A Study of An Open Source IP Multimedia Subsystem Test Bed

Jin Tang
Department of Electrical and Computer Engineering
Illinois Institute of Technology
Chicago, Illinois 60616
jtang9@iit.edu

Carol Davids
Center for Professional Development
Illinois Institute of Technology
Chicago, Illinois 60616
davids@iit.edu

Yu Cheng
Department of Electrical and Computer Engineering
Illinois Institute of Technology
Chicago, Illinois 60616
cheng@iit.edu

ABSTRACT

In this paper, we present the creation and characterization of an open source Internet Protocol Multimedia Subsystem (IMS) test bed. We built this test bed with the intention and motivation of doing research related to the performance of its various functional components as they cooperate to provide IMS features. The IMS specifications describe a set of functional components, the interfaces between those components, and the protocols and procedures to be used across those interfaces. The test bed is built based on the "Open IMS Core", which is an open source set of IMS functional components. We verify that the operation of the test bed is in conformance with the message sequence charts provided by the IMS standards. Moreover, we define four metrics that quantify the overhead associated with IMS implementations, and present the observed values of these metrics collected on the test bed under the conditions of no background traffic. This provides necessary and valuable data for us in order to study the performance when the test bed is under various background traffic loads.

Categories and Subject Descriptors

C.2.2 [**Computer-Communication Networks**]: Network Protocols

General Terms

Conformance Performance

Keywords

IMS, Open IMS Core, test bed

1. INTRODUCTION

The Internet Protocol Multimedia Subsystem (IMS) defines an architecture intended to standardize the delivery of next generation networks and the many applications, features, functions and services they promise [1]. Included among these are fixed-mobile convergence, converged service development and service control, authentication, accounting, location and access. The IMS architecture describes a set of functional components, the interfaces between these components, and the protocols and procedures to be used across those interfaces. The definitions are intended to support interoperability of networks and products. The reasons for the development and adoption of this architecture are many, and are the subject of much speculation, discussion and debate. Some believe that the future of telecommunications requires such a unified approach [4]. Others believe that the various functions and features that IMS is designed to deliver can be achieved with satisfactory results outside this architecture [5]. The current study of the performance of a system based on the IMS architecture is intended to add data to this discussion. Today many telephone service providers consider IMS functionality and conformance to be important to the future development of their businesses and networks and vendors are currently offering for sale, devices that incorporate one or more of the functions defined in the IMS architecture. There are also several IMS implementations available for deployment on enterprise networks. The current study of the performance of a system based on the IMS architecture and using a vendor-neutral platform is intended to support these deployments by providing comparative data without advocating a particular vendor.

To create our IMS Test Bed we considered three options. First, we could build a network using the functional components available on the market from various vendors. We rejected this solution since it introduces the risk that we would be testing the interoperability of the components as much as the performance of the test bed. The second option was to develop our own code, basing our development work on the 3GPP standards. This course was not taken since it would consume too many research hours. Instead, we chose to use the Open IMS Core software, available freely at [6] because it is not identified with a particular vendor and because it supplies code for four inter-operable core functional components in the IMS architecture.

The following documents are especially relevant to this work. [7] provides an overview of the Open IMS Core and the related Open IMS Playground as part of the German 3G beyond national test bed. [8] further describes the Open IMS Playground and the activities in progress for establishing works on open issues within IMS research and develop-

QShine 2008, July 28-31, 2008, Hong Kong, Hong Kong.
Copyright 2008 ICST ISBN 978-963-9799-26-4
DOI 10.4108/ICST.QSHINE2008.3952

ment. [9] discusses the benefits and challenges in integrating the IMS test beds in a community. Performance measurements describing IMS network signaling latency associated with various access networks are presented in [10]. IMS uses Session Initiation Protocol [11] as its signaling control protocol. SIP end-to-end network performance benchmarks are described in [13]. SIP device performance benchmarks are described in [14]. Benchmarks for measuring the characteristic behavior and performance of IMS networks and IMS devices are provided in [15]. These TISPAN metrics have been incorporated into running code available at the SIPp [16] website. We plan to evaluate and possibly use them in future work. More study on signaling delay can be found in [17, 18].

Since our focus is on the behavior and performance of the nodes and links in an IMS environment, we installed the four core IMS components on four separate hardware platforms then verified that the core functional components behaved in conformance with the call flows that the 3GPP standards specify for registration and session establishment [2, 3]. Verification was done using Wireshark traces of the messages exchanged during registration and session establishment, matching them to the message sequences specified in the 3GPP requirements. Next we characterized the behavior of the test bed under conditions of no-stress. We defined a set of four metrics and recorded their values for ten experiments that we performed manually. The metrics are registration time, initial response time, initial ringing time and disconnect request time. They were chosen to reflect the experience of the end-user and the distinction between them and the metrics identified in [13, 15] is also made. The no-stress condition provides necessary and valuable data for us to study the performance when the test bed is under various background traffic loads.

The remainder of this paper is organized as follows. Section 2 introduces the IMS core components and procedures. Section 3 describes the test bed development and operation. Section 4 shows how the characteristics of the test bed conform to the 3GPP IMS standards. Section 5 identifies four metrics and shows the associated results for sake of the performance of the test bed. Section 6 concludes the paper.

2. IMS CORE COMPONENTS AND PROCEDURES

The core functional components of the IMS are the Call Session Control Functions (CSCF) and the Home Subscriber Server (HSS) [1]. There are three different kinds of CSCFs: Proxy-CSCF (P-CSCF), Interrogating-CSCF (I-CSCF) and Serving-CSCF (S-CSCF). They are responsible for routing signal and managing sessions. The HSS is a database containing all user identities, location information and service-related data. It is responsible for Authentication, Authorization and Accounting (AAA). The HSS shares a long-term secret with each of the user elements for mutual authentication and for establishing Security Associations (SA) during registration. These four core functional components are the heart of the IMS network control plane and the test bed is developed to implement all of them. Besides these core functional components, there are a number of additional functional components that may be included in an IMS network. Examples include: (1) the Policy Decision Function (PDF), that is responsible for media resource reservation and (2) the Application Server (AS), that is in charge of different multimedia applications and services [12]. The IMS environment created by the interworking of these functional components enables two basic types of procedures: registration and session control. In the IMS environment, User Elements (UE) need to register on the network and then set up sessions to communicate with each other. We give a more detailed description of the four core functional components and the two basic procedures in the following.

2.1 Core Components

Proxy-Call Session Control Function (P-CSCF) is the first functional component the UE contacts on the IMS network. It maintains IPsec SAs with the UEs and applies integrity and confidentiality protection for SIP signaling. It is also able to inspect the contents of the Session Description Protocol (SDP) [20] in the body of the SIP signaling messages, to identify improper media types or codecs. The P-CSCF can also interact with the PDF to derive authorized IP QoS information that is used to reserve bandwidth resources for the upcoming media streams.

Interrogating-Call Session Control Function (I-CSCF) queries the HSS for the name of the S-CSCF assigned to serve a user who is registering. If there is no previously assigned S-CSCF, the I-CSCF receives S-CSCF capabilities. Based on this information, the I-CSCF assigns a suitable S-CSCF for the user and routes incoming register requests further to the assigned S-CSCF. The I-CSCF can also act like a Topology Hiding Inter-network Gateway (THIG) that hides the topology of the internal network from the external networks.

Serving-Call Session Control Function (S-CSCF) acts as a SIP registrar. It handles registration processes, downloads authentication data from the HSS, generates challenges to the UE, verifies the response from the UE and then accepts the registration. It also makes routing decision during session setup.

Home Subscriber Server (HSS) is the main data storage entity. It stores private user identities, public user identities, user authentication vectors, roaming authorization and allocated S-CSCF names. It also shares a long-term secret with every UE for the establishment of IPsec SAs and for mutual authentication between the UE and the network during registration.

Three different kinds of *Interfaces* are defined between each two of the core components for their communication. The *Gm* interface connects the UE to the IMS, by means of the P-CSCF. It carries SIP signaling messages between the UE and the P-CSCF. The *Mw* interface connects different CSCFs. The protocol used for communication between the CSCFs is also SIP. The *Cx* interface is the interface between the CSCFs and the HSS. The main difference between Cx and the other two interfaces is that the messages carried across this interface use Diameter [21] protocol rather than SIP.

The relationship between these core functional components and the UE as well as the interfaces is shown in Figure 1.

2.2 Procedures

Registration and Session are generally the two categories of IMS procedures.

2.2.1 Registration

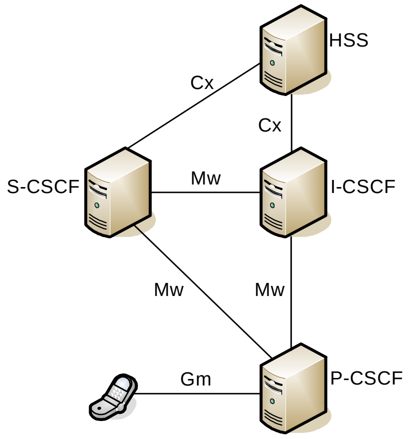


Figure 1: IMS Core Functional Components

The user *registration* procedure provides mutual authentication between the UE and the IMS network using the 3GPP Authentication and Key Agreement (3GPP-AKA) mechanism. The AKA is accomplished by the S-CSCF challenging the UE with a 401 Unauthorized message after receiving the initial REGISTER message from the UE and the UE responding with a second REGISTER message. In addition, the IMS access security begins with the establishment of IPsec SAs between the UE and the P-CSCF. These provide both integrity and confidentiality protection services for the traffic going through between the two parties. The IPsec SAs are established during the initial authentication, and take effect after the authentication procedure is finished. Protected server ports and protected client ports for both the UE and the P-CSCF are set for all the SAs [22].

2.2.2 *Session*

Two UEs can set up sessions such as voice calls or video sharing between each other after both of them successfully register on the IMS network. The session setup is achieved by SDP offer/answer exchanges carried in SIP messages. The IMS capability to create innovative multimedia applications and services derives from this ability to establish and manage different multimedia sessions.

3. THE TEST BED DEVELOPMENT AND OPERATION

We aimed to build a test bed that conforms to the 3GPP IMS standards using an open source solution. IMS is based primarily on extensions to SIP for registration and session control to make it easily integrated to the existing networks. Thus it would be natural to try to build an IMS system by extending existing SIP proxies. There are open source SIP proxies that have been around for some years, and the SIP Express Router (SER) is one of the most commonly used among them. The call session control functions of the Open IMS Core are extended from the SER to deal with extensions of SIP for IMS. This is done by adding dynamically loadable modules to the SER. Additionally, since IMS requires us to query the HSS for information related to user registration and the routing of signaling messages, an HSS is necessary and developed as part of the Open IMS Core based on MySQL and the JavaDiameterStack to communicate with the CSCFs through the Diameter protocol. These CSCFs and HSS together provide the basis for an open source solution of IMS.

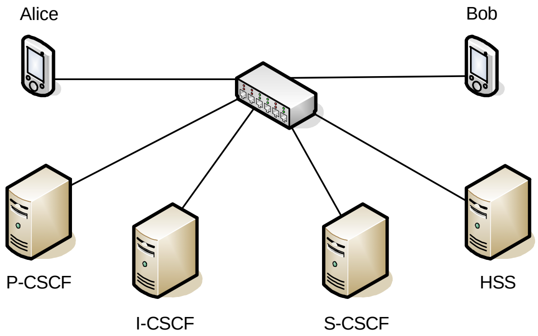


Figure 2: Test Bed Architecture

In the test bed, our core network consists of four Open IMS Cores installed on four separate Linux systems that are all connected to a hub. Its compilation and installation require as a prerequisite the additional installation of supporting software including GCC, JDK, ant and other packages as indicated in [19]. The Core itself consists of all four IMS core functional components. They are controlled by their own startup scripts and can be turned on either all together or separately. A web console comes with the HSS to add, delete and manage user profiles. Although all the functional components can run on one Linux system, we chose to separate them to create an environment that more closely resembles a real-world installation. We started one function on each of the four systems in order to make them collaborate to function as an IMS core network. The four-in-one setup was still helpful to get us familiar with the characteristics of the Open IMS Core.

We chose OpenIC Lite as the user element since it was also from Fraunhofer FOKUS [23]. It is open source and provides the basic functional behaviors as an IMS user element such as SIP-based signaling and AKA authentication. The OpenIC Lite also provides an ISIM profile simulation feature. An ISIM is an application in which a user element shares a long-term secret with the HSS for purposes like authentication and session key establishment. The application usually resides on a tamper resistent hardware, such as a USIM card, but the simulation feature gives us a convenient software solution. We configured two OpenIC Lite user elements, Alice and Bob, and added them to the HSS through the web console. In this case we manually set a shared secret between each of the users and the HSS. The two UEs were configured to communicate with the P-CSCF, the first contact point of the IMS core network. The four core functional components and two user elements together are the IMS functional part of the test bed and are illustrated in Figure 2.

To make the host names of the core functional components such as “pcscf.open-ims.test” recognizable to others in the network, we set up a minimal DNS server using BIND9. Note that the DNS server here only provided basic host name-IP address mapping function and was not yet used for P-CSCF address discovery [12]. We also set up a MySQL server, since the HSS needed a database to store data such as private user identities, public user identities and user authentication vectors. Additionally, we connected another Linux system running the Wireshark network protocol analyzer in order to sniff traffic in the network and analyze the characteristics and performance of the test bed. All BIND9, MySQL and Wireshark are open source software and provide necessary services to the IMS core network.

The test bed was up and running after we started the core functional components as well as the DNS and MySQL services. We let the two OpenIC Lite user elements, Alice and Bob, register on the network. Next, We initiated a session from Alice to Bob using Bob's public identity sip:bob@open-ims.test. Bob answered the call and a voice call session was established. We traced both the registration and session setup procedures using Wireshark.

To further develop the test bed and continue our research, we plan to add more domains to our environment in order to test handoff and voice continuity. To achieve this, we will add security gateways between the domains. To test the resilience of the Open IMS Core, we plan to add a traffic generator in the environment. After the IMS test bed is characterized, we plan to introduce commercial components, one at a time, to replace the IMS Core functional components and to repeat our characterization tests and observe variations in performance.

4. THE TEST BED CONFORMANCE

In order to verify that the test bed is IMS conformant we compared the traces collected during the UE registration and session establishment with the message flows specified in the IMS standards. The detailed results of the comparison are described below.

4.1 Registration

In IMS, the user *registration* procedure performs mutual authentication between the UE and the IMS network using the 3GPP Authentication and Key Agreement (3GPP-AKA) mechanism, based on a long-term secret shared by the UE and the HSS.

During authentication, a user, Alice, sends her identity in the initial REGISTER request to the IMS network. This request is finally routed to the S-CSCF. Upon receipt of the REGISTER request, the S-CSCF downloads Alice's Authentication Vector (AV) from the HSS. The AV includes five parameters generated from the shared secret, namely RAND, XRES, AUTN, IK and CK. The S-CSCF removes the XRES and sends a 401 Unauthorized response, including the other four parameters, towards the UE through I-CSCF and P-CSCF. Upon receipt of the 401 response, the P-CSCF removes the IK and the CK and sends AUTN and RAND to the UE. The AUTN is verified by the UE based on the shared secret. If this verification is successful, the IMS network is authenticated to the UE. The RAND is used to calculate RES. The IK and the CK are calculated by the UE based on the shared secret. Thus, two session keys are known and shared by the UE and the P-CSCF at this point. These two session keys are used for the two parties to set up a set of IPsec SAs. Subsequent traffic will go through the newly established protected server and client ports rather than through the original, unprotected ports. The UE then sends a second REGISTER request including the RES mentioned above to the S-CSCF. The S-CSCF compares the received RES to the XRES. If the two values are equal, the UE is authenticated to the IMS network, and this is indicated to the UE by a 200 OK response. After mutual authentication is accomplished, the network registers the UE as a legitimate user. The procedure is illustrated in Figure 3. HSS is not included for simplicity.

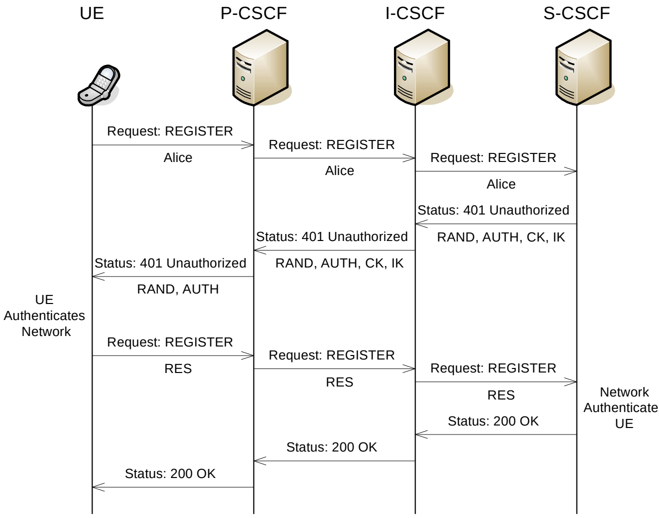
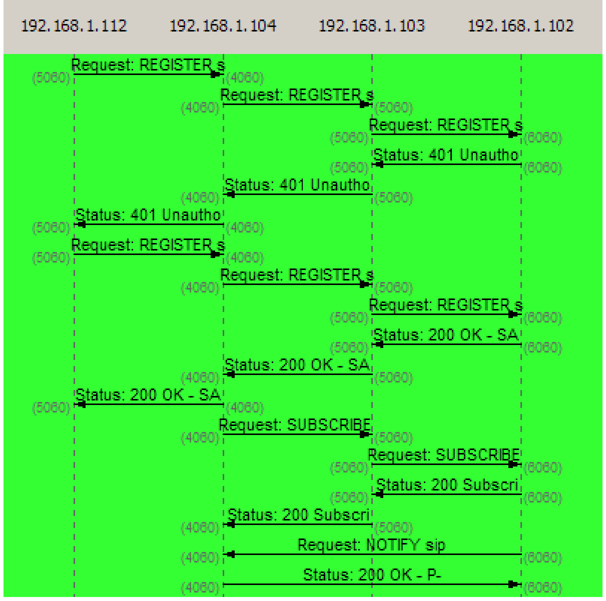


Figure 3: Standard Registration Procedure

We captured traces on the test bed during the registration process and generated message flow charts using the graph tool in Wireshark. The resulting graph is shown in Figure 4. There we observe that the S-CSCF of the Open IMS Core sent a 401 Unauthorized message to the user Alice after the initial REGISTER message and Alice responded with a second REGISTER message followed by a 200 OK. This challenge-response procedure is identical to the standard 3GPP-AKA mechanism and we conclude that mutual authentication occurred according to the IMS standards. If we look further into the header fields of the SIP messages in the trace, the WWW-Authentication header is inside the 401 Unauthorized messages to challenge Alice, while the Authorization header is in the first REGISTER message to initiate the authentication and in the second REGISTER message to respond to the challenge. The presence and population of these header fields is also in conformance with the IMS specifications.

Continuing to examine Figure 4, we see that no IPsec security association was set up between Alice and the P-CSCF, since the second REGISTER message was both sent and received through the same UDP ports as the initial REGISTER message, rather than through a new protected client port at the UE side and a protected server port at the P-CSCF side. IPsec is mandatory for IMS access security in the standards and is used to protect all traffic between the UE and the P-CSCF afterwards. We looked into the header fields of the SIP messages and found out Alice did not include the Security-Client header in the initial REGISTER message which was supposed to initiate the IPsec SA negotiation. Thus the failure of IPsec was due to the limitation of the OpenIC Lite user element at this point.

The last thing we got from the message flows was that the P-CSCF associated with Alice sent a SUBSCRIBE message through the I-CSCF to the S-CSCF right after the registration and authentication of Alice was successful. This message was used for the P-CSCF to subscribe to the registration-



```
192.168.1.112   192.168.1.104   192.168.1.103   192.168.1.102
(5060) Request: REGISTER s (4060)
(4060) Request: REGISTER s (5060)
(5060) Request: REGISTER s (6060)
(5060) Status: 401 Unautho (6060)
(4060) Status: 401 Unautho (5060)
(5060) Status: 401 Unautho (4060)
(5060) Request: REGISTER s (4060)
(4060) Request: REGISTER s (5060)
(5060) Request: REGISTER s (6060)
(5060) Status: 200 OK - SA (6060)
(4060) Status: 200 OK - SA (5060)
(5060) Status: 200 OK - SA (4060)
(4060) Request: SUBSCRIBE (5060)
(5060) Request: SUBSCRIBE (6060)
(5060) Status: 200 Subscri (6060)
(4060) Status: 200 Subscri (5060)
(4060) Request: NOTIFY sip (6060)
(4060) Status: 200 OK - P- (6060)
```

Alice: 192.168.1.112 P-CSCF: 192.168.1.104
I-CSCF: 192.168.1.103 S-CSCF: 192.168.1.102

Figure 4: Test Bed Registration Procedure

state event package of Alice. The subscription was acknowledged by a 200 OK message. Afterwards, the S-CSCF sent a NOTIFY message back to the P-CSCF. The body of this NOTIFY message was of XML format, and included Alice's actual registration-state information. This procedure too is in conformance with the IMS standards.

4.2 Session Setting Up

In IMS, the *session* establishment procedure sets up a session, for example, a voice call or video sharing, between two communicating parties using handshake messages. According to the 3GPP IMS standards, media negotiation should be done through two offer/answer SDP exchanges to ensure a single codec per media stream in a session. The messages exchanged are (1) INVITE, 183 Session Progress, and (2)PRACK, 200 OK. Resource reservation can also be provided through preconditions by means of a third offer/answer SDP exchange of UPDATE, 200 OK. The information supplied in this exchange ensures that there are enough resources available to establish the session. The called party is not supposed to ring until both of the two parties confirm successful resources reservation. This procedure is illustrated in Figure 5. The IMS network core components are not shown in the figure since they mainly act as routers in the procedure.

We captured traces on the test bed during the session establishment process and generated message flow charts using the graph tool in Wireshark. The resulting graph is shown in Figure 6. In this test, Alice originates a call to Bob following the standard IETF SIP procedure [11]. The CSCFs in the core network routed and processed all the existing session establishing signaling messages in conformance with the IMS specifications [1, 2, 3]. The option to reserve bandwidth for the session was not used because it was not available in the OpenIC Lite UE. As a result, the calling party, Alice, did not include any preconditions line in the first SDP offer and Bob, the called party, did not perform any resource reservation upon receipt of the initial INVITE. Thus, we cannot report whether this optional function of an IMS network is performed by our test bed in accordance with the specifications. In future work on this test bed we will need to incorporate a UE with the capability to perform resource reservations. While the lack of such capability is not a problem on wired networks as in the current test bed, it would be unacceptable in wireless and other environments where bandwidth is a scarce commodity. We plan to measure performance of IMS systems that enable fixed mobile convergence and we will identify and incorporate a UE that includes the option to use resource reservation.

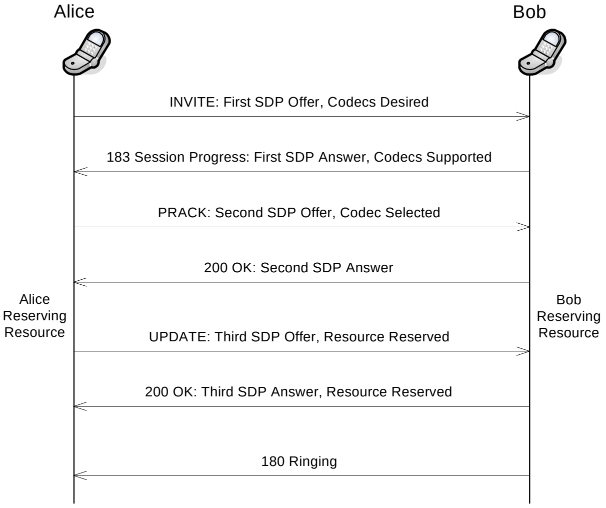
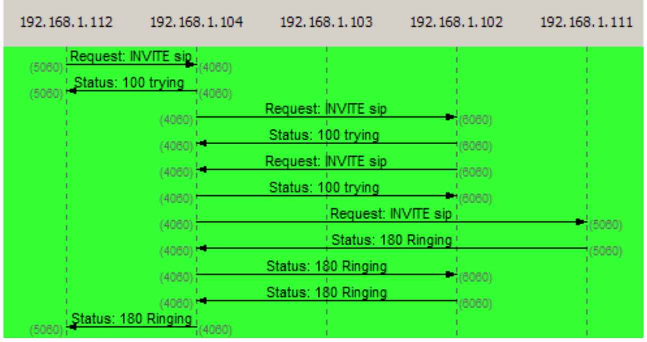

Alice
Bob
INVITE: First SDP Offer, Codecs Desired
183 Session Progress: First SDP Answer, Codecs Supported
PRACK: Second SDP Offer, Codec Selected
200 OK: Second SDP Answer
Alice Reserving Resource
Bob Reserving Resource
UPDATE: Third SDP Offer, Resource Reserved
200 OK: Third SDP Answer, Resource Reserved
180 Ringing

Figure 5: Standard Session Set Up Procedure



```
192.168.1.112   192.168.1.104   192.168.1.103   192.168.1.102   192.168.1.111
(5060) Request: INVITE sip (4060)
(5060) Status: 100 trying (4060)
(4060) Request: INVITE sip (6060)
(4060) Status: 100 trying (6060)
(4060) Request: INVITE sip (6060)
(4060) Status: 100 trying (6060)
(4060) Request: INVITE sip (5060)
(4060) Status: 180 Ringing (5060)
(4060) Status: 180 Ringing (6060)
(4060) Status: 180 Ringing (6060)
(5060) Status: 180 Ringing (4060)
```

Alice: 192.168.1.112 Bob: 192.168.1.111
P-CSCF: 192.168.1.104 I-CSCF: 192.168.1.103
S-CSCF: 192.168.1.103

Figure 6: Test Bed Session Set Up Procedure

5. PERFORMANCE

In order to benchmark how the test bed performs, we identified four metrics, applied them and recorded the associated results. We performed the registration and session establishment ten times with no background traffic and captured the associated traces using Wireshark. All the results we present are calculated by taking the differences between

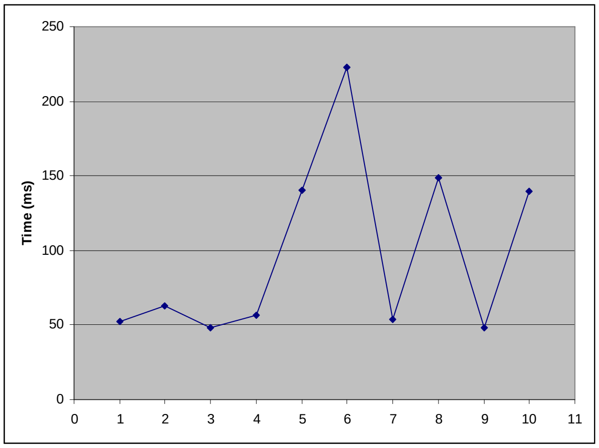


Figure 7: Registration Time

time stamps found in the traces. The data point "n" on each of the four metric-value graphs corresponds to a value measured on the "nth" Wireshark trace. The number of trials is small, yet there is already a pattern in the data. For each metric we saw what appears to be a single anomaly and for all but one metric, the data cluster around two distinct values. Bulk, automated call generation in the next phase of this study should help clarify if this bimodal behavior is characteristic. Further study of the traces for other network activity is needed to help explain the anomalies.

5.1 Registration Time

The registration time is defined as the time elapsed between the sending of the initial REGISTER request and the appearance of the 200 OK associated with this REGISTER from the P-SCSF. This metric represents a lower limit on the minimum time it takes for a user to gain access to the IMS network. It is comparable with the "Processing time for the first register request" defined in [15] and with the "successful Registration Request Delay" defined in [13]. The results are shown in Figure 7. The average time elapsed for registration was 97.4206 milliseconds. The data cluster around two distinct values, 50 ms and 150 ms, with an anomaly of over 200 ms in experiment 6.

5.2 Initial Response Time

The initial response time is defined as the time elapsed between the sending of the initial INVITE and the appearance of the first response to the INVITE, usually a 100 Trying from the P-CSCF. This metric indicates the amount of time it takes the access links and nodes to respond to a session initiation request. The results are shown in Figure 8. The average initial response time was 0.5124 milliseconds. The data cluster around two distinct values, .4 ms and .6 ms with an anomaly in experiment 9 of 1.2 ms. There is no obvious correlation between the low values in initial response time and the low values in registration times. For example, there is a high initial response time recorded for experiment 7 and a low registration time in the same experiment. Meanwhile, there is a low initial response time recorded for experiment 8 and a high registration time for this experiment.

5.3 Initial Ringing Time

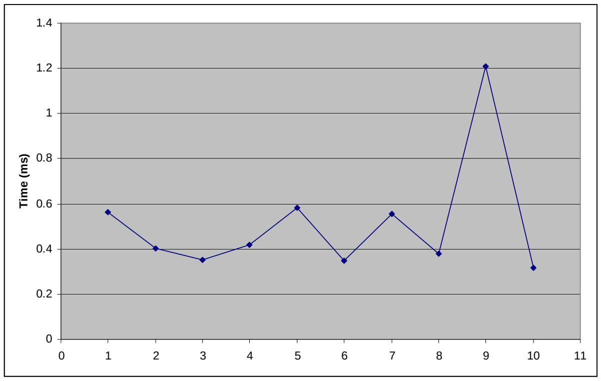


Figure 8: Initial Response Time

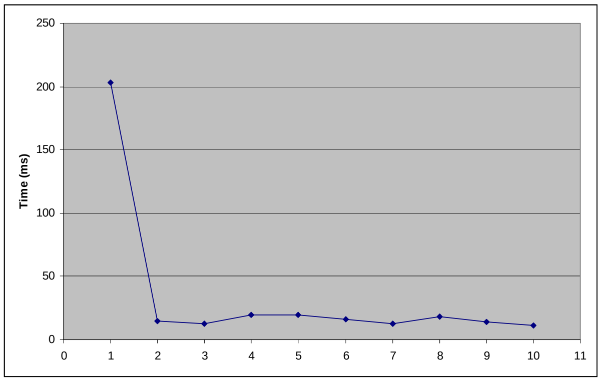


Figure 9: Initial Ringing Time

The initial ringing time is defined as the time elapsed between the sending of the initial INVITE and the appearance of the first 180 ringing associated with that INVITE. This metric represents a lower limit on the "time to ring", the time elapsed between the sending of the initial sessions set up request and the receipt of the ring-back signal. It is comparable to the "successful session setup delay" defined in [13], but not with the metrics in [15] which include final responses. The results are shown in Figure 9. The average initial ringing time was 34.1147 milliseconds. In this case, the data settled at about 12 ms after an initial spike to 200 ms in experiment 1.

5.4 Disconnect Request Time

The disconnect request time is defined as the time elapsed between the appearance of the first BYE request and the appearance of the 200 OK associated with that BYE. This metric indicates how long it takes to end a session after one of the parties sends a disconnect request. It reflects the use of end-user bandwidth and processor resources and so provides information about the end-user experience. It is comparable to the "time to release the resource" in [15] and to the "successful session disconnect delay" defined in [13]. The results are shown in Figure 10. The average time to disconnect a call was 48.9686 milliseconds. The data cluster around 10 ms and 50 ms, with an anomaly of about 210 ms in experiment 8.

6. CONCLUSIONS

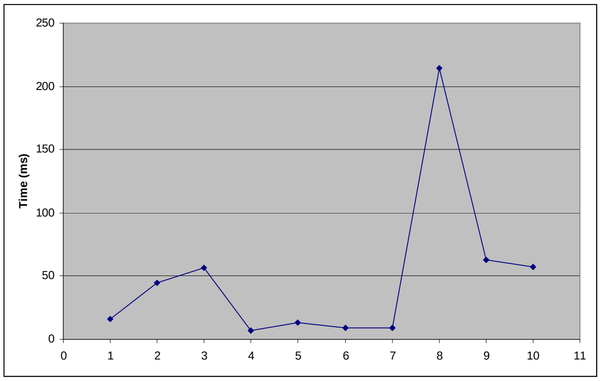


Figure 10: Disconnect Request Time

In this paper we first describe the IMS functional components and procedures specified by the 3GPP standards. Next we show the creation and characterization of an IMS compliant test bed using the Open IMS Core. The four core functional components available from Open IMS Core proved to work in conformance to the IMS standards under the test conditions described. We also identified a set of metrics, applied them and recorded the associated results to describe the behavior of the test bed under the no-load condition. The metrics were designed to indicate the end-user experience and we compare them to other performance metrics in use and under development. We conclude that the metrics described here will add new information to the study of the end-user experience in IMS networks, and that automating their collection will yield useful results. Besides automating their collection, next steps will include the addition of various background loads and the use of different communication options and a comparison of the new results to those obtained under no-load and minimal options. In this manner we plan to contribute to the literature that explains the trade-offs between security and resource reservation features on the one hand and device, network and application performance on the other, particularly as they are experienced by the end-user.

7. REFERENCES

[1] 3GPP. IP Multimedia Subsystem version 7. 3G TS 23.228, December 2005.

[2] 3GPP. Signaling flows for the IP multimedia call control based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3. 3G TS 24.228, 2006.

[3] 3GPP. Internet Protocol (IP) multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3. 3G TS 24.229, 2006.

[4] G. Camarillo and M. García-Martín. The 3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular Worlds, Second Edition. Wiley, February 2006.

[5] J. Waclawsky. IMS 101: What You Need To Know Now. IMS 101: Business Communications Review, June 2005.

[6] Open IMS Core. [Online:]`http://www.openimscore.org`.

[7] T. Magedanz, D. Witaszek and K. Knuettel. The IMS Playground @ FOKUS-An Open Testbed for Next Generation Network Multimedia Services. The First International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities, February 2005.

[8] F.C. de Gouveia, T. Magedanz, R. Good and N. Ventura. The Role of Open IMS Testbeds in Complex Service Delivery Platforms. AFRICON, September 2007.

[9] D. Witaszek, F.C. de Gouveia, S. Wahle and T. Magedanz. IMS Playground in Pan-European Network of Testbeds; Benefits and Challenges. The Third International Conference on Testbeds and Research Infrastructure for the Development of Networks and Communities, May 2007.

[10] D. Vingarzan and P. Weik. IMS Signaling over Current Wireless Networks: Experiments Using the Open IMS Core. Vehicular Technology Magazine, IEEE, March 2007.

[11] J. Rosenberg, H. Schulzrinne and G. Camarillo. SIP: Session Initiation Protocol. IETF RFC 3261, June 2002.

[12] M. Poikselka, A. Niemi, H. Khartabil and G. Mayer. The IMS: IP Multimedia Concepts and Service, Second Edition. Wiley, March 2006.

[13] D. Malas. SIP End-to-End Performance Metrics. IETF PMOL Draft-ietf-pmol-sip-perf-metrics-00, February 25, 2008.

[14] S. Poretsky, V. Gurbani and C. Davids . Terminology for Benchmarking Session Initiation Protocol (SIP) Networking Devices. IETF BMWG Draft-poretsky-sip-bench-term-04, November 18, 2007.

[15] ETSI TISPAN. IMS/NGN Performance Benchmark. TS 186 008, 2007.

[16] IMS Bench SIPp. [Online:]`http://sipp.sourceforge.net/ims_bench/index.html`.

[17] I. Curcio and M. Lundan. SIP Call Setup Delay in 3G Networks. In Proc. IEEE Symp. Comput. Communications '02, Taormina, Italy, July 2002.

[18] M. Melnyk, A. Jukan and C. Polychronopoulos. A Cross-Layer Analysis of Session Setup Delay in IP Multimedia Subsystem (IMS) with EV-DO Wireless Transmission. IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 9, NO. 4, JUNE 2007.

[19] FOKUS. Open IMS Core Installation Guide. [Online:]`http://www.openimscore.org/installation_guide`.

[20] J. Rosenberg and H. Schulzrinne. An Offer/Answer Model with Session Description Protocol (SDP). IETF RFC 3264, June 2002.

[21] P. Calhoun, J. Loughney and E. Guttman. Diameter Base Protocol. IETF RFC 3588, September 2003.

[22] 3GPP. Access Security for IP-based Services, V8.1.0. 3GPP TS 33.203, December 2007.

[23] The Open IMS Client (OpenIC). [Online:]`http://www.open-ims.org/openic`.