Vernemq cluster HA improvement test

1   Test Framework

The original test framework was first adapted to facilitate long-running test sessions comprised of successive
testruns and automatic reporting.

Test session results are captured in a table .

Each session contains a number of successive testruns

In each testrun we perform the following steps :

• check/wait until cluster is healthy

• trigger tx client : txm starts publishing a keyspace of 1000 messages

• kill the node where rx client is connected to

• wait until tx_eos : all messages delivered to broker (End Of Sequence)

• wait until cluster is healthy again

• get message analysis results from rx client (stowaways, admitted, duplicates, missing) and log results as row in the table.

• start next testrun ...

Notes :

• each successive keyspace sequence of 1000 messages gets a "sequence id"

• the rx client expects only messages with the correct sequence id corresponding to each testrun

• the rx client flags messages with a wrong sequence_id as "stowaways"

• stowaways are not considered when checking for missing/duplicate rx messages

2   3-node VerneMQ 1.11.0

2.1 Test session results

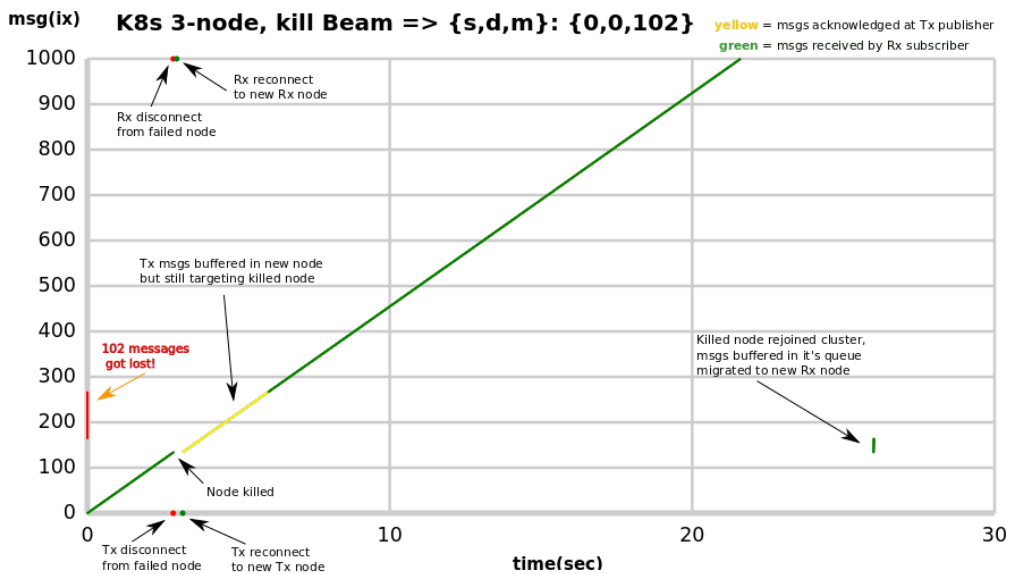### VerneMQ HA Test on K8s 3-node cluster, Failure = killing active Beam

- Runtime : tx_trigger to cluster healed
- Txtime : tx_trigger to tx_eos
- Gaptime : kill to tx_eos
- Downtime : kill to cluster healed

Note that only "admitted" messages are analysed for "duplicates" and "missing" message indexes
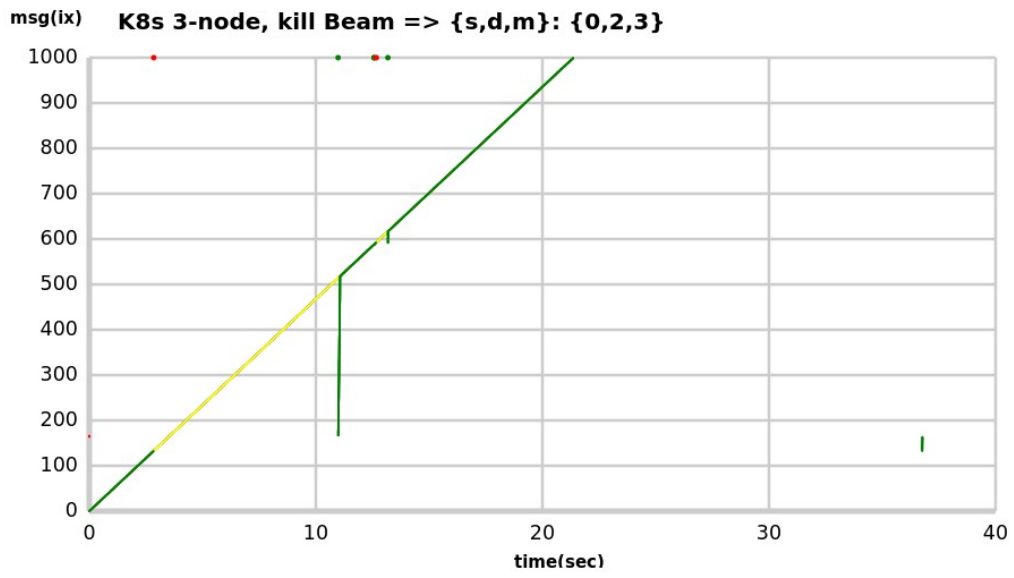
Run	Seq_ix	Killed node	Stowaways	Admitted	Duplicates	Missing	Runtime	Txtime	Gaptime	Downtime	tx_downtime	rx_downtime	Rx error	Txm	Rxm
1	1	myvernemq-2	0	1000	0	0	106	21	19	104	NA	0	nil	myvernemq-1	myvernemq-2
2	2	myvernemq-1	0	1000	0	0	127	22	19	124	0	0	nil	myvernemq-1	myvernemq-1
3	3	myvernemq-2	0	999	0	1	116	21	19	114	NA	3	nil	myvernemq-0	myvernemq-2
4	4	myvernemq-1	0	1000	0	0	126	21	19	124	NA	0	nil	myvernemq-0	myvernemq-1
5	5	myvernemq-2	0	1000	1	1	136	21	19	134	NA	7	nil	myvernemq-0	myvernemq-2
6	6	myvernemq-0	0	915	0	85	97	22	20	95	0	1	nil	myvernemq-0	myvernemq-0
7	7	myvernemq-2	0	999	0	1	147	22	19	144	NA	0	nil	myvernemq-1	myvernemq-2
8	8	myvernemq-0	0	999	2	3	116	21	19	114	NA	10	nil	myvernemq-1	myvernemq-0
9	9	myvernemq-1	0	1000	0	0	98	28	26	96	10	0	nil	myvernemq-1	myvernemq-1
10	10	myvernemq-0	0	998	0	2	137	27	23	133	NA	0	nil	myvernemq-2	myvernemq-0
11	11	myvernemq-2	0	924	0	76	141	21	19	139	0	0	nil	myvernemq-2	myvernemq-2
12	12	myvernemq-1	0	999	0	1	126	21	19	124	NA	0	nil	myvernemq-0	myvernemq-1
13	13	myvernemq-0	0	1000	0	0	145	25	23	143	4	2	nil	myvernemq-0	myvernemq-0
14	14	myvernemq-1	0	1000	0	0	131	21	19	129	NA	1	nil	myvernemq-2	myvernemq-1
15	15	myvernemq-0	0	994	0	6	186	21	19	184	NA	0	nil	myvernemq-2	myvernemq-0
16	16	myvernemq-1	0	1000	0	0	146	21	19	144	NA	5	nil	myvernemq-2	myvernemq-1
17	17	myvernemq-0	0	994	0	6	256	21	19	254	NA	0	nil	myvernemq-2	myvernemq-0
18	18	myvernemq-1	0	1001	1	0	186	21	19	184	NA	88	nil	myvernemq-2	myvernemq-1
19	19	myvernemq-2	0	898	0	102	97	22	19	94	0	0	nil	myvernemq-2	myvernemq-2
20	20	myvernemq-1	0	1000	0	0	271	21	19	269	NA	0	nil	myvernemq-0	myvernemq-1
21	21	myvernemq-0	0	1000	0	0	97	22	19	94	0	0	nil	myvernemq-0	myvernemq-0

2.2 Example testrun captures

This is a capture of testrun 19 where both Tx and Rx were connected to myvernemq-2 :

This is a capture of testrun 8 :

K8s 3-node, kill Beam => {s,d,m}: {0,2,3}

3 3-node VerneMQ with "Cluster HA improvements"

3.1 Test session results

VerneMQ HA Test on K8s 3-node cluster, Failure = killing active Beam

- Runtime : tx_trigger to cluster healed
- Txtime : tx_trigger to tx_eos
- Gaptime : kill to tx_eos
- Downtime : kill to cluster healed

Note that only "admitted" messages are analysed for "duplicates" and "missing" message indexes

Run	Seq_ix	Killed node	Stowaways	Admitted	Duplicates	Missing	Runtime	Txtime	Gaptime	Downtime	tx_downtime	rx_downtime	Rx error	Txm	Rxm
1	1	myvernemq-2	0	1000	0	0	102	22	19	99	NA	0	nil	myvernemq-1	myvernemq-2
2	2	myvernemq-0	0	1000	0	0	102	22	20	100	NA	0	nil	myvernemq-1	myvernemq-0
3	3	myvernemq-2	0	1000	0	0	122	22	19	119	NA	0	nil	myvernemq-1	myvernemq-2
4	4	myvernemq-1	0	1000	0	0	102	22	19	99	0	0	nil	myvernemq-1	myvernemq-1
5	5	myvernemq-2	0	1000	0	0	127	22	19	124	NA	0	nil	myvernemq-0	myvernemq-2
6	6	myvernemq-1	0	1000	0	0	117	22	19	114	NA	0	nil	myvernemq-0	myvernemq-1
7	7	myvernemq-0	0	1000	0	0	102	22	19	99	0	0	nil	myvernemq-0	myvernemq-0
8	8	myvernemq-1	0	1000	0	0	132	22	19	129	NA	0	nil	myvernemq-2	myvernemq-1
9	9	myvernemq-0	0	1000	0	0	127	22	19	124	NA	0	nil	myvernemq-2	myvernemq-0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1699	1699	myvernemq-2	0	1000	0	0	132	22	19	129	NA	0	nil	myvernemq-1	myvernemq-2
1700	1700	myvernemq-1	0	1000	0	0	152	22	20	150	0	0	nil	myvernemq-1	myvernemq-1
1701	1701	myvernemq-2	0	1001	1	0	142	22	19	139	0	0	nil	myvernemq-2	myvernemq-2
1702	1702	myvernemq-0	0	1000	0	0	102	22	19	99	NA	0	nil	myvernemq-1	myvernemq-0
1703	1703	myvernemq-2	0	1000	0	0	193	22	20	190	NA	0	nil	myvernemq-1	myvernemq-2
1704	1704	myvernemq-1	0	999	0	1	102	22	19	99	0	0	nil	myvernemq-1	myvernemq-1
1705	1705	myvernemq-0	0	1000	0	0	122	22	19	119	NA	0	nil	myvernemq-2	myvernemq-0
1706	1706	myvernemq-1	1	1000	0	0	122	22	19	119	NA	0	nil	myvernemq-2	myvernemq-1
1707	1707	myvernemq-2	0	1000	0	0	152	22	19	149	0	0	nil	myvernemq-2	myvernemq-2
1708	1708	myvernemq-1	0	1001	1	0	132	22	19	129	NA	0	nil	myvernemq-0	myvernemq-1
1709	1709	myvernemq-2	0	1000	0	0	124	24	21	121	NA	0	nil	myvernemq-0	myvernemq-2
1710	1710	myvernemq-0	0	1001	1	0	102	22	19	99	0	0	nil	myvernemq-0	myvernemq-0
1711	1711	myvernemq-1	0	1000	0	0	152	22	19	149	NA	0	nil	myvernemq-2	myvernemq-1
1712	1712	myvernemq-0	0	1001	1	0	123	23	20	120	NA	0	nil	myvernemq-2	myvernemq-0
1713	1713	myvernemq-1	0	1000	0	0	187	22	19	184	NA	0	nil	myvernemq-2	myvernemq-1
1714	1714	myvernemq-2	0	1000	0	0	97	22	19	94	0	0	nil	myvernemq-2	myvernemq-2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1739	1739	myvernemq-1	0	1000	0	0	117	22	19	114	0	0	nil	myvernemq-1	myvernemq-1
1740	1740	myvernemq-2	0	1000	0	0	127	22	19	124	NA	0	nil	myvernemq-0	myvernemq-2
1741	1741	myvernemq-1	0	1001	1	0	137	22	19	134	NA	0	nil	myvernemq-0	myvernemq-1
1742	1742	myvernemq-0	0	1000	0	0	107	22	19	104	0	0	nil	myvernemq-0	myvernemq-0
1743	1743	myvernemq-2	0	1000	0	0	142	22	19	139	NA	0	nil	myvernemq-1	myvernemq-2
1744	1744	myvernemq-0	0	1000	0	0	122	22	19	119	NA	0	nil	myvernemq-1	myvernemq-0
1745	1745	myvernemq-2	0	999	0	1	182	22	19	179	NA	0	nil	myvernemq-1	myvernemq-2
1746	1746	myvernemq-1	0	1000	0	0	97	22	19	94	0	0	nil	myvernemq-1	myvernemq-1

3.2 Example HA testrun

Note that this example is a capture of one of the rare testruns that still shows a single lost message.
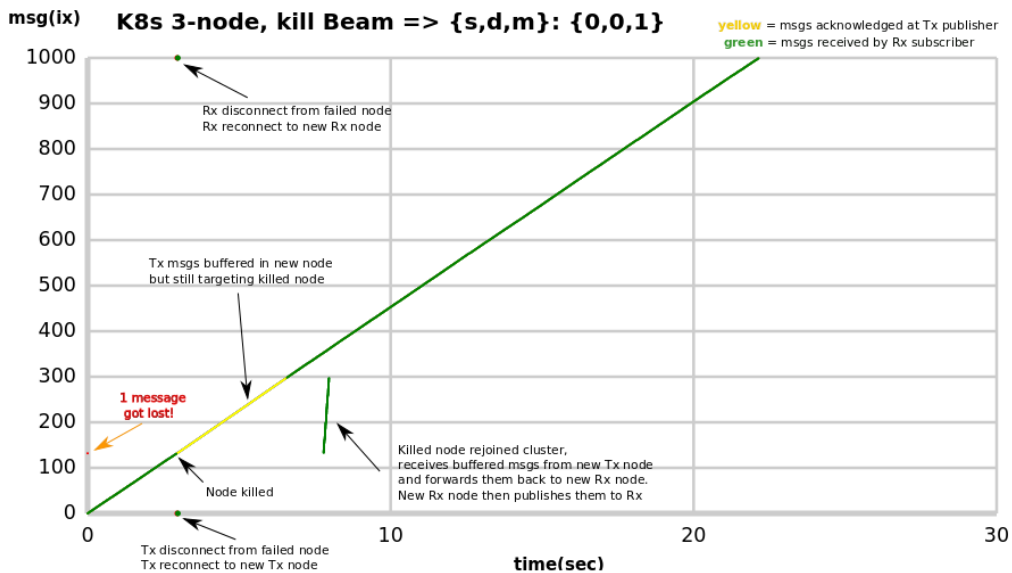
4   Conclusion

- **The "Cluster HA improvement" almost completely resolved the issue of lost messages in QoS1.**

- "almost" because we still can get 1 message lost in about 1000 testruns. (as can be seen in the last plot above)

- This remaining message loss is a result of unfortunate timing and it's probability of occurrence is therefore directly related to message rate.

- Note that in order to avoid message loss, the VerneMQ parameter **"outgoing_clustering_buffer_size" must be set sufficiently high.**
  This parameter limits the size of the buffer on the inter-node tcp link.
  It's value depends on :
  - The maximum expected message rate between 2 broker nodes
  - The size of the messages
  - The maximum expected time between a node crash and restart + rejoin
    (in the example above this is about 7s, but sometimes it took even longer)

LikeBe the first to like this

- No labels

- Edit Labels

Write a comment…