This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2010 proceedings.

ENUM based Service Discovery Architecture for 6LoWPAN

Fatima Muhammad Anwar‡, Muhammad Taqi Raza†, Seung-Wha Yoo‡, Ki-Hyung Kim‡
‡Information and Communication Department
‡Ajou University, South Korea
†USN Networking Research Team
†Electronics and Telecommunications Research Institute (ETRI), South Korea
‡{fatima, swyoo, kkim86}@ajou.ac.kr
†taqi@etri.re.k

Abstract— **Service discovery in Ubiquitous Sensor Networks has been targeted mostly for services available within certain proximity, but the service availability only in close vicinity is no longer feasible in the pervasive and ubiquitous era. In order to address the ubiquity in service discovery, we have proposed a framework that makes use of the Electronic Number Mapping (ENUM) protocol. Our network architecture consists of sensor nodes associated with few relatively powerful nodes called master nodes. Only master nodes within IPv6 enabled Low power Personal Area Networks (6LoWPANs) are assigned unique E.164 numbers so that the services offered by the network could be accessed globally. Services destined for sensor nodes first reach the master node to which they are associated to, using the E.164 number of the master node. The gateway of the network performs the task of converting attribute-value pair based human readable queries to E.164 numbers. Also the gateway facilitates its network by running ENUM protocol for service sharing like multimedia, mail, web and many other services. Moreover, a significant improvement in service discovery cost in terms of latency and traffic overhead is observed.**

Keywords - Service discovery; Ubiquitous Sensor Networks; ENUM; 6LoWPAN; E.164 numbers

I. INTRODUCTION

Ubiquitous Sensor network (USN) is a new technology emerging on the horizon today, whereas, the internet has already established mature infrastructure. In order to share the physical world data extracted by the sensors with the whole world, Internet provides the best platform to do so. The web integrates diverse information sources and diverse human generated information. It provides extensive interoperability, access control and firewall mechanisms for security, naming, addressing, translation and lookup. Services follow discovery, lease, use, and discard model. This IP domain's mature infrastructure could aid many sensor network applications. For instance, the main aspect of Habitat monitoring applications [1] is management at-a-distance. Therefore, monitoring of sensor networks over the Internet is the prime requirement for the remoteness of the field sites in habitat monitoring. The gateway of the sensor network connects to database replicas across the Internet so that the services offered by sensor nodes at the remote location could be accessed by scientists through any user interface. In this way, services could be reliably accessed from locations previously inaccessible.

While talking about the Wireless Sensor Networks (WSNs), the nascent "IPv6 over Low power Personal Area Networks" standardized as 6LoWPAN [2] by the IETF, transmits IPv6 packets over LoWPAN. The work done related to SD in 6LoWPAN is still immature to meet the requirements of worldwide SD. One of the efforts include "Proxy based service discovery architecture for 6LoWPAN" [3] which targets services mostly in its close vicinity while putting network elements in a hierarchal manner. It provides a complex mechanism for accessing services from the Internet resulting in reduced performance and cost burden. To overcome these discrepancies, we propose an architecture for SD that makes use of the well known Electronic Number mapping (ENUM) protocol [4] to provide global service discovery. ENUM makes use of the Domain Name System (DNS) infrastructure that provides Naming Authority Pointer (NAPTR) records [5] for finding out the path to the requested service. The purpose of choosing ENUM is to use already available robust Internet DNS infrastructure instead of deploying some new complex architecture in parallel just for the purpose of SD. Our proposed network architecture consists of few relatively powerful nodes called master nodes and only they are assigned unique service identifiers called E.164 numbers [6]. These numbers provide numbering scheme for services in the ENUM protocol. The sensor nodes initiate queries based on attribute-value (av) pairs and the master node is responsible to forward this query to the gateway. The service being requested could be provided by any node in some sensor domain or it could be any service provided on the web. The gateway converts the av-pair based service strings to E.164 numbers and then perform domain name conversion procedure. This domain name has been sent to the DNS server for look up. The server provides NAPTR records against the domain name entry. These records are resolved into specific Uniform Resource Identifiers (URIs) which provides the path to the desired service. Our proposed SD framework employing ENUM is quite simple and robust.

The rest of the paper is organized as follows. Section II describes the Related Work. Section III gives the motivation of using ENUM and basic concept about the protocol followed by the main idea of the proposed architecture in Section IV. Section V presents the performance evaluation of the proposed architecture. Section VI gives the results and their discussion followed by the future work in section VII and finally section VIII concludes the paper.

II. RELATED WORK

In general there are three types of architecture based SD approaches: centralized directory based, hierarchal architecture based and fully distributed architecture based approaches. In large infrastructure networks, centralized directory based approach is used where a dedicated server is maintained. DNS-based Service Discovery (DNS-SD) is a way of using standard DNS programming interfaces, servers and packet formats to browse the network for services. It ensures wide area access**.** The protocol, Multicast DNS service discovery (mDNS-SD) [7] uses the multicast DNS and service resource records (SRV) by utilizing the public DNS servers in the Internet. It uses a part of the DNS namespace, which is available for local use. Moreover, Jini [8], UPNP [9] and Bonjour [10] protocols require a hosting environment that must be provided by a resource rich device. The Portal based SD [11] uses the concept of regional locals to perform local SD through master nodes and data servers in the network. Proxy-Enabled Service Discovery Architecture [3] is hierarchal in nature and finds services in its close vicinity. It supports the same framework as SLPv2 and SSLP, i.e., based on User Agents (UAs) and Service Agents (SAs). Directory Proxy Agent (DPA) is deployed in 6LoWPAN. DPA is a node working as a proxy for the Directory Agent (DA). It gets services information and contextual information, from DA and peer DPAs, and caches it to use in the local PAN. SSLP introduces the Translation Agent (TA) which performs the translation of messages for the interoperability with SLPv2. The TA works in a gateway that binds the 6LoWPAN to the IP network. The major drawback of this approach is complexity and the large overhead incurred through the message translation from SLPv2 to SSLP compatible type and vice versa.

The Cluster based service discovery maintains fully distributed service architecture. DMAC and Clustering for SD (C4SD) clustering algorithms [12] exploit a cluster overlay and the protocol uses an underlying clustering structure, where the cluster forms a distributed directory of service descriptions. But the problem of chain reaction limits its use and implies higher maintenance overhead. Moreover, re-clustering and re-registration incurs higher discovery overhead, making it not suitable for networks with large number of clusters. Cross layer SD (OLSR-mDNS) [13] and Localized distance sensitive Service Discovery (iMesh) [14] are confined to their home mesh cells. There is a need for the SD to be targeted in the domain of WSNs specifically for 6LoWPAN. The previous efforts serve as building blocks for designing an efficient service discovery scheme that meet our goals.

III. ENUM MOTIVATION AND DESCRIPTION

In surveillance applications, for example the proximity or video sensors deployed at the borders or any remote site need to send data or offer services to the clients at the far away monitoring stations [15]. Internet connectivity, both for the sensor node and the client interface, solves the problem of wide area access. In order to make use of the services globally, there should be some common SD protocol for the interoperability between a client connected to the Internet and a service provider node in the 6LoPWPAN. The heterogeneity in terms of network and SD protocols should be considered. Little work has been done in this regard which offers some complex SD protocol translation mechanism for the integration of both environments. For example a protocol running for the SD in the 6LoWPAN domain has to be completely translated to some other SD protocol which runs on the web, for the interoperability [3]. These protocol translation mechanism and rules are really complex. In order to avoid complexities, we propose to make use of ENUM as a SD protocol because of its property of simple integration between the both environments. The ability of ENUM to use the DNS infrastructure and its resource records to look up for services makes it suitable for the pervasive environment.

Related to ENUM, we will make use of two key terms frequently, E164 numbers and Naming Authority Pointer Records (NAPTR) [5]. The standard E164 numbers [6] constitute a numbering and addressing plan for ENUM. These are unique identifiers associated with services or service providers globally. The E164 numbers are converted to a domain name, in order to be sent on the internet towards the DNS. The format of a unique E164 number and its conversion to a domain name is shown below in TABLE I.

TABLE I. NUMBER STRUCTURE FOR GLOBAL SERVICES AND CONVERSION TO DOMAIN NAME

E164 number		**E164 number converted to domain name**
Country Code (ac)	**Subscriber Number**	1. Reverse the number 2. Separate them with dots 3. Add the e164.arpa suffix
cc = 3 digits	Maximum = (15–cc) digits	
+82	31219001	*1.0.0.9.1.2.1.3.8.2.e164.arpa*

The NAPTR records are the resource records of the DNS found after matching with the domain name. These records publish URIs and it provides the available ways for contacting a specific service. Now we could formally define ENUM as a protocol that maps E.164 numbers into services (URLs, E-mail address, multimedia sessions, IP phone etc.). Internet connectivity is essential in our approach both for the clients within the sensor domain and outside the sensor domain, as

1. 6LoWPAN comprises of sensor nodes that are IPv6 enable devices.
2. The main concern of our approach is SD in ubiquitous era which requires Internet connectivity to have a global visibility.

ENUM as a service discovery protocol in 6LoWPAN offers many advantages. Some of the advantages are listed below,

1. ENUM ties services to unique identifiers called E164 numbers and provides global visibility to the services.
2. Unlike the URI resolution application (I2*) of the NAPTR records, the ENUM uses ENUM to URI (E2U) services provided by the NAPTR records. The E2U service is relatively light weight because its “rewrite rules” specify the flags in such a way so that the rule is terminal i.e. the result of first look up is always the end result URI [5]. As the sensor domain is totally resource constrained so a light weight protocol like ENUM is required.
3. Also the purpose of choosing ENUM is to use already available robust Internet DNS infrastructure instead of deploying some new complex architecture in parallel just for the purpose of service discovery.

IV. Proposed Architecture

Our proposed architecture incorporates ENUM as a SD protocol in WSNs. The main components are shown in the general framework in Fig. 1.

The architecture is composed of several components within a PAN. There are many resource constrained sensor nodes, few relatively powerful master nodes, and a single gateway performing some SD computations and acting as the edge router for the master node. Each sensor node is associated with a particular master node. The master node assists the sensor node by forwarding its service requests to gateway and also entertains the sensor nodes with services destined for them. The sensor nodes are not computationally strong devices so it would be a big burden on them to perform data extraction, data and control messages routing and computations. Only the master nodes can communicate with the gateway where the gateway provides the Internet connectivity.

A. Service Discovery using ENUM

We proposed that only the master node is assigned a unique E.164 number where this E.164 number is used to provide services to the entire PAN. In order to understand the assignment of E.164 numbers to services and the SD procedure, we would describe a general scenario. Suppose a client (it may be a sensor node or some user connected to the Internet) makes a service request by using some human understandable string. If the sensor has originated the service request, the master node will check its local cache for the service. On cache miss, the master node will forward the query to the gateway where the name string is converted into integers based string termed as E.164 number. The gateway does this conversion by making use of its own database servers or it may get it resolved by a server on the Internet. This E.164 number has been changed into a domain name (as described in Section III) and it has been sent towards the DNS server for NAPTR records which matches against the domain name. These NAPTR records are resolved into URIs which in turn is returned to the client. The general ENUM based SD model is shown in Fig. 2.

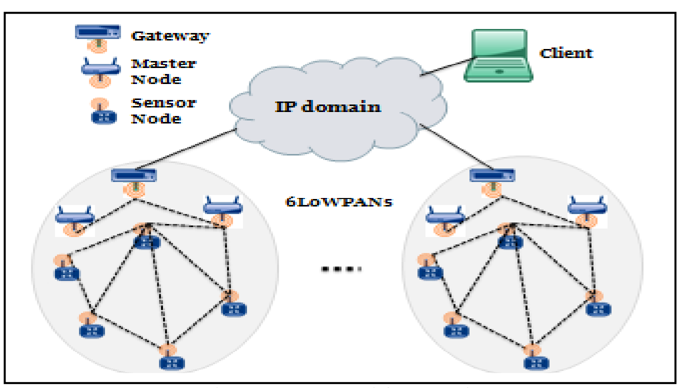

Figure 1. General framework

B. Service Query – (Attribute-Value pair (av-pair) to E.164 Mapping)

ENUM protocol is used in IP telephony and the general requirement in making an ENUM based IP call is to have a number which could be dialed easily by the user with their hand phones. But the problem with the number based query in WSNs is that it is not easy for the client to remember and understand numbers. The sensor nodes in WSNs are resource constrained so it is not feasible for the node client to maintain the service to service number mapping in order to initiate a request. Therefore, in WSNs, it is necessary that service queries should be based on attributes which are human readable. As a result, ENUM has been modified based on the initial query from the client. The client will make a query based on human readable strings constructed from attribute-value pairs (av-pairs), where, attribute is a category in which an object can be classified and value is the object's classification within that category [16]. For example, for the fire detection purpose, a client wants to have the temperature sensor data from our lab which is situated in building = Sanhak Won - Ajou University, City = Suwon and Country = South Korea, then the query string would be like,

[Country = Korea [city = Suwon [building = Sanhak Won-Ajou University [service = temperature]]]]

But in the sensor domain, the bandwidth or processing power required for handling names is a concern, so the av-pair string is converted into integers. We will do the conversion from a human readable string into integers based string which we would refer to, as the E.164 number. The conversion mechanism is very simple as we will use universal codes for the values of the attributes. The number assigned to the end service, the temperature service in our example, should be globally unique and it could be done by the ENUM Registry which is on the top of National ENUM authority pyramid. Conversions of one human readable service query to its respective E.164 number is shown in the TABLE II.

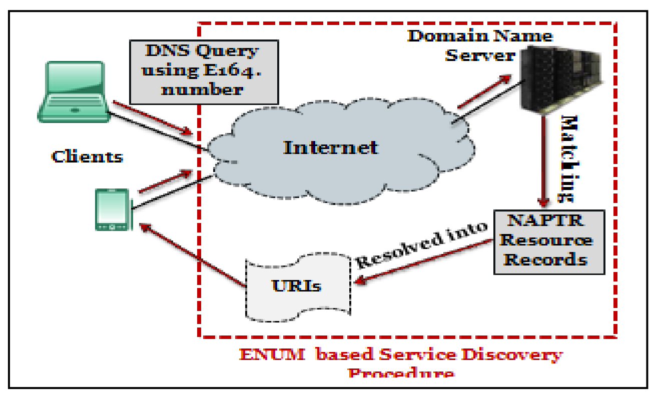

Figure 2. ENUM based service discovery model

Moreover, the direct assignment of E.164 numbers to every sensor node in the PAN is not feasible while incorporating ENUM in WSNs. As we know that the nodes within a PAN constantly leave and join the PAN and the topology keeps on changing. If every sensor node has been assigned the numbers statically, the dynamic nature of the network will be lost and we have to work on a fixed topology. Also, there is no hard and fast rule to assign the new joining nodes, some numbers. To avoid any complex assignment procedure and static topology, we have just assigned the E.164 Number to the master node. Moreover, we have also defined the service in human understandable form for the sensor node, so that every new node could also make the same service request without the need of any number and complex rules. The sensor node will send its av-pair based service request to the master node which in turn will forward it to the gateway. Now the gateway will do the E.164 number conversion and the rest of the procedure. We have made the number conversion and SD task totally transparent to the sensor node and transferred all the computational tasks to the master nodes and the gateway.

While using ENUM in WSNs, we have encountered one more problem due to service query initiation based on E.164 numbers. Lets suppose, we use the E164 number to request the two same services but present at multiple different places. There is a chance that same E.164 numbers would be assigned to both of the services. The av-pair based service query solves this problem. For example, there is a PAN present some place in US which is dedicated for the purpose of video surveillance. The same kind of PAN containing video sensors would be present at some place in Korea. The service in both the PANs is the same i.e. video streaming but our av-pair based service query can easily differentiate between the two in order to access the video service from either of the two PANs.

TABLE II AV-PAIR TO INTEGER (E.164 NUMBER) MAPPING

Human readable Service String	**Attribute**	Country	City	Building	Service
	Value	Korea	Suwon	Sanhak Won - Ajou University	temperature
Integers based Service String	***E.164 number***	*+82*	*31*	*219*	*001*

After the conversion from the av-pair based string to an E.164 number, this number is converted into a domain name. The conversion procedure is already described in Section III i.e. reversing the number, separating it with dots and adding the e164.arpa suffix. The number is being reversed because of the use of Address and Routing Parameter Area (ARPA) domain. This domain performs the reverse DNS lookup for IP addresses based on the defined standards for the ARPA domain. ARPA is the Internet Top Level Domain (TLD) whereas; e164.arpa is the second level domain specific for ENUM lookup and mapping of numbers into DNS. The ENUM based client does not notice anything of this reversal and the DNS database look-up, as this is done automatically behind the scene using a client agent terminal or the gateway.

Let's suppose a client wants to query for all the LoWPANs i.e all types of services within specific region, say our lab Sanhak Won – Ajou University. The av-pair based string would be of the following form,

[Country = Korea [city = Suwon [building = Sanhak Won - Ajou University]]]

We have made the service field empty because the client is not querying for any particular service, instead it wants information regarding all types of LoWPANs in that specific area (building). After conversion into integers based string, the E164 number would be of the form, "*+8231219*" and it would be converted to a domain name of the form, "*9.1.2.1.3.2.8.e164.arpa*". The URI returned as the end result would contain the address of the Light weight Directory Access Protocol (LDAP) server [17] where the information about those groups of LoWPANs has been stored. LDAP server is chosen for our approach in LoWPAN because it provides fast look ups which are desired in resource constrained environments.

C. *Inter LoWPAN Service Discovery*

There could be different scenarios in which the Inter LoWPAN communication could be desired e.g. in surveillance application, lets suppose that the proximity sensors are present in one LoWPAN and the video sensors are placed in some other LoWPAN. Upon some intrusion, the proximity sensors may want to invoke the video sensors to have visual confirmation [15]. Therefore, if a node in one LoWPAN wants to access some service from a node in some other LoWPAN, they would do so by using their respective master nodes. The master node will check its local cache. Upon cache miss it will resolve the av-pair based query and convert it to an E.164 number targeted for the video service in the other LoWPAN. This E.164 number is actually the number assigned to the master node in the targeted service LoWPAN. Through the general procedure of ENUM look up described above, the URIs of that service is obtained and returned to the master node which in turn will entertain the client node. The step by step procedure is described in Fig. 3.

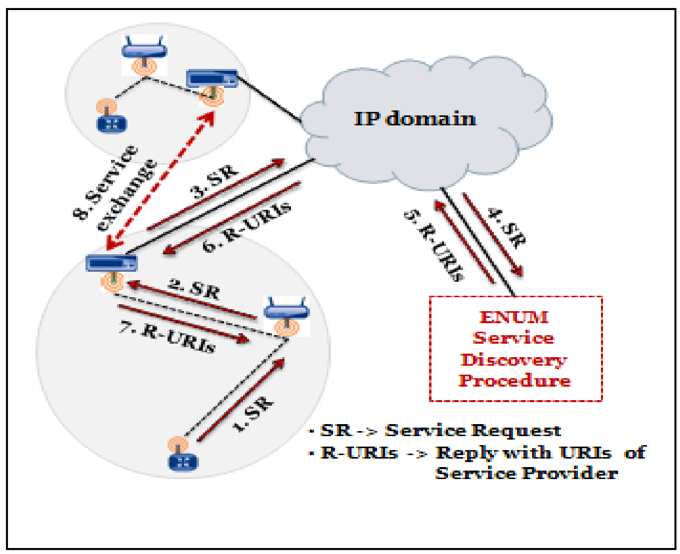

Figure 3. Inter LoWPAN service discovery

D. *Basic Assumptions*

Below are some of the basic assumptions necessary for the working of our scheme.

1. The NAPTR record should always be found whenever DNS is looked up using the domain name.

2. The sensor nodes which are multi hop distance away from the master node, must know the route towards the master node.

3. If a new node joins in, it finds the route towards master node through its neighbors (not in the scope of service discovery).

V. Performance Evaluation

We will first describe an example scenario and then derive equations for two performance metrics as mentioned before. Suppose a client node wants to query some service. The request from the client will go to the master node of the PAN to which the client belongs. The request would propagate towards the master node in a multi hop fashion. The master node first checks its internal cache. If there is a cache hit in the master node for the particular service queried, the cached information is used for the service reply and there is no need to go through the SD process. The probability that the service could be found in the cache is P*cacheHit*.

The list of variables needed for the evaluation purpose is shown below in the TABLE III.

TABLE III VARIABLES REQUIRED FOR PERFORMANCE ANALYSIS

Variable	Description
N	Total number of nodes
R*req*	Service Request rate i.e. number of queries per client node
L	Average number of hops from client node towards master node
P*cacheHit*	Probability of Cache hit for the queried service
D*h*	One hop delay
D*t*	Transmission delay in the Internet
λ	Data rate (queries per second)

O*cache* packets are sent in the cache hit case and T*cache* is the round trip time required. If there is cache miss, the network must go through all the service discovery process. In order to perform the service discovery, the master node will forward the initial service request to the gateway where the gateway perform conversions and SD task as described in Section IV. The total number of packets generated during the whole process are (O*cache* + O*sd)* and total time taken is *(*T*cache* + T*sd))*, where O*sd* is the number of packets generated for the SD purpose and T*sd* is total time taken for cache miss SD.

E. Service Look up Latency

$T_{exp} = 2 (D_h \times R_{req} / \lambda \times L \times N) + (1- P_{cacheHit}) \times (N \times D_t \times R_{req} / \lambda)$ ... (1)

Equation 1 gives the expected time needed for a complete service discovery process. The client node is not at a single hop distance from the master node so the requests from the client to the master node propagate in a multi hop fashion. We assume that the client node already has the path route information towards the master node. The number of hops from client to master node and a single hop delay affects the overall services look up time. Moreover, the probability of finding a service in the cache also affects the time delay because that way we could avoid the transmission delay in the Internet.

F. Traffic Overhead

$\mathbf{O}_{exp} = (2 - P_{cacheHit}) \times (2 \times N \times R_{req})$ … (2)

Equation 2 provides the expected traffic overhead in the network when all the clients query for services. In order to provide the bandwidth efficient solution the average traffic should always be less than the worst case scenario.

In order to show the efficacy of our proposed scheme, we will compare our results with two different SD schemes. As we know that not much work has been done in the field of service discovery for WSNs, especially the 6LoWPAN domain, so it was a challenge to find the exact same scenario as for our approach in order to do the comparison of our results. We have compared our work with those approaches that perform SD as the main functionality e.g. the 'Proxy based hierarchal infrastructure service discovery' and the 'Portal based service discovery' so that a fair comparison could be performed.

VI. Results And Discussion

The results are obtained through the analytical evaluation as derived in the previous Section V. In our performance evaluation, the metrics in equations (1) and (2) have been used. We have generated large number of service requests from the client nodes in order to find the average time delay for the complete SD procedure, starting from the service request till the service reply. A single client node is taken and we increase the number of service requests per client to estimate the total round trip look up time. Fig. 4 shows the effect of increasing service request rate per client node on the total latency i.e. round trip time delay. Our proposed scheme shows slightly linear response whereas, the portal based SD has a constant behavior. This shows that the portal based SD entertains less service request rate in a degraded fashion by incurring large network delay because the services not in its regional local has to go through the internet transmission and congestion delays irrespective of the number of service requests being made. So the response is almost constant. Portal based SD is basically more optimized for the services available within the regional local of a client and incurs large delay for services not in its close vicinity. As we are targeting the applications requiring management and monitoring at a greater distance, the portal based SD provides a large and somewhat constant delay irrespective of the amount of traffic generated. Our approach proves to be more time efficient especially at lower request rates. The service being requested is usually retrieved from the multiple replicated DNS servers which do not require the request to go to the root server for the look up purpose. Hence internet congestion and heavy load conditions could be avoided which are the main hindrances to achieve timely service discovery. Also the proxy based SD shows linear response but the look up time increases more significantly with even greater slope. It is because of the service look ups performed by many layers of its hierarchal framework which adds more time to the discovery process.

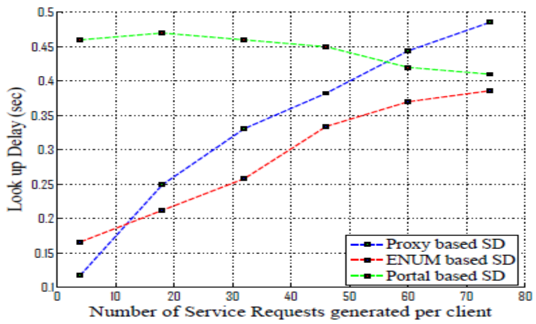

Figure 4. End to end latency versus service request generation rate

One of the factors that could affect the round trip time for the service discovery process is the path length. Number of hop counts from the source to destination determines the path length. One hop delay is small but it cannot be ignored in the multi hop scenario in the sensor network domain. Figure 5 compares the three protocols based on the hop count and its effect on the latency. The service look up time in Portal based SD is independent from the hop count because the protocol does not take into account the path followed by the packet from the client towards the master node. Provided the same delay per hop, our proposed scheme outperforms the proxy based SD because the path length in the proxy based SD is elongated due to the additional device layers of hierarchy.

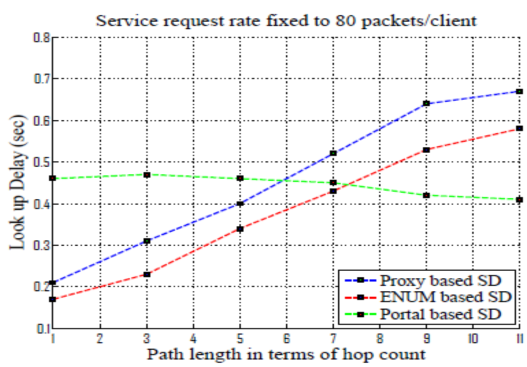

Figure 5. End to end delay versus path length

The traffic overhead in the senor domain should be reduced because of the less bandwidth requirements for the sensor networks. Figure 6 shows the comparison based on the bandwidth utilization. Sensor domain has limited total network bandwidth of say 256kbps. Portal based SD has almost the fair share of bandwidth because of the nearly constant end to end delays irrespective of the number of packets. Our proposed scheme shows significant improvement over portal based SD at the lower delays i.e. nearly 70% improvement but as the packet generation rate increases and there is more and more traffic load in the network, the delay increases and so does the bandwidth consumption but not at an increased rate as the proxy based SD. Hence, our proposed scheme out performs the proxy based SD especially at the increased service request generation rate.

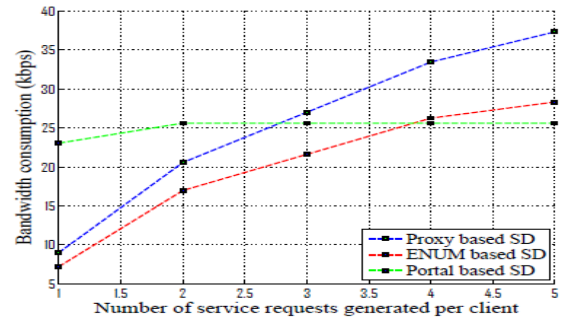

Figure. 6 Bandwidth utilization versus service request generation rate per client

VII. Conclusion

ENUM protocol offers simple mechanism to perform SD on global scale and results in lower service access network delay and lower bandwidth utilization. It is also observed that there is a significant improvement in results over previous work while accessing the services which are not within the proximity of the client.

References

[1] Alan Mainwaring, Joseph Polastre, Robert Szewczyk, David Culler, John Anderson, "Wireless Sensor Networks for Habitat Monitoring" Proceedings of *WSNA'02,* September 28, 2002, Atlanta, Georgia, USA.

[2] E. Kim, D. Kaspar, N. Chevrollier, JP. Vasseur, 2009, "http://tools.ietf.org/html/draft-ietf-6lowpan-usecases-04".

[3] Shafique Ahmad Chaudhary, Won Du Jang, Chaudhary sajjad hussain, Ali Hammad Akbar, Ki Hyung Kim, "A Proxy-Enabled service discovery architecture to findproximity based services in 6LoWPAN" EUC 2006, LNCS 4096, pp. 956 – 965, 2006 [IFIP International Federation for Information Processing 2006].

[4] P. Faltstrom, M. Mealling, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)" 2004, "http://tools.ietf.org/html/rfc3761".

[5] M. Mealling, R. Daniel, "The Naming Authority Pointer (NAPTR) DNS Resource Record" 2000, "http://tools.ietf.org/html/rfc2915".

[6] Available at, http://en.wikipedia.org/wiki/E.164.

[7] Åke Östmark ; Jens Eliasson ; Per Lindgren ; Lianne Meppelink ; Aart van Halteren,"An infrastructure for service oriented sensor networks" Journal of Computers, vol. 1, no. 5, August 2006.

[8] Sun microsystems, Jini Architectural Overview, Technical White Paper, 1999.

[9] Avaialble from, *http://en.wikipedia.org/wiki/Universal_Plug_and_Play*.

[10] Åke Östmark, Per Lindgren, Aart van Halteren, Lianne Meppelink,"*Service and Device Discovery of Nodes in a Wireless Sensor Network*" in Proc. Of IEEE CCNC 2006.

[11] Muhammad Taqi Raza, Seung-Wha Yoo, Ki-Hyung Kim, Seong-Soon Joo and Wun-Cheol Jeong, "Design and Implementation of an Architectural Framework for Web Portals in a Ubiquitous Pervasive Environment" published in *Sensors* 2009, *9*(7), 5201-5223; doi:10.3390/s90705201.

[12] Marin-Perianu R.; Scholten H.; Havinga P.; Hartel P., "Energy-Efficient Cluster-Based Service Discovery in Wireless Sensor Networks" *Local Computer Networks, Proceedings 2006 31st IEEE Conference on* , vol., no., pp.931-938, 14-16 Nov. 2006.

[13] Martin Krebs, Karl-Heinz Krempels, Markus Kucay, "Service Discovery in Wireless Mesh Networks" appeared in WCNC 2008 proceedings.

[14] Xu Li; Santoro N.; Stojmenovic I., "Localized Distance-Sensitive Service Discovery in Wireless Sensor and Actor Networks" Computers, IEEE Transactions on Volume 58, Issue 9, Sept. 2009 Page(s):1275 - 1288.

[15] Ian F. Akyildiz, Tommaso Melodia, Kaushik R. Chowdhury, "A survey on wireless multimedia sensor networks" Appeared in Computer Networks Journal, Volume 51, Issue 4, March 2007.

[16] William Adjie-Winoto, Elliot Schwartz, Hari Balakrishnan, and Jeremy Lilley, "The design and implementation of an intentional naming system" SOSP '99, 34(5):186–201, Dec. 1999.

[17] Avaialable at, http://en.wikipedia.org/wiki/LDAP