Data Transmission in Hybrid Distributed Environment

Anton Pavlovich Teykhrib
Company Naumen (Nau-Service), Russian Federation

Article Info	ABSTRACT
	This article discusses a hybrid distributed environment including two levels of nodes: the first level which has high quality channels between the nodes, and the second one in which the nodes are directly involved in communication and the communication is performed through the first-level nodes. For the indicated distribution scheme, the questions of selection of the best path between the first-level nodes are solved based on such communication channel characteristics as delay in packet transmission, jitter in transmission, packet loss in transmission, and choosing a method for data transmission through the selected best path: protocols of transport and network layers have been compared, and the use of network layer has been preferred.

Corresponding Author:

Anton Pavlovich Teykhrib,
Company Naumen (Nau-Service),
4th floor, Tatischeva st. 49a, Ekaterinburg-620028, Russian Federation.
Email: ateyhrib@naumen.ru

1. INTRODUCTION

Telecommunications are widely used nowadays in different areas of human live, for example they may be used in telemedicine, as shown in [1]. As shown in [2] distributed communications also significantly increased in recent years. In accordance with the architecture defined in [2] and illustrated in Figure 1, there is a routing server layer in a hybrid distributed communication scheme; these servers are connected with high quality communication channels and transmit data between the second level nodes (these nodes refer to the end users). To accomplish successfully this task, it is necessary to transmit data between the routing servers through the best path. Data transmission through the best path comprises two subtasks: the first one is to find the best path, and the second one is to ensure data transmission through the selected best path. Both subtasks are being discussed in the article.
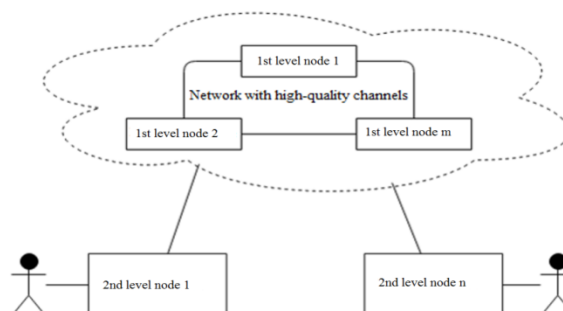

Figure 1. The Interaction Pattern of Nodes in Hybrid Network

2.   FINDING THE BEST PATH BETWEEN ROUTING SERVERS

In accordance with the considered architecture, the routing servers build a full mesh graph in which the routing servers are vertices, and physical paths between them are the edges. Thus, each physical path has a number of characteristics affecting the quality of the data streaming:

a. Transmission delay, measured in milliseconds;
b. Jitter of transmission delay, measured in milliseconds;
c. Transmission loss, measured in percentage of the amount of lost packets.

Thus, the graph is weighted, but the weight is a tuple of three characteristics. The general view of the graph is shown in Figure 2.
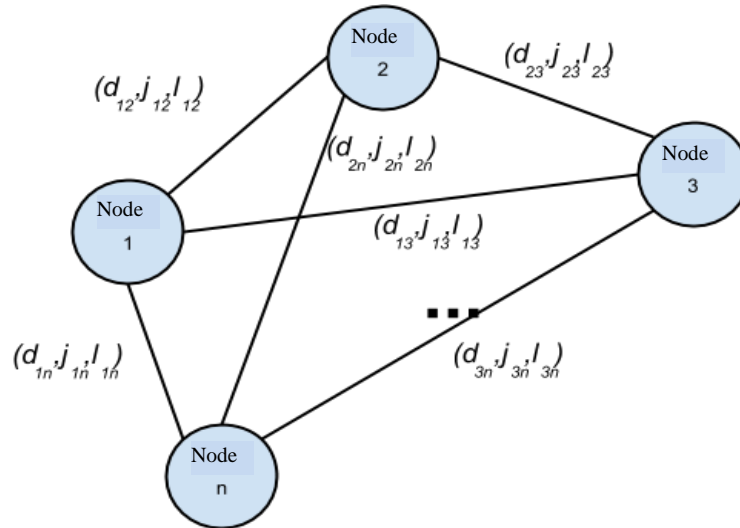

Figure 2. Data Transmission Paths Graph

where:

a. d_{km} is delay of transmission between node k and m, at the same time due to the bi-directional data exchange, we assume that this value is symmetric and $d_{km} = d_{mk}$;
b. j_{km} is jitter of transmission delay between node k and m, at the same time due to the bi-directional data exchange, we assume that this value is symmetric and $j_{km} = j_{mk}$;
c. l_{km} is transmission loss between node k and m, at the same time due to the bi-directional data exchange, we assume that this value is symmetric and $l_{km} = l_{mk}$.

The task of building the best path is a classic problem of the optimization theory, and it should be solved by methods and means of this theory. This problem must be solved for each pair of vertices. There are several commonly accepted algorithms to solve this problem:

a. Floyd-Uorshella algorithm, presented in [3].
b. Johnson's algorithm, presented in [4].
c. Common algorithms to find the shortest distance from one vertex to the others made for each vertex. An example of such an algorithm is Dijkstra's algorithm, designed to find the shortest path from one of the vertices of the graph to all the others, and presented in [5].

All of these algorithms can be applied only to the weighted graph. In this case, the graph is also weighted, but edge weight represents a tuple, as a result, weighting function must be developed so that it converts unambiguously tuple into edge weight.

In its general form, this function can be represented as a formula:

$$f_{mk} = w_d \cdot d_{mk} + w_j \cdot j_{mk} + w_l \cdot l_{mk} \qquad (1)$$

where

a. w_d is weight of delay component in the total weight of the edge;
b. w_j is weight of jitter component in the total weight of the edge;
c. w_l is weight of loss component in the total weight of the edge.

In this case, if the parameters of the delay, jitter, and loss can be measured objectively by using different software, such as MTR [6], the weights of these parameters must be determined under this section. To determine the weights, consider the limiting values of these parameters:

a. delay should not exceed 150 milliseconds;
b. jitter of delay should not exceed 75 milliseconds;
c. loss should not exceed 5%.

When parameters under consideration achieve any of these values, it should mean equivalent weight value, thus we obtain:

a. $w_d = 1/150$;
b. $w_j = 1/75$;
c. $w_l = 1/5$.

Having calculated weight of each edge based on the measured parameters of functioning of the physical paths between the routing servers, any of these algorithms can be used. Moreover, based on the fact that it is necessary to build a path from a specific node to the others when streaming, it is more expedient to calculate a path from each node locally-that is, the third option. This group includes the following algorithms:

a. the Dijkstra's algorithm requires only positive weights of the edges;
b. the Levit's algorithm can be used with negative weights;
c. the Bellman-Ford algorithm can be used with negative weights

It is also worth noting the need to monitor network status, as routing state may change over time, and the network state has to be updated. To update, network status should be measured periodically, followed by recalculation of the path. The frequency of polls should be determined on the basis of a compromise between the necessity to respond quickly to changing network conditions and network load caused by monitoring packets.

Another question is dissemination of information on the network state since, in order to find the best path, it is necessary to have idea about the paths between all the pairs of nodes. Because of the absence of synchronization (introduction of synchronization leads to additional complication of a common scheme), each routing server will carry out a poll again at arbitrary points in time. Eventually, the best way is a gradual upgrade of network status, in which each routing server will periodically update the status of their paths to the other servers and will inform the other routing servers about updating results. Thus, the final protocol which finds the paths for streaming will be as follows:

a. Each newly added routing server measures the characteristics of the paths to the other routing servers.
b. It calculates weight of each path on the basis of the developed formula.
c. It informs the other routing servers about its status.

Upon receipt of the status of paths from another server, each routing server rebuilds its paths; in this case, there is a danger that the changeover will be carried out continuously as a result of the continuous arrival of updates. This should also be taken into account when choosing polling intervals.

3.   DISTRIBUTED STREAMING OVER SELECTED PATH

Another aspect of the operation process of distributed data streaming protocol is a task for the data streaming over the generated paths. There are two basic approaches to solving this problem, which operate at different layers of 7-layered OSI model:

a. Operation at the transport layer.
b. Operation at the network layer.

3.1. Transport layer protocols

Currently, the following protocols are used for streaming at the transport layer:

a. RTP, used in SIP, WebRTC, RTSP, is the underlying protocol, currently used in streaming on the Internet, and it is defined in RFC 3550. Example is shown in [7].
b. Adobe RTMFP is used in streaming between two Adobe Flash endpoints, which has replaced another protocol Adobe RTMP; since 2014, it has become an open protocol, and it is defined in RFC 7016
c. Closed protocols such as Skype and others.

In terms of actual implementation, both protocols are identical and depend on the final customer, which supports either one or the other technology.

Only open protocols are suitable for the further use - that is, RTP and RTMFP, because they can be legally implemented in software products. Thus, software has to be developed that would execute data reception and streaming in this node for data transmission at the transport layer on each routing server.

3.2. Network layer protocols
At the network layer, an example of such a protocol is the following:
a. OSPF, defined in RFC 2328. This protocol has the following advantages in accordance with [8]:
1) High convergence speed in comparison with the distance-vector routing protocols.
2) Variable-length subnet masking (VLSM).
3) The optimal use of capacity with construction of the shortest path tree based on Dijkstra's algorithm.
As can be seen, the problem of finding the best path is solved in the framework of this protocol, which has to be solved autonomously in case of use of the transport layer.
b. IS-IS, defined in ISO / IEC 10589: 2002; essentially this protocol is similar to OSPF and enables similar operation performance; it is different from OSPF being more low-layer since ISIS runs directly over layer 2 of OSI model, and OSPF does over the IP protocol, wherein the protocol requires support at the levels of hardware and hardware management for configuration, while OSPF can be used for a standard network hardware.
c. BGP, defined in RFC 1163, is a basic dynamic routing protocol in the Internet, similar to OSPF; however, it does not allow tracking the status of paths.
d. EIGRP, defined in [9] enhanced distance-vector dynamic routing protocol, developed by Cisco. This protocol is also similar to OSPF, but it is recommended for use in not very distributed solutions in accordance with [10].
Thus, all other protocols apart from OSPF have significant drawbacks; consequently, OSPF will be selected for the further study.

Before starting to use one or another approach, the inability to influence the routing tables of Internet service providers has to be taken into account; it means that it is necessary to use a certain virtual network, under which the above mentioned approaches will be applied. In accordance with [11] and [12], the best approach for organizing a virtual network is GRE protocol defined in RFC 2784.

Further development of streaming protocol should be based on the selection of the layer, which will determine the transmission over the best path: at the transport or network layers.

4.   RESULT AND DISCUSSION
Main result of this article is selection between protocols of network and transport layers. To determine the most appropriate solution, it is necessary to choose the criteria in order to evaluate solutions. The selection criteria will be as follows:
a. Ease of development and maintenance.
b. The ability to use the developed path evaluation function.
c. The ability to work with arbitrary internet service providers.
d. Computational ease which is minimization of computational cost of processing packages.
e. The ability to work with the customer.
The results of comparison of streaming protocols under consideration are presented in Table 1.

Table 1. Streaming Protocols Comparison Results

Comparison criterion	Transport layer protocols	Network layer protocols
Development ease	-	+
	It requires the development of an independent streaming data forwarding module, and support of its operation on each node	There are open source software implementing this technology
Support of developed path evaluation function	+	+
Support of arbitrary Internet service providers	+	+/- depending on the particular implementation
Computational ease	-	+
		by operating at a lower layer of OSI model
Ability to work with the client	+	-
		since there is no possibility to customize network hardware at client workplace

Thus, if we exclude the point of client support, the most appropriate is the use of network layer protocols to organize distributed streaming protocol. However, there is a possibility to combine the approaches due to the distributed operation scheme, and to use the transport layer protocols between the client and the routing servers and the network layer protocols between routing servers.

The use of network layer protocols modifies a little the best path definition protocol. Due to the fact that its own algorithm finding the best path is used within OSPF, a mechanism has to be developed that will change the value of paths (value is the weight according to OSPF terms) on the basis of the developed weighting function. Afterwards OSPF will automatically rebuild the paths.

5.   CONCLUSION

Thus, as a result of the research of questions of finding the best path between routing servers, the function was designed which determinates the weight of each path between routing servers. According to research of the questions of distributed streaming over the selected path, two main directions of solutions were identified:

1) At The Transport Layer.
2) At The Network Layer.

As a result of comparison by the selected criteria, the solution at the network layer was selected for further use, based on the use of OSPF and GRE protocols. The obtained results can be used afterwards for practical realization.

ACKNOWLEDGEMENTS

REFERENCES

[1]   L. Wang and C. A. Alexander, "Telemedicine Based on Mobile Devices and Mobile Cloud Computing," *International Journal of Cloud Computing and Services Science (IJ-CLOSER),* vol/issue: 3(1), pp. 26-36, 2014.
[2]   A. P. Teykhrib, "Review of Decentralized Communications Technologies," *Asian Journal of Information Technology*, vol. 14, pp. 162-165, 2015.
[3]   R. W. Floyd, "Algorithm 97: Shortest Path," *Communications of the ACM,* vol/issue: 5(6), 1962.
[4]   D. B. Johnson, "Efficient algorithms for shortest paths in sparse networks," *Journal of the ACM,* vol/issue: 24(1), pp. 1–13, 1977.
[5]   E. W. Dijkstra, "A note on two problems in connection with graphs," *Numerische Mathematik,* vol. 1, pp. 269-271, 1959.
[6]   MTR. [Electronic resource] – Access mode: http://www.bitwizard.nl/mtr/
[7]   S. Peng, "Audio and Video Communication Software Design Based on SIP," *Indonesian Journal of Electrical Engineering and Computer Science*, vol/issue: 12(11), pp. 7946-7951, 2014.
[8]   OSPF. [Electronic resource] – Access mode: https://ru.wikipedia.org/wiki/OSPF
[9]   R. White, *et al.*, "Enhanced Interior Gateway Routing Protocol," 2014.
[10]   Routing Protocol Choice OSPF vs. EIGRP. [Electronic resource] – Access mode: http://www.h3c.com/portal/Products___Solutions/Products/Other_Products/Routers/Quidway_AR18-2X_Series_Routers/White_Paper/200701/194231_57_0.htm#_Toc87242195
[11]   GRE Tunnel used as a OSPF Virtual-Link. [Electronic resource] – Access mode: http://cisqueros.blogspot.ru/2013/05/ospf-gre-tunnel.html
[12]   Connect to OSPF area 0 over GRE tunnel. [Electronic resource] – Access mode: https://heggel4.wordpress.com/2015/02/26/connect-to-ospf-area-0-over-gre-tunnel/

BIOGRAPHY OF AUTHOR

Anton Teykhrib received the degree in Ural Federal University in 2007. Currently he is a researcher in company Naumen. His interests are in VoIP, including SIP and WebRTC, and cloud services.