PORT KNOCKING SENSITIVE NAT AND ITS TRAVERSAL METHOD

YU-CHUN HWANG[1], WEI-YUAN CHENG[2], YU-TING HWANG[3], HONG-YI LIAO[4] AND CHAO-RONG CHEN[1]

[1]Department of Electrical Engineering, National Taipei University of Technology, Taipei, TAIWAN, ROC
[2]ICT Integration Department, Chunghwa Telecom Laboratories Taoyuan, TAIWAN
[3]Department of Electrical Engineering, National Chung Hsing University, Taichung, TAIWAN, ROC
[4]Department of Electrical Engineering, National Yang Ming Chiao Tung University, Hsinchu, TAIWAN, ROC
E-mail: lovelive8911@gmail.com ,markov@cht.com.tw, huangxiaode22@gmail.com,
xul6106@gmail.com, crchen@ntut.ee.edu.tw

Abstract:

In real-time video streaming systems, NAT(Network Address Translator) traversal technology is very important, as it can significantly reduce system operating bandwidth costs.

In past studies, four NAT classifications and nine NAT classifications were proposed for NAT behavior. At the same time, STUN, TURN, P2P, ICE, ALG, C2C, etc. protocols were proposed to implement NAT traversal. Although the above NAT classifications and traversal methods have effectively solved the current problems, there are still unknown NAT classifications waiting to be discovered and traversed.

This paper discovers a new NAT behavior for the first time and proposes an effective NAT traversal method. This NAT behavior is called Port Knocking Sensitive (PKS) NAT. The PKS NAT will block the mapping port for external active invading packets, which helps prevent DDoS(Distributed Denial of service) attacks. Regardless, this newly discovered NAT behavior will render existing NAT traversal techniques inoperable.

This paper also proposes a Synchronous Client to Client(SC2C) NAT traversal protocol technology for the first time. This protocol first requires both parties to test and predict the Mapping Port of their own NAT and notify the other party. The server then coordinates both parties to send NAT traversal packets synchronously, which can successfully traverse this newly discovered NAT. The experimental results prove that this new NAT does block the existing NAT traversal protocol, and also confirm that the SC2C NAT traversal protocol proposed in this paper can indeed effectively traverse this newly discovered NAT.

Keywords:

Video Streaming, NAT Traversal, Port Knocking Sensitive(PKS), Synchronous Client to Client(SC2C)

1. Introduction

The popularity of the Internet has led to a variety of network applications, especially those that require high-bandwidth image streaming, such as video streaming, online video conferencing, real-time video communication, real-time video monitoring, etc. In addition, although IPv4 network addresses only use four digits, which is completely insufficient to supply the huge number of users around the world, IPv4 is still the mainstream standard in the world. Therefore, NAT[1] was born to meet the needs of the times and solve the problem of insufficient IP locations. However, NAT also brings about the problem of network inaccessibility, especially for point-to-point video streaming applications, which has a huge impact.

NAT uses IP address and port conversion technology to allow many virtual IP users to share one real IP address to solve the problem of insufficient IP locations. Under NAT(NAT1), virtual IP users can still enable the Client to Server (C2S) service, but users in the real Internet environment, or users in another different NAT(NAT2) environment, cannot actively connect directly to users in the NAT1 environment.

In many video streaming applications, real-time video and audio intercommunication applications such as: Video Surveillance, Video Phone, ... etc., video and audio packets will be blocked by NAT devices, resulting in the failure of video and audio streaming protocols including: RTP, RTSP, RTCP, RTMP, SIP, H.323, ... etc.

In previous research, TURN[2] proposed using server switching technology to enable audio and video communication between two parties in different NAT environments. This approach indeed solved the problem of audio and video communication being unable to be communicated due to NAT, but it also brought about the problem of requiring huge operating bandwidth costs.

In previous studies, ICE[3], uPnP[4], ALG[5] and others proposed NAT traversal methods, which directly implement algorithms on NAT devices. This method enables direct communication between two parties under different NATs without the need for additional servers. However, these methods must be built on a special NAT that supports ICE, uPnP, and ALG.

In previous studies, STUN[6~7] used STUN Server to observe NAT Port and exploited un-rigorous NAT to achieve direct communication between two parties under different NATs without the need for additional servers. However, this method still cannot traverse the rigorous Symmetric NAT. Furthermore, due to security considerations, almost all NATs currently on the market are Symmetric now.

In previous studies, C2C[8~12] used the server to help predict the NAT port change rules of its own end, and then informed the other party of the predicted port, so that the two parties under different NATs can communicate directly without the need for additional servers. This method can also traverse Symmetric NAT.

In the past, there have been many studies on NAT behavior analysis and classification. Among them, STUN[6~7] divides NAT into four categories. Although the STUN method cannot traverse the most rigorous Symmetric NAT, the C2C method can successfully traverse it. Takeda[13] also proposed nine NAT classifications, and [14-18] also proposed various NAT behavior analyses and classifications. Among them, the Asymmetric NAT proposed by [14] is similar to the nine categories proposed by [13], and also proposed a NAT traversal method. [9,18] proposed ICMP Packet Sensitive(IPS) NAT behavior and related traversal methods.

This paper proposes a new NAT, called Port Knocking Sensitive(PKS) NAT, and a new NAT traversal method, called Synchronous Client to Client(SC2C) NAT Traversal Protocol, which will be described in detail in the next chapter.

2. The Proposed System

The SC2C-based NAT traversal protocol proposed in this paper is shown in Figure 1 below. The figure includes: the server, the calling end, the called end, and the NATs of both ends. In order for the calling and called parties to complete NAT traversal and directly communicate with each other through video streaming, they must go through four procedures of the SC2C protocol, including: registration procedure, port prediction procedure, synchronization procedure, and NAT traversal procedure, which are described in detail below:

2.1. Registration Session

The registration process has two main purposes: first, authentication, and second, pre-establishing a connection to facilitate communication between users, especially parameter transmission between two points of the SC2C protocol. The registration program uses TCP connection and maintains a online connection to avoid affecting the subsequent port prediction program and interfering with the prediction of NAT Port. The registration process is not the focus of this paper, so the security mechanism and the connection maintenance mechanism are not considered. The detailed registration packet is shown in steps 1 to 4 of Figure 1. The contents of the registration packet are as follows:

Step 1: Client “John” sends a registration message “REGISTER|John” to the server

Step 2: Server responds with a successful registration message "REGISTER_200|John" to "John"

Steps 3 and 4 are the registration process for "Peter".

After the registration process is completed, control signal of video streaming and NAT traversal commands, such as call commands (INVITE, INVITE_200), RTP synchronization commands (SYN, SYN_200), etc., will be transmitted and exchanged through this registration process. As long as the registration process maintains a regular connection status, NAT will not configure a new port when transmitting and exchanging messages to avoid affecting and interfering with subsequent NAT Port predictions.

2.2. Port Prediction Session

Because NAT traversal has a higher success rate in UDP mode, it is almost impossible to succeed in TCP mode. Furthermore, each terminal does not know the NAT status of its own location, so the terminal needs the help of the server to inform the IP and Port of the NAT it sees. In addition, NAT traversal is a Client to Client (C2C) architecture, because both communicating parties are clients and both actively send packets to each other. This communication mode is different from the traditional Client to Server (C2S) architecture. Therefore, both parties in the C2C architecture must send out completely symmetrical packets simultaneously. The so-called complete symmetry means that the Source IP and Port of the packets of both parties are exactly the same as the Destination IP and Port. If both parties are in a real IP environment, it is very easy for both parties to send completely symmetrical packets under the C2C architecture. However, if both parties are under different NATs, it is very difficult for both parties to send completely symmetrical packets under the C2C architecture because the Source Port of the packets of both parties is controlled and configured by NAT, which is uncontrollable by the client parties. Although the Source Port of the client packet after traversing NAT cannot be controlled, it can be observed and predicted. This section will describe the port prediction process in detail.

The main purpose of the port prediction program is to predict the NAT communication port in your environment, and then inform the other party of the predicted communication port through the registration connection, so that the other party can send UDP packets to traverse your own NAT and achieve the goal of NAT traversal.

When the calling party intends to communicate directly with the called party, both parties must first test and predict the NAT Port change rules in their own environment, and predict the Port communication port that their NAT will use for the next new UDP session. Then, they will inform each other of their own NAT IP and predicted Port value, and both parties can enter the NAT traversal process. The detailed steps 5~6 of the Port Prediction Session are as follows:

Step 5: Client "John" sends a new UDP packet "TEST|John" to the server. The Source IP and Port of this packet are (IP1,), and the Destination IP and Port are (IPC,C). Because it is a new UDP session, NAT-1 will configure a new Port value p1, so the Source IP and Port of the packet will be changed to (IPA, p1).*

Step 6: After receiving the test packet from the Client, the Server will respond to the Client "John" with the Source IP and Port value (IPA, p1) of the packet. The packet content is "TEST_200|IPA|p1". Where IPA is the IP address of NAT-1 where "John" is located, and p1 is the modified Source Port of NAT-1.

Steps 5 and 6 are executed repeatedly. Step 5 will send a new UDP program whose Source Port is completely different from the previous UDP session. Therefore, NAT will configure a new Source Port. The user end can predict the next port value (p3) of NAT by repeating steps (5,6), (7,8),... Then, through steps 9 and 10, the IP and port value (IPA, p3) of the NAT on the own end are passed to the called end. The detailed steps are listed below:

Step 9: The client sends its own NAT IP (IPA) and predicted port (p3) to the server. The packet content is "INVITE|John|Peter|IPA|p3".

Step 10: The server transfers the data sent by "John" in step 9 to "Peter".

After receiving the data from "John" in step 10, the called party "Peter" immediately start port prediction which is as same as steps 5~8 at steps 11~14.

2.3. Synchronization Session

In a C2C communication architecture, even if the two communicating parties send completely symmetrical packets, due to lack of synchronization, one party may deliver the packet to the other party's NAT door earlier and the packet may be blocked or even returned, resulting in NAT traversal failure in the C2C communication architecture.

Figure 2 below lists two actual cases. In Case-1, NAT-2 is a Port Knocking Sensitive(PKS) NAT. When the packet arrives at NAT-2 in advance, NAT-2 will block the knocked port (q3). The new UDP session sent subsequently will skip Port (q3) and use other Ports (q4) to avoid the risk of being attacked by DDOS packets, which will cause the NAT traversal in the subsequent step 21 to fail.

In Case-2, NAT-1 is an ICMP (Internet Control Message Protocol) packet-sensitive NAT. When the packet arrives at NAT-2 in advance, NAT-2 will usually return it with an ICMP packet. NAT-1 is an ICMP Packet Sensitive(IPS) NAT. After receiving the ICMP packet, it will immediately close the used Port (p3) and no longer use it. This behavior is designed for network security to avoid hacker intrusion, but it also causes the subsequent step 21 NAT traversal to fail.

Therefore, under the C2C communication architecture, both parties must synchronize their time first and then send out symmetric packets together to successfully complete NAT traversal. Therefore, the main purpose of steps 17 to 19 is to synchronize the two communicating parties and send out symmetric UDP packets at the same time to allow UDP to successfully traverse NAT. The detailed description is as follows:

Step 17: The client sends a synchronization packet "SYN|John|Peter|IPA|p3|IPB|q3" to the server, where (IPA,p3) is the IP and port predicted by Client "John" and (IPB,q3) is the IP and port predicted by Client "Peter".

Step 18: The server sends a synchronization packet "SYN_200|Peter|IPB|q3" to the client "John". After receiving the packet, the client "John" immediately sends a NAT traversal packet in step 20.

Step 19: The server sends a synchronization packet "SYN_200|John|IPA|p3" to the client "Peter". After receiving the packet, the client "Peter" immediately sends a NAT traversal packet in step 21.

2.4. NAT Traversal Session

The NAT traversal program uses the UDP protocol, with both communicating parties sending symmetric packets to the other party's NAT. As long as the original and destination IP and Port of the packets of both parties are symmetric, and they are sent synchronously, they can traverse each other's NAT and achieve the goal of intercommunication. This communication method is the C2C mode, which is different from the traditional C2S mode. Steps 20~21 will be explained in detail below.

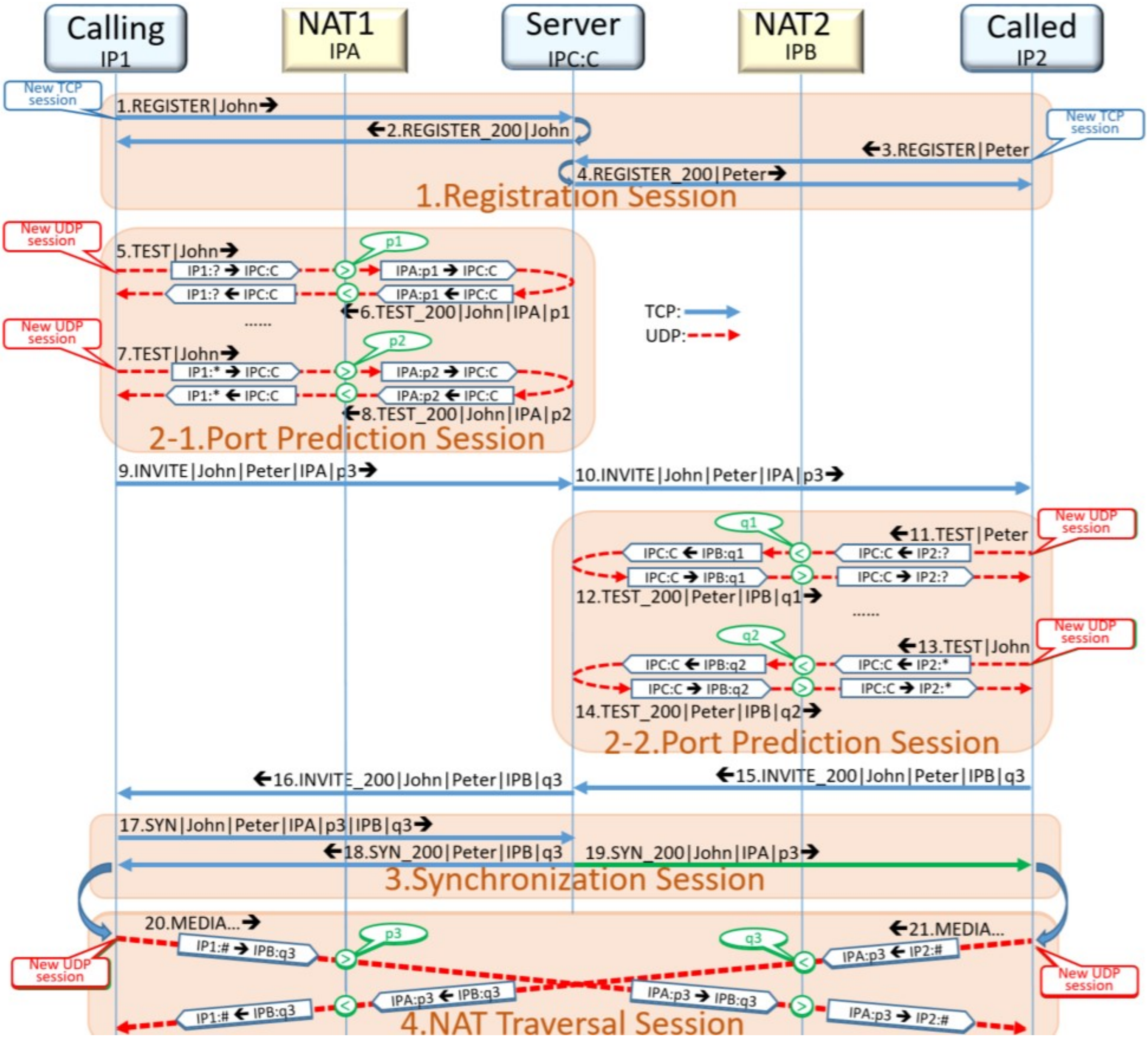

Fig.1 The Synchronous Client to Client(SC2C) Protocol.

Step 20: Client "John" sends a new UDP packet to NAT-2 where Client "Peter" is located. The destination IP and Port of the packet are (IPB,q3). After the packet passes through NAT-1, the original IP and Port of the packet will be changed to (IPA,p3).

Step 21: Client "Peter" sends a new UDP packet to NAT-1 where Client "John" is located. The destination IP and Port of the packet are (IPA, p3). After the packet passes through NAT-2, the original IP and Port of the packet will be changed to (IPB, q3).

Steps 20 and 21 will send out new UDP packets at the same time. These two UDP packets are completely symmetrical, so they can complete NAT traversal and establish a client to client(C2C) communication channel. Then the RTP video packet can be transmitted through this C2C channel, completely eliminating the bandwidth operation cost of forwarding through the server.

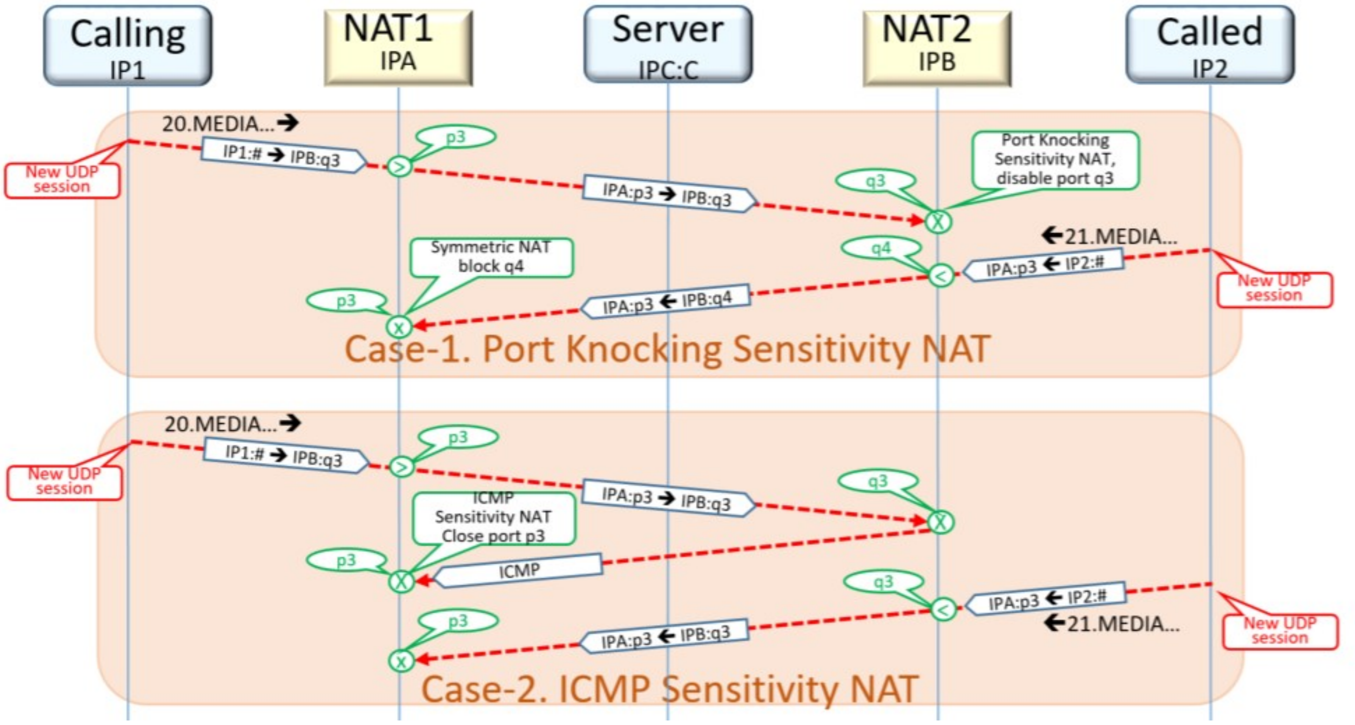

Fig.2 Two cases of NAT Traversal.

3. Experimental Results and Discussions

This paper proposes a synchronous C2C NAT traversal technology and uses 134 commercially available NAT models for experiments. Table-1 lists the classification results of the 134 NAT models based on STUN protocol.

Table.1. NAT type statistics based on STUN protocol.

NAT Type(STUN)	Number	Percentage
Full Cone	20	14.9%
Restricted Cone	31	23.1%
Port Restricted Cone	79	58.95%
Symmetric Cone	4	2.98%
Total	134	100%

From the above table, we can see that according to the STUN standard classification, most NATs belong to PRC, among which Symmetric NAT cannot be traversed by the STUN method. The synchronous C2C NAT traversal technology proposed in this paper is applicable to all NAT types, including Symmetric NAT.

For new UDP or TCP Session, NAT will configure a new Port Number. Each NAT has different rules for configuring new Ports. Table 2 below lists the statistics of NAT Port change rules. Most of them are +1, so the success rate of Port Prediction Session is very high. Only three are Random, including: AboCom (FSM410), ASUS (Rx3081) and Octtle (SP4220).

Table.2. NAT Port change rule statistics

Port Changed Rule	Number	Percentage
+1	126	94.03%
+2	5	3.73%
Random	3	2.24%
Total	134	100%

In the analysis and statistics of Port Knocking Sensitivity(PKS) and ICMP Packet Sensitivity(IPS), only 28 of the 134 NAT models belong to PKS NAT, and 2 belong to IPS NAT. The brand names of only the 2 IPS models are IO DATA (NP-BBRM) and AboCom (CAS4047B). The 28 PKS models include brands such as AboCom*1, ASUS*1, D-Link*11, EDIMAX*7, GigaByte*1, SAPIDO*5, SMC*1, and ZyXel*1. Table.3 list the statistics number of PKS and IPS NAT.

Table.3. NAT behavior statistics

NAT Behavior	Number	Percentage
PKS	28	20.90%
IPS	2	1.50%
Others	104	77.61%
Total	134	100%

In order to verify the performance of the synchronous C2C protocol, three NATs were selected for testing. The following table lists the success rate of the interaction test. Each case was tested 100 times. SC2C is the synchronous C2C proposed in this paper, and C2C is the asynchronous C2C with steps 17 to 19 deleted. The data in the table shows that synchronous C2C has good performance for IPS and PKS.

Table.4 Successful Rate of NAT Traversal with C2C and SC2C protocols.

Called Calling	Others		IPS		PKS	
	SC2C	C2C	SC2C	C2C	SC2C	C2C
Others	98%	90%	96%	10%	98%	92%
IPS	99%	92%	96%	5%	98%	90%
PKS	97%	1%	99%	2%	96%	2%

4. Conclusions

This paper discovers the new behavioral NATs for the first time, names as Port Knocking Sensitive(PKS) NAT. This PKS NAT will make many NAT traversal methods proposed in the past invalid. This paper also proposes a synchronous C2C(SC2C) NAT traversal protocol for the first time, which enables NAT traversal to be successful. Experimental data also confirms the superiority of the synchronous NAT traversal protocol.

Acknowledgements

This paper is supported by the ICT Integration Department, Chunghwa Telecom Laboratories Taoyuan, TAIWAN. The project No. is TL-113-K201.

References

[1] P. Srisuresh, and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC-2663, Aug. 1999.

[2] R. Mathy, P. Matthews, and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", RFC-5766, Apr. 2010.

[3] A. Keranen, C. Holmberg, and J. Roseberg, "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal", RFC-8445, Jul. 2018

[4] M. Boucadair, R. Penno, and D. Wing, "Universal Plug and Play (UPnP) Internet Gateway Device - Port Control Protocol Interworking Function (IGD-PCP IWF)", RFC-6970, Jul. 2013

[5] I. van Beijinum"An FTP Application Layer Gateway (ALG) for IPv6-to-IPv4 Translation", RFC-6384, Oct. 2011

[6] J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy, "STUN – Simple Traversal of User Datagram Protocol(UDP) Through Network Address Translators(NATs)", pp.1~47, Network Working Group, IETF, RFC3489, Mar., 2003.

[7] J. Rosenberg, R. Mahy, P. Matthews, and D. Wing, "Session Traversal Utilities for NAT (STUN), IETF RFC-5389, Oct. 2008.

[8] Shaw-Hwa Hwang and Cheng-Tu Yeh, "Session Traversal Utilities for Network Address Translator (STUN)-based Traversal Approach Using Port Assignment Prediction Mechanism", Sensors and Materials, Vol.34, No.5 pp.1791-1801, 2022.

[9] "Traversal method for ICMP-sensitive NAT", US Patent No.9042376, 2015.5.26

[10] "SIP Communication Protocol", US Patent No.8700785, 2014.4.15

[11] "NAT traversal method in session initial protocol", US Patent No.8676933, 2014,3,18

[12] "Modified NAT firewall traversal method for SIP communication", US Patent No.7751387, 2010.7.6

[13] Y. Takeda, "Symmetric NAT Traversal using STUN", pp.1~23, Internet Draft, IETF, Jun., 2003

[14] Chi-Long Huang, Shaw-Hwa Hwang, "The Asymmetric NAT and Its Traversal Method", 2009, IFIP, Apr. 2009

[15] S. P. Shieh, F. S. Ho, Y. L. Huang, and J. N. Luo, "Network Address Translators: Effects on Security Protocols and Applications in the TCP/IP Stack", IEEE Internet Computing, pp.42~49, Nov. 2000.

[16] A. Muller, and G. Carle, "Behavior and Classification of NAT Devices and Implications for NAT Traversal", Technische Universitat Munchen, Andreas Klenk, Universitat Tubingen, 2008.

[17] F. Audet and Jennings C, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP," IETF RFC 4787, Jan. 2007.

[18] P. Srisuresh, B. Ford, S. Sivakumar, and S. Guha, "NAT Behavioral Requirements for ICMP," BCP 148, RFC 5508, Apr. 2009