Network Working Group
Internet-Draft
Intended status: Informational
Expires: July 16, 2009

M. Boucadair, Ed.
J-L. Grimault
P. Levis
A. Villefranque
France Telecom
January 12, 2009

Behaviour of BitTorrent service in an IP Shared Address Environment
draft-boucadair-behave-bittorrent-portrange-02.txt

Status of this Memo

This Internet-Draft will expire on July 16, 2009.

Copyright Notice

Abstract

This memo describes the behaviour of BitTorrent service in the context of IP shared addresses. It provides an overview of the used testbed and main results of the tests that have been conducted in order to assess the limitations of an architecture based on shared IP addresses.

Table of Contents

1. Introduction

Recently, several proposals have been disseminated within IETF to contribute to solve the IP exhaustion problem. These solutions may be grouped into two categories:

(1) Solutions which propose the introduction of a second level of NAT (Network Address Translator), denoted also as Carrier Grade NAT (CG-NAT). This node is located in the Service Provider domain. Private addresses are assigned to end-user CPEs, which still perform their own NAT. The CG-NAT is responsible for translating IP packets issued with private addresses to ones with publicly routable IPv4 addresses (especially when exiting the domain of the Service Provider). [ID.durand-softwire-dual-stack-lite] is a variant of these solutions where there is only one NAT hosted in the Service Provider's network.

(2) Solutions which avoid the introduction of a NAT in the Service Provider's network. Examples of these solutions are [ID.ymbk-aplusp], [ID.boucadair-port-range], [ID.despres-sam] and [ID.bajko-v6ops-port-restricted-ipaddr-assign]. These solutions allocate the same IP public address to several customers at the same time. They also allocate a restricted port range to each customer so that two customers with the same IP address have two different port ranges that do not overlap.

Both the above listed categories are based on sharing an IP address between several machines. In this context, the delivery of some services may be impacted, especially those enforcing a restriction based on the source IP address.

This memo focuses on BitTorrent as an example of application which applies a restriction based on IP address. This memo describes a testing campaign that has been carried out to assess the impact of IP shared address on BitTorrent.

Testing activities are conducted using a testbed which is configured according to the solution described in [ID.boucadair-port-range].

2. BitTorent Overview

2.1. BitTorrent at a Glance

BitTorrent is a distributed file sharing infrastructure. It is based on P2P (Peer to Peer) techniques for exchanging files between connected users. Three parties are involved in a BitTorrent

architecture as detailed hereafter:

1. The Server: The server into which, has been uploaded the torrent file.

2. The Tracker: Maintains a list of clients which have the file or some portions of that file.

3. The Client: Entities which are downloading and/or uploading portions of the file. Two categories of clients may be distinguished:

   A. Leechers: Clients which are currently downloading the file but do not yet detain all the portions of the file. As for the portions already obtained, the leechers upload them towards requesting clients;

   B. Seeders: Clients which detain all the portions of the file and are uploading them to other requesting clients.

A torrent file is a file which includes the meta-data information of the file to be shared: the file name, its length, a hash and the URL of the tracker. In order to download a given file, a BitTorrent client needs to obtain the corresponding torrent file. Afterwards, it connects to the tracker to retrieve a list of leechers and seeders. Then, the client connects to those machines and downloads the available portions of the requested file. It uploads also the portions already obtained towards requesting clients.

2.2. Software Configuration

This section provides an overview of installed tools.

2.2.1. BitTorrent Client

Various BitTorrent clients are available for public use. The following one has been installed for the purposes of our testing activities:

URL: www.bittorrent.com

The installed version is 6.1.

2.2.2. BitTorrent Server

The BitTorrent server that has been used is the following:

URL: www.metro-torrent.com

2.2.3. BitTorrent Tracker

The BitTorrent tracker that has been used is the following:

URL: www.metro-torrent.com/announce.php

3. Testbed Overview

3.1. Testbed Description

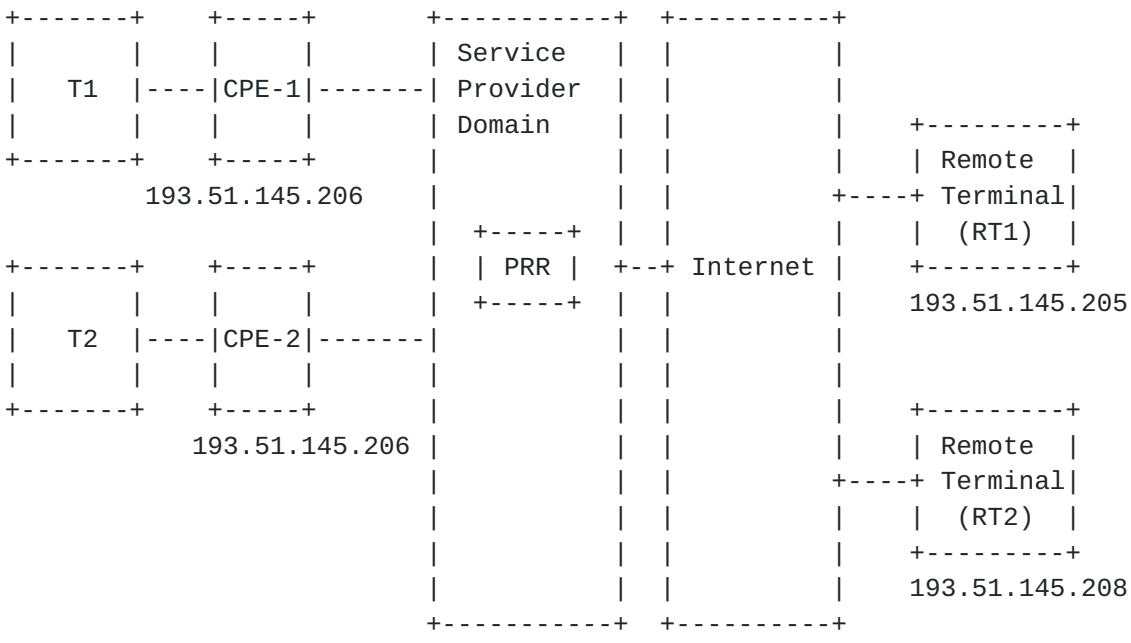
The testbed used to conduct the testing activities, described in Section 5, is the same as the one described in Section 4 of [ID.boucadair-port-range].

- The PRR (Port Range Router) is responsible to implement a port-driven routing so as to be able to route incoming traffic to the appropriate machine among those having the same IP address.

- CPE-1 and CPE-2 are two CPEs to which the same IP address is assigned. In order to discriminate these CPEs, distinct Port Masks [ID.boucadair-dhc-port-range] are assigned to each of these CPE.

- T1 (respectively T2) is a machine located in the LAN behind CPE-1 (respectively CPE-2).

- RT1 and RT2 are remote machines reachable through Internet.



3.2. Files

The following table lists the files available in each machine:

Machine' s name	Available files
T1	TestCaenF1 and TestCaenFa
T2	TestCaenF1 and TestCaenFb
RT1	TestCaenFRT1 and TestCaenFRTa
RT2	TestCaenFRT1 and TestCaenFRTb

Available files

4. Description of Tests

This section lists the tests that have been conducted.

4.1. Connection to Overlay Test Group

This table lists the test to assess the ability of distinct machines having the same IP address to connect to BitTorrent overlay.

Test Index	Test Title	Purpose	Description
Test_1	Connection to BitTorrent Overlay	Check if two terminals, having the same public IP address, are able to connect to BitTorrent overlay network	Check if BitTorrent client installed on T1 and T2 machines are able to use the same tracker and that no problems are experienced to use the same tracker by T1 and T2.

Connecting to Overlay Test Group

4.2. Upload Test Group

This test group aims at checking if upload operations are not impacted/restricted due to the presence of several machines with the same IP address.

Test Index	Test Title	Purpose	Description
Test_2	Uploading distinct files using the same BitTorrent tracker and server	Check if two terminals, having the same public IP address, are able to upload torrent files (referring to distinct files) using the same tracker and same server	Check if torrent files may be uploaded from T1 and T2 using the same tracker. On T1 (resp. T2), generate a torrent file TestCaenFa.torrent (resp. TestCaenFb.torrent) referring to the file TestCaenFa (resp. TestCaenFb) and pointing to the tracker TRA. From T1 (resp. T2) try to put TestCaenFa.torrent (resp. TestCaenFb.torrent) onto server S. Check if the upload operation has succeeded
Test_3	Uploading torrent files referring to the same file	Check if two terminals, having the same public IP address, are able to upload torrent files, which refer to the same file, using the same tracker	On T1 (resp. T2), generate a torrent file TestCaenF1.torrent (resp. TestCaenF1.torrent) referring to the file TestCaenF1 and pointing to the tracker TRA. From T1 (resp. T2) try to put TestCaenF1.torrent (resp. TestCaenF1.torrent) onto server S. Check if the upload operation has succeeded

Upload Test Group

4.3. Mutual Download Test Group

The purpose of this test group is to check if mutual downloading operations can occur between machines having the same IP address.

Test Index	Test Title	Purpose	Description
Test_4	Mutual Downloading between machines sharing the same IP address	Check if two terminals having the same public IP address can download a file from each another	Check if T1 can download the file uploaded by T2 (ref. Test_2) and vice versa. Three scenarios are to be tested: (1) T1 downloads TestCaenFb but T2 does not download any file from T1, (2) T2 downloads TestCaenFa but T1 does not download any file from T2, (3) T1 downloads TestCaenFb and T2 downloads TestCaenFa at the same time

Mutual Download Test Group

4.4. Simultaneous Download Test Group

This test group aims at checking if simultaneous downloading operations from remote seed(s)/leecher(s) can be performed by several machines sharing the same IP address.

Test Index	Test Title	Purpose	Description
Test_5	Downloading distinct files	Check if two terminals, having the same public IP address, are able to download distinct files available on BitTorrent infrastructure	Check if distinct files available on BitTorrent infrastructure may be downloaded by T1 and T2 simultaneously

Test_6	Downloading the same file located on several seeders	Check if two terminals, having the same public IP address, are able to download the same file located on several seeders	Check if a file available on several seeders may be downloaded from T1 and T2 simultaneously. As an example, check if T1 and T2 can download the same file located in RT1 and RT2 (referred to as TestCaenFRT1)
Test_7	Download the same file available on a single machine	Check if two terminals having the same public IP address are able to download, at the same time, the same file available on a single seed	Check if T1 and T2 can download the same file uploaded by RT1 (referred to as TestCaenFRTa) concurrently. In case the test fails, one of the two host is called the "waiting client"

Test_8	Simultaneous downloading from the same seeder	Check if it is not precluded that a different file can be downloaded by the waiting client from the same seeder	In case Test_7 fails, check that it is not precluded that a different file can be downloaded by the waiting client (T1 or T2) from the same seeder (RT1) at the same time the other terminal (respectively T2 or T1) is downloading TestCaenFRTa. Execute Test_7 in launching on T1 the downloading of TestCaenFRT1 and just few seconds afterwards in launching on T2 the downloading of TestCaenFRT1 and TestCaenFRTa. Check that while T1 is downloading TestCaenFRT1 that does not preclude T2 to concurrently download TestCaenFRTa.
Test_9	Downloading distinct files from the same seeder	Check if the two terminals having the same public IP address are able to download at the same time two distinct files from the same seeder	Check if T1 (respectively T2) can download files uploaded by RT1 (referred to as TestCaenRF1 and TestCaenFRTa) concurrently. Particularly, check if T1 can download TestCaenFRT1 and T2 can download TestCaenFRTa simultaneously
Test_10	Download the same file located on machines having the same IP address	Check if the same file can be downloaded by a given machine from seeders having the same IP address	in RT1, launch the downloading of TestCaenF1. Check that RT1 is downloading portions of TestCaenF1 at the same time from T1 and T2

	Test_11	Automatic query to download the same file available on a single machine	Check if the terminal which was waiting can finally download the file once the other terminal has finished	In case Test_7 fails, check that the terminal which was waiting can finally download the file once the other terminal has finished
	Test_12	Download distinct files from two machines having the same IP address	Check if distinct files can be downloaded by the same machine from seeders having the same IP address	Check if RT1 can download simultaneously TestCaenFa (from T1) and TestCaenFb (from T2)

Simultaneous Download Test Group

5. Results

BitTorrent client can be configured to accept multiple connections using the same IP address. A dedicated parameter can therefore be positioned. This parameter is called: bt.allow_same_ip. Possible values that can be taken by this parameter are: FALSE (0) or TRUE (1).

For the testing activities, two configurations have been tested:

1. First Configuration: bt.allow_same_ip == TRUE

2. Second Configuration: bt.allow_same_ip == FALSE

The following sub-sections describe the obtained results for each configuration.

5.1. First Configuration: Multiple Connections with the same IP address are enabled

The following table summarises the results of the aforementioned tests as performed using the testbed described in Section 4. Note that bt.allow_same_ip is positioned to TRUE.

Test Identifier	Results	Comments
Test_1	No problems have been experienced	None
Test_2	Both T1 and T2 are able to upload distinct torrent files using the same tracker and the same server	None
Test_3	Only one machine can upload a torrent file referring to the same file	The server ensures that only one single torrent file corresponding to the same file is listed in its base
Test_4	Three scenarios have been tested: (1) T1 downloads TestCaenFb but T2 does not download any file from T1 (2) T2 downloads TestCaenFa but T1 does not download any file from T2 (3) T1 downloads TestCaenFb and T2 downloads TestCaenFa in the same time. For all these scenarios, no problems have been encountered. The downloading operations have succeeded	None
Test_5	Both T1 and T2 are able to download distinct files from the BitTorrent infrastructure	None
Test_6	Both T1 and T2 are able to download the same file located in several seeders. No particular problem has been encountered	None
Test_7	No problem has been encountered. Both T1 and T2 are able to download TestCaenFRTa from RT1 simultaneously. Note that at the same time, mutual downloading by T1 of portions of TestCaenFRTa already downloaded by T2 (and vice versa) have been noticed	None
Test_8	Not applicable	None

Test_9	No problem has been encountered. Distinct files located in RT1 have been successfully downloaded by T1 (respectively T2)	None
Test_10	No problem has been encountered	None
Test_11	Not applicable	Not applicable
Test_12	No problem has been encountered. RT1 has succeeded to download simultaneously TestCaenFa (from T1) and TestCaenFb (from T2)	None

First Configuration Obtained Results

5.2. Second Configuration: Multiple Connections with the same IP address are disabled

The following table summarises the results of the aforementioned tests as performed using the testbed described in Section 4. Note that bt.allow_same_ip is positioned to FALSE.

Test Identifier	Results	Comments
Test_1	No problems have been experienced	None
Test_2	Both T1 and T2 are able to upload distinct torrent files using the same tracker and the same server	None
Test_3	Only one machine can upload a torrent file referring to the same file	The server ensures that only one single torrent file corresponding to the same file is listed in its base

Test_4	Three scenarios have been tested: (1) T1 downloads TestCaenFb but T2 does not download any file from T1 (2) T2 downloads TestCaenFa but T1 does not download any file from T2 (3) T1 downloads TestCaenFb and T2 downloads TestCaenFa in the same time. For all these scenarios, no problems have been encountered. The downloading operations have succeeded	None
Test_5	Both T1 and T2 are able to download distinct files from the BitTorrent infrastructure	None
Test_6	Both T1 and T2 are able to download the same file located in several seeders. No particular problem has been encountered	When TestCaenFRT1 is used as example. T1 and T2 are able to download the same file. But for each file it is sending (here TestCaenFRT1) RT1 can allow no more than one unique connection to the same address IP. This is the same behaviour for RT2. T1 and T2 exchanges the portions of the files they stored

Test_7	Both T1 and T2 are able to download the file but only one single connection is accepted by RT1 at the same time	This is because for each file it is sending (here TestCaenFRTa) RT1 can allow no more than one unique connection to the same address IP. The result is that, once T1 (or T2) has begun to download TestCaenFRTa, the other terminal (T2 or respectively T1) cannot get any portion of TestCaenFRTa directly from RT1 till the other (T1 or respectively T2) has completed the downloading of TestCaenFRTa. However, that does not preclude the waiting terminal (T2 or T1) to download from the other terminal (T1 or T2) portions of TestCaenFRTa already downloaded from RT1
Test_8	The test 8 has succeeded	While T1 has been downloading TestCaenFRT1 from RT1, T2 could download TestCaenFRTa from RT1 and in addition it can get portions of TestCaenFRTa already downloaded by T1
Test_9	No problem have been experienced	None
Test_10	Both T1 and T2 are able to upload the file, but only one connection is accepted by RT1 at the same time	The test failed because, once RT1 has begun to download portions of TestCaenF1 from T1 (respectively T2) it cannot accept additional connection with T2 for the same file
Test_11	The test succeeded	Once T1 has completed its downloading from RT1, T2 has been able automatically to connect to RT1 for receiving the portions of TestCaenFRTa it has not already got from T2
Test_12	No problem has been encountered. RT1 has succeeded to download simultaneously TestCaenFa (from T1) and TestCaenFb (from T2)	None

Results: Multiple connections desiabled

6. Conclusions

This memo describes the main behaviour of BitTorrent service in an IP shared address environment. Particularly, the tests have been carried out on a testbed implementing [ID.boucadair-port-range] solution. The results are, however, valid for all IP shared address based solutions.

Two limitations were experienced. The first limitation occurs when two clients sharing the same IP address want to simultaneously retrieve the SAME file located in a SINGLE remote peer. This limitation is due to the default BitTorrent configuration on the remote peer which does not permit sending the same file to multiple ports of the same IP address. This limitation is mitigated by the fact that clients sharing the same IP address can exchange portions with each other, provided the clients can find each other through a common tracker, DHT, or Peer Exchange. Even if they can not, we observed that the remote peer would begin serving portions of the file automatically as soon as the other client (sharing the same IP address) finished downloading. This limitation is eliminated if the remote peer is configured with bt.allow_same_ip == TRUE.

The second limitation occurs when a client tries to download a file located on several seeders, when those seeders share the same IP address. This is because the clients are enforcing bt.allow_same_ip parameter to FALSE. The client will only be able to connect to one seeder, among those having the same IP address, to download the file (note that the client can retrieve the file from other seeders having distinct IP addresses). This limitation is eliminated if the local client is configured with bt.allow_same_ip == TRUE, which is somewhat likely as those clients will directly experience better throughput by changing their own configuration.

Mutual file sharing between hosts having the same IP address has been checked. Indeed, machines having the same IP address can share files with no alteration compared to current IP architectures.

7. IANA Considerations

This document raises no IANA considerations.

8. Security Considerations

This memo does not introduce any security issue.

9. Acknowledgements

The authors would like to thank Dan WING for suggesting the writing of this memo and for the review of the text.

10. References

10.1. Normative References

[RFC2026] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

10.2. Informative References

[ID.bajko-v6ops-port-restricted-ipaddr-assign]
Bajko, G. and T. Savolainen, "Port Restricted IP Address Assignment", Internet Draft, draft-bajko-v6ops-port-restricted-ipaddr-assign-02.txt (work in progress), November 2008.

[ID.boucadair-dhc-port-range]
Boucadair, M., Grimault, J-L., Levis, P., and A. Villefranque, "DHCP Options for Conveying Port Mask and Port Range Router IP Address", Internet Draft, draft-boucadair-dhc-port-range-01.txt (work in progress), October 2008.

[ID.boucadair-port-range]
Boucadair, M., Grimault, J-L., Levis, P., and A. Villefranque, "Provider-Provisioned CPE: IPv4 Connectivity Access in the context of IPv4 address exhaustion", Internet Draft, draft-boucadair-port-range-00.txt (work in progress), October 2008.

[ID.despres-sam]
Despres, R., "Stateless Address Mappings (SAMs) IPv6 & extended IPv4 via local routing domains - possibly multihomed", Internet Draft, draft-despres-sam-01.txt (work in progress), November 2008.

[ID.durand-softwire-dual-stack-lite]
Durand , A., Droms, R., Haberman, B., and J. Woodyatt, "Dual-stack lite broadband deployments post IPv4 exhaustion", Internet Draft, draft-durand-softwire-dual-stack-lite-01.txt (work in progress), November 2008.

[ID.ymbk-aplusp]
Maennel , O., Bush, R., Cittadini, L., and S. Bellovin , "The A+P Approach to the Broadband Provider IPv4 Address Shortage", Internet Draft, draft-ymbk-aplusp-01.txt (work in progress), November 2008.

Authors' Addresses

Mohamed Boucadair (editor)
France Telecom
42 rue des Coutures
BP 6243
Caen Cedex 4 14066
France

Email: mohamed.boucadair@orange-ftgroup.com

Jean-Luc Grimault
France Telecom

Email: jeanluc.grimault@orange-ftgroup.com

Pierre Levis
France Telecom

Email: pierre.levis@orange-ftgroup.com

Alain Villefranque
France Telecom

Email: alain.villefranque@orange-ftgroup.com