ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 7, July 2017

Alleviate HOL Blocking and DoS Attack in IMS Based NGN

Jashim Uddin ATM

Krishi Gobeshona Fondation, AIC Building, BARC Complex, Farm Gate, Dhaka, Bangladesh

Abstract: IMS based NGN uses TCP or UDP as transport layer protocol to carry multimedia data across different networks and even to carry signal among core components such as Call/Session Control Functions (CSCF). Due to using of TCP or UDP, IMS based NGN is facing TCP's SYN Denial of Service (DoS) attack, Head of Line (HOL) blocking crisis. To overcome these troubles the SCTP with its features can be used in IMS based NGN.

Keywords: Intercontinental marketing services; Transmission control protocol; Network; Denial of service attack; Head of line, Blocking problem; IMS based NGN; Stream control transmission protocol; Transmission control protocol etc.

I. INTRODUCTION

IMS based NGN convergence different types of access networks through common IP transport plane. It has separated signalling and media plane. The signalling plane handles the session control, authorization, security and policy based Quality of Service (QoS). The media plane handles message media encoding and transport issues [1]. It is consist of several SIP servers and other elements. The SIP servers are collectively referred to as Call/Session Control Functions (CSCF) such as Proxy-CSCF (P-CSCF), Interrogating-CSCF (I-CSCF) and Serving-CSCF (S-CSCF). The IMS signalling elements communicate over WAN via SIP over UDP or TCP. SIP does not have intrinsic security mechanism to ensure message reliability, privacy or validate the source. Due to uses of TCP or UDP as transport layer protocol, IMS facing DoS attack and HOL blocking crisis whereas adaptation of SCTP with its features can prevent the these tribulations in network [2].

II. METHODOLOGY

We have used qualitative and quantitative research methodologies to diagnose the existing system to prevent DoS attacks, HOL blocking and Spoof message crisis caused by TCP/UDP in the signaling core of the IMS-based NGN. Therefore, we will observe and compare unconventional transport layer protocol with its features to mitigate DoS and HOL blocking problems in Network. Hence we will use reputed publish journals and text book to collect secondary data [3].

III. LITERATURE REVIEW

3.1 IMS based NGN

Many organization and body confirmed and initiated IMS as the perfect architecture to meet the requirements for designing NGN environment. It allows real-time services and makes sure end-to-end QoS unlike traditional IP-based Networks. The IMS based NGN is a packet-based network based on SIP signaling protocol capable to provide multimedia services and quality of service enabled transport technologies. Service related functions are independent from the underlying transport-related technology [4]. It converge all kinds of network into a common IP platform. A user can get several services by one signing and authentication process.
IMS has two planes, functions independently:
Signalling Planes: controls QoS, security, authorization and session control and use SIP, SDP, UDP or TCP protocols.
Media Plane: controls message media encoding and transport issues and use following protocols:
For real-time media (audio/video): RTP or RTCP is used to carry media over UDP.
For near real-time streaming media: RTSP is used to carry media over UDP.
For non-real-time media: HTTP, FTP, SMTP deliver non real-time media over TCP.

There are three essentials elements in the IMS construction, Proxy (P), Serving (S) and Interrogating (I)-Call/Session Control Function. These Functions involved in registration and session establishment process and shape SIP routing machinery [5-7]. There are other elements in IMS architecture such as PDF, HSS, AS, BGCF, MGCF etc.

a) Home subscriber server (HSS):

Home Subscriber Server (HSS) is the central database that holds user's complete information like profiles, policies, subscriptions, preferences, etc. for a particular user and user identification (Authentication, Authorization and Accounting (AAA)). It keeps track of currently assigned S-CSCF of a user. All the Call/Session Control Functions (CSCFs) and Application Server can access the HSS [8-10].

b) Proxy-call/session control function (P-CSCF):

It is the entry point for users/UE to access the IMS core, placed at edge point of the IMS core. It authenticates users by checking central database resided in HSS and handles users' requests for accessing services [11]. After validating SIP signaling messages, it forwards SIP messages to other appropriate CSCF or discards it.

c) Serving-call/session control function (S-CSCF):

It is the main component of IMS core, performs session control services for IMS UEs. It is located at the users' home domain which registers users. When a user wants to access the application servers, it provides registration and filtering criteria services. Single registration is enough to access multiple services [12]. When a registered user requests for services, it checks the media parameters from the users' service profile and local policy resided in HSS. If it is okay, provides services or discard the communication. It is a place for routing the 'call' and triggering service. When needs to route the 'call', the S-CSCF receives "call" message from I-CSCF for routing as per destination address [13].

d) Interrogating-call/session control functions (I-CSCF):

When user registration comes up, it assigns the right S-CSCF for the User Equipment's. For registration, the I-CSCF cross-examines the HSS to be ensured the capabilities of S-CSCF and user profile [14]. When SIP request comes from other networks towards its S-CSCF, it acts as proxy and route that SIP request to the appropriate S-CSCF. When S-CSCF needs to route a call to another service provider, the I-CSCF retrieve the address of the targeted provider's I-CSCF from DNS, defined in (International Telecommunication Union).

3.2 Session Initiation Protocol (SIP)

SIP is the application layer transaction-oriented text based signalling protocol. It solves the primary inconvenience in establishing of real-time communication session. It develops communication systems and provides a framework to build different type of communications and large scale carrier next generation network such as IMS. SIP establishes, modifies and terminates the multimedia sessions among participants. It does not depend on the media transport. It uses TCP or UDP as the transport layer protocol for passing signal and uses RTP or RTCP over UDP for transporting media data [15]. It also provides transaction-level state machines and timers for reliable transportation by invoking retransmission if packet is lost.

3.3 Introduction to TCP

TCP is a connection oriented protocol which carries reliable end-to-end byte stream over untrustworthy Internetworking. It supports multiplexing, error recovery, flow control, connection establishment and termination, end-to-end ordered data transfer and segmentation etc. TCP supported machine has TCP transport entity which accepts data streams from local process and split the data streams into pieces of 1500 bytes for being fitted in a single Ethernet frame (including IP and TCP header). When an IP datagram with TCP data arrives at a machine, the IP datagram handed over to the TCP entity that reconstructs the original byte of streams. The IP does not guarantee the correct delivery of data whereas TCP guarantees the delivery of data properly by retransmission or time out process as needed. The TCP makes segments either from several writes into one segment or can split one write into several segments. Each TCP segment contains 20-byte header [16].

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 7, July 2017

<table>
<tr><td colspan="32">&lt;-------------------32 Bits-------------------&gt;</td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td colspan="16">Source Port</td><td colspan="16">Destination Port</td></tr>
<tr><td colspan="32">Sequence Number</td></tr>
<tr><td colspan="32">Acknowledgement Number</td></tr>
<tr><td colspan="4">TCP Header</td><td colspan="3">Reserved</td><td>NS</td><td>CWR</td><td>ECE</td><td>URG</td><td>ACK</td><td>PSH</td><td>RST</td><td>SYN</td><td>FIN</td><td colspan="16">Window Size</td></tr>
<tr><td colspan="16">Checksum</td><td colspan="16">Urgent Pointer</td></tr>
<tr><td colspan="32">Options (0 or more 32 bits words)</td></tr>
<tr><td colspan="32">Data (Optional)</td></tr>
</table>

Table 1: TCP header format.

Every segment starts with 20 bytes fixed header format. After the fixed header format, there might have 0-32 bits length's "options" header. After the "options" header, there will have optional location for data in segment or not and the highest length of data will be 65,535-20-20=64,495 data bytes, where the first 20 bytes is for IP header and second 20 bytes is for TCP header (Fixed) (Table 1).

3.3.1 TCP data transfer:

The TCP entities use sliding window protocol and timer. TCP starts timer when a sender starts to transmit data. The sender and receiver's transport entity buffer data. Suppose, if sender transmits 2048 byte segment and if it is received accurately, the receiver will acknowledge the segment. If the application does not move data from buffer that implies the receiver still has another 2048 bytes buffer space. As a result the receiver will announce a window of 2048 byte starting at the next expected byte. This process can be run unless the receiver's buffer is full (receiver's window size=0). In this circumstance, the sender will not transmit any data unless the urgent flag is used or sender needs to transmit 1 byte segment to convince the receiver for re-announcing next expected byte and window size.

3.4 User Datagram Protocol (UDP)

UDP is connectionless transport protocol. It does not need to establish prior connection before data transfer begins. It provides unreliable, unordered data transfer among the participants, and uses multiplexing capabilities. It does not support congestion control mechanism, error recovering and retransmission upon receiving of bad segment. UDP transmits segment and each segment has 8 byte header followed by the payload (optional).

3.5 Security and Congestion Control Problems

SIP is the core signalling protocol in the IMS used to communicate among core components (P-CSCF, S-CSCF, I-CSCF and HSS etc.) over TCP or UDP over IP. The IMS signalling core elements communicate others elements over WAN by using UDP due to network performance which making the IMS signalling core hottest area to be attacked. Since SIP does not have intrinsic security mechanism to ensure message reliability, privacy or validate the source which urging to provide sufficient internal security.

To prevent these situations the following mechanism can be used in the IMS signalling core system:

1. Firewall can be used on top of the IMS signaling core. It prevents network based attacks but not prevents harmful SIP messages which pass through security devices and reduces large network's performance.
2. ALG can be used on top of the IMS signaling core. It provides nearly same solution as firewall along with prevention of passing of harmful SIP messages through IMS core but reduces network's performance.
3. IPSEC among IMS signaling core elements can be used to provide secrecy, reliability and source authentication services between IMS core elements. But the usage of IPSEC causes the network traffic invisible to the service provider.
4. TCP can be used instead of UDP for SIP traffic: UDP is stateless, so it is more attackable by deceiving messages. TCP uses three way handshakes procedure that will protect deceived SIP message to proceed. But it reduces the network performance.
5. Interruption detection or prevention system can be introduced: but it is costly and need expert to operate.

In conclusion, any single procedure explained above cannot ensure the IMS core elements' security. If an unpredicted and significant multimedia traffic is increased over the framework without end-to-end congestion control (for UDP) it might not be scaled and could raise congestion collapsed situation. To stay away from this situation, the IMS has introduced end-to-end Quality of Service (QoS). But the following reasons may lead severe traffic congestion.

1. The Head-Of-Line (HOL) Blocking
2. Denial of Service (DoS) Attack

3.5.1 Head of line (HOL) blocking:

The web browser displays web pages from the web server. HTTP/1 supports constant and channel connections. The web browser establishes a new connection and sends HTTP GET request with the desired URI to the server. Then server replied the HTTP GET request by sending the page with contents to the web browser. But there might have multiple independent embedded objects with different URIs in the received page. As a result, the web browser parses the contents of these different URIs and HTTP sends pipelined HTTP GET request for each URI to the server again and server responses with contents accordingly [17]. TCP and UDP are commonly used as the transport layer protocol. TCP offers single sequential byte stream to an application. The TCP buffers the data and wait until the buffer is full. The usage of TCP for the web transport may block the transfer of other successfully received independent web objects to the web browser. Suppose if a TPDU contains an independent embedded web object and it is lost in the network, the lost TPDU may block the delivery of other successful received independent web objects. This problem is known as Head-Of-Line (HOL) blocking since the TCP does not have ability to differentiate the independent embedded application level objects in its transport and delivery mechanisms. When missed TPDU is being successfully retransmitted then the data in the receiver buffer is ordered and delivered to the application. On the other hand, if the next received TPDU belongs to a different application object not the earlier lost TPDU(s), which is increasing the buffer space with the loss probability in the transmission path and increasing the number of independent objects to be transferred [18]. So the unnecessary filling of the receiver's end buffer space causing Head-Of-Line (HOL) blocking problem. If data loss rate is high or channel bandwidth is low, the Head-Of-Line (HOL) blocking problem is increased severely in domains. Uses of web browser and other web applications on mobile phones increasing the Head-Of-Line (HOL) problem significantly. As a result, delay is increasing dramatically towards users. To mitigate HOL blocking, the web browser generally open several TCP connections with a web server. Therefore, the web browser sends HTTP GET request via these multiple TCP connections to the server to stay away from HOL blocking problem between the matching responses. But these established several TCP connections between the web browsers and server do not remove the HOL blocking problem because there are still multiple independent objects which are being transmitted through one of the parallel TCP connections. In addition, if the sender uses several TCP connections for transmitting a single application's data then many negative consequences for both the application and the network may be introduced:

1. **Aggressive behaviour during congestion:** TCP uses window mechanism to control the network congestion. If network congestion is detected then the sender reduces the congestion window to half of its size. If a single application uses multiple TCP connections for transmitting data, then these connections gets an unfair share of the available bandwidth in the path because all of the TCP connection may not face loss of data during network congestion in the transmission path.

2. **Absence of integrated loss detection and recovery:** Since the web objects generally smaller size, so for each HTTP response from the server side sends just a few TPDU. Since there is no adequate number of duplicate ACKs in the server side, so it will not be able to trigger a fast retransmit. As a result, the web server will use expensive timeout to recover the lost TPDU. Since the TCP connection is persistent and pipelined requests, that is why the multiple separate connections cannot share ACK information for loss recovery.

3. **Increased load on web server:** For each connection there is Transmission Control Block (TCB), stores information regarding connection in server. Usage of parallel connections between server and client increases TCB processing load on the server. As a result, high load in server can lead to reject incoming TCP connection due to lack of available memory resources.

4. **Increased connection establishment latency:** TCP uses three way handshake procedures to set up a connection before data transfers. This handshake wastes one roundtrip to open every connection to the same server. Since TCP apply timeout event to recover the loss data or retransmission then if any data is loss during the connection setup then it could be expensive. So increase of number of parallel connections is increasing the loss possibility during connection establishment. Consequently, it is increasing the overall average transfer time [19].

3.5.2 Denial of service (dos) attack:

Denial of Service attacks (DoS) generally arrived when hackers send lot of requests to the server at a time causing server out of service is called denial of service attacks. TCP uses three way handshakes procedure to establish a connection between the participants i.e. server-client architecture. The server side executes LISTEN and ACCEPT primitives and wait passively for incoming connection and client side execute CONNECT primitive with SYN=1 and FIN=0 flags and wait for server's SYN/ACK response. After receiving the client's SYN request, the web server allocates memory space for storing client's SYN request information and send back SYN/ACK to the client and wait for client's ACK in replying to server's SYN/ACK. During this gap, the TCP connection is half open state and will remain half open if server's SYN/ACK expires with no client's ACK message. Therefore, the server is using resources for those TCP half open connections unnecessarily [20]. So if a cruel/malicious user manipulates a coordinated SYN attack, means if a lot of (1000's) nasty hosts sends SYN flood with IP-Spoofed SYN request to a predestined web server, the web server needlessly allocates memory space for a lot of half open TCP connections. Consequently the servers' memory space will be filled up by these fictitious SYN requests, which are causing the server to deny even legal users to use its resources. UDP sends large number of fake UDP packets which consume bandwidth. In IMS architecture, the S-CSCF is responsible for providing services to the users, so it needs to maintain the signalling path. Hence any kinds of attack on S-CSCF will hamper in providing services to users. In addition, when a device is try to be registered with P-CSCF, the I-CSCF perform SIP registration and fix S-CSCF Thus, DoS attacks on I-CSCF can causes severe condition in IMS Network such as hamper user registration process etc. As SIP is the signalling protocol in IMS therefore the IMS components must trust the SIP messages and process the SIP messages from possible attackers. Therefore, DoS attacks can be form of abnormal packets, manipulated SIP states and simple REGISTER or INVITE flooding.

3.6 Stream Control Transmission Protocol (SCTP)

SCTP initially introduced for carrying the PSTN signaling over IP networks. It is consists of the shared features of TCP and UDP. Unlike TCP and UDP, the SCTP has unique characteristics such as Multi-streaming and Multi-homing and enhanced error detection and correction capabilities. SCTP is a reliable, message stream oriented end-to-end transport layer protocol which supports TCP's ordered delivery of data and congestion control mechanism and UDP's unordered delivery of data but in reliable way and can identify message boundary.

3.6.1 SCPT's packet format:

12 Bytes is common header and 4 Bytes is for each chunk. So for one chunk, the total memory overhead is 12+4=16 Bytes (Table 2).

<table>
<tr><td colspan="32"><----------------32 Bits----------------------></td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td colspan="16">Source Port</td><td colspan="16">Destination Port</td></tr>
<tr><td colspan="32">Verification Tag</td></tr>
<tr><td colspan="32">Checksum</td></tr>
<tr><td colspan="8">Chunk 1 Type</td><td colspan="8">Chunk1 Flag</td><td colspan="16">Chunk 1 Length</td></tr>
<tr><td colspan="32">Chunk1 Data</td></tr>
<tr><td colspan="32">_ _ _</td></tr>
<tr><td colspan="8">Chunk N Type</td><td colspan="8">Chunk N Flag</td><td colspan="16">Chunk N Length</td></tr>
<tr><td colspan="32">Chunk N Data</td></tr>
</table>

Table 2: SCTP header format.

3.6.2 SCTP association/connection establishment:

Figure 1, the SCTP uses four way handshakes procedure to establish an association/connection between server and client in server/client architecture by using INIT, INIT-ACK, COOKIE ECHO and COOKIE-ACK events. Here, Host A is sending an INIT to B. Then Host B replies INIT-ACK to Host A with cookie along with additional information. Upon receiving INIT-ACK, Host A sends back COOKIE-ECHO to Host B with A's application data. At last the HOST B verifies the received COOKIE-ECHO and send back COOKIE-ACK to Host A, allocates resources and establish the association. "With SCTP's four-way handshake, a web client that initiates an association must maintain state before the web server does, avoiding spoofed connection request attacks.

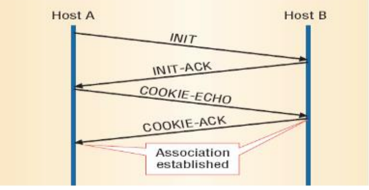

Figure 1: SCTP association.

3.6.3 SCTP's multi-streaming feature

In Figure 2, SCTP can establish multiple streams on an association/connection between server and client. A stream is a unidirectional channel that supports ordered/unordered delivery of messages. The multi-streaming feature maintains logical separation of independent object which is transmitted through different streams. Each stream maintains or share same type of congestion control mechanism of TCP [21]. There may have 'n' numbers of independent streams under an association/connection against 'n' numbers of embedded objects which reside in web server. When a stream faces data losses, other unaffected streams transfer data continuously to the corresponding application. In SCTP, each stream among multi-stream has its own Streaming Sequence Number (SSN). SCTP contains U flag that determines ordered or unordered option of data chunk. Therefore, if U flag is '1', the receiving end does not need to wait for receiving any ordered or any lost data packet. SCTP uses global TSN across streams to detect and recover loss data. If there are loss of data in one stream, SCTP uses ACKS for detecting that lost data on other streams.

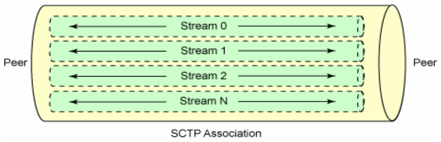

Figure 2: SCTP association.

IV. RESULTS AND CALCULATION

4.1 Prevention of Head of Line Blocking (HOL)

To prevent HOL blocking, the network system can use SCTP with its multi-streaming functionality instead of TCP/UDP. So if any one stream among multiple streams in one association faces data loses, the other unaffected streams can continue to transfer and deliver data to the corresponding application. TCP stores information for each successful/unsuccessful connection through Transmission Control Block (TCB) in web server (Table 3). To proof the SCTP's multi-streaming feature that can magnify the network performance and prevent HOL blocking in network, we are comparing TCB of both TCP and SCTP's connections. Generally the size of TCP TCB is very high and is about ~700 bytes and the size of One (1) SCTP TCB= 2* TCP TCB in Byte.... (1). the memory overhead for a pair of SCTP stream is 16+16=32 bytes=1.6~2 parallel TCP connection's overhead (here, for one TCP connection and one SCTP stream's memory overheads are 20 Byte and 16 Byte respectively). SCTP is more useful while TCP creates a lot of parallel connection to deliver several independent embedded objects in web in client-server architecture [22]. So memory requirements in terms of several parallel TCP connections and One (1) SCTP association/connection of several streams are as follows which has been collected from P Natarajan.

Formula 1:
The size of One (1) SCTP TCB= 2* TCP TCB in Byte
Formula 2:
For “n” number of parallel TCP connections:
= [n*(TCP TCB size)] bytes
Formula 3:
For One SCTP association with “n” pairs of streams
= [(SCTP TCB size) + (n* 32)] Bytes
= [(2* TCP TCB size) + (n*32)] Bytes

Comparison of memory requirements in terms of several parallel TCP connections and One (1) SCTP association of several streams.

Let TCP TCB Size in Byte	No. of Independent Embedded Objects Reside in Server.	No. of TCP’s Parallel Connection between Server and Client (n).	Formula 2= [n* TCP TCB Size] Bytes for TCP’s Connection	No. of SCTP’s Several Streams in one (1) Association between Server and Client (n).	Formula 3= [(SCTP TCB size) + (n* 32)] Bytes =[(2* TCP TCB size) + (n*32)] Bytes for SCTP’s one Association
700	1	1	700	1	1432
700	2	2	1400	2	1464
700	3	3	2100	3	1496
700	4	4	2800	4	1528
700	5	5	3500	5	1560
700	6	6	4200	6	1592
700	7	7	4900	7	1624
700	8	8	5600	8	1656
700	9	9	6300	9	1688
700	10	10	7000	10	1720
700	11	11	7700	11	1752
700	12	12	8400	12	1784
700	13	13	9100	13	1816
700	14	14	9800	14	1848
700	15	15	10500	15	1880
700	16	16	11200	16	1912
700	17	17	11900	17	1944
700	18	18	12600	18	1976
700	19	19	13300	19	2008
700	20	20	14000	20	2040
700	21	21	14700	21	2072
700	22	22	15400	22	2104
700	23	23	16100	23	2136
700	24	24	16800	24	2168
700	25	25	17500	25	2200
700	26	26	18200	26	2232
700	27	27	18900	27	2264
700	28	28	19600	28	2296
700	29	29	20300	29	2328
700	30	30	21000	30	2360
700	31	31	21700	31	2392
700	32	32	22400	32	2424
700	33	33	23100	33	2456
700	34	34	23800	34	2488
700	35	35	24500	35	2520
700	36	36	25200	36	2552

Table 3: Memory consumption comparison of SCTP and TCP for independent embedded objects in server while the size of TCP TCB is ~700 bytes.

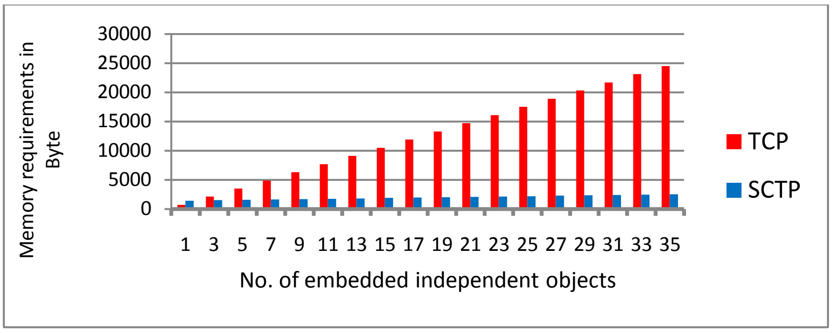

Figure 3: Graphical representation of memory consumption comparison of SCTP and TCP for independent embedded objects in server while the size of TCP TCB is 700 bytes.

To compare the performance of TCP and SCTP, paired sample t-test is used. The sample consists of required memory space of using TCP and SCTP. Figure 3 we have used 700 Bytes for the size of TCP TCB. It is also possible to use different size of TCP TCB such as 800, 900, 1200 etc. bytes. Their relative performance is assessed using the test. The result is as follows:

		Paired Differences					t	df	P-value
		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference				
					Lower	Upper			
Pair 1	TCP - SCTP	10958.0	7037.817	1172.969	8576.745	13339.255	9.342	35	0.000

Table 4: Paired samples test.

The P-value is 0.000 which indicates the test is highly significant. Therefore, required memory space for TCP and SCTP is significantly different. And the sign of the mean difference of space's requirement of TCP and SCTP is positive which means that TCP requires more space than that of SCTP for same number of embedded objects in web server (Table 4).

4.2 Solution of DoS Attack Caused by TCP's SYN Flood Attack

To prevent TCP's DoS/SYN flag attack in server/client architecture, SCTP can be used as a perfect transport layer protocol instead of TCP in IMS based NGN.

Unlike TCP's connection establishment procedure, the SCTP uses four way handshakes procedure to establish an association. Server sends back COOKIE-ACK (Server's side message executed at 4th level) in response to COOKIE-ECHO (Client's side message executed at 3rd level) and hence establishes connection and allocates memory for successful incoming connection request which has been described in section 3.6.2. So there is no room for half open state, no chance to allocate memory space for unsuccessful connection attempt, hence no chance to make TCP's SYN attack scenario to allocate server's memory space unnecessarily that will lead to prevent DoS attack in network, defined in section 3.5.2.

V. DISCUSSION

According to the comparison of memory requirements in terms of several parallel TCP connections and one (1) SCTP association with several streams shown in Table 1, usage of SCTP with its multi-streaming feature can reduce the memory consumption tremendously for establishing connection for several embedded objects rather than that of TCP. As a result the SCTP's multi-streaming feature that can magnify the network performance and can prevent HOL blocking problem in network. Also, four ways handshakes procedure of SCTP prevents half open connection state or TCP's SYN flag flood attacks condition ensuring not to take place Denial of Service (DoS) attacks in networking system. Therefore, we can conclude that using of SCTP instead of TCP or UDP in IMS based NGN can reduce the security and congestion control problem in network.

VI. REFERENCES

1. A Aliya, F Muddassar, et al. Attack analysis & bio-inspired security framework for IP multimedia subsystem. Conference on Genetic and Evolutionary Computation, 2008; 161-162.
2. B Nilanjan, A Anup, et al. Enabling SIP-based sessions in ad hoc networks. Wireless Networks 2007; 13: 461-479.
3. B Gavin, B Declan, Developing multiparty conferencing services for the NGN: towards a service creation framework. International Symposium on information and Communication Technologies 2004; 90: 50-55.
4. JL Chiang, NGN: Next Generation Network. MNET Lab Meeting 2006.
5. F Ali, N Heiko, et al. A cooperative SIP infrastructure for highly reliable telecommunication services. International Conference on Principles, Systems and Applications of IP Telecommunications 2007; 29-38.
6. EL Filho, GT Hashimoto, et al. An IMS Control and Access Layer PR-SCTP Based Network. Networking and Services 2008; 63-66.
7. HA David, B Jason, et al. Issues with network address translation for SCTP. SIGCOMM Computer Communication 2009; 39: 23-33.
8. HT Michael, CJ Russell, et al. Security issues with the IP multimedia subsystem (IMS). Middleware for Next-Generation Converged Networks and Applications, MNCNA 2007; 1-6.
9. International Telecommunication Union. Next Generation Networks Global Standards Initiative (NGN-GSI) 2008.
10. K Humaira, P Brad, et al. SCTP versus TCP for MPI. Conference on Supercomputing, Conference on High Performance Networking and Computing IEEE Computer Society 2005; 1-14.
11. M Mehdi, C Noel, Adopting IMS in WiFi technology. Mobile Technology, Applications, and Systems and the 1st international Symposium on Computer Human interaction in Mobile Technology 2007; 325-331.
12. MT Masonta, OJ Oyedapo, et al. Mobile Client for the Next Generation Networks. Broadband Communications, Information Technology and Biomedical Applications 2008; 274-279.
13. N Preethi, AD Paul, et al. Multi-streamed web transport for developing regions. Networked Systems for Developing Regions 2008; 43-48.
14. N Preethi, IR Janardhan, et al. SCTP: an innovative transport layer protocol for the web. World Wide Web, New York, USA 2006; 615-624.
15. O Wendell, CCNA INTRO Exam Certification Guide. Indianapolis, USA: Cisco Press 2003; 142-167.
16. P Podhradsky, E Mikoczy, et al. NGN platform architecture and its adaptation to the evolution trends. IEEE Systems, Signals and Image Processing 2007; 331-334.
17. S Noëmie, Y Chunyang, et al. An NGN middleware based on an enhanced IMS. Middleware for Next-Generation Converged Networks and Applications 2007; 1-7.
18. S Robert, SIP: basics and beyond. Queue 2007; 5: 22-33.
19. S William, Data and Computer Communications. Prentice-Hall 2008; 663-702.
20. AS Tanenbaum, Computer Networks. Pearson Education 2007; 524-555.
21. T Feng Cheng, C Huang Lin, et al. Improvement of SCTP Performance during Handshake Process. Advanced information Networking and Applications IEEE Computer Society 2008; 445-450.
22. T Ivan, IP Multimedia Subsystem (IMS) signaling core security. Information Security Curriculum Development 2008; 59-63.