TRAM Working Group
Internet Draft

A.Wang
China Telecom
B.Liu
Huawei Technologies
J.Uberti
Google

Intended status: Standard Track
Expires: January 7, 2017

July 7,2016

**Operator-Assisted Relay Service Architecture(OARS)**
**draft-wang-tram-oars-00.txt**

Status of this Memo

This Internet-Draft will expire on January 7, 2009.



Abstract

This document proposes a new relay-based NAT traversal architecture called OARS which could simplify the data communication process between two hosts that locates behind some non-BEHAVE compliant [RFC4787] [RFC5382] NAT devices. The key mechanism in OARS is the newly defined "Couple" operation (using STUN [RFC5389] message format) which allows the Relay servers to be easily incorporated into existing CGN/CDN nodes which are already deployed within the network in a distributed manner.

Table of Contents

# 1. Introduction

## 1.1. Motivations

This document proposes a new relay-based NAT traversal architecture called OARS based on the following motivations.

1) Leverage ISPs' infrastructures

Currently, the deployment of TURN [RFC5766] is very limited and most of the application providers use their own platform to transfer the data between two hosts that behind NATs and to translate the communication packets between two hosts in different address families.

The relay devices deployed centrally by various application providers often lead to inefficient data transmit between two hosts and it must deal with complex network layer problems which the application providers are not familiar with.

On the other hand, service providers have deployed many CGN/CDN nodes in a distributed manner within their networks. If the service providers can use these CGN devices/CDN nodes as the relay devices for communication between two hosts behind NATs or that from different address family, and provide their data translation/forwarding capability to the application providers, the host to host communication will be more efficient. Given most of the CGNs are capable of translating packets between IPv4 and IPv6, the adoption of IPv6 technology will also be accelerated.

2) Simplify the communication procedures

OARS needs less communication procedures than TURN of which the procedures are considered very complex to be integrated into the ISPs' infrastructure, for example:

o TURN solution has to closely interact with ICE
Within current TURN solution, there are scenarios where the ICE or other NAT-hole punching procedures must be included for the success of communication via TURN servers. The key point is that TURN allocates different relay transport address-port

pairs for different hosts.

Each client must first use TURN allocation request to get their transport relay address-port pairs, and then must use ICE procedure (connectivity check) or other similar signaling to punch holes for these transport relay addresses on the alongside NAT devices. Or else the relayed UDP/TCP packet will be blocked.

Even with the above procedures, there still exist some risks that the packets be rejected by TURN server due to the permission list that created by client via "CreatePermission Request" before it sending data to the peer. In order to mitigate such issues, current TURN solution requires the TURN servers only check the IP address part of the relay transport address, and ignore the port portion. But this will again introduce some attack risks because different host may share one public IP address when the CGN device is deployed within network.

o IPv4/IPv6 Relay Address/Port Reservation and Allocation

Another drawback of different relay transport addresses for different host is that the TURN server must reserve some IPv4/ IPv6 address block for the allocation and plan the TCP/UDP port usage for each host. When TURN servers are deployed in a distribute manner (For example when they are incorporated into the CGN devices), there will be much coordination work to do for the address/port reservation and allocation on the TURN servers.

o Simultaneous TCP/UDP connections burden on TURN server

Current TURN solution requires the TURN servers to open and listen on many TCP/UDP ports at the same time, Under TURN solution for TCP[RFC6062], each host requires two connections to the TURN server. This will increase the burden on TURN server and the complexity to incorporate them into the CGN/CDN devices.

o Different procedures for TCP/UDP communication

Current TURN solution adopts different procedures for the TCP and UDP communication channel. So for one TURN server to provide the TCP/UDP relay function, it must implement two different procedures. This again increases the complexity of the TURN server implementation, especially in CGN devices.

o Communication complexity between two different TURN servers

Current TURN solution cannot assure two hosts select the same TURN server, and then it must deal with the communication situation between two different TURN servers. This scenario has not been covered by the current TURN related drafts. The client must reuse the XOR-PEER-ADDRESS attribute to include the relay address of the peer to reach the second TURN server.

On the other hand, because the hosts select their own TURN server, there is no mechanism to assure the relayed path is most optimal for them. The application latency will be increased when this occurs.

OARS solution will simplify the above mentioned complexity and make the TCP/UDP data relay function be easily incorporated into the existing distributed CGN devices or other kinds distributed devices i.e. the CDN nodes etc.

## 1.2. Relationship with TURN

This document doesn't intent to replace TURN with OARS, but consider OARS as a complementary solution along with TURN for some specific scenarios.

If one SP wants to open its infrastructure to accelerate their customers' (mainly regarding to application providers) client-to-client communications within the SP's domain, OARS could be a good candidate.

## 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 [RFC2119].

O Relay Selector: which is in charge of selecting a proper relay device (CGN or CDN nodes) for the communicating hosts behind NATs. Normally, the RS is a function located in the network's management plane and possibly a part of the NMS server

O OARS: Operator-Assisted Relay Service. Compared with the relay services that implemented independently by each TURN client, OARS

can simplify the relay procedures in central control mode via the assistance of network operator.

- OARS Client: The client that initiated the ''Couple'' command to bind two TCP/UDP sessions on one relay device or two different relay devices.

.

- OARS Server: The server that implemented the ''Couple'' command to bind two TCP/UDP sessions on one relay device or two different relay devices.

## 3. Solution Overview

### 3.1. Reference Architecture of OARS

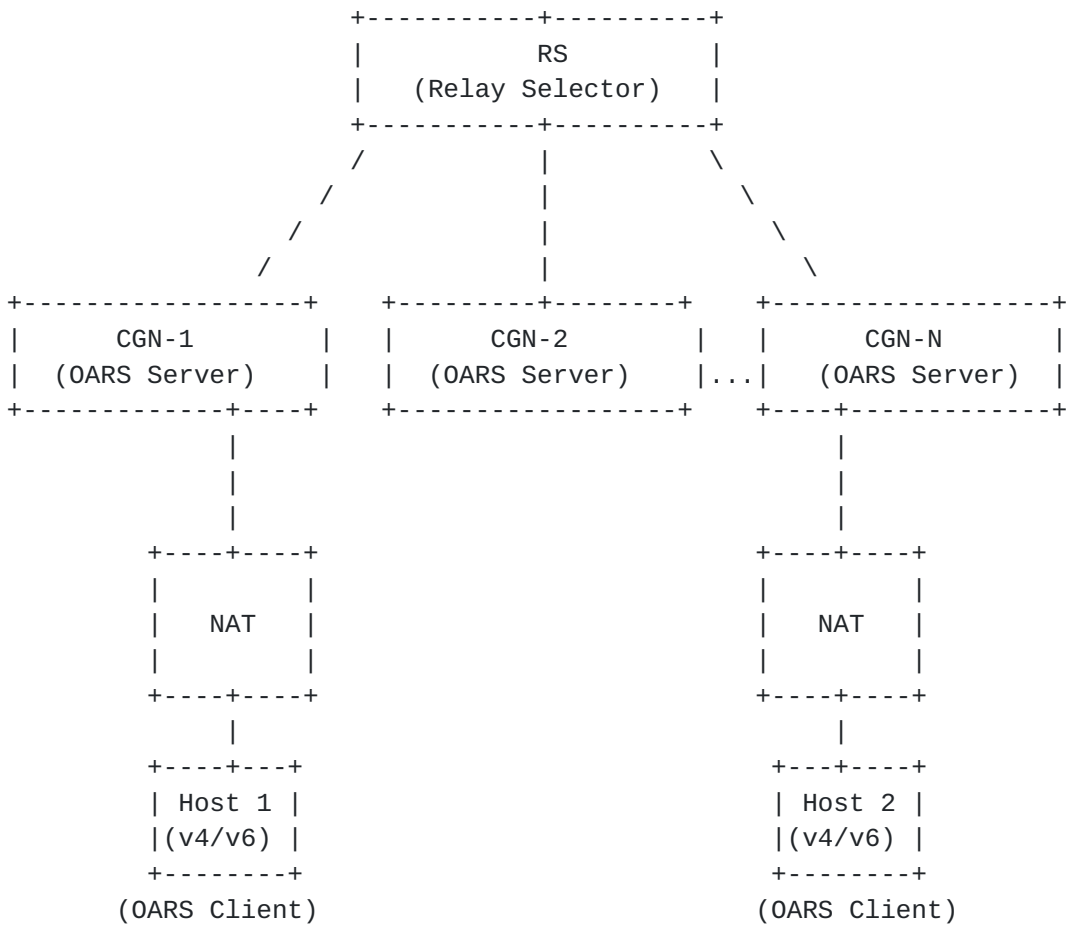



Fig. 3-1: OARS Architecture

As depicted in above figure, the application clients that located on hosts act as the OARS clients while the CGNs act as OARS Servers. There is a Relay Selector (RS) for choosing a proper CGN to relay traffic between the two hosts. In practice, the RS could be a dedicated server or a function located in the management plane servers such as NMS server.

RS has the intelligent route selection capability to choose a proper CGN for a given host pair. RS sends the data relay indication to the selected CGN devices/CDN node via the newly defined "Couple" method.

BEHAVE compliant and non-BEHAVE compliant NAT traverse [RFC4787] [RFC5382] is supported in OARS. IPv6 and IPv4 host communication is also supported.

### 3.2. Solution Rationale

The solution could be briefly described in the following sections.

#### 3.2.1. Relay Selector Reflection and Selection

Each host that wants to communicate with the other host should send STUN message to the RS (Relay Selector), and get their reflex addresses to the RS (here we refer to REFLX-RS).

The application provider needs to select a suitable RS and informs it to the hosts (e.g. via application specific client-server protocol). The detailed RS selection mechanism and criteria are out of the scope of this document, but some general considerations are as the following.

- If the hosts locate in the same ISP/administrative domain, then the RS selection is fairly easy since normally there is only one RS in one ISP; even there are multiple RSes in one ISP, the application provider should also be able to select a suitable RS based on the addresses of the two hosts.

- If the hosts locate in two ISPs/administrative domains (assuming both of the ISPs providing OARS service), the application provider can select one RS based on pre-defined policies (the simplest way is just arbitrarily choosing one RS in one of the ISPs).

- The application provider can also select two RS to deal with the communication between two hosts that located in different service

provider. Under such situation, the application provider will send one extend ''Couple'' command to each RS, let the RS tunnel the customer's data to another RS. The detail process of this situation will be provided further. Currently, we focus only the one ISP scenario.

### 3.2.2. Relay Selection

If two hosts want to communicate, one of them will send the two hosts' REFLX-RS addresses to the selected RS, to let the RS select one appropriate relay device to relay the traffic.

Generally, the RS can select the appropriate relay device based solely on the REFLX-RS addresses of these two hosts, for example, select one relay device that locate in the middle of the communication path. This approach is possible since the relay behavior is within one ISP's domain that the RS could be possible to learn the knowledge of all CGNs (relays) within that domain.

The selection criteria can also be expanded to include other factors, such as the privacy concern of the communication peers, the linkage usage information between the host and the relay device etc. For example, RS can select one relay device that locates far from the communication peer to hide the location of the peer. This might sacrifice the communication efficiency but increase the protection of the host privacy. Anyway, RS has more flexible control over the relay selection, upon the requirement of communication hosts, or the consideration of relay service provider.

After the relay device selection, the RS will respond the IP address of the selected relay device to the communication peer, together with the well known port that used by every relay device. The combination of this relay IP address and the well-known port form the relay transport address of the communication peers, each peer will use this relay transport address to communicate.

When two hosts located within one administration domain, the centralized relay point selection and control architecture can easily achieve one low latency communication path because it knows the whole network condition of its own. When two hosts located within different administration domains, the OARS solution will also work except that some end-to-end communication efficiency might be sacrificed unless there is some coordination between these two administration domains.

### 3.2.3. Forming "Couple" Command

Each host will send again one STUN message to the selected relay transport address, get the new reflex address(here we refer to REFLX-Relay) to the selected relay device, and reports it to the RS, together with the previous reflex address to the RS (which is REFLX-RS).

The RS will use the REFLX-RS addresses to find out which two peers will communicate (such communication pair information is gotten from Section 3.2.2). RS will retrieve the corresponding REFLX-Relay address of the communication peer, forms the "Couple" command based on such information, and sends the "Couple" command to the selected relay transport address.

Upon receiving the "Couple" command, the relay device will add one item to its forwarding table. The forwarding table will bind the reflex addresses of the two peers, the required transport protocol and other additional information.

### 3.2.4. Data Relay

Each host will then send the data traffic directly to the unique relay transport address. The source address of this packet will be changed by the alongside NAT devices that located between the host and the relay device.

When this packet arrives to the relay address, its source address will be one of the RFLEX-Relay addresses. The relay device will search the forwarding table that formed in Section 3.2.3. If the REFLX-Relay address in one item match the source address of the received packet, then the other REFLX-Relay address will be retrieved and be used as the destination address of the application packet, the packet's source address will be changed to the relay transport address.

After the conversion, the packet will be sent by the relay device. This packet will be routed to the corresponding peer, according to its REFLX-Relay address.

The application return packet will be sent again back to the same relay device via the relay transport address. The similar search process and convert work will be done by the relay device. The converted return packet will then be routed to the packet originator.

## 4. New STUN Method Definition

In order to let the CGN device to build one Couple item upon the request of RS, it is needed to define one general Couple message to transfer the related information.

### 4.1. Couple Operation

The Couple request defines the relationship between two TCP or UDP half-connections, the translation rule that converts both the source address and destination address of pass through packet in both directions.

Couple Opcode: It defines how to bind two half-connections that ends at the CGN's well-known relay transport address together. When CGN device receives the Couple request, it will create one map table item that includes the reflex IP address/port [REFLX-Relay] of both hosts that lies behind the NAT device and the protocol that the host will use to communicate.

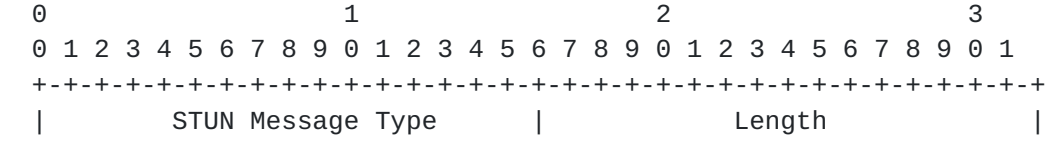
When the CGN device receives the packet from one host, it will use the reflex source address/port to lookup the map table item; converts the source address/port of this packet to the relay transport address of the CGN device and converts the destination address/port of this packet to the reflex address [REFLX-Relay] that results from the map table lookup action.

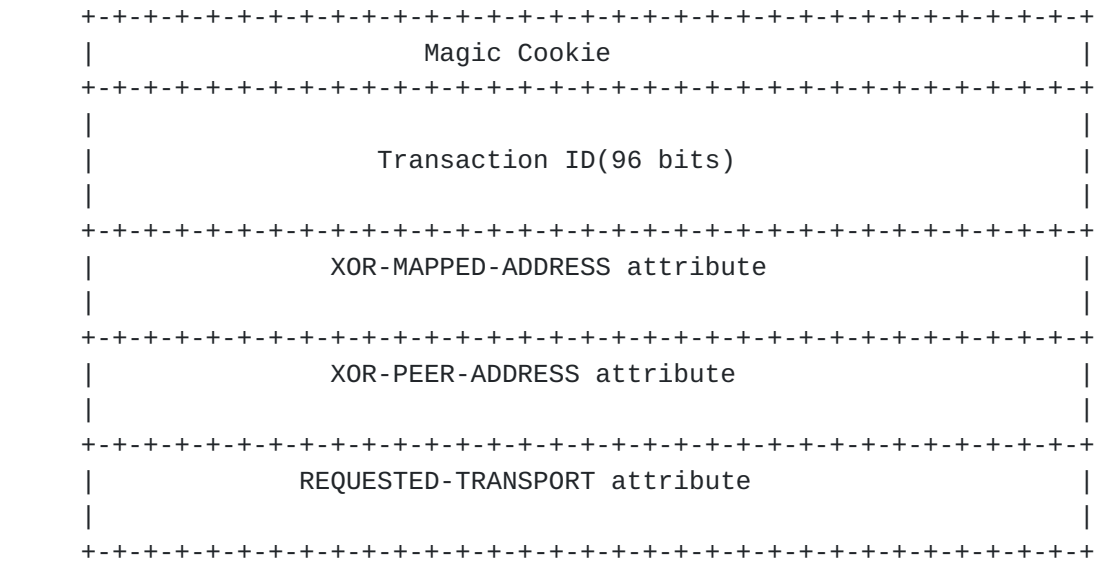
The converted packet will be routed to NAT side of the other host, converted by the NAT device and then to the other host. The return packet will be delivered to the relay transport address of CGN/CDN device and be converted in reverse accordingly.

### 4.2. Couple Operation Packet Format

The Couple Opcode allows RS to create a new explicit couple table on the CGN device(OARS Server), instructs the CGN device to accomplish the related translation work.

The following diagram shows the Opcode layout for the Couple Opcode request/response format.






| | |
|---|---|
| STUN Message Type | Couple method: value TBD.<br>only request/response semantics<br><br>Decouple method: value TBD.<br>only request/response semantics |
| Length | The same definition as STUN prococol<br>[RFC5389] |
| Magic Cookie | The same definition as STUN prococol<br>[RFC5389] |
| Transaction ID | The same definition as STUN prococol<br>[RFC5389] |
| XOR-MAPPED-ADDRESS | The same definition as STUN prococol<br>[RFC5389]. The value should be the<br>RFLX-Relay address of the host. |
| XOR-PEER-ADDRESS | The same definition as TURN prococol<br>[RFC5766]. The value should be the<br>RFLX-Relay address of the peer. |
| REQUESTED-TRANSPORT | The same definition as TURN prococol<br>[RFC5766]. the value of the<br>"protocol" fiel should be TCP or UDP. |

Fig.4-1: Couple Opcode Request/Response Format

## 5. Detailed Example

### 5.1. Procedures of Communication Traversing Symmetric NATs

When one of the communication hosts located behind the symmetric NAT device, the host-to-host communication must via one relay device. Below are the key procedures of OARS solution, we use the Fig3-1 to describe the example.

Please note the communication procedures between the hosts and the application server are out of the scope of this document, we only focus on the key procedure proposed by this document.

1) If Host 1 and Host 2 want to communicate with each other, they will send STUN binding message to the RS IPv4 address/port, get their reflex address to RS[REFLX-RS].

2) RS will select one CGN device to relay the packet, based on the RS addresses information of the two peers. Here we assume it select CGN-1 as the relay device. RS will notify Host 1 and Host 2 of their relay transport address, both will use the same relay IP address/port on CGN-1.

3) Host 1 and Host 2 will send STUN binding message to CGN-1, get their relay address to CGN-1[REFLX-Relay] and report them to RS, together with RS addresses gotten in step 1). Here we assume the [REFLX-Relay] address of Host 1 is 192.0.2.1:7000, and [REFLX-Relay] address of Host 2 is 192.0.2.150:32102.

4) RS will form the "Couple" message, which include mainly the [REFLX-Relay] addresses of Host 1 and Host 2 and their communication protocol, here we assume they use TCP to communicate.

5) Upon receiving the "Couple" message, the CGN-1 device will form one forwarding table that look like below:

| Reflextive transport address of Host1 | Reflextive transport address of Host2 | Transport Protocol |
|---|---|---|
| 192.0.2.1:7000 | 192.0.2.150:32102 | TCP |

Table 5-1: Couple Table Example (symmetric case)

6) Host1 will send the application data to the relay transport address in CGN-1.

7) CGN device will look up the Couple table, use the source address of received packet(192.0.2.1:7000 in this example) to get the reflex IPv4 address of Host 2.

8) It then will change the source address of the packet to the relay transport address in CGN device, the destination address of this packet to the IPv4 reflex address of Host 2. The translated packet will be forwarded to Host 2.

9) The return traffic will also be sent to the same relay transport address in CGN-1, converted by the CGN device, and sent back to Host 1.

### 5.2. Procedures of IPv4 and IPv6 Host Communication

If Host 1 is one IPv4 node and Host 2 is one IPv6 node. The communication process are similar, except the relay address that is sent to the Host 1 is the IPv4 address of the CGN-1 and the relay address that is sent to the Host 2 is the IPv6 address of the CGN-1. Host 1 and Host 2 will not notice that they are talking to one node that in different address family.

The relay device selection process is same as the above example. Here we describe the procedure from step 3.

3) Host 1 and Host 2 will send STUN binding message to CGN-1, get their relay address to CGN-1[REFLX-Relay] and report them to RS, together with RS addresses gotten in step 1). Here we assume the [REFLX-Relay] address of Host 1 is 192.0.2.1:7000, and [REFLX-Relay] address of Host 2 is 2001:c68:300:105::1[49191].

4) RS will form the "Couple" message, which include mainly the [REFLX-Relay] addresses of Host 1 and Host 2 and their communication protocol, here we assume they use TCP to communicate.

5) Upon receiving the "Couple" message, the CGN-1 device will form one forwarding table that look like below:

| Reflextive transport address of Host1 | Reflextive transport address of Host2 | Transport Protocol |
|---|---|---|

| | | |
|---|---|---|
| 192.0.2.1:7000 | 2001:c68:300:105::1[49191] | UDP |

Table 5-2: Couple Table Example (different address families case)

6) Host1 will send the application data to the relay transport address in CGN-1.

7) CGN device will look up the Couple table, use the source address of received packet(192.0.2.1:7000 in this example) to get the reflex IPv6 address of Host 2.

8) It then will change the source address of the packet to the relay transport IPv6 address in CGN device, the destination address of this packet to the IPv6 reflex address of Host 2. The translated packet will be forwarded to Host 2.

9) The return traffic will also be sent to the same relay transport address in CGN-1, converted by the CGN device, and sent back to Host 1.

## 6. OARS Benefits

Comparing to TURN, OARS could provide following benefits:

o Decoupled from ICE

TURN is tightly coupled with ICE. Operations like NAT punching, create permission .etc all require ICE connectivity check packets. Benefited from the couple operation, OARS doesn't necessarily need ICE interaction.

o Avoid the Create Permission issues in TURN

In the OARS solution, each communication pair will use the same relay server and the same relay address. The relay permission action required by TURN solution is replaced with the "Couple" command. There is no ambiguity for the relay permission because "Couple" command use the ip address and port information of the communication pair simultaneously. There are also no possible attacks due to the loose control of the current TURN permission treaments.

o Less Relay Address/Port Consumption and Management

OARS doesn't need to allocate different address-port pair for each session initiated from the hosts. Thus, it could obviously reduce the resource consumption and the human burden for planning the resource allocation.

o Simplified Procedures

Theoretically, it requires only two commands to accomplish the relay function, compared with over eight commands that required by TURN solution. Due to every host communicate with the well-known relay address, there is no additional requirement for punching holes in the NAT devices, which is indispensable for the current TURN solution.

| | TURN Solution | OARS Solution |
|---|---|---|
| Required Commands | 1. Binding | 1. Binding |
| | 2. Allocate | 2. Couple |
| | 3. Send | |
| | 4. Data | |
| | 5. Channel Bind | |
| | 6. Connect | |
| | 7. ConnectionBind | |
| | 8. ConnectionAttempt | |

Table 6-1: Procedures comparison between TURN and OARS

o Unified solution for TCP/UDP and IPv4(6)-IPv6(4) data relay
OARS supports the data relay for the communication betweentwo hosts, uses same mechanism for TCP and UDP transport protocol, even for the communication between different address families.

o Support for optimal relay selection

Because of the central deployed RS could have the whole network's routing/path knowledge, OARS is more likely to achieve an optimal relay (OARS server) selection based on various information such as the reflective transport addresses of the two communicating peers, the link usage information between two peers and the load status of the candidate TURN-Lite servers etc.

With the RS's knowledge, OARS is also more likely to achieve better relay selection for some specific requirements.For example, if one peer wants to hide its ip address to protect its privacy, the RS in OARS architecture could possibly select one OARS server that located far away from the host.

## 7. OARS Deployment Considerations

The OARS Server can be deployed in distributed manner. The most appropriate devices for incorporating this function are the CGN devices that have been deployed distributed by the service provider. Each distributed OARS Server has one unique public IPv4/IPv6 transport address.

The RS can select the appropriate OARS Server based on the proximity of the OARS server with the communication hosts and can also use other criteria to influence the selection procedure, as described in Section 3.

## 8. Security Considerations

The additional requirement of OARS is authenticating the couple operation from the RS. When the communication channel between the RS and the OARS server is secured, such security risks can be mitigated accordingly.

## 9. IANA Considerations

This draft requires IANA to allocate following STUN methods:

Couple: value TBD.

Decouple: value TBD.

## 10. Conclusions

Currently, the communication between two hosts that located behind NAT devices, especially that behind the symmetric NAT devices is emerging. With the development of IPv6 technology, the communication between two hosts that in different address families needs also be considered. Under the OARS architecture, the communication requests for IPv4/IPv4, IPv4/IPv6 scenario can be met in one general solution. Such solution can alleviate the burden of various CP/SP to deploy the TURN server by themselves, exploit and open the capabilities of CGN device that deployed by service provider to the third party(CP/SP), make the host-

to-host communication more efficient.

## 11. Acknowledgements

Many valuable comments were received from Brandon Williams, Oleg Moskalenko, Jonathan Rosenberg, and Dan Wing etc.

This document was produced using the xml2rfc tool [RFC2629].

## 12. References

### 12.1. Normative References

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

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.

[RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", RFC 5766, April 2010.

### 12.2. Informative References

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

[RFC5382] Guha, S., Biswas, K., Ford, B., Sivakumar, S., and P. Srisuresh, "NAT Behavioral Requirements for TCP", BCP 142, RFC 5382, October 2008.

[RFC6062] Perreault, S. and J. Rosenberg, "Traversal Using Relays around NAT (TURN) Extensions for TCP Allocations", RFC 6062, November 2010.

Authors' Addresses

Aijun Wang
<Wang,Liu&Uberti>

China Telecom

Southern Zone of Future Science & Technology City, Beiqijia Town, Changping District

Beijing, 102209

Email: wangaj@ctbri.com.cn

Bing Liu

Huawei Technologies

Q14, Huawei Campus, No.156 Beiqing Road, Hai-Dian District

Beijing, 100095

P.R. China

Email: leo.liubing@huawei.com

Justin Uberti

Google

747 6th Ave S

Kirkland, WA 98033

USA

Email: justin@uberti.name