Debugging Report

7 Aug 2019

By Kaleem Peeroo

Contents

[Debugging Summary 2](#_Toc18418248)

[Step 1 Summary 2](#_Toc18418249)

[Step 2 Summary 2](#_Toc18418250)

[Combination 1.1 2](#_Toc18418251)

[Combination 1.2 2](#_Toc18418252)

[Combination 2.1 2](#_Toc18418253)

[Combination 2.2 2](#_Toc18418254)

[Combination 3.1 2](#_Toc18418255)

[Combination 3.2 2](#_Toc18418256)

[Combination 4.1 3](#_Toc18418257)

[Combination 4.2 3](#_Toc18418258)

[Combination 5.1 3](#_Toc18418259)

[Combination 5.2 3](#_Toc18418260)

[Combination 6.1 3](#_Toc18418261)

[Combination 6.2 3](#_Toc18418262)

[Step 3 Summary 3](#_Toc18418263)

[Combination 1 3](#_Toc18418264)

[Combination 2 3](#_Toc18418265)

[Combination 3 3](#_Toc18418266)

[Combination 4 3](#_Toc18418267)

[Combination 5 4](#_Toc18418268)

[Combination 6 4](#_Toc18418269)

[Step 1: Run 1 pub, 1 sub on single machine on all machines 4](#_Toc18418270)

[Windows Machine 1 (nsq905ap): 4](#_Toc18418271)

[Windows Machine 2 (nsq906ap): 5](#_Toc18418272)

[Windows Machine 3 (nsq907ap): 5](#_Toc18418273)

[Windows Machine 4 (nsq908ap): 6](#_Toc18418274)

[Step 2: Run 1 pub, 1 sub on separate machines in different pairs/combinations 6](#_Toc18418275)

[Combination 1: VM 1 + VM 2 (nsq905ap + nsq906ap): 7](#_Toc18418276)

[Combination 1.1 7](#_Toc18418277)

[Combination 1.2 7](#_Toc18418278)

[Combination 2: VM 1 + VM 3 (nsq905ap + nsq907ap): 8](#_Toc18418279)

[Combination 2.1 8](#_Toc18418280)

[Combination 2.2 8](#_Toc18418281)

[Combination 3: VM 1 + VM 4 (nsq905ap + nsq908ap): 9](#_Toc18418282)

[Combination 3.1 9](#_Toc18418283)

[Combination 3.2 9](#_Toc18418284)

[Combination 4: VM 2 + VM 3 (nsq906ap + nsq907ap): 10](#_Toc18418285)

[Combination 4.1 10](#_Toc18418286)

[Combination 4.2 11](#_Toc18418287)

[Combination 5: VM 2 + VM 4 (nsq906ap + nsq908ap): 11](#_Toc18418288)

[Combination 5.1 11](#_Toc18418289)

[Combination 5.2 12](#_Toc18418290)

[Combination 6: VM 3 + VM 4 (nsq907ap + nsq908ap): 12](#_Toc18418291)

[Combination 6.1 12](#_Toc18418292)

[Combination 6.2 13](#_Toc18418293)

[Step 3: Run step 1 simultaneously 13](#_Toc18418294)

[Combination 1: VM 1 + VM 2 (nsq905ap + nsq906ap): 14](#_Toc18418295)

[Machine 1 14](#_Toc18418296)

[Machine 2 14](#_Toc18418297)

[Combination 2: VM 1 + VM 3 (nsq905ap + nsq907ap): 15](#_Toc18418298)

[Machine 1 15](#_Toc18418299)

[Machine 3 15](#_Toc18418300)

[Combination 3: VM 1 + VM 4 (nsq905ap + nsq908ap): 16](#_Toc18418301)

[Machine 1 16](#_Toc18418302)

[Machine 4 16](#_Toc18418303)

[Combination 4: VM 2 + VM 3 (nsq906ap + nsq907ap): 17](#_Toc18418304)

[Machine 2 17](#_Toc18418305)

[Machine 3 17](#_Toc18418306)

[Combination 5: VM 2 + VM 4 (nsq906ap + nsq908ap): 18](#_Toc18418307)

[Machine 2 18](#_Toc18418308)

[Machine 4 19](#_Toc18418309)

[Combination 6: VM 3 + VM 4 (nsq907ap + nsq908ap): 19](#_Toc18418310)

[Machine 3 19](#_Toc18418311)

[Machine 4 20](#_Toc18418312)

# Debugging Summary

The debugging tests consists of 3 steps:

1. Step 1: Run 1 pub, 1 sub on single machine on all machines
2. Step 2: Run 1 pub, 1 sub on separate machines in different pairs/combinations
3. Step 3: Run step 1 simultaneously

The following key words are of the same:

**Machine 1 = nsq905ap = VM 1**

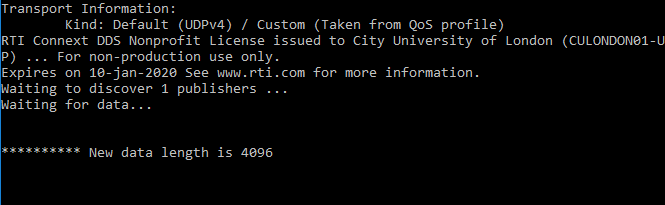
**Machine 2 = nsq906ap = VM 2**

**Machine 3 = nsq907ap = VM 3**

**Machine 4 = nsq908ap = VM 4**

If a test has Ended then on the console it is shown that the test is complete by displaying the cursor for the console input.

If a test has Continued then within the console the same view has been shown as the beginning of the test. The tests will start and continue to look like the following:



In all cases the publisher ends the tests as shown below:



All data files are stored in the zip file named “Debugging Tests.zip”.

## Step 1 Summary

|  |  |  |
| --- | --- | --- |
| **Machine** | **Publisher Status** | **Subscriber Status** |
| 1 (nsq905ap) | Ended | Continued |
| 2 (nsq906ap) | Ended | Ended |
| 3 (nsq907ap) | Ended | Ended |
| 4 (nsq908ap) | Ended | Continued |

As seen from the table above it seems that machines 2 and 3 can run the tests fine and in both cases the tests end. However, from initial inspection the first and last machines seem to have a common case of the tests continuing.

## Step 2 Summary

### Combination 1.1

|  |  |  |
| --- | --- | --- |
|  | **Machine 1 (Publisher)** | **Machine 2 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 1.2

|  |  |  |
| --- | --- | --- |
|  | **Machine 2 (Publisher)** | **Machine 1 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 2.1

|  |  |  |
| --- | --- | --- |
|  | **Machine 1 (Publisher)** | **Machine 3 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 2.2

|  |  |  |
| --- | --- | --- |
|  | **Machine 3 (Publisher)** | **Machine 1 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 3.1

|  |  |  |
| --- | --- | --- |
|  | **Machine 1 (Publisher)** | **Machine 4 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 3.2

|  |  |  |
| --- | --- | --- |
|  | **Machine 4 (Publisher)** | **Machine 1 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 4.1

|  |  |  |
| --- | --- | --- |
|  | **Machine 2 (Publisher)** | **Machine 3 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 4.2

|  |  |  |
| --- | --- | --- |
|  | **Machine 3 (Publisher)** | **Machine 2 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 5.1

|  |  |  |
| --- | --- | --- |
|  | **Machine 2 (Publisher)** | **Machine 4 (Subscriber)** |
| **Status** | Ended | Ended |

### Combination 5.2

|  |  |  |
| --- | --- | --- |
|  | **Machine 4 (Publisher)** | **Machine 2 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 6.1

|  |  |  |
| --- | --- | --- |
|  | **Machine 3 (Publisher)** | **Machine 4 (Subscriber)** |
| **Status** | Ended | Continued |

### Combination 6.2

|  |  |  |
| --- | --- | --- |
|  | **Machine 3 (Publisher)** | **Machine 4 (Subscriber)** |
| **Status** | Ended | Continued |

## Step 3 Summary

### Combination 1

|  |  |  |
| --- | --- | --- |
|  | **Publisher** | **Subscriber** |
| **Machine 1** | Ended | Ended |
| **Machine 2** | Ended | Continued |

### Combination 2

|  |  |  |
| --- | --- | --- |
|  | **Publisher** | **Subscriber** |
| **Machine 1** | Ended | Ended |
| **Machine 3** | Ended | Continued |

### Combination 3

|  |  |  |
| --- | --- | --- |
|  | **Publisher** | **Subscriber** |
| **Machine 1** | Ended | Ended |
| **Machine 4** | Ended | Ended |

### Combination 4

|  |  |  |
| --- | --- | --- |
|  | **Publisher** | **Subscriber** |
| **Machine 2** | Ended | Ended |
| **Machine 3** | Ended | Ended |

### Combination 5

|  |  |  |
| --- | --- | --- |
|  | **Publisher** | **Subscriber** |
| **Machine 2** | Ended | Ended |
| **Machine 4** | Ended | Ended |

### Combination 6

|  |  |  |
| --- | --- | --- |
|  | **Publisher** | **Subscriber** |
| **Machine 3** | Ended | Continued |
| **Machine 4** | Ended | Ended |

For more details about the tests read further.

# Step 1: Run 1 pub, 1 sub on single machine on all machines

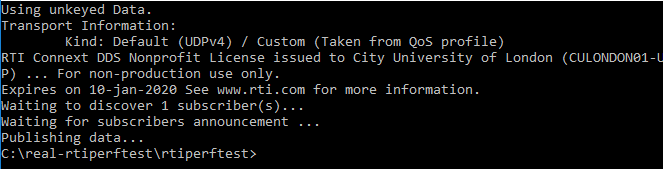
Here is the script I used for all the tests in step 1:

start cmd /k perftest\_java.bat -pub -dataLen 4096 -domain 2 -latencyCount 10000 -numIter 10000000 -bestEffort -multicast

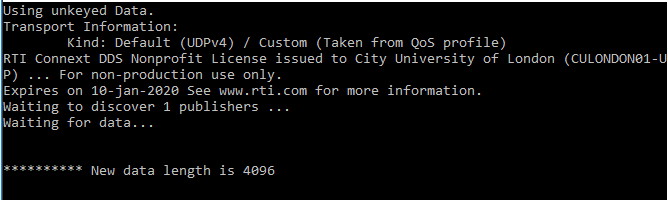
start cmd /k perftest\_java.bat -sub -dataLen 4096 -domain 2 -latencyCount 10000 -numIter 10000000 -bestEffort -multicast -fileName sub1

## Windows Machine 1 (nsq905ap):

Publisher Ended:

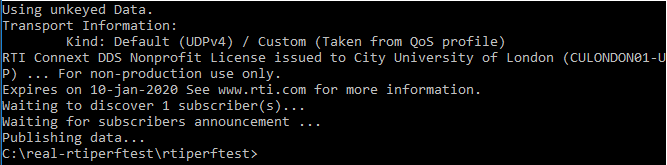


Subscriber Continued:

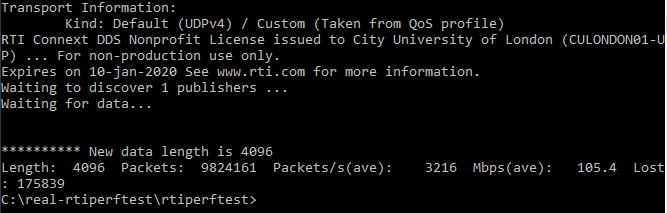


## Windows Machine 2 (nsq906ap):

Publisher Ended:

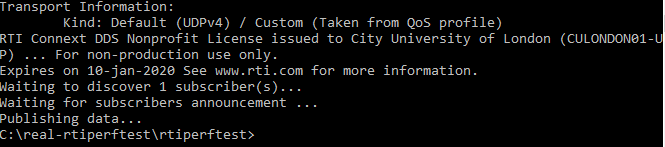


Subscriber Ended:

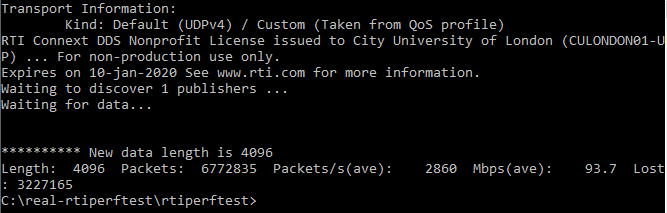


## Windows Machine 3 (nsq907ap):

Publisher Ended:

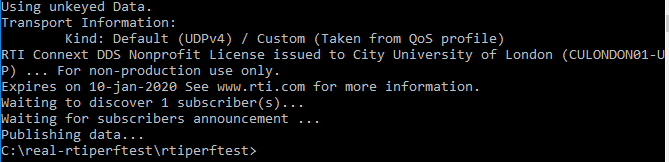


Subscriber Ended:

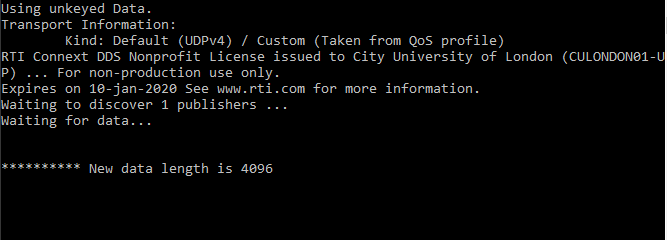


## Windows Machine 4 (nsq908ap):

Publisher Ended:



Subscriber Continued:



# Step 2: Run 1 pub, 1 sub on separate machines in different pairs/combinations

Here is the script for the subscriber used in all tests in step 2:

start cmd /k perftest\_java.bat -sub -dataLen 4096 -domain 2 -latencyCount 10000 -numIter 10000000 -bestEffort -multicast -fileName sub1

Here is the script for the publisher used in all tests in step 2:

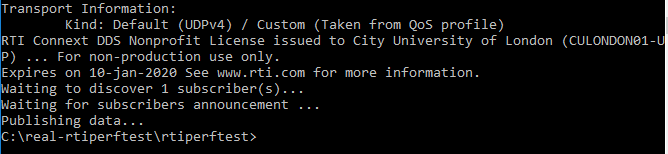
start cmd /k perftest\_java.bat -sub -dataLen 4096 -domain 2 -latencyCount 10000 -numIter 10000000 -bestEffort -multicast -fileName sub1

## Combination 1: VM 1 + VM 2 (nsq905ap + nsq906ap):

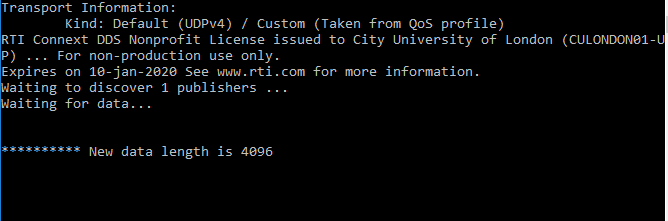
### Combination 1.1

|  |  |
| --- | --- |
| Publisher: | VM 1 |
| Subscriber: | VM 2 |

Publisher Ended:



Subscriber Continued:



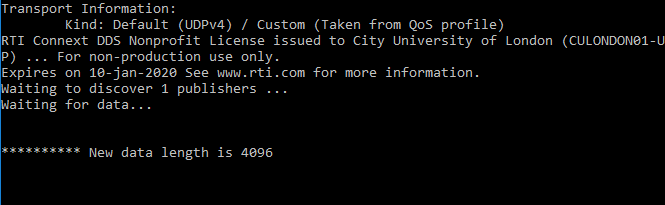
### Combination 1.2

|  |  |
| --- | --- |
| Publisher: | VM 2 |
| Subscriber: | VM 1 |

Publisher Ended:



Subscriber Continued:



## Combination 2: VM 1 + VM 3 (nsq905ap + nsq907ap):

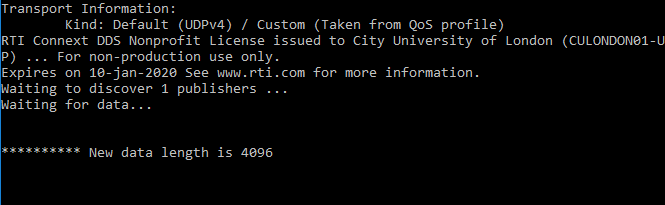
### Combination 2.1

|  |  |
| --- | --- |
| Publisher: | VM 1 |
| Subscriber: | VM 3 |

Publisher Ended:



Subscriber Continued:



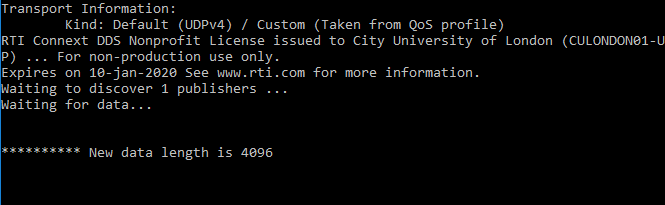
### Combination 2.2

|  |  |
| --- | --- |
| Publisher: | VM 3 |
| Subscriber: | VM 1 |

Publisher Ended:



Subscriber Continued:



## Combination 3: VM 1 + VM 4 (nsq905ap + nsq908ap):

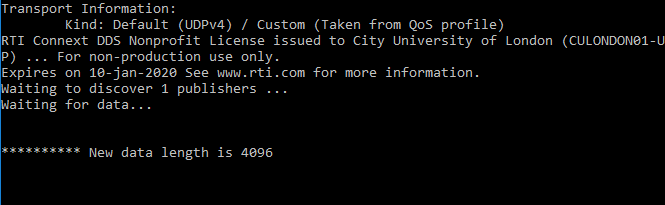
### Combination 3.1

|  |  |
| --- | --- |
| Publisher: | VM 1 |
| Subscriber: | VM 4 |

Publisher Ended:



Subscriber Continued:



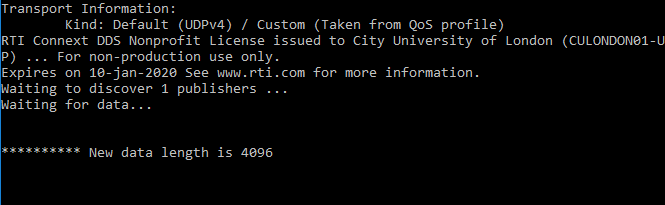
### Combination 3.2

|  |  |
| --- | --- |
| Publisher: | VM 4 |
| Subscriber: | VM 1 |

Publisher Ended:



Subscriber Continued:



## Combination 4: VM 2 + VM 3 (nsq906ap + nsq907ap):

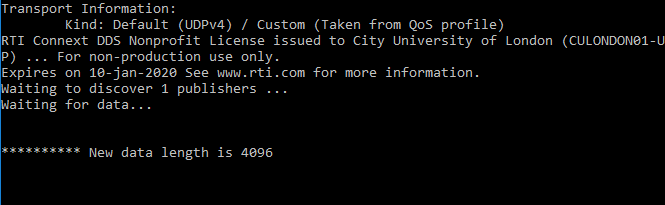
### Combination 4.1

|  |  |
| --- | --- |
| Publisher: | VM 2 |
| Subscriber: | VM 3 |

Publisher Ended:



Subscriber Continued:



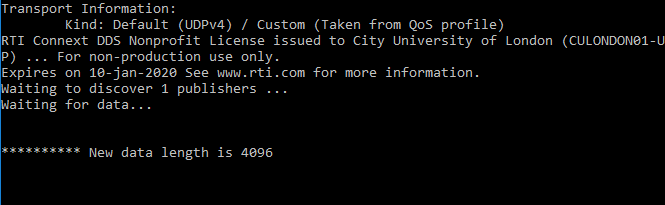
### Combination 4.2

|  |  |
| --- | --- |
| Publisher: | VM 3 |
| Subscriber: | VM 2 |

Publisher Ended:



Subscriber Continued:



## Combination 5: VM 2 + VM 4 (nsq906ap + nsq908ap):

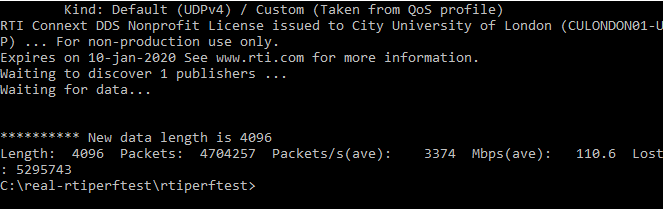
### Combination 5.1

|  |  |
| --- | --- |
| Publisher: | VM 2 |
| Subscriber: | VM 4 |

Publisher Ended:



Subscriber Ended:



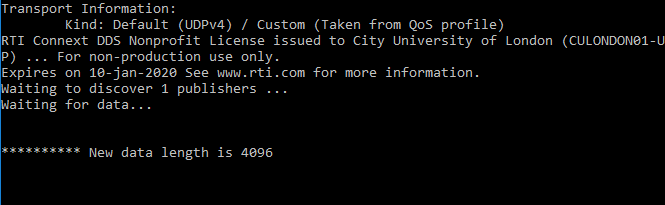
### Combination 5.2

|  |  |
| --- | --- |
| Publisher: | VM 4 |
| Subscriber: | VM 2 |

Publisher Ended:



Subscriber Continued:



## Combination 6: VM 3 + VM 4 (nsq907ap + nsq908ap):

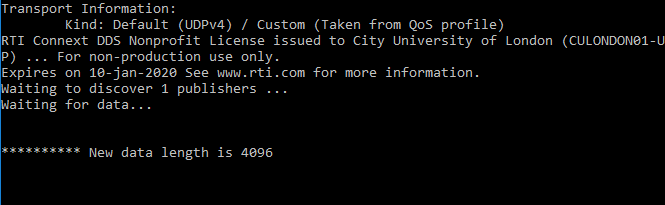
### Combination 6.1

|  |  |
| --- | --- |
| Publisher: | VM 3 |
| Subscriber: | VM 4 |

Publisher Ended:



Subscriber Continued:



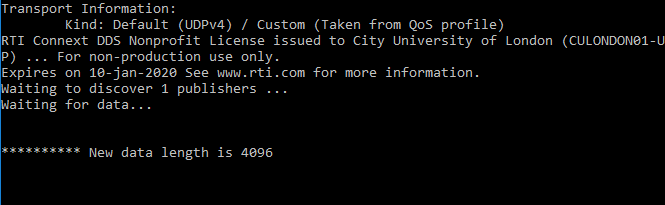
### Combination 6.2

|  |  |
| --- | --- |
| Publisher: | VM 4 |
| Subscriber: | VM 3 |

Publisher Ended:



Subscriber Continued:



# Step 3: Run step 1 simultaneously

In this step 1 pub, 1 sub would be run on one machine while another machine is also running the same type of test just with a different domain.

Here is the script I used for all the tests in step 3:

start cmd /k perftest\_java.bat -pub -dataLen 4096 -domain 2 -latencyCount 10000 -numIter 10000000 -bestEffort -multicast

start cmd /k perftest\_java.bat -sub -dataLen 4096 -domain 2 -latencyCount 10000 -numIter 10000000 -bestEffort -multicast -fileName sub1

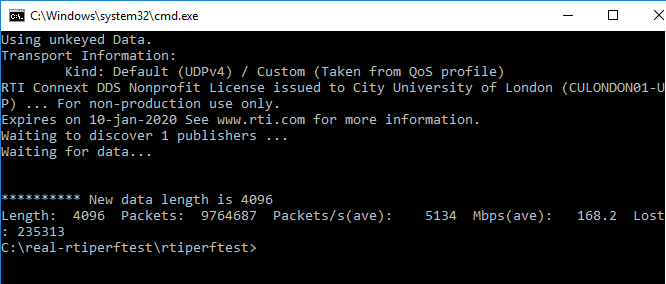
## Combination 1: VM 1 + VM 2 (nsq905ap + nsq906ap):

### Machine 1

Publisher Ended:



Subscriber Ended:

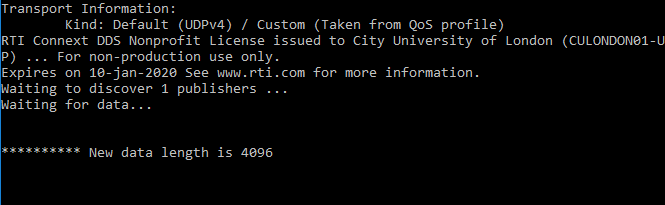


### Machine 2

Publisher Ended:



Subscriber Continued:



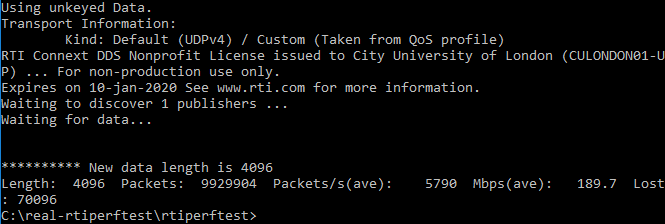
## Combination 2: VM 1 + VM 3 (nsq905ap + nsq907ap):

### Machine 1

Publisher Ended:



Subscriber Ended:

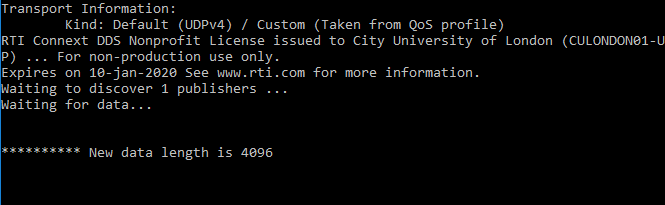


### Machine 3

Publisher Ended:



Subscriber Continued:



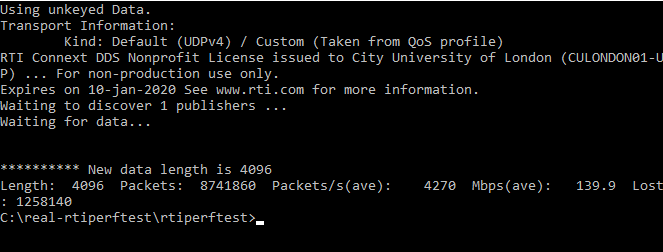
## Combination 3: VM 1 + VM 4 (nsq905ap + nsq908ap):

### Machine 1

Publisher Ended:



Subscriber Ended:

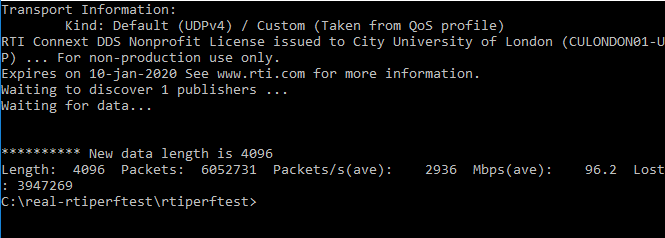


### Machine 4

Publisher Ended:



Subscriber Ended:



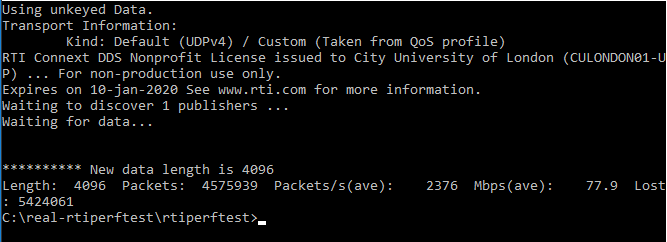
## Combination 4: VM 2 + VM 3 (nsq906ap + nsq907ap):

### Machine 2

Publisher Ended:



Subscriber Ended:

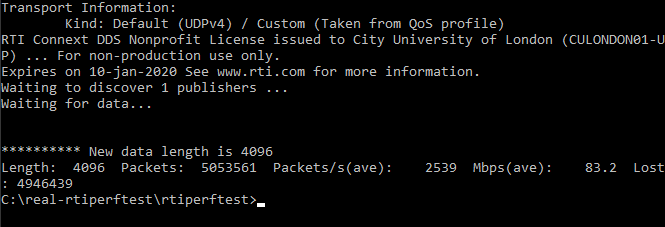


### Machine 3

Publisher Ended:



Subscriber Ended:



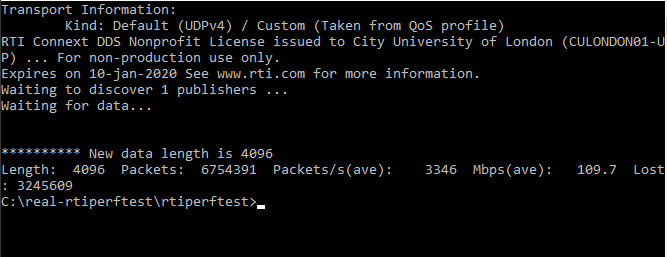
## Combination 5: VM 2 + VM 4 (nsq906ap + nsq908ap):

### Machine 2

Publisher Ended:



Subscriber Ended:

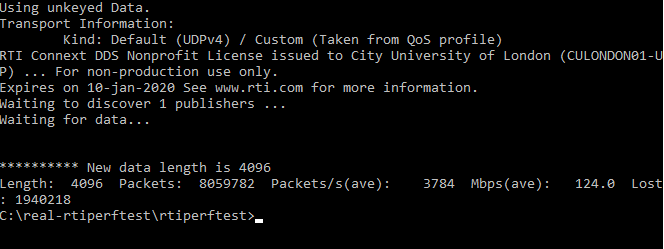


### Machine 4

Publisher Ended:



Subscriber Ended:



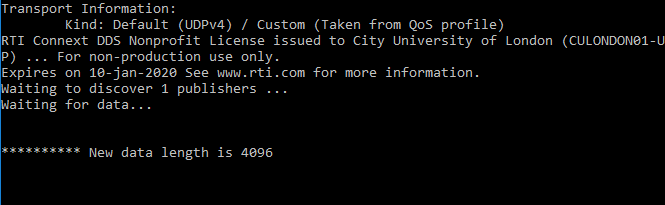
## Combination 6: VM 3 + VM 4 (nsq907ap + nsq908ap):

### Machine 3

Publisher Ended:



Subscriber Continued:



### Machine 4

Publisher Ended:



Subscriber Ended:

