily

missingok

rotate 15

notifempty

create 640 nginx adm

sharedscripts

postrotate

[ -f /var/run/nginx.pid ] && kill -USR1 `cat /var/run/nginx.pid`

rm -rf /var/log/nginx/\*.log- && rm -rf /var/log/nginx/json\_access/\*.log-

systemctl stop rsyslog.service

systemctl start rsyslog.service

endscript

}

**Rsyslog systemctl status log cua cache49 chay binh thuong =>** watch descriptor 6 khong tim thay giup cho syslog nhan ra file moi va ban binh thuong

Sep 10 10:48:18 cache49.prod.hcm.fplay systemd[1]: Started System Logging Service.

Sep 11 03:21:01 cache49.prod.hcm.fplay rsyslogd[424250]: imfile: internal error? inotify provided watch descriptor 6 which we could not find in our tables - ignored [v8.37.0 try http://www.rsyslog.com/e/2175 ]

Sep 11 09:31:13 cache49.prod.hcm.fplay systemd[1]: Stopping System Logging Service...

## T4 Sep-12

Loi rsyslog msg lon vuot MAXMSGSIZE

Sep 12 05:46:27 cache06.prod.hcm.fplay rsyslogd[31172]: message too long (8424) with configured size 8096, begin of message is: { "receive\_time": "2018-09-12T05:46:27+07:00", "services": "playstats", "source\_ [v8.37....og.com/e/2445 ]

Sep 12 08:49:59 cache15.prod.hcm.fplay rsyslogd[49986]: message too long (22533) with configured size 8096, begin of message is: { "receive\_time": "2018-09-12T08:49:59+07:00", "services": "playstats", "source\_ [v8.37...og.com/e/2445 ]

Sep 12 10:55:45 cache15.prod.hcm.fplay rsyslogd[49986]: message too long (22526) with configured size 8096, begin of message is: { "receive\_time": "2018-09-12T10:55:45+07:00", "services": "playstats"

## T5 Sep-13

Kafka full đĩa

**So lieu luc kafka full disk ngày 2018-09-13**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | kafka1 | kafka2 | kafka3 | sum/topic |
| **st\_fplay** | **45.6** | **45.6** | **33.4** | **129.6667** |
| **ts\_htv** | **25.8** | **26** | **14.5** | **69.16667** |
| **ts\_playtv\_other** | 231 | **231** |  | 488 |
| **ts\_playtv\_box** | 71.6 | **71.6** |  | 151 |
| **drm\_token** | 20.4 | **20.4** |  | 43 |
| **vod** | **10.9** | **11** | **13.5** | **36.61111** |
| Available | 388 | 388 | 458 |  |
| sum/host | 405.3 | 405.6 | 61.4 |  |

## CN Sep-16

Zookeeper gặp vấn đề, Thạnh thay đổi những thông số sau

|  |  |  |
| --- | --- | --- |
| Config | Cũ | Mới |
| initLimit | 20 | 40 |
| syncLimit | 10 | 20 |
| tickTime |  | 6000 |

Log lỗi

[2018-09-15 02:44:14,476] ERROR Unexpected exception causing shutdown while sock still open (org.apache.zookeeper.server.quorum.LearnerHandler)

java.io.EOFException

at java.io.DataInputStream.readInt(DataInputStream.java:392)

at org.apache.jute.BinaryInputArchive.readInt(BinaryInputArchive.java:63)

at org.apache.zookeeper.server.quorum.QuorumPacket.deserialize(QuorumPacket.java:83)

at org.apache.jute.BinaryInputArchive.readRecord(BinaryInputArchive.java:99)

at org.apache.zookeeper.server.quorum.LearnerHandler.run(LearnerHandler.java:559)

[2018-09-15 02:44:14,543] WARN \*\*\*\*\* GOODBYE /183.80.199.4:58878 \*\*\*\*\*\* (org.apache.zookeeper.server.quorum.LearnerHandler)

[2018-09-15 02:44:14,543] INFO Got user-level KeeperException when processing sessionid:0x165d0b8f90f000a type:setData cxid:0xab zxid:0x10000072a txntype:-1 reqpath:

/brokers/topics/drm\_token/partitions/1/state Error Path:null Error:KeeperErrorCode = Session expired (org.apache.zookeeper.server.PrepRequestProcessor)

[2018-09-15 02:44:14,551] INFO Notification: 1 (message format version), 1 (n.leader), 0x100000721 (n.zxid), 0x2 (n.round), LOOKING (n.state), 1 (n.sid), 0x1 (n.peer

Epoch) LEADING (my state) (org.apache.zookeeper.server.quorum.FastLeaderElection)

[2018-09-15 02:44:15,000] INFO Expiring session 0x365d0b8f921000c, timeout of 6000ms exceeded (org.apache.zookeeper.server.ZooKeeperServer)

[2018-09-15 02:44:15,000] INFO Processed session termination for sessionid: 0x365d0b8f921000c (org.apache.zookeeper.server.PrepRequestProcessor)

[2018-09-15 02:44:16,207] INFO Accepted socket connection from /118.69.190.39:41976 (org.apache.zookeeper.server.NIOServerCnxnFactory)

[2018-09-15 02:44:16,207] INFO Client attempting to establish new session at /118.69.190.39:41976 (org.apache.zookeeper.server.ZooKeeperServer)

[2018-09-15 02:44:17,446] INFO Shutting down (org.apache.zookeeper.server.quorum.Leader)

[2018-09-15 02:44:17,446] INFO Shutdown called (org.apache.zookeeper.server.quorum.Leader)

java.lang.Exception: shutdown Leader! reason: Not sufficient followers synced, only synced with sids: [ 3 ]

at org.apache.zookeeper.server.quorum.Leader.shutdown(Leader.java:517)

at org.apache.zookeeper.server.quorum.Leader.lead(Leader.java:492)

at org.apache.zookeeper.server.quorum.QuorumPeer.run(QuorumPeer.java:950)

## T2 Sep-17

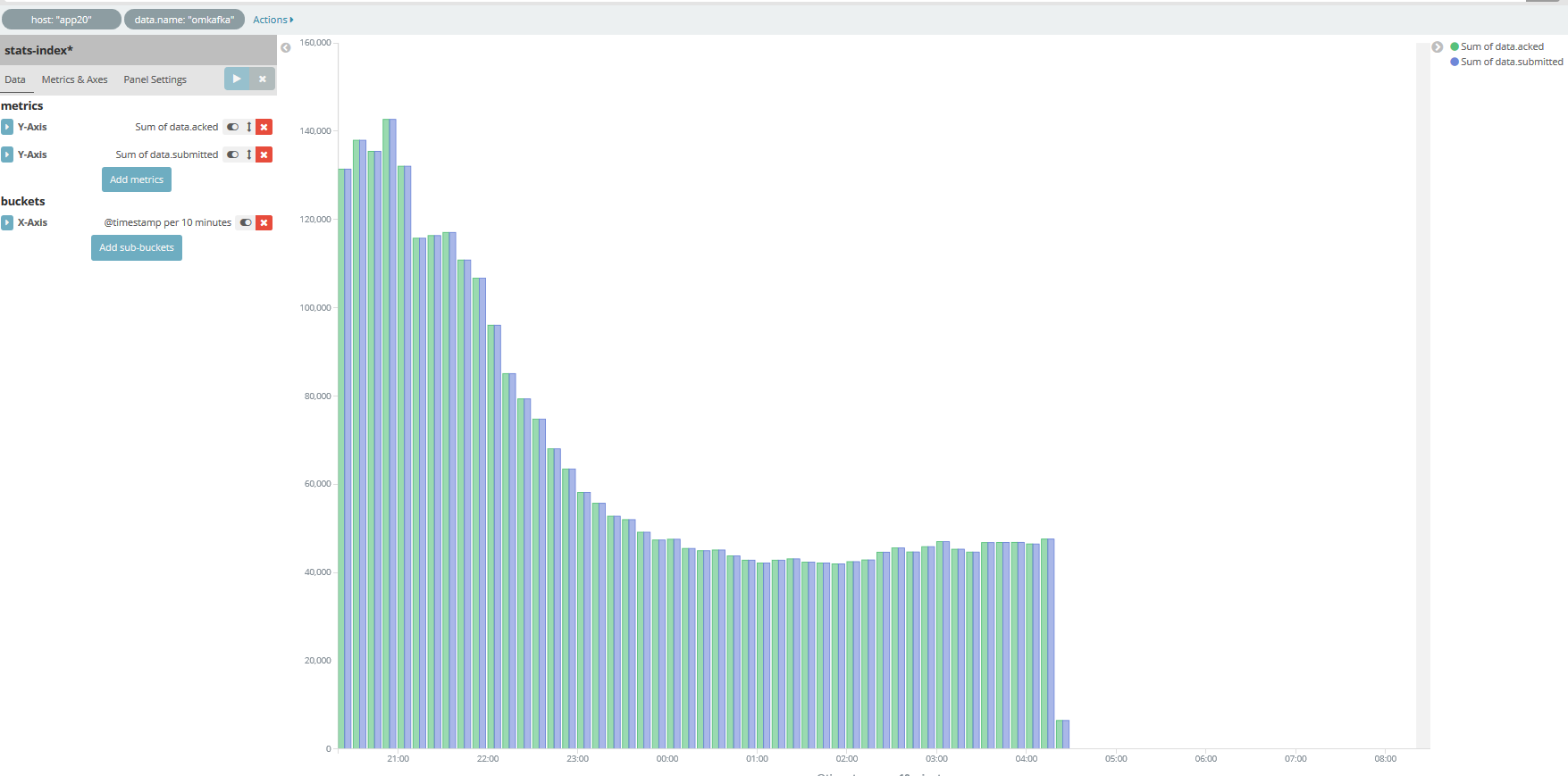
Fix: rsyslog imfile fails to detect log rotation

Add option: reopenOnTruncate

https://github.com/rsyslog/rsyslog/issues/2659

## T2 Sep-25

Hinh the hien rsyslog – omkafka output cua app20 khong ban sau 4h20.



# Vấn đề hay gặp

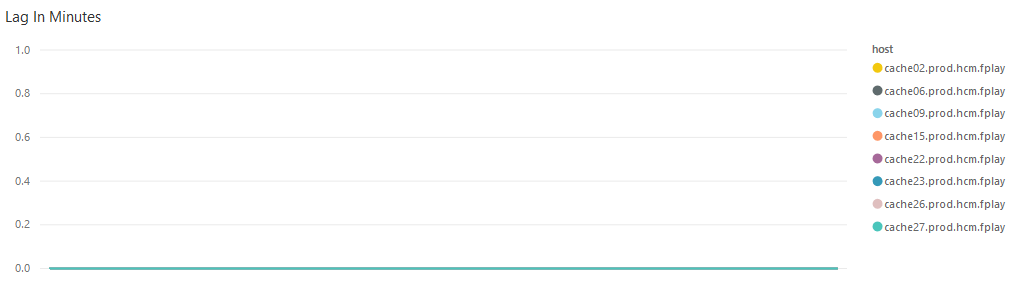
## Rsyslog

1. Rsyslog không chạy như mong muốn (ví dụ không bắn log lại qua kafka) sau khi đổi config để sửa lỗi gì đó. =>Cần gửi –HUP signal hoặc restart rsyslog process để config mới được load và áp dụng.

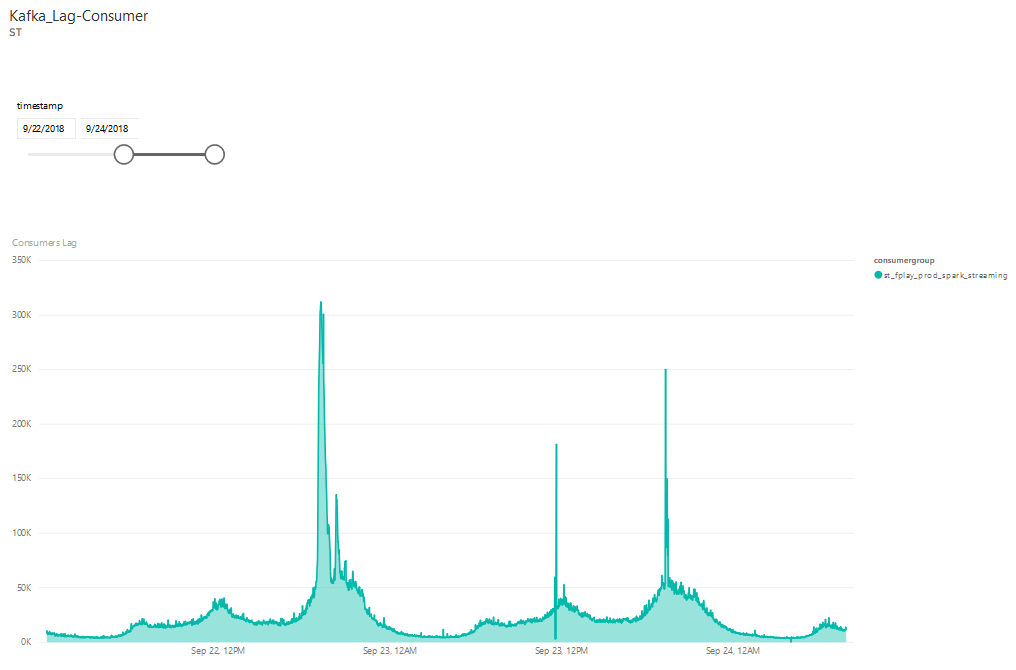
## Kafka

# Monitoring

1. Monitor độ lag của log st trên Kafka.
   * Viết 1 consumer riêng để lấy message mới nhất trong 1 Kafka topic mỗi khoảng thời gian định trước (ví dụ 1 phút). Parsed ra phần receive\_time của các log st, so sánh với thời gian hiện tại để biết được độ trễ của log trên kafka.



1. Monitor lag của Spark Streaming consumers.
   * Lag của consumer = offset mới nhất trên kafka – offset hiện tại của consumer.
   * Lag thể hiện độ trễ của consumer group so với message hiện tại trên kafka.
   * Ví dụ lag là 100,000 và offset mới nhất trên kafka là 1 triệu thì consumer đã xử lý được tới message 900,000



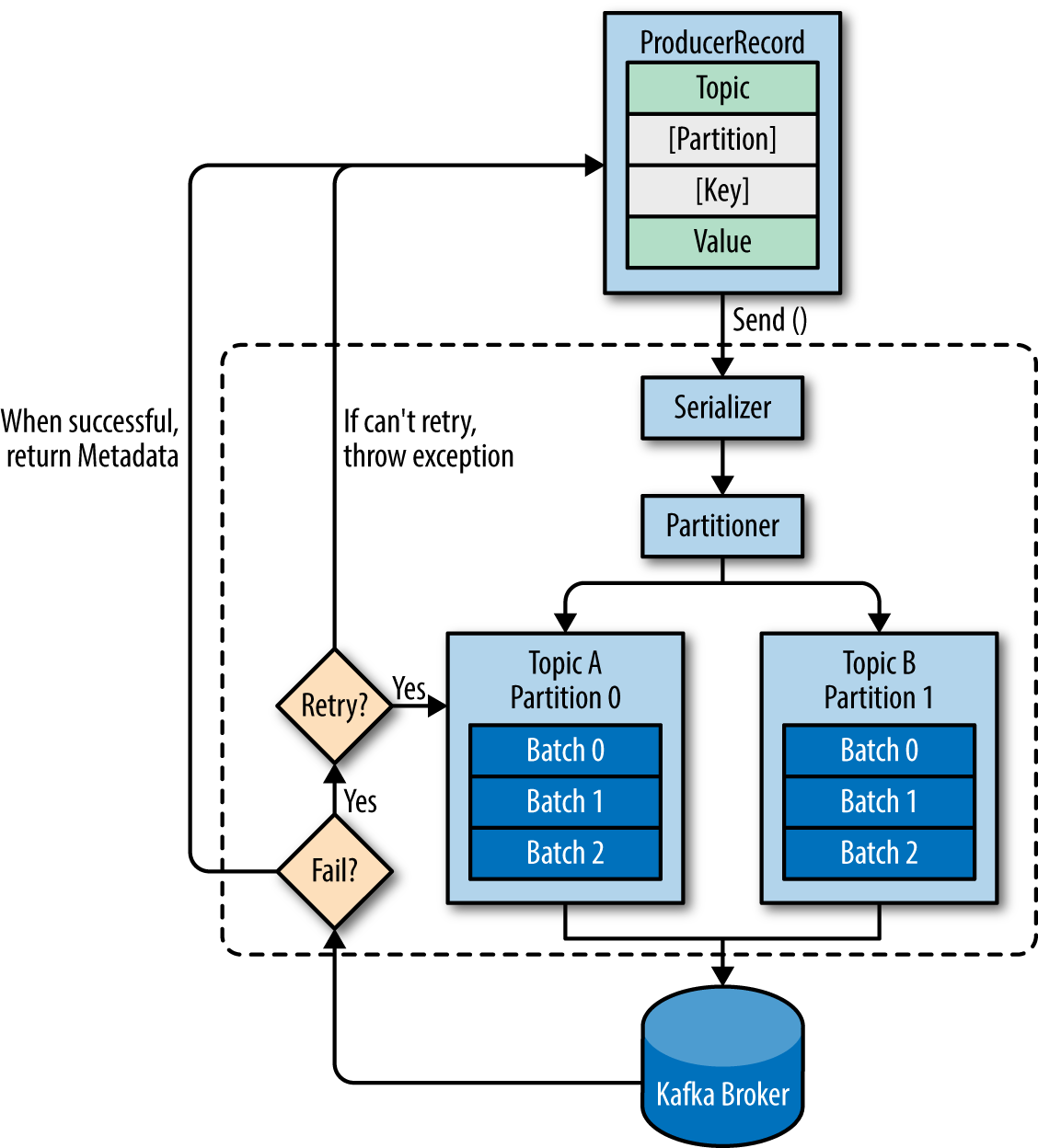
# Kafka

## Workflow

### Pub-sub

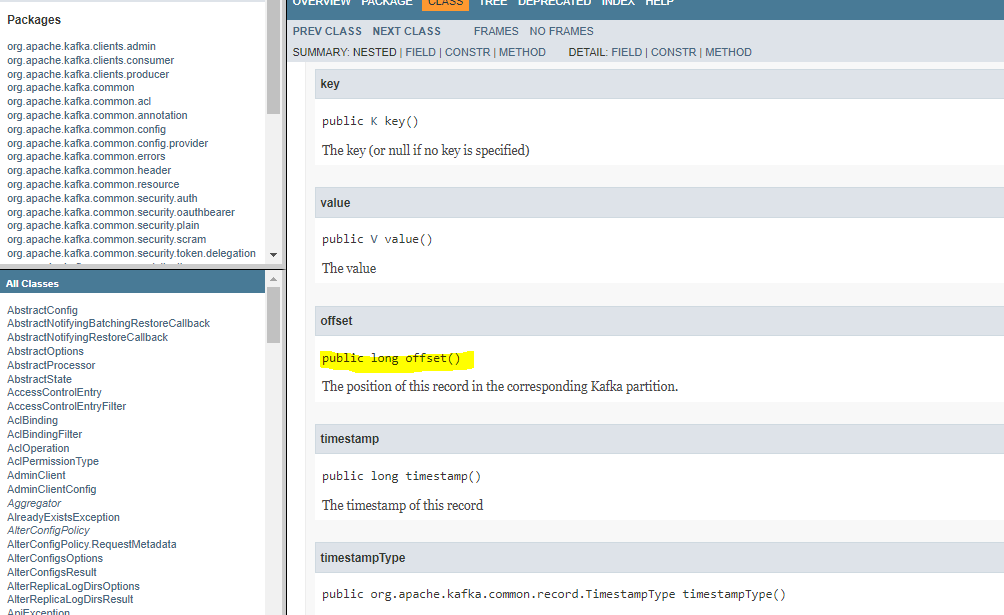
1. Consumer subscribe to specific topic
2. Once the consumer subscribes to a topic, Kafka will provide the current offset of the topic to the consumer and also saves the offset in the Zookeeper ensemble. (interval or only at the beginning?)
3. Consumer will request the Kafka in a regular interval (like 100 ms) for new messages.
4. Once Kafka receives the messages from producers, it forwards these messages to the consumers.
5. Consumer will receive the message and process it.
6. Once the messages are processed, consumer will send an acknowledgement to the Kafka broker. (Spark Direct Stream does not send this)
7. Once Kafka receives an acknowledgement, it changes the offset to the new value and updates it in the Zookeeper. Since offsets are maintained in the Zookeeper, the consumer can read next message correctly even during server outrages.
8. This above flow will repeat until the consumer stops the request.
9. Consumer has the option to rewind/skip to the desired offset of a topic at any time and read all the subsequent messages.

## Producing



## Offset

* Offset được tính trên mỗi partition của topic, offset là type long – java, 64 bit => giá trị tối đa là khoảng 18 tỉ tỉ. ([tham khảo doc kafka](https://kafka.apache.org/20/javadoc/index.html?org/apache/kafka/clients/consumer/KafkaConsumer.html) v2.0.0 – tìm offset trong page)
* Kafka thiết kế offset sẽ được tăng lên mãi mãi nếu topic và partition được giữ nguyên.



# Rsyslog

# Tham khao

1. [Apache Kafka best practices](https://medium.com/real-time-streaming/apache-kafka-best-practices-d9fac5c483c0) – medium

Các loại log:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Loai log | 1 log entry size | Servers | Host |
|  | St – playstats log, proxy log | 1425 chars ~ 1.4KB | Cache 02, 06, 09, 15, 22, 23, 26, 27 | St.fptplay.net |
|  | Ts\_playtv\_other | 718 char |  |  |
|  | Ts\_playtv\_box | 1045 char ~ 1KB | Cache 10, cache32 | Box-livecdn.fptplay.net |
|  | Drm\_token | 940-1074 ~ 1KB |  |  |
|  | Playapi | 1170 char ~ 1.14 KB | App 08, 09 | Api.fptplay.net  Hostname: app08.prod.hcm.fplay |