Conference Papers

School of Science and Computing

2008

A Study of SIP Based Instant Messaging Focusing on the Effects of Network Traffic Generated Due to Presence

Finnian McKeon
Institute of Technology, Tallaght, finnian.mckeon@ittdublin.ie

Follow this and additional works at: http://arrow.dit.ie/ittscicon

Part of the Computer Engineering Commons

A STUDY OF SIP BASED INSTANT MESSAGING FOCUSING ON THE EFFECTS OF NETWORK TRAFFIC GENERATED DUE TO PRESENCE

Finnian McKeon
Institute of Technology, Tallaght
Dublin, Ireland

ABSTRACT

The increasing usage of Instant Messaging(IM) applications worldwide continues to grow year on year. However, larger numbers of users creates problems with greater network traffic, interoperability and proprietary protocol dependence. Network traffic due to presence information being exchanged between users and servers needs to be considered, but is it possible to estimate or measure this traffic? There have been very few studies performed on the impact of presence traffic. The purpose of this research is to implement a SIP/SIMPLE network environment that will allow the evaluation of key metrics such as the latency and throughput of the network under a number of scenarios. It will be shown that the amount of network traffic generated due to presence subscriptions can grow large, and the effects of this on network performance will be analysed. From this analysis, it will be shown that SIP/SIMPLE presence traffic has a steady, predictable impact on the performance of the network. It will also be demonstrated that other nodes on the network that are transmitting non SIP/SIMPLE traffic do not suffer significant throughput degradation due to SIP/SIMPLE presence traffic.

1. INTRODUCTION

The popularity of Instant Messaging continues to grow year on year, with an ever increasing number of devices that support the technology. However there is also an increasing amount of different protocols that offer Instant Messaging.

A protocol that is gaining widespread attention recently is the Session Initiation Protocol(SIP), due to its open nature and its suitability to services such as Voice over IP (VoIP). One of the concerns identified with SIP based Instant Messaging is the amount of traffic that is generated due to presence messages being sent across domains and the corresponding responses.

In this paper a study of SIP based Instant Messaging will be carried out with particular emphasis on presence traffic. It will be shown that the amount of network traffic generated due to presence subscriptions can grow large but that it has a predictable impact on the performance of the network.

It will be shown that SIP based Instant Messaging does not significantly effect other nodes on the network that are transmitting non-SIP traffic. It will also be established that `rush hour' traffic had a minimal impact on the overall throughput of the network in the test environment implemented.

The amount of cross domain traffic captured during the simulations for this research will be analysed. This traffic will be compared to network traffic estimations that have been published for SIP based Instant Messaging.

2. SIP/SIMPLE

There have been many specifications written that relate to Instant Messaging and presence using SIP. Rosenberg (2007) describes how together these specifications are known as SIP for Instant Messaging and Presence Leveraging Extensions(SIMPLE). SIP/SIMPLE can be seen as a broad catch all term that covers a large number of specifications.

As SIP/SIMPLE is based on SIP, this means that clients can either use the Transmission Control Protocol(TCP) or the User Datagram Protocol(UDP) to connect to SIP servers, (Rosenberg et al. 2002). One of the primary goals of TCP is to provide reliable ordered data transmission (Postel 1981), however this reliability can have a significant impact on performance. TCP checks for errors in packets and if errors are found, packets are then retransmitted. UDP by contrast does not check for errors and does not guarantee delivery, this enables it to offer greater speed than TCP.

Nahum et al. (2007) found that using TCP can reduce performance by a factor of three compared to using UDP. However, it is not advisable to recommend UDP over TCP for every SIP/SIMPLE environment, in a mission critical scenario such as a hospital, it would be more preferable to guarantee the delivery of a message than to increase the speed of message transmission.

2.1 SIP/SIMPLE Presence Traffic

SIP was originally designed as a protocol to establish sessions between one or more parties, (Rosenberg et al. 2002). In general, the majority of SIP messages are sent at the start of the communication to initiate the session and at the end to terminate the session. SIP was not designed for sending large amounts of messages throughout the session. However, for SIP/SIMPLE sessions, a large amount of presence change messages can be sent during the session depending on a users availability. As mentioned previously, each presence change by a user requires a PUBLISH message being sent to the SIP/SIMPLE server and multiple NOTIFY messages being sent by the SIP/SIMPLE server. In addition to the above messages, each successfully delivered message in SIP is replied to by a 200/OK message.

In SIP/SIMPLE a PUBLISH message is made up of a standard SIP message with XML content to indicate the new presence of the user. The XML must adhere to the Presence Information Data Format (PIDF) as specified by Sugano et al. (2004). For example, in order for UserA to indicate to a SIP/SIMPLE server that they have changed their presence they need to send a PUBLISH message. Niemi (2004) defines the PUBLISH message as an extension of the SIP protocol in RFC 3903. Once the user has published its presence state using the PUBLISH message, the SIP/SIMPLE server will send notifications of the state change using RFC3856, see Figure 1 below.
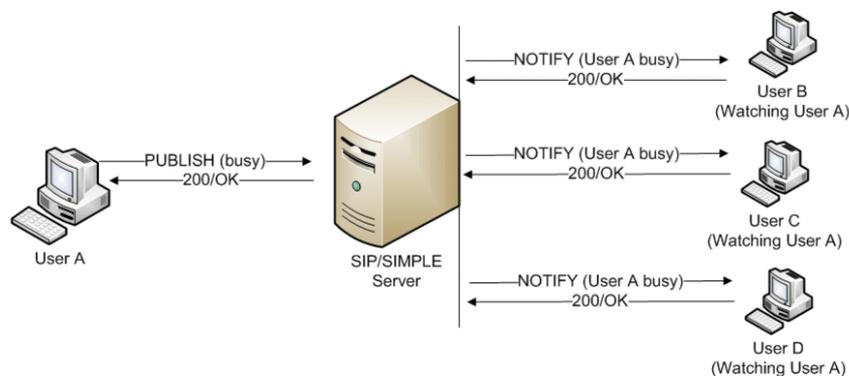

Figure 1. User A changes presence, users watching are notified

2.2 SIP/SIMPLE Network Environment Setup

It was the intention of this research to generate presence data by simulating multiple users logged on to SIP/SIMPLE servers and to provide an analysis of the presence data generated between the two domains. The effect of this presence traffic on the network throughput and latency was measured.

In order to analyse the network traffic generated by SIP/SIMPLE presence messages more closely, it was necessary to construct a SIP/SIMPLE network environment. It was the intention to create a network that simulated a real life scenario as closely as possible. This environment was made up of two separate IP networks, on each network there was one SIP/SIMPLE server running, this allowed the creation of clients on one network that were subscribed to the presence changes of clients on the other network, (see Figure below). This resulted in cross network/domain traffic, which was measurable.
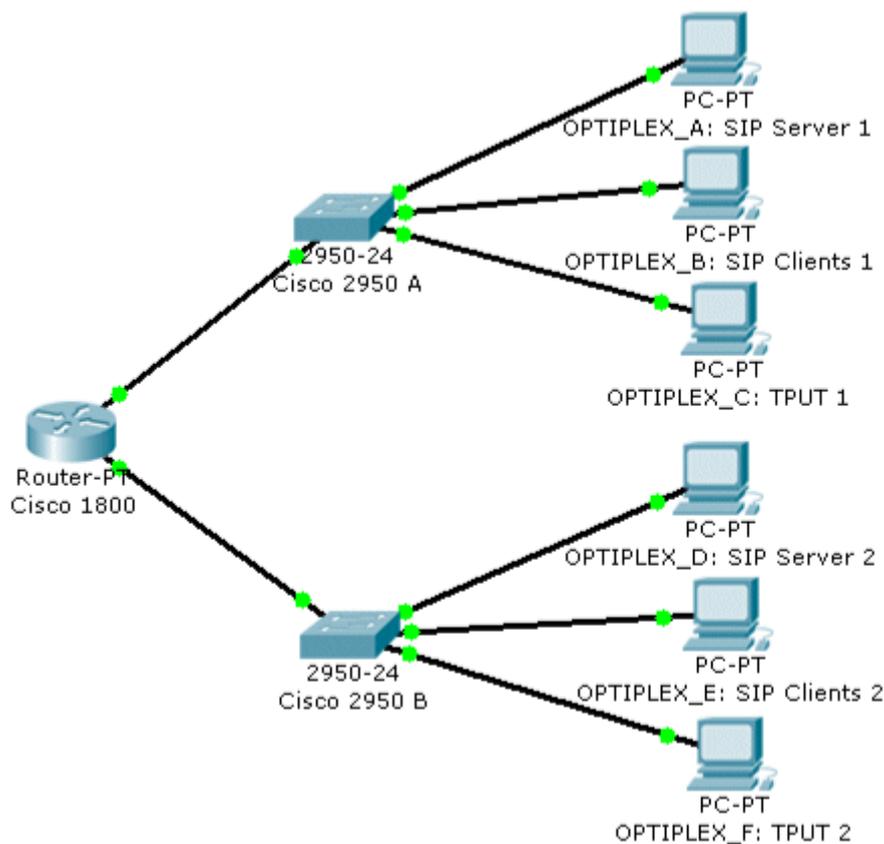

Figure 2. SIMPLE test network layout

The network consisted of the following components:
- 1 Cisco 1800 Series Router
- 2 Cisco 2950 Series Switches
- 6 Dell OPTIPLEX GX620 workstations

In order to minimise unrelated network traffic, the test network used for these experiments was not connected to any other networks or servers apart from those displayed in Figure 2. All cabling used for network connections were CAT 5E network cables. The theoretical maximum throughput of the network was

100Mbits/sec, however this is never obtainable in a real life scenario due to a number of factors such as time taken to route the packets, and line noise. The maximum possible network throughput recorded with no SIP/SIMPLE users in this environment was 94.51Mbits/sec.

2.3 SIMPLE Presence Traffic Generation Application

A Java command line application was written which simulates multiple users logging on to a SIP/SIMPLE network, adding users and changing their presence at random intervals. The random interval is generated using the java.util.Random class, which generates pseudo-random numbers. This application was written using the open source Java API for Integrated Networks (JAIN) SIP library, version 1.2 which is specified in Java Specification Request (JSR) 32.

Houri et al. (2007) identified that the presence traffic generated between domains can grow very large due to the number of users changing presence and the amount of messages being transferred between domains. Houri et al. (2007) have outlined a number of scenarios in their document to illustrate this. The authors defined the parameters of typical instant messaging activity over a period of time from their extensive experience with instant messaging systems.

In order to make the simulations in this research as close as possible to real life scenarios, the parameters used in the simulations run have been based on these figures wherever possible.
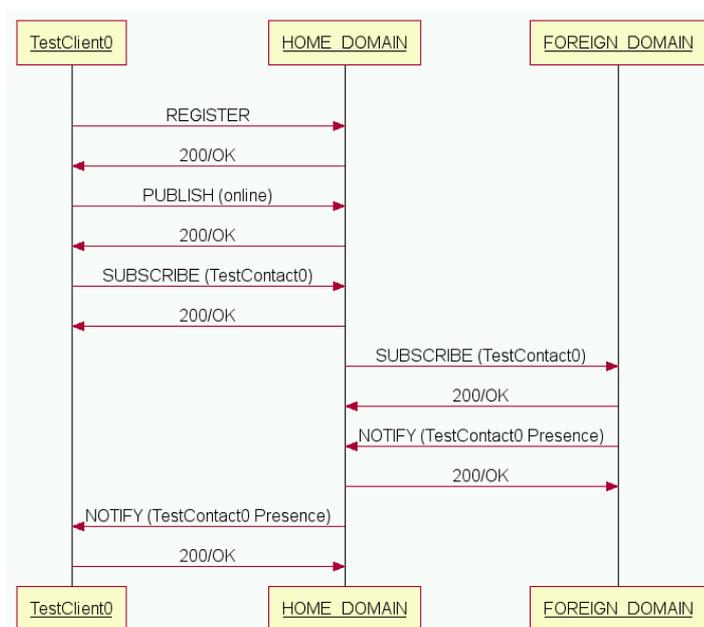

Figure 3. Test client sequence diagram

The sequence diagram in Figure 3 illustrates the activity of each client thread during the initial stages of the simulations. The simulations start with a two minute warm up period, this allows the operator to ensure everything is in place before the client threads start creating activity on the system. Following the warm up period, Figure 3 shows the sequence of events for a client thread, the client registers with the Home Domain, publishes their status as online, subscribes to a user on the Foreign Domain and receives notification of their presence status.

The key difference between the parameters used in the simulations in this research and those estimated by Houri et al. (2007), is the timeline. In order to run meaningful simulations, it was decided that after the initial two minute warm up period, one second of time in the simulation should be equivalent to one minute. For

instance, after registering with the SIP/SIMPLE server, each client thread subscribes to four users from the foreign domain and each client thread changes their presence status three pseudo-random times per minute. Each client thread refreshes their subscription and registration after sixty seconds. The total length of the subscription is eight minutes, which is equivalent to eight hours in estimations of Houri et al. (2007). After the eight minutes, each client un-subscribes from their four subscriptions and un-registers from the SIP/SIMPLE server. The total length of the simulations is ten minutes, a two-minute warm period followed by eight minutes of activity.

3. ANALYSIS OF SIP/SIMPLE PRESENCE TRAFFIC

3.1 Network Throughput

Houri et al. (2007) discuss the need to add `rush hour' calculations to their problem statement for the start and the end of a day. A 'rush hour' period for a SIP/SIMPLE system is the period of high message transfer activity, this period occurs at the start and end of a session.

The 'rush hour' scenario was modeled in this research by examining the traffic after the two minute warm up period, this was the period where the client threads were registering with the SIP/SIMPLE server, subscribing to four contacts on the foreign domain and receiving their current presence status. It is equivalent to 9am in an office block when most users log in to applications in the morning and get the current status. As the application developed for this research included a warm up period and displayed a number of counters, it was clear when the 'rush hour' period was going to start.

To calculate the network throughput during normal SIP/SIMPLE operation, the network throughput was measured five times after the `rush hour' period. During this period some client threads may have been changing their presence status and receiving updates, but not all of them as during the 'rush hour'. The average of these five calculations is used to compare against the network throughput during 'rush hour'. It was established that as the number of users on the network increases there is a small but noticeable decrease in the network throughput. It was also established that during the `rush hour' period the network throughput figures are marginally more affected than the figures for network throughput after the `rush hour' period, see diagram below:
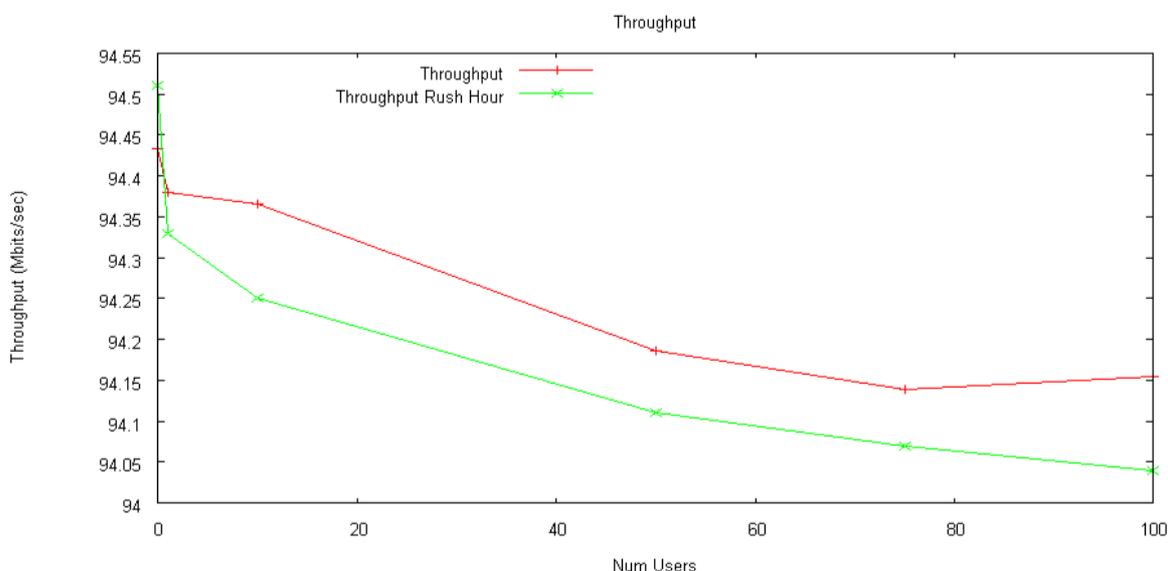

Figure 4. SIP/SIMPLE network throughput

In the simulations run for this research, above the level of seventy five users per domain the SIP/SIMPLE servers began to struggle with the amount of processing that was being carried out. The workstations hosting the SIP/SIMPLE server application experienced maximum levels of CPU usage and this could explain the marginal increase in throughput at this level.

Although there is a slight degradation of network throughput as the number of users increases, it is not a significant enough amount to act as deterrent to using a SIP/SIMPLE deployment.

3.2 Message Latency

The message latency of SIP/SIMPLE in this environment was examined in order to determine the effect of the increase in users on the length of time taken to receive a response to a SIP/SIMPLE message. For the purposes of this research message latency is calculated from the time a request leaves a client to the time the response arrives back at the client from the requested server.

In Figure 5 below, the latency for each message type is plotted; REGISTER, SUBSCRIBE and PUBLISH. The average latency of the three messages is also plotted. It can clearly be seen that as the number of users increases the latency of the messages increases. Up to the point of sixty users per domain, the latency increases at predictable and steady level, however after this point the latency increases sharply and continues to increase at high rate.

Above the level of sixty users, the SIP/SIMPLE server began to struggle to handle the amount of messages being sent to it by the client applications. It is difficult to determine exactly why the latency increased to such a level, the SIP/SIMPLE server used was proprietary software so it was not possible to examine the code. However, high levels of CPU usage were observed on the workstations that hosted the SIP/SIMPLE servers, so it is likely that the workstations were not powerful enough to handle the high level of traffic and this led to longer message latency as the number of users increased.

It can also be seen in Figure 5 that the message that has the largest latency is the SUBSCRIBE message in this system. This can probably be attributed to the fact when the SIP/SIMPLE server receives a SUBSCRIBE message, it needs to do extra processing to find out the current presence of the user that the subscription is for in order to generate a NOTIFY message.
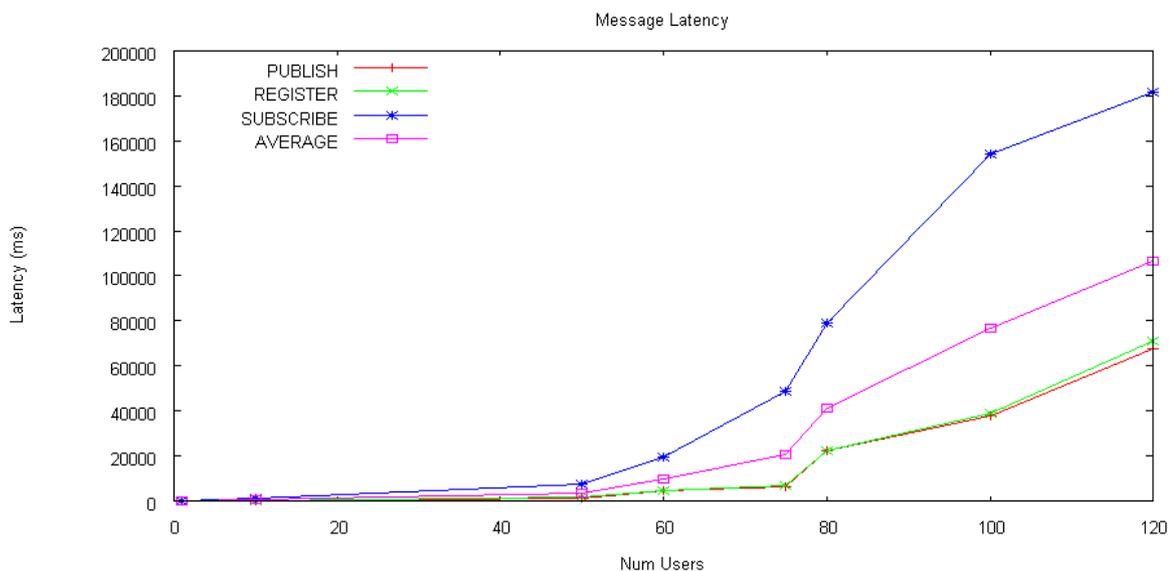

Figure 5. SIP/SIMPLE message latency

While this system was operating correctly, it can be said that as the number of users registered and generating presence traffic on the SIP/SIMPLE servers increases, the message latency also increases.

3.3 Cross Domain Traffic Analysis

During the execution of the experiments for this research, the traffic that passed between the domains was captured using Wireshark. The main reason for this was to compare the figures estimated by Houri et al. (2007) with the amount of actual cross domain traffic captured in this test environment.

Figure 6 shows the amount of data transmitted between the two domains in Megabits. It can be seen in the graph that the amount of data transferred between domains increases in a linear and predictable manner.
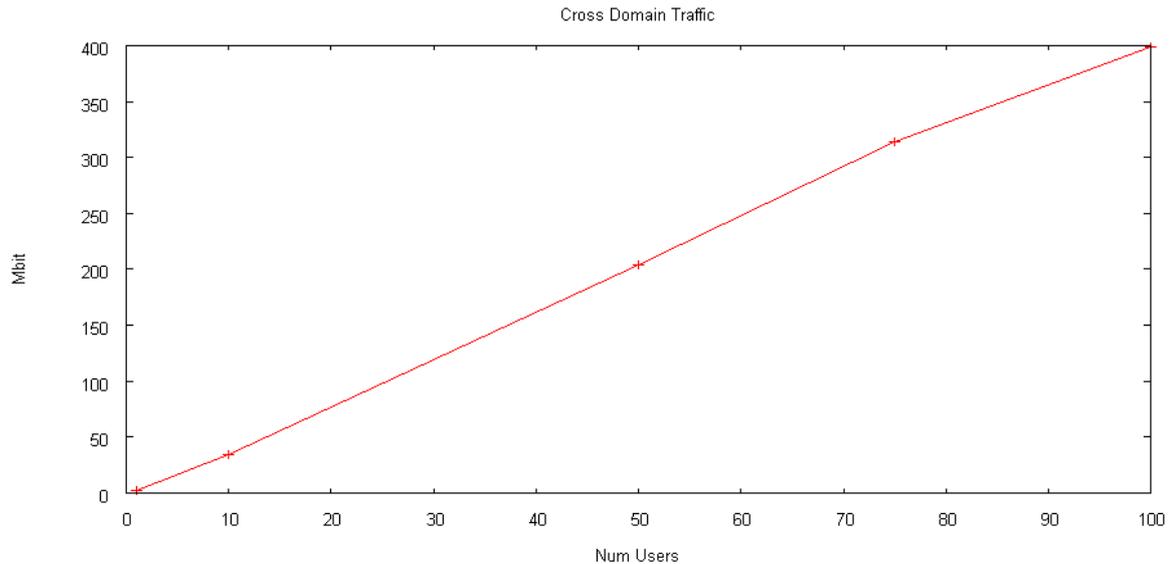

Figure 6. SIP/SIMPLE cross domain traffic

The figure estimated by Houri et al. (2007) for the number of bytes per second on the network wire is 830578 for 20,000 users in each domain. Taking the figures recorded in this research, it is possible to calculate the number of bytes per second on the network wire for the simulations performed.

 The cross domain traffic for 20,000 users in the environment setup for this research would be between 313801 and 381051 bytes. This differs significantly to the figure of 830578 bytes estimated by Houri et al. (2007). The main reason for the differences appears to be Houri et al. (2007) estimations that each SUBSCRIBE message and corresponding 200/OK message would be 1024 bytes and that each NOTIFY message and corresponding 200/OK message would be 4096 bytes.

 In order to calculate the average size of the SUBSCRIBE pair and NOTIFY pair in the environment setup for this research, the Wireshark data captured was analysed further. Five SUBSCRIBE and 200/OK pairs were examined to find the average size, this was calculated to be 1400 bytes, which is marginally greater than estimation of Houri et al. (2007). However, five NOTIFY and 200/OK pairs were then examined, the average size of these pairs was calculated to be 1493 bytes. This figure is significantly lower than the estimation of Houri et al. (2007), which is 4096 bytes.

 The difference in size of the NOTIFY and 200/OK pair estimated by Houri et al. (2007) compared to the size of the NOTIFY and 200/OK experienced in this research would cause a significant difference in cross domain traffic calculations.

4. CONCLUSION

In this paper a study of Instant Messaging with particular emphasis on presence traffic was carried out. SIP was recognised as a protocol that is gaining widespread attention recently, due to its open nature and its suitability to services such as VoIP. One of the concerns identified with SIP based Instant Messaging was the amount of traffic generated due to presence messages being sent across domains.

For the purposes of this research, a network environment was implemented where presence traffic could be generated by simulating large numbers of users connected to SIP/SIMPLE servers. Using network performance monitoring tools and logging data, it was shown that the amount of network traffic generated due to presence subscriptions grew large. However, it was also shown that SIP/SIMPLE presence traffic has a steady, predictable impact on the latency of the network, up to a certain level.

It was important to establish the effect that SIP/SIMPLE traffic has on the overall performance of the network and in particular to nodes that were on the same network but were not sending or receiving SIP/SIMPLE data. It was shown that other nodes on the network that were transmitting non SIP/SIMPLE traffic did not suffer significant throughput degradation due to SIP/SIMPLE presence traffic. It was also established that `rush hour' traffic had a minimal impact on the overall throughput of the network in this test environment.

The amount of cross domain traffic captured during the simulations for this paper was analysed. This traffic was compared to estimations that had been published by members of the SIP/SIMPLE working group. It was found that the amount of traffic captured during this research was less than the amount that was expected by the SIP/SIMPLE working group. The main reason established for this was due to the size of the NOTIFY and 200/OK pair estimated by the SIP/SIMPLE working group being significantly greater than experienced in this research.

It has been established that SIP/SIMPLE is one of the leading open standards based protocols available at the moment. One of the criticisms of it has been the large amount of traffic that is generated by its presence messages and their corresponding 200/OK messages. The research carried out for this paper has shown that in the environment created, network traffic does grow large but that it does not have a significant impact on the overall performance of the network and should not discourage deploying a SIP/SIMPLE network.

REFERENCES

Houri, A., Rang, T., Aoki, E., Singh, V. & Schulzrinne, H. (2007), `Problem statement for SIP/SIMPLE'.

Nahum, E. M., Tracey, J. & Wright, C. P. (2007), `Evaluating SIP Server Performance', SIGMETRICS Perform. Eval. Rev. 35(1), 349/350.

Niemi, A. (2004), `Session Initiation Protocol (SIP) extension for event state publication, RFC 3903'.

Postel, J. (1981), `Transmission Control Protocol(TCP), RFC 793'.

Rosenberg, J. (2007), `Simple made simple: An overview of the IETF specifications for instant messaging and presence using the Session Initiation Protocol (SIP)'.

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. & Schooler, E. (2002), `Sip: Session initiation protocol, RFC 3261'.

Sugano, H., Fujimoto, S., Fujitsu, Klyne, G., by Nine, N., Bateman, A., Vision Tech, Carr, W., Intel, Peterson, J. & NeuStar (2004), `Presence Information Data Format (PIDF), RFC3863'.