Investigating Indirect Impacts of TCP Connection on IMS Network

Ali Abdulrazzaq K. [1*]

[1*]Department of Computer Science, Education College for Pure Science, University of Mosul, Mosul, Iraq

E-mail: [1*] aliabd@uomosul.edu.iq

Abstract

The IP Multimedia Subsystem (IMS) has the major architectural framework to be involved in the Next Generation Network. IMS works to bridge multimedia communication among a variety of applications over the Internet. IMS bears its multimedia signals and streams through different means of transport protocols; TCP, UDP, and SCTP. TCP is a connection-oriented protocol that provides reliable data delivery and congestion control. To setup connection with TCP, IMS entities require extra operations to be completed, and those operation processes called (worker process) cost the multimedia server extra overload and delay. This paper investigates the indirect impacts of TCP connections that result from the Call Session Control Function (CSCF) servers when dealing with video communication using a local network setup with wired connection. Two parameters are evaluated in the experiment: CPU usage and response time, in two different scenarios. The experimental results show that the outbound scenario performs better than the inbound scenario due to the extra operations required to setup new TCP connection for the inbound.

Keyword: IMS, TCP, Video Communication, Call Session Control Function

<div dir="rtl">

التحقيق في التأثيرات الغير مباشرة لأتصال بروتوكول ال TCP على شبكة ال IMS

علي عبدالرزاق خضر [1*]

[1] قسم علوم الحاسوب، كلية التربية للعلوم الصرفة، جامعة الموصل، الموصل، العراق

الخلاصة:

من المتوقع حديثا ان تصبح انظمة الوسائط المتعددة الهيكل الاساسي في تطوير شبكات الجيل القادم. تعمل انظمة الوسائط المتعددة كموصل بين اتصالات الوسائط المتعددة مع انواع عديدة من تطبيقات الانترنيت. من الناحية التقنية تُحمل انظمة الوسائط المتعددة اشاراتها

</div>

و بياناتها من خلال بروتوكولات وسائل نقل المتعددة مثل بروتوكول التحكم بالنقل او بروتوكول مخطط المستخدم او بروتوكول نقل التحكم في التدفق. بالنسبة لبروتوكول التحكم في النقل يعد بروتوكولا مهياً للاتصال من شأنه يوفر إمكانية عالية من حيث موثوقية تسليم البيانات والتحكم في ازدحام البيانات. من أجل الحصول على اتصال مزود بجميع خاصيات بروتوكول التحكم في النقل, تحتاج انظمة الوسائط المتعددة في نقل بياناتها عبر الإنترنت عمليات إضافية لإكمالها، ولكن هنالك معوقات تكلف هذه العمليات التشغيلية خوادم الوسائط المتعددة بتحميل زائد و تأخير اضافي. من أجل ذلك, تبحث هذه الورقة البحثية في التأثيرات غير المباشرة الناتجة من بروتوكولات النقل في اجهزة وخوادم نقل البيانات عبر الانترنيت وخصوصا عندما تتعامل مع اتصالات فيديوية. من خلال تطبيق التجارب يتم تقييم عاملين اساسيين في التجربة وهما اداء المعالج المركزي و وقت الاستجابة ضمن سيناريوهين مختلفين. يُظهر الاختبار التجريبي أن السيناريو الصادر يؤدي أداءً أفضل من السيناريو الوارد بسبب العمليات الإضافية المطلوبة التي من شأنها ان تبطيء و تقلل من اداء خوادم الجيل القادم.


الكلمات المفتاحية : IMS، TCP، أتصالات الفيديو, السيطرة على جلسات الاتصال

1. Introduction

The Internet has grown rapidly in the last decade with a big involvement of multimedia communication in the use of end user devices. Multimedia communication has become a daily requirement through the continuous increase of user communication via social media, education, businesses, applications, among others. To control multimedia communication there should be a control system that can handle video communication, video conferencing, and video streaming [1]. The Internet Protocol multimedia subsystem (IMS) is an architecture that has been used to work and handle the most usable services and applications on the Internet [2].

IMS is a standard found by the Third Generation Partnership Project (3GPP), which is considered one of the major standards of organizations that utilize 3G networks. IMS systems allow service providers to provide multimedia communication (video, audio, messages), security, and multisession applications between end users [3]. In addition, IMS systems provide quality, roaming, and authentication in real time communications. In order to provide acceptable service and flexibility between end to end devices IMS networks can work with application layer, control layer, and access connectivity layer [4].

IMS systems provide the fundamental concepts of a multimedia session including registration, user location, media capability, presence service, and session negotiation and management. Devices in the IMS systems are connected through different means of networks and transmit data using transport protocol such as TCP or UDP [5] (See IMS architecture layers in Figure 1).
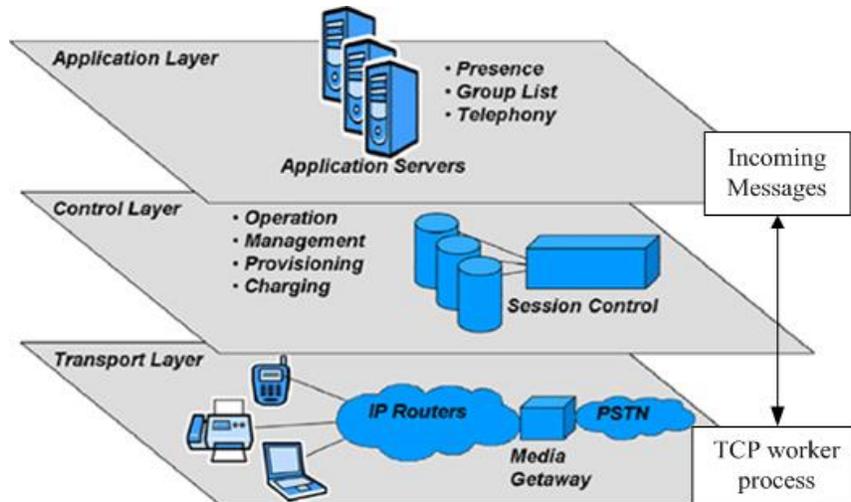
Figure 1. IMS architecture layers

Call Session Control Function (CSCF) servers, which is the element that handles the control layer, uses different means of transport protocols, including UDP and TCP to provide end to end data transmission [6]. TCP is known as connection-oriented protocol that can provide reliable delivery and congestion control, while UDP (not considered) is connectionless protocol. It is worth mentioning that each of (TCP and UDP) have their own advantage. In most common practice, UDP is widely utilized in IMS communication instead of TCP, due to the additional complexity nature of TCP [7]. TCP connection uses an extra operation process for handling end to end connectivity, that is, end up with significantly lower overhead and throughput caused from the direct TCP structure expenses and/or indirect from the system that the TCP implemented in. It is still an issue that the indirect factor of TCP effects the quality of multimedia communication. In literature, there are a number of studies that have been conducted on TCP and video communication [1], UDP and TCP in signaling messages [7], and transport protocol performance in multimedia communication [12]. In sum, TCP is used for signaling and UDP for media transfer.

The aim of this paper is to investigate the indirect impacts of TCP connection that result from the Call Session Control Function (CSCF) servers when dealing with video communication. The investigation is conducted by evaluating two relative parameters: video call response time and CPU usage as CPU can directly show the performance of the device itself. Previous studies have already tested other network parameters such as throughput, delay, and packet delivery ratio.

The following section is a background of IMS. Section 3 presents TCP structure in general. Section 4 describes the relation between IMS and TCP connectivity. Experimental setup along with investigation scenarios are discussed in Section 5. Finally, Section 6 concludes the paper.

2. IP multimedia subsystem (IMS)

IMS is found to be an IP-based overlay network which provides the access and control for IP-based signaling. The major concern of IMS is involved in fixed-networks, WiMAX, or Wireless LAN to provide a guaranteed end-to-end service with acceptable quality due to several problems such as mobile handover and wireless signal fading. [8][9].

The IMS signaling core is architected from the CSCF (Call Session Control Function). As shown in Figure 2, P-CSCF (Proxy CSCF) is one of the important parts of IMS which is located at the edge of the IMS network. Also, P-CSCF is used to handle user authentication, receive requests, and send responses. P-

CSCF also plays a role of security by parsing all messages arrived from the upstream and match them with the standard [10].

Another logical component is called S-CSCF (Serving CSCF) which is used for routing all incoming requests and provides the main service in that system. The S-CSCF is connected with a direct links to the backend database and end user. Additionally, it is responsible for handling other kind of services such as video call charging and billing systems.

The I-CSCF (Interrogating CSCF) serves as the entity for handling peering between two different service providers. S-CSCF co-works with DNS server when there is a requirement to route requests to another service provider at a different network. In other words, I-CSCF queries the assigned local database and retrieves the desired address for further actions.

In this paper, Kamailio server [11] is used as CSCF to control video signaling setup with the ability to use all kinds of transport protocols. By nature, Kamailio server supports TCP connection.

3. Transmission Control Protocol (TCP)

In this paper only the main TCP processes related to CSCF are discussed. The real structure of TCP code is neglected as other researches have already discussed it in detail [12]. TCP is a connection-based protocol meaning that all connections will be ready first for further message communication. TCP also maintains connections across one or multiple transactions, one transaction can be shared by multiple workers. TCP is managed by Supervisor Process which is mainly responsible for managing and controlling all TCP connections. In order to make the connections synchronous with session messages, a shared hash table is used to handle all connection objects [13].
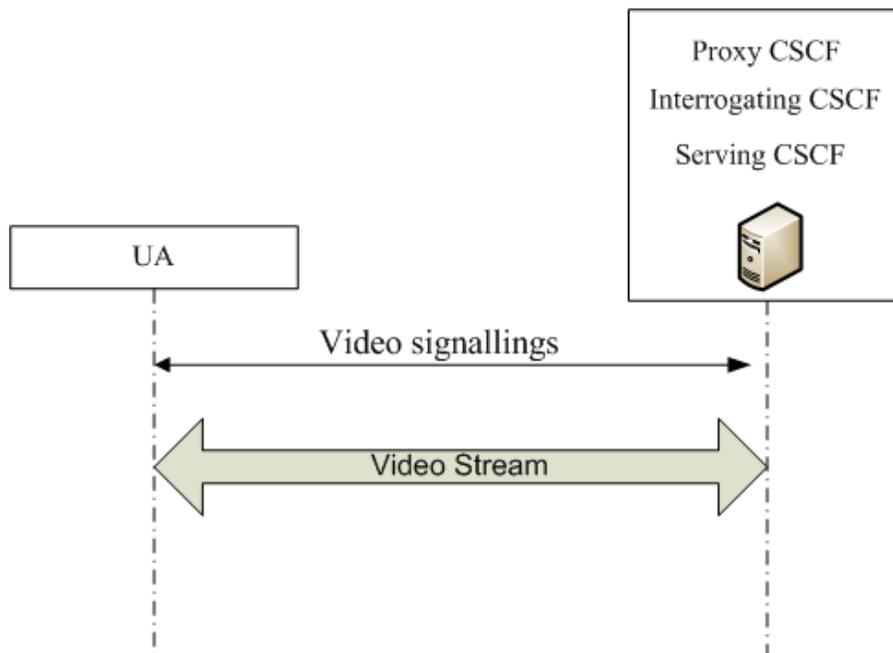
Figure 2. IMS Components

Once the supervisor accepts a new connection it eventually creates a new TCP connection object to be added into the hash table. The new object is ready to receive new messages through the already opened worker process, as illustrated in Figure 3. From this point, all the messages will be sent and received at the same specified worker until the connection is idle and closed [14].
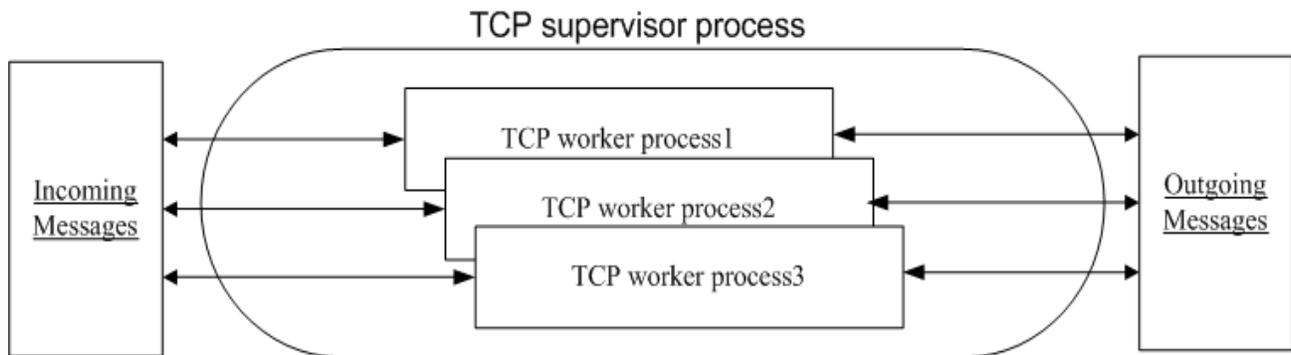
Figure 3. TCP workers process and supervisor

4. Indirect TCP Impacts

In this section indirect impacts that are caused by the use of TCP connection are illustrated as follows:

Shared Transaction: one complete communication may be achieved by two or more transactions each of which can use a separate worker process. To synchronize the transactions a Shared Transaction status is used to control the complete call. However, these operations will cost the CSCF server time and overhead.

Socket file descriptor: the worker needs to request the socket file descriptor which corresponds to the current TCP connection object directly from the supervisor. Immediately, the supervisor process makes a copy of all the opened TCP connection files for synchronization. The copy and synchronous operations are time consuming and require extra CPU calculation.

Idle and close connection: Closing a TCP connection in Kamailio server is achieved through two operations; closing all descriptor files of each open socket and terminating the TCP connection objects. Frequent checks for idle state also require the server to use some operations which in turn require extra CPU usage.

5. Experimental and Investigation

In this section an experimental setup is presented to show the hardware and parameter setup. Also, in this section an investigation discusses the results found from the conducted experiment to show the indirect factors that affect Kamailio server, Figure 4 shows the network topology for the experiment. To investigate

and test the Kamailio server two parameters use video Call Response Time and CPU utilization in two different scenarios; inbound and outbound. In this research video stream is used in TCP rather than UDP, because TCP connection has many different operations which affect the performance of network device; To achieve the desired results a lab setup is conducted with the major network element involved in this research. It is worth to mention that the conducted video calls are handled only with session setup time and there is no video stream exchange between end users. Table 1 presents the hardware specifications.
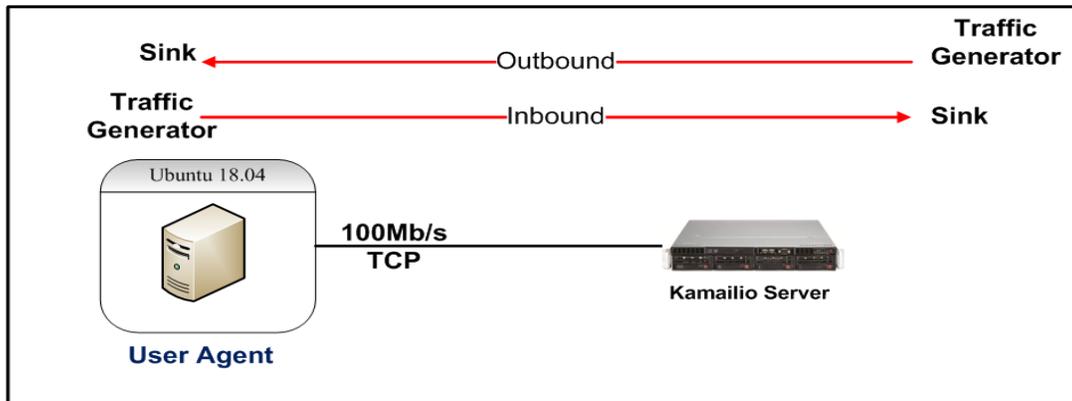
**Figure 4.** Network topology for the experiment

Table 1. Hardware specifications

Entity	CSCF Server	User Agent
Machine Model	Dell-Vostro (PC)	Dell-Vostro Laptop
Operating System	Debian Stretch v9	Ubuntu 18.04 server
Kernel	4.9	4.15
RAM	2 Giga Byte	2 Giga Byte
CPU	3.00GHz (deactivate other core processors)	900@ 2.20GHz
SIP software	Kamailio 5.3	Traffic Generator
Ethernet	100Mb/s	100Mb/s

The structure of TCP is different from the one in UDP connection, so this paper performs the impact of TCP operations only. Figure 5 shows the CPU percentage that has been utilized by the CSCF server during video call handling. Note that in this experiment there is only one CPU core active, the remaining cores available in the machine are deactivated for better results gain from CPU usage.

In this figure inbound calls at the 200 cps use 1.7% of total CPU for both inbound and outbound video calls and when the load increases to 600 cps the CPU starts to hit the 2.4% because it is run in local

network and it is isolated from the external factors. This is because the new connection setup for TCP and the supervisor process require extra operations to open the connection for each transaction. For inbound call the CPU usage reaches 2.65% to handle 800 cps while it reaches 2.60% for outbound that is because for outbound call the connections of calls are already open and no authentication is required that means the CPU will require to handle less operations to settle. In case of a higher call load 1000 cps, the inbound call spend 2.88% of CPU to use for handling 1000 calls. For outbound call the CPU hit 2.69% at the same amount of calls.
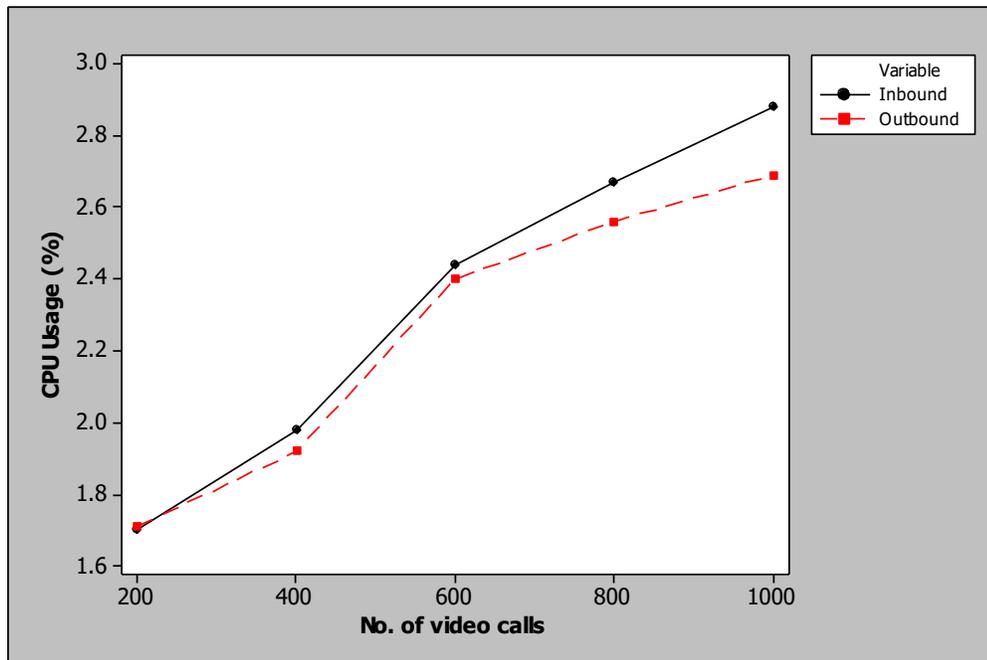

Figure 5. CPU usage for inbound and outbound

In figure 6 the video call response time is calculated from the moment the call is generated from the source traveling to destination and back again to source. That time includes all the processes that have been taken to open TCP connection and received signaling messages. The time spent for handling 200 cps (for one call) for inbound call is 20 ms. When the number of calls increase the time also increases by 2ms, which means for the extra 200 call there is only 2 ms penalty of the operation of TCP connections. For inbound all the connection have to be created and open for the messages that arrive from the upstream. From the figure, the state of 800 calls will require 26 ms to handle the individual call. When the heavy load is applied to the server the call starts to spend 28 ms. In figure 7 the 200 call takes 15 ms to handle individual call which is 5ms less than the inbound call that due to the already opened TCP connection. When the call is 400 and 600 the server is able to handle the calls in only 16 ms as for outbound call. The supervisor process have ready description files to be used in outbound that is due to the already opened TCP connection when the first call setup is made.
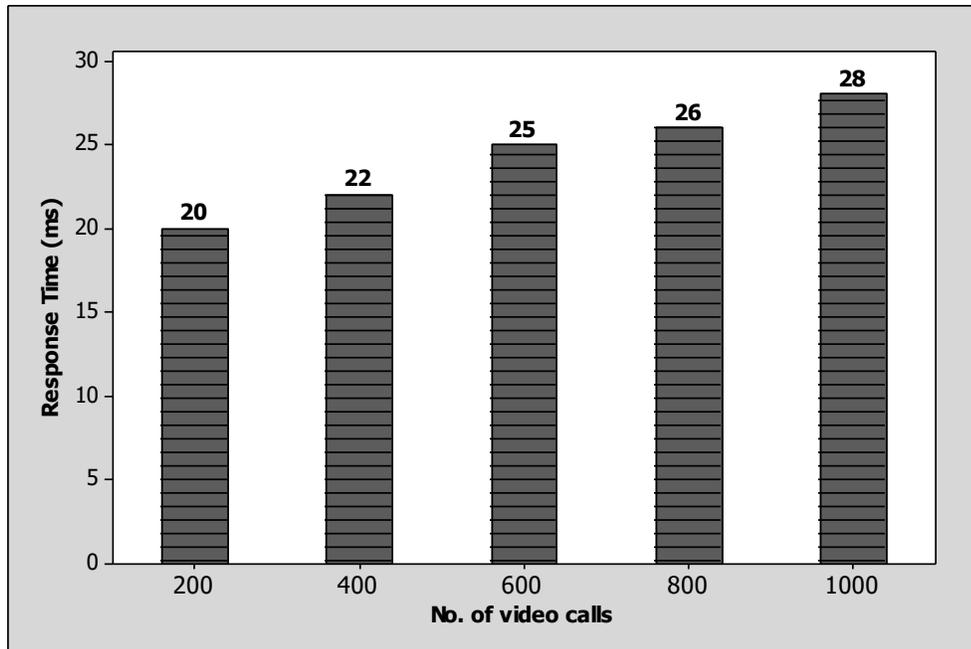
Figure 6. Video Call response time for inbound call
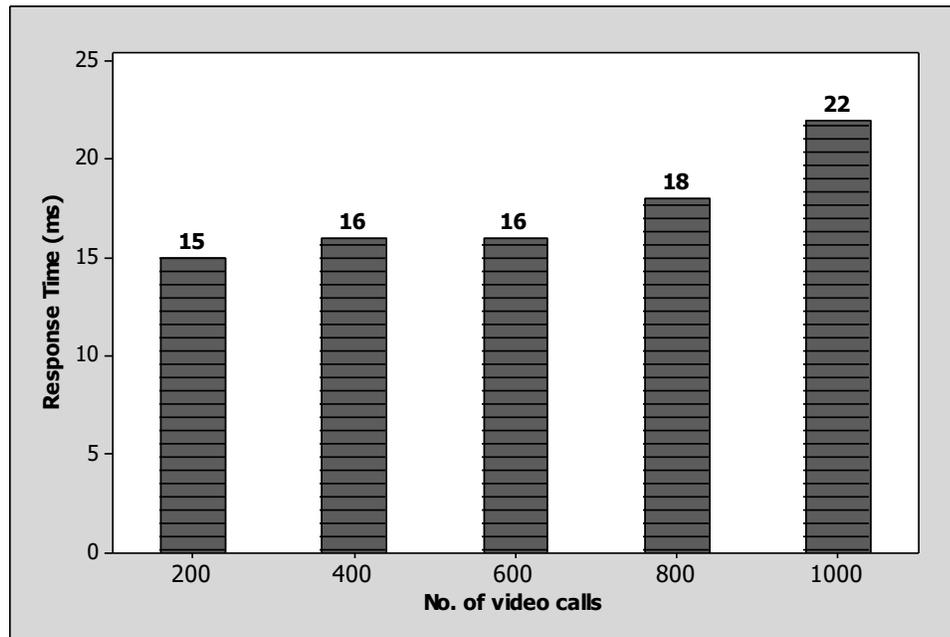
Figure 7. Video Call response time for outbound call

6. Conclusion

IMS technologies have witnessed a wide deployment in the last decade, involved as a control system for all multimedia communication. CSCF server is a main element in the IMS network which is responsible for controlling all the sessions between client and server. This kind of network element resides at the edge

of their network and has been responsible to handle inbound and outbound calls using different means of transport protocols. TCP is a connection-oriented protocol that has been utilized to carry multimedia data from source to destination. This protocol by nature requires extra operations compared to UDP in order to handle the connection. The extra operation is costly in term of CPU utilization and the time spent for call handling. This paper investigates the indirect impacts caused by the CSCF server and TCP supervisor. The experiments have been conducted with the multimedia server Kamailio 5.3 to be tested with two different parameters; CPU utilization and video call response time. The investigation uses one CPU core to avoid distribution processing and thread scaling which impacts CPU usage in real-time. Results show that for inbound video calls at the 200 cps use 1.7% of total CPU for both inbound and outbound video calls. The CPU also uses 2.65% to handle 800 cps while it reached 2.60% for outbound. For inbound scenario, the time spent for handling 200 cps (for one call) for inbound call is 20 ms. At the other side when the number of calls increase, the time also increases by an average of 2ms. To conclude, Kamailio server performs well when the outbound scenario is used in both parameters CPU and call response time. The limitation of this paper is in the test of TCP only, no UDP connections have been used. Also, there is no video stream in the connection, it only tests the signaling messages of the connection.

7. Acknowledgements

8. References

[1]   Lopes Filho, Edmo, Gilberto Tadayoshi Hashimoto, and Pedro Frosi Rosa. "An IMS control and access layer PR-SCTP based network." In *Fourth International Conference on Networking and Services (icns 2008)*, pp. 61-66. IEEE, 2008.

[2]   Khlifi, Hechmi, and Jean-Charles Gregoire. "IMS application servers: roles, requirements, and implementation technologies." *IEEE Internet Computing* 12, no. 3 (2008): 40-51.

[3]   Fabini, Joachim, Norbert Jordan, Peter Reichl, Alexander Poropatich, and Rainer Huber. "" IMS in a Bottle": Initial Experiences from an OpenSER-based Prototype Implementation of the 3GPP IP Multimedia Subsystem." In *2006 International Conference on Mobile Business*, pp. 13-13. IEEE, 2006.

[4]   Vargic, R., M. Krhla, S. Schumann, and I. Kotuliak. "IMS interworking using IBCF." In *2008 Third International Conference on Convergence and Hybrid Information Technology*, vol. 2, pp. 89-94. IEEE, 2008.

[5]   Tirado, Ivan. "IP Multimedia Subsystem (IMS) signaling core security." In *Proceedings of the 5th annual conference on Information security curriculum development*, pp. 59-63. 2008.

[6]   Khoirunnidzom, Nizam, Dadiek Pranindito, and Syariful Ikhwan. "Analisis QoS Open IMS Core berbasis Network Function Virtualization pada Protokol TCP." *Jurnal Informatika* 4, no. 01 (2019).

[7]   Miyazawa, K., Yamaguchi, S. and Kobayashi, A., Throughput Fairness Improvement among TCP BBR Connections. *IEICE Technical Report; IEICE Tech. Rep.*, *120*(327), pp.24-29.(2021)

[8]   Abualhaj, Mosleh M., Mayy M. Al-Tahrawi, and Sumaya N. Al-Khatib. "A new method to improve Voice over IP (VoIP) bandwidth utilization over Internet Telephony Transport Protocol (ITTP)." In *Proceedings of the 2019 8th International Conference on Software and Information Engineering*, pp. 192-195. 2019.

[9]   Praptodiyono, S., Hasbullah, I.H., Kadhum, M.M., Wey, C.Y., Murugesan, R.K. and Osman, A., Securing duplicate address detection on IPv6 using distributed trust mechanism. *Int J Simulation—Systems, Sci Technol*, *17*(26). 2016.

[10]   Romanov, O.I., Nesterenko, M.M., Veres, L.A. and Hordashnyk, Y.S, September. IMS: Model and calculation method of telecommunication network's capacity. In *2017 International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo)* (pp. 1-5). IEEE, 2017

[11]   Moravcik, Marek, and Martin Kontsek. "Proposal of VoIP infrastructure and services for academia-case study." In *2019 17th International Conference on Emerging eLearning Technologies and Applications (ICETA)*, pp. 540-545. IEEE, 2019.

[12]   Zhang, M., Polese, M., Mezzavilla, M., Zhu, J., Rangan, S., Panwar, S. and Zorzi, M., Will TCP work in mmWave 5G cellular networks?. *IEEE Communications Magazine*, *57*(1), pp.65-71, 2019.

[13]   Xu, Z., Tang, J., Yin, C., Wang, Y. and Xue, G., Experience-driven congestion control: When multi-path TCP meets deep reinforcement learning. *IEEE Journal on Selected Areas in Communications*, *37*(6), pp.1325-1336, 2019.

[14]   Mohammadi, R., Javidan, R. and Conti, M., SLICOTS: An SDN-based lightweight countermeasure for TCP SYN flooding attacks. *IEEE Transactions on Network and Service Management*, *14*(2), pp.487-497. 2017.