Behavior Engineering for Hindrance

IPv6 IPv4 translation FTP considerations
draft-liu-behave-ftp64-03.txt

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on February, 2010.

Copyright Notice

Copyright (c) 2009 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents in effect on the date of publication of this document (http://trustee.ietf.org/license-info). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.
Abstract

The File transfer protocol, which is defined by the RFC 959, is widely used. RFC 2428 define IPv6 extensions of FTP, introducing EPRT and EPSV command.

In the IPv6-IPv4 translation scenario, considerations should be applied to FTP client, server and translation box to ensure FTP protocol work properly. There already have some work to address this problem, such as "draft-van-beijnum-behave-ftp64-05" etc, but this document provides a different approach.

Table of Contents

1. Introduction...................................................3
2. Conventions used in this document............................4
3. Client considerations........................................4
4. Server considerations........................................4
5. ALG considerations..........................................4
6. Existing solutions and comparison............................5
7. Security Considerations......................................6
8. IANA Considerations..........................................6
9. Acknowledgments.............................................6
10. References................................................6
    10.1. Normative References................................6
    10.2. Informative References.................................7
Author’s Addresses............................................7
1. Introduction

Figure 1 illustrated the IPv6-IPv4 translation FTP scenario.

The IPv6 FTP client situated in an IPv6 network and tries to communicate with an IPv4 server that situated in an IPv4 network using a translation box in the middle.

FTP has two operation modes: passive mode and active mode. In passive mode, the server provides port used for the client to connect to. In active mode, the server connect back to the client, using the IP address and port number which provide by the client.

RFC 2428 specifies IPv6 extension of FTP. Two new commands, EPRT/EPSV are specified. The EPRT command is an extension of PORT, it could provide IPv6 address and port number to the server. The EPSV command is an extension of PASV, when issue this command, the server should responses its port number used for the client to connect.

Many serves do not support EPSV command, but most of them could support PASV mode (draft-van-beijnum-behave-ftp64-05). This document provides guidelines for client and server to avoid the problems that IPv6 FTP client communication with an IPv4 server through a translation box.
2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119.

3. Client considerations

It is required that all IPv6 FTP clients MUST support both EPSV and PASV command.

When the client tries to connect to a server using IPv6 connection, it should use EPSV command first. If the server response that it does not support this command or encounters an error, it MUST retry with PASV command. The server will respond to PASV command with a message that contains an IPv4 address and port number that used for the client to connect to. The client MUST ignores the IPv4 address provided in the response; it should use the control connection’s IP address to connect to the server to establish the data connection.

This approach could not only simply the FTP client software’s implementation but also can avoid the problems caused by using the IPv4 address that included in the response message. For example, if the FTP client has a private IPv4 connection and a public IPv6 connection, if it tries to use the IPv4 connection to establish data connection with the server, it will never success.

4. Server considerations

All FTP servers MUST support EPSV and PASV command. All FTP severs MUST could respond with error message to EPSV command if it does not support it. The FTP server MUST allow the client to retry with PASV command when it fails with EPSV command. Also, the server must allow the client to use the control connection’s IP address to establish data connection when it retries with PASV command.

5. FTP ALG considerations

5.1. FTP ALG limitations

Implementing FTP ALG in the translation box may have some limitations, such as:

1) FTP ALG may case to increase the complexity of translation box, since FTP ALG needs to understand FTP protocol and translate the application layer payload and update the header of FTP control packets.
2) For most of current Network Processor based translation box, ALG processing may cause its performance decreased significantly. The reason is that FTP ALG processing can not take the advantage of Network Processor, which is designed and optimized for processing regular packets (such as header translation).

3) From the evolution perspective, if the network continues to provide support of FTP ALG all the time, the users will have no motivation to upgrade their FTP client software. If things evolve toward this direction, the ALG function of the translation box will become more and more complex. In reality, upgrading FTP client software is a more easy way to solve the ALG issue compared with requiring that all the translation box to implement FTP ALG.

5.2. Solution to avoid FTP ALG

This document suggests that the translation box which situated between the IPv6 network and IPv4 network should not implement FTP ALG. It is depend on the client and server that comply with this specification to avoid the FTP ALG issue.

The reason that this document does not encourage translation box to implement FTP ALG is that since the FTP ALG problem can be totally avoid by defining the behavior of the client and server, it is not necessary to implement it at all. This approach can reduce the translation box’s complexity. Also, the FTP client and server’s communication without ALG will significantly improve its performance.

For the legacy IPv6 FTP client, this document suggests that the legacy client should behave like RFC1579 recommendation which is that "vendors convert their FTP clients programs to use PASV instead of PORT". For IPv6 FTP client, this recommendation should be "the users or vendors should convert their FTP clients programs to use EPSV instead of EPRT". This can be done by configuration and do not require upgrading of the client software for most of current IPv6 FTP client. This solution may require that the FTP server should support EPSV command.

6. Existing solutions and comparison

[I-D.draft-van-beijnum-behave-ftp64-05] provides a solution that addresses same problem as this document, the major differences between the two approaches are:

1. ALG considerations of the translation box
[I-D.draft-van-beijnum-behave-ftp64-05] does not speak out in favor or against the deployment of an FTP application layer gateway. However, this document specifies that the translation box should not implement FTP ALG.

The main concern of not recommending ALG is that FTP ALG could dramatically decrease the performance of the translation box due to the stateful application layer processing. ALG could be avoided by the FTP client and server’s implantation that complies with this document. The argument here is that it is much easier for the client/server software to upgrade than implementation of ALG in the translation box. Eliminating the ALG function in the translation box will simply the protocol operation and avoid unexpected errors.

2. Behavior of FTP client when retrying with PASV command

[I-D.draft-van-beijnum-behave-ftp64-05] recommends that the client should use the IPv4 address in the PASV response message if it has IPv4 connectivity and use the control connection’s IP address if it does not have IPv4 connectivity. However, this document specifies that the client should use the control channel’s IP address without determination whether it has IPv4 connectivity or not. This will simplify the client software, besides, if the client has IPv4 connectivity, the control channel will use its IPv4 address instead of using its IPv6 address to connect to the IPv4 server. This approach can avoid the problems that maybe caused by using the client’s IPv4 connection as described in section 3.

7. Security Considerations

TBD

8. IANA Considerations

None

9. Acknowledgments

TBD

10. References

10.1. Normative References


10.2. Informative References


Author’s Addresses

Dapeng Liu
China Mobile research institute
Unit2, 28 Xuanwumenxi Ave, Xuanwu District,
Beijing 100053, China

Phone: (8610)13911788933
Email: liudapeng@chinamobile.com

Zhen Cao
China Mobile research institute
Unit2, 28 Xuanwumenxi Ave, Xuanwu District,
Beijing 100053, China

Phone: (8610)15120015799
Email: caozhen@chinamobile.com